A Reinforcement Learning Approach to Improve Low-Resource Machine Translation Leveraging Domain Monolingual Data

Hongxiao Zhang[1]*, Mingtong Liu[2], Chunyou Li[1], Yufeng Chen[1]†, Jinan Xu[1], Ming Zhou[2]
[1]Beijing Key Lab of Traffic Data Analysis and Mining,
Beijing Jiaotong University, Beijing, China
[2]Beijing Lanzhou Technology Co., Ltd., Beijing, China.
{hongxiaozhang,chunyouli,chenyf,jaxu}@bjtu.edu.cn
{liumingtong,zhouming}@langboat.com

Abstract

Due to the lack of parallel data, the mainstream fine-tuning-based domain adaptation methods have the overfitting problem in the translation of low-resource domains, and it is difficult for the model to learn the in-domain generalization knowledge. To address the above issue, in this work, we propose a novel **R**einforcement **L**earning **D**omain **A**daptation method for **N**eural **M**achine **T**ranslation (**RLDA-NMT**) in the low-resource domain. RLDA-NMT utilizes in-domain source monolingual data to make up for the lack of parallel data, and reinforces domain features learning to make the translation model learn the domain-specific knowledge more fully. Specifically, we first train a ranking-based model with a small-scale in-domain parallel corpus, and then adopt it as the reward model to select higher-quality generated translations for reinforcement when fine-tuning pre-trained NMT model using in-domain source monolingual data. We conduct experiments on *Education*, *Laws*, *Thesis*, and *Patent* domains of Chinese⇔English translation tasks. Experimental results demonstrate that RLDA-NMT can alleviate overfitting and reinforce the NMT model to learn domain-specific knowledge. Additionally, the results also show that RLDA-NMT and back-translation (BT) are nicely complementary to each other, where combining RLDA-NMT with BT can further improve translation quality.

Keywords: Low-Resource Machine Translation, Reinforcement Learning, Domain Adaptation

1. Introduction

General neural machine translation (NMT) models often perform poorly on specific domains (Koehn and Knowles, 2017; Chu and Wang, 2018; Saunders, 2022), while some low-resource domains do not have enough parallel training data. Domain adaptation is one of the promising solutions for the NMT task to deal with data scarcity in low-resource domains (Koehn and Schroeder, 2007; Daumé Iii and Jagarlamudi, 2011; Yang et al., 2018), which uses a large-scale out-of-domain parallel corpus and a small-scale in-domain parallel corpus to improve in-domain translation performance. A basic and mainstream domain adaptation method is continuous training, also known as fine-tuning, using in-domain parallel data (Luong and Manning, 2015) or a mixture of in-domain and out-of-domain parallel data (Chu et al., 2017) to continue training the model pre-trained with out-of-domain data. On this basis, some studies have explored modifying the optimization objectives (Khayrallah et al., 2018) or model architecture (Thompson et al., 2019; Shao and Feng, 2022) to further optimize the utilization of in-domain data.

However, for those domains with fewer resources, only using bilingual data to fine-tune the model is prone to forgetting and overfitting (Chu et al., 2017, Saunders, 2022), and it is difficult for the model to learn the generalization knowledge of the domain. Compared with the scarcity of bilingual data, domain monolingual corpus is much easier to obtain (Sun et al., 2019; Zhang et al., 2022). Therefore, research on improving domain translation performance using domain monolingual data has been extensive in the recent literature (Gulcehre et al., 2015; Sennrich et al., 2016; Dou et al., 2019a). The mainstream approach in these studies is to augment the training data with synthetic in-domain corpora generated through back-translation (BT; Sennrich et al., 2016), copying (Currey et al., 2017), or word-by-word translation (Hu et al., 2019). The problem faced by the above schemes is that the generated data is usually noisy, and they only use the target monolingual data, without fully exploring the knowledge of the source monolingual data.

In this work, we propose a **R**einforcement **L**earning **D**omain **A**daptation method for **N**eural **M**achine **T**ranslation (**RLDA-NMT**), which explores mining the knowledge of in-domain source monolingual data to improve domain adaptation through reinforcement learning (RL). Specifically, RLDA-NMT is a self-supervised learning method that can automatically enhance high-quality domain translations with a reward-based mechanism. Firstly, we use

* Contribution during internship at Beijing Lanzhou Technology Co., Ltd., Beijing, China.
†Corresponding author.

LREC-COLING 2024, pages 1486–1497
20-25 May, 2024.

the out-of-domain parallel data to pre-train an NMT model and in-domain parallel data for preliminary fine-tuning to get the base NMT model. Secondly, we train a translation quality estimation (QE) model with the ranking dataset constructed by in-domain parallel data, where high-quality translations are associated with high scores predicted by the QE model. Finally, we further fine-tune the base NMT model with domain source monolingual data using the Proximal Policy Optimization (PPO; Schulman et al., 2017), where the output of the QE model is used as a reward score.

We evaluate the proposed method on four domains of the Chinese⇔English translation tasks, namely *Education*, *Laws*, *Thesis*, and *Patent*. Experimental results demonstrate that RLDA-NMT can improve translation performance, alleviate overfitting, and strengthen the learning of domain features in low-resource domains. In addition, we also explore and prove the complementarity of our method with the mainstream back-translation (BT) method. Combining the proposed method with BT can further improve the performance of the model.

In summary, our contributions can be summarized as follows:

- We propose an RL method for domain adaptation of NMT (RLDA-NMT), leveraging domain source monolingual data to improve translation performance in low-resource domains through a reward-based mechanism.
- We conduct extensive experiments on four domains of Chinese⇔English translation tasks. Experimental results show that our method improves in all domains.
- We demonstrate the complementarity of the proposed method to the mainstream back-translation (BT) method. The combination of RLDA-NMT and BT can make the model achieve the optimal effectiveness.

2. Related work

2.1. Domain Adaptation of NMT

Domain adaptation is a significant approach for improving the performance of machine translation in low-resource domains (Wu et al., 2018; Uc-Cetina et al., 2023). The most basic approach to domain adaptation is to fine-tune a general pre-trained model using small-scale in-domain parallel data (Chu et al., 2017; Freitag and Al-Onaizan, 2016; Luong and Manning, 2015). Some have also investigated improvements based on fine-tuning: Khayrallah et al. (2018) propose a solution that minimizes the cross-entropy between the output word distribution of the model and the out-of-domain model. Thompson et al. (2019) address the catastrophic forgetting problem in domain adaptation for NMT via Elastic Weight Consolidation. And Shao and Feng (2022) solve the problem of unbalanced training through Complementary Online Knowledge Distillation. Although the above schemes have been proven to be effective, for low-resource domains with fewer corpora, the method of fine-tuning only using bilingual data is prone to overfitting, and it is difficult for the model to learn a common representation of the domain.

Existing mainstream works on domain adaptation in the aforementioned scenarios are data-centric approaches (Wu et al., 2018), which explore selecting training data from out-of-domain parallel data (Cuong and Sima'an, 2014; Durrani et al., 2015) or constructing pseudo-parallel data using in-domain monolingual corpora (Sennrich et al., 2016; Hu et al., 2019) to fine-tune pre-trained NMT models. Sennrich et al. (2016) use back-translation to construct in-domain pseudo-parallel sentence pairs, and similarly, Chinea-Rios et al. (2017) use forward-translation approach to generate them, but it has been proven to introduce noise to the decoder in low-resource scenarios (Haddow et al., 2022a). To further improve the quality of pseudo-parallel data, Hoang et al. (2018) introduce an iterative process to continuously optimize the NMT model and generated data. On this basis, Kumari et al. (2021) apply classifiers to filter the synthetic data, and Zhang et al. (2022) proposed Iterative Constrained Back-Translation to incorporate in-domain lexical knowledge. In addition, there are also some model-centric studies that modify the objective function (Dou et al., 2019a; Wei et al., 2020) and model architecture (Gulcehre et al., 2015; Cheng et al., 2016; Dou et al., 2019b) to improve the adaptability of the NMT model.

2.2. Reinforcement Learning for NMT

Recently, some researchers have explored the application of reinforcement learning (RL) in natural language processing (NLP) tasks, such as dialogue systems (Crook et al., 2014; Zhao et al., 2019; Chen et al., 2020), text generation systems (Shi et al., 2018; Keneshloo et al., 2020; Ouyang et al., 2022), text summarization (Stiennon et al., 2020), machine translation (Wu et al., 2018; Lam et al., 2019; Keneshloo et al., 2020) and so on.

We focus on the application of reinforcement learning in NMT, which is originally proposed to solve these two problems (Keneshloo et al., 2020): (1) the token-level maximum likelihood estimation (MLE) objective function during training is inconsistent with the sequence-level evaluation metrics (such as BLEU (Papineni et al., 2002)) at test time, and (2) exposure bias, where inference during training relies on the input golden sentences, while during the test it relies on the outputs of the model.

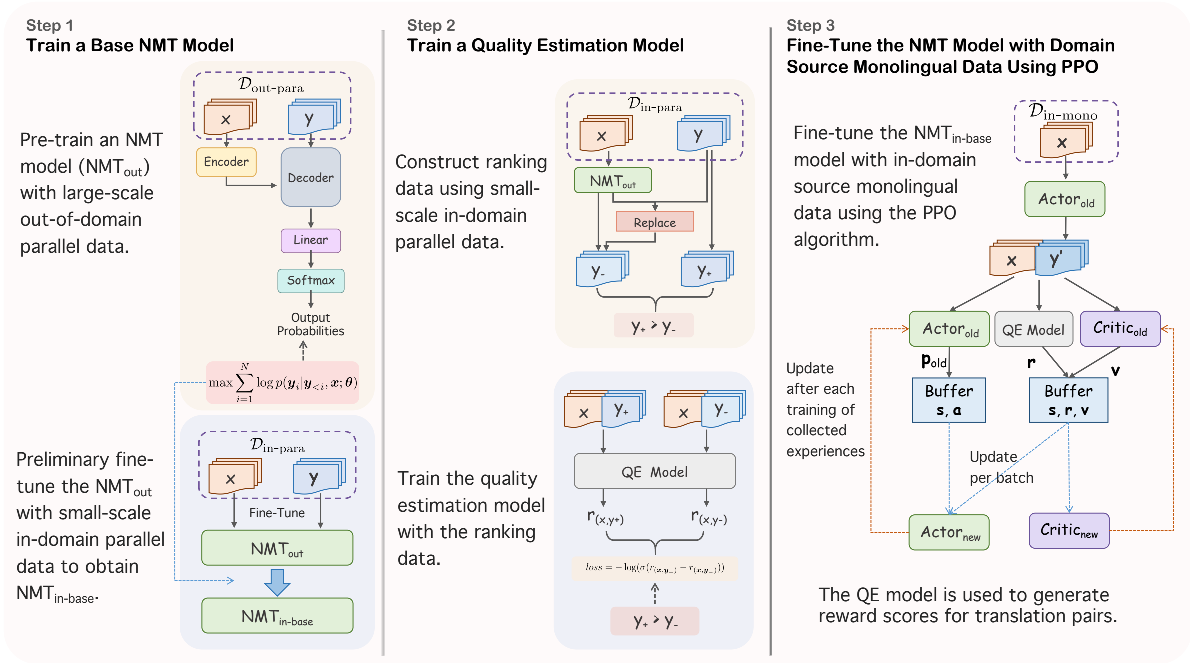

Figure 1: The main pipeline of the proposed method (RLDA-NMT), which mainly includes 3 steps: (1) pre-training and preliminary fine-tuning of the NMT model (§ 3.1), (2) training of the quality estimation model with ranking dataset constructed by in-domain parallel data (§ 3.2), and (3) fine-tuning of the NMT model with in-domain source monolingual data using reinforcement learning (§ 3.3). In Step 3, $\text{Actor}_{\text{old}}$ and $\text{Actor}_{\text{new}}$ are initialized by $\mathrm{NMT}_{\text{in-base}}$, $\text{Critic}_{\text{old}}$ and $\text{Critic}_{\text{new}}$ are initialized by the QE model.

Shen et al. (2016) propose a minimum-risk training method to directly optimize model parameters for any evaluation metric. Wu et al. (2017) apply policy optimization methods to sequence prediction tasks including NMT. Wu et al. (2018) are the first to explore the practical application of RL in NMT in real-world systems based on transformer-based NMT models and huge datasets. The above studies focus on general-domain translation, for which a large amount of parallel data is available. In contrast, this work focuses on the application of reinforcement learning when transferring translation models to low-resource domains. We delve into the utilization of monolingual data when fine-tuning NMT, while paying attention to terminological translation accuracy, domain-style translation, etc., which are not available in previous studies.

3. Our Approach

In this work, we propose a Reinforcement Learning Domain Adaptation method for Neural Machine Translation (RLDA-NMT), which mainly consists of three steps (Figure 1). We start with a pre-trained NMT model (§ 3.1), use the available small-scale in-domain parallel data as the seed data to train a ranking-based QE model (§ 3.2), and reinforce domain-specific translations when fine-tuning the NMT model with domain source monolingual data (§ 3.3), so as to improve the translation performance of the NMT model in low-resource domains.

3.1. Train a Base NMT Model

As shown in Figure 1, in the beginning, out-of-domain parallel data ($\mathcal{D}_{\text{out-para}}$) is used to train a baseline NMT model ($\mathrm{NMT}_{\text{out}}$). We follow the neural machine translation architecture proposed by Vaswani et al. (2017), which builds an encoder-decoder architecture based on attention mechanism. The parameters of the NMT model are optimized by maximizing the objective function:

$$\mathcal{L}_{nmt} = \sum_{k=1}^{K} \sum_{i=1}^{N^k} \log p\left(y_i | \boldsymbol{y}_{<i}^k, \boldsymbol{x}^k; \boldsymbol{\theta}\right) \tag{1}$$

where $(\boldsymbol{x}^k, \boldsymbol{y}^k) \in \mathcal{D}_{\text{out}-\text{para}}$, K is the number of out-of-domain parallel sentence pairs, and N^k is the number of tokens in $\boldsymbol{y}^k$.

After pre-training, a small-scale in-domain parallel corpus ($\mathcal{D}_{\text{in-para}}$) is used to initially fine-tune the pre-trained NMT model ($\mathrm{NMT}_{\text{out}}$) through the objective function in Equation 1. We define the fine-tuned model as $\mathrm{NMT}_{\text{in-base}}$.

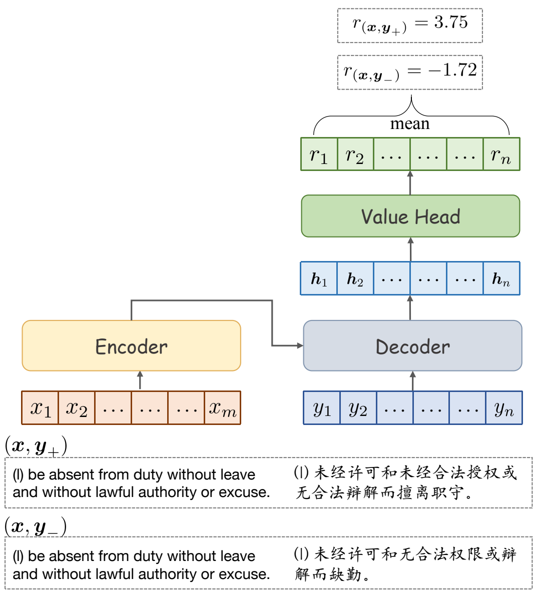

Figure 2: The structure of the quality estimation (QE) model. It is based on the encoder-decoder architecture, and a linear layer Value Head is connected after the Decoder layer to generate the score of each token of the target sentence. The score of the translation pair is the average of the scores of tokens.

3.2. Train a Ranking-Based Model for Translation Quality Estimation

The quality estimation (QE) model is used to score translated sentence pairs. It learns the ability that more domain-specific translations should be scored higher. We utilize the small-scale in-domain parallel corpus ($\mathcal{D}_{\text{in-para}}$) mentioned in Section 3.1 to construct ranking dataset to train the QE model.

Construct the ranking data. As shown in Step 2 of Figure 1, for each sentence pair $(\boldsymbol{x}, \boldsymbol{y}_+)$ in $\mathcal{D}_{\text{in-para}}$, we construct a poor sentence pair $(\boldsymbol{x}, \boldsymbol{y}_-)$ for it. Considering domain adaptation, we focus on two aspects when constructing data: *translation style* and *terminological translation accuracy*. First, NMT_{out} is used to translate $\boldsymbol{x}$ into $\boldsymbol{y}'$ to construct $(\boldsymbol{x}, \boldsymbol{y}')$. The ground truth $(\boldsymbol{x}, \boldsymbol{y}_+)$ is an in-domain style translation pair, compared to $(\boldsymbol{x}, \boldsymbol{y}')$ which is out-of-domain translation style. Ranking data structured in this way can strengthen domain-style translation. Considering to strengthen the translation of domain-specific terms, we substitute corresponding terms with generic expressions in the target sentences of translation pairs, *i.e.*, $(\boldsymbol{x}, \boldsymbol{y}_+)$ and $(\boldsymbol{x}, \boldsymbol{y}')$. In particular, we utilize an in-domain terminological bilingual dictionary[1] to match the translation pairs. For each matched term pair, we replace its position in the target sentence with a generic expression, generated by NMT_{out}. In this manner, we generate negative translation pairs $(\boldsymbol{x}, \boldsymbol{y}_r)$ and $(\boldsymbol{x}, \boldsymbol{y}'_r)$ for domain-specific terms by operating on $(\boldsymbol{x}, \boldsymbol{y}_+)$ and $(\boldsymbol{x}, \boldsymbol{y}')$, respectively. Overall, the final negative translation pairs $(\boldsymbol{x}, \boldsymbol{y}_-)$ consist of three parts, *i.e.*, $(\boldsymbol{x}, \boldsymbol{y}')$, $(\boldsymbol{x}, \boldsymbol{y}_r)$, and $(\boldsymbol{x}, \boldsymbol{y}'_r)$.

Train the quality estimation model. The structure of the QE model is shown in Figure 2. Unlike Ouyang et al. (2022) who build with a language model, we choose the encoder-decoder architecture to build our QE model, which is more suitable for translation tasks. The Encoder layer and Decoder layer of the QE model are initialized by NMT_{out}, and the Value Head layer is randomly initialized. For an input translation pair $(\boldsymbol{x}, \boldsymbol{y}_+)$ or $(\boldsymbol{x}, \boldsymbol{y}_-)$, the model first generates its hidden representation $(\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_n)$ through the Encoder and Decoder layers. Then, a linear layer Value Head with a latitude of $(\text{hidden\_size}, 1)$ is used to map the hidden representation $(\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_n)$ into the reward score $(r_1, r_2, \ldots, r_n)$. Finally, $r_{(\boldsymbol{x},\boldsymbol{y}_+)}$ or $r_{(\boldsymbol{x},\boldsymbol{y}_-)}$ is obtained by averaging the reward scores of each token. Following Stiennon et al. (2020), we optimize the QE model with the preference loss:

$$\mathcal{L}_{qe} = -\log\left(\sigma\left(r_{(\boldsymbol{x},\boldsymbol{y}_+)} - r_{(\boldsymbol{x},\boldsymbol{y}_-)}\right)\right) \quad (2)$$

where σ is the *sigmoid* function.

3.3. Fine-Tune the NMT Model with Domain Source Monolingual Data Using PPO

We use the QE model to generate reward scores for translation pairs, and use in-domain source monolingual data ($\mathcal{D}_{\text{in-mono}}$) to fine-tune $\text{NMT}_{\text{in-base}}$ with the PPO (Schulman et al., 2017). PPO includes two optimization networks, namely: 1) ActorNet, which predicts the probability of actions in each state, and 2) CriticNet, which predicts the score of each state and participates in calculating the scores of actions. In this work, the $\text{NMT}_{\text{in-base}}$ model is used as the ActorNet, and the CriticNet is initialized by the QE model.

Specifically, as shown in Step 3 of Figure 1, for each input sentence $\boldsymbol{x} \in \mathcal{D}_{\text{in-mono}}$, the $\text{Actor}_{\text{old}}$ model generates its translation $\boldsymbol{y}'$, and the QE model outputs the score r of the translation pair $(\boldsymbol{x}, \boldsymbol{y}')$. The process of fine-tuning the NMT model using the PPO algorithm is shown in Algorithm 1. The objective function of optimizing ActorNet parameters through PPO is:

$$\begin{aligned}\mathcal{L}_{pg} = &\hat{\mathbb{E}}_t\left[\min\left(ratio_t \cdot A_t,\right.\right.\\ &\left.\left. clip\left(ratio_t, 1-\epsilon, 1+\epsilon\right) \cdot A_t\right)\right].\end{aligned} \quad (3)$$

where $ratio_t = \dfrac{p_{new}(a_t|s_t)}{p_{old}(a_t|s_t)}$ is the probability ratio of the new ActorNet to the old ActorNet, and A_t is

[1]The in-domain terminological bilingual dictionary used in this work is internal unpublished data.

Algorithm 1 Fine-Tuning NMT model using PPO

Input: base NMT model $\mathrm{NMT}_{\text{in-base}}$, in-domain source monolingual data $\mathcal{D}_{\text{in-mono}}$, QE model, collect size N.
Output: fine-tuned translation model

```
Actor_new ← NMT_in-base; Critic_new ← QE;
for iter = 1, 2, 3 ... do
    Actor_old ← Actor_new; Critic_old ← Critic_new
    for n < N do
        sample an source sentence x
        translate x to y'
        get the probability p_old(a_1|s_1), p_old(a_2|s_2), ..., p_old(a_T|s_T) from Actor_old
        get the state value v_1, v_2, ..., v_T from Critic_old
        get reward score r_1, r_2, ..., r_T from QE
        experience = (x, y', p_old, v, r)
    end for
    for mini-batch in experience batches do
        calculate A_1, A_2, ..., A_T for each experience
        get p_new from Actor_new and v^θ from Critic_new
        optimize Actor_new and Critic_new
    end for
end for
return fine-tuned NMT model
```

the advantage of the action a_t. When $A_t > 0$, the ratio $ratio_t$ is strengthened, otherwise it is weakened. $clip$ is used to limit the update range of $ratio_t$ to enhance the stability of reinforcement learning. Following Schulman et al. (2017), we define A_t as:

$$A_t = \delta_t + (\gamma\lambda)\delta_{t+1} + \cdots + (\gamma\lambda)^{T-t+1}\delta_{T-1} \quad (4)$$

where $\delta_t = r_t + \gamma v_{t+1} - \gamma v_t$, v_t is the value score generated by $\mathrm{Critic}_{\text{old}}$, and r_t is the reward score generated by the QE model. The CriticNet is optimized through a squared-error loss function:

$$\mathcal{L}_v = \left(v_t^{\theta} - v_t^{targ}\right)^2 \quad (5)$$

where v_t^{θ} is the value predicted by the $\mathrm{Critic}_{\text{new}}$, and v_t^{targ} represents the real return value, which is calculated by:

$$v_t^{targ} = A_t + v_t \quad (6)$$

In order to further stabilize the RLDA-NMT training process, we combine the NMT supervised training objective and PPO objective (Wu et al., 2018). We maximize the following combined objective function to optimize the ActorNet:

$$\mathcal{L}_{com} = \mathcal{L}_{pg} + \alpha\mathcal{L}_{nmt} \quad (7)$$

where α is a hyperparameter that controls the weight of $\mathcal{L}_{nmt}$.

4. Experiments

4.1. Data and Setup

Datasets. We evaluate the proposed approach on Chinese⇔English (Zh⇔En) translation tasks.

Domain	Train	Dev	Test	Mono
UN	17731615	–	–	–
Education	29989	3000	790	208500
Laws	29832	3000	456	415893
Thesis	29973	3000	625	133500
Patent	30000	3000	4382	282751

Table 1: Corpus Statistics for our experiments.

The UN Parallel Corpus V1.0 (Ziemski et al., 2016) is used as the out-of-domain training dataset. For in-domain parallel corpus, we use the *Laws*, *Education*, and *Thesis* domain data from UM-corpus (Tian et al., 2014), and use the *Patent* domain data from ParaPat (Soares et al., 2020)[2]. To simulate NMT in the low-resource scenario, for each domain, we randomly sample 30K parallel sentence pairs as a training set, and 3K pairs as a validation set, and the rest are used to construct the monolingual dataset. The data similar to the test set are filtered out for fair comparison. Note that there is no public test dataset in the *Patent* domain, so we sample 4K test data that does not overlap with the training and validation datasets.

We follow Hu et al. (2019) to construct the non-parallel monolingual datasets. Specifically, for each domain, the rest of the parallel data is randomly divided into two equal parts. Then, the source sentences of the former part and the target sentences of the latter part are taken as our monolingual datasets. In addition, for the *Laws* domain, we extract some monolingual data from the general data using a publicly available similarity retrieval algorithm[3] because of its small size. After cleaning and filtering, the statistics of the data we finally used are shown in Table 1.

Implementation Details. We implement the proposed method on OpenNMT-py (Klein et al., 2017). We adopt the Transformer model with *Transformer-base* setting as defined by Vaswani et al. (2017). All data is pre-processed through *SentencePiece* library (Kudo and Richardson, 2018). The optimizer used for both NMT training and RL training is Adam (Kingma and Ba, 2015), with initial *learning rate* is 2, *decay_method* is Noam, and *warmup_steps* is 15000. The *label_smoothing* and *dropout* are set to 0.1. During pre-training, the batch size is set to $8192 \times 8 \times 16$, where 8192 is the maximum number of tokens on each GPU, 8 is the number of GPUs, and 16 is the accumulative count of the gradient. The NMT model in each direction is trained for 20K steps (about 30 epochs). For fine-tuning of NMT in

[2]We choose these domain based on the quality of the data.

[3]https://github.com/shibing624/text2vec

	Zh-En							
	Education		Laws		Thesis		Patent	
	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET
Unadapted	27.92	0.603	46.06	0.604	18.15	0.072	26.55	0.232
Fine-Tuning	29.22	0.614	50.81	0.685	21.12	0.114	38.08	0.327
Forward-Translation	29.33	0.611	50.20	0.676	19.10	0.103	30.31	0.270
RLDA-NMT$^{(Ours)}$	**30.81**	**0.633**	**51.12**	**0.712**	**21.96**	**0.137**	**38.57**	**0.364**
Back-Translation	29.48	0.615	**51.68**	0.712	21.67	0.141	38.94	0.328
RLDA-NMT+BT$^{(Ours)}$	**30.92**	**0.656**	51.58	**0.716**	**22.22**	**0.152**	**39.41**	**0.330**

	En-Zh							
	Education		Laws		Thesis		Patent	
	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET
Unadapted	34.26	0.712	54.10	0.867	29.19	0.229	33.14	0.276
Fine-Tuning	36.77	0.745	**63.58**	0.959	34.48	0.425	47.43	0.469
Forward-Translation	35.76	0.725	59.74	0.933	31.70	0.336	37.98	0.348
RLDA-NMT$^{(Ours)}$	**36.92**	**0.757**	63.40	**0.971**	**34.50**	**0.427**	**47.68**	**0.478**
Back-Translation	36.70	0.767	64.87	0.968	36.29	0.452	**49.02**	0.484
RLDA-NMT+BT$^{(Ours)}$	**37.02**	**0.786**	**64.91**	**0.975**	**36.53**	**0.465**	49.01	**0.488**

Table 2: Experimental results for Zh-En and En-Zh translation tasks on four low-resource domains.

each domain, the batch size is set to $4096 \times 1 \times 4$[4], and the model in each domain is trained for 10K steps with the *learning rate* is 0.0002. We save the intermediate checkpoints every 1K steps and choose the best-performing model as a baseline for comparison. The checkpoint model after fine-tuning 3K steps is used as the $\mathrm{NMT}_{\text{in-base}}$. We train the QE model for 10 epochs in each domain with a learning rate of 0.000005. And for RLDA-NMT training, we set the collection size N to 5K, the threshold ϵ of the $clip$ function to 0.2, the γ in Equation 4 to 1.0, λ in Equation 4 to 0.95, and α in Equation 7 to 15 in all experiments. We train *Education* RLDA-NMT models with the *learning rate* is 0.000005 and train the other RLDA-NMT models with 0.00005. Different learning rates are set to stabilize the training process. SacreBLEU[5] python package is used to calculate the BLEU score. For Chinese, we calculate the BLEU at the character granularity. We also report COMET (Rei et al., 2020) scores, which are shown to have higher correlation with human judgment, to further evaluate translation quality. The `eamt22-cometinho-da` model (Rei et al., 2022) is used to generate the COMET scores.

Experimental Comparison. Our goal is to empirically test whether the proposed reinforcement learning method is effective for translation in low-resource domains and explore its complementarity with data-centric approaches. Therefore, we compare the following methods:

- **Unadapted.** The NMT model is trained with the out-of-domain parallel corpus and directly evaluated with the in-domain test dataset.
- **Fine-Tuning** (Chu et al., 2017). The NMT model is fine-tuned by mixing the out-of-domain parallel data and in-domain parallel data in equal proportions after pre-training.
- **Forward-Translation** (Chinea-Rios et al., 2017). A method that fine-tunes pre-trained NMT model using synthetic in-domain parallel data generated by translating source monolingual data with a source-to-target NMT model.
- **Back-Translation** (Sennrich et al., 2016). A method similar to *Forward-Translation*, but using synthetic in-domain parallel data generated by translating target monolingual data with a target-to-source NMT model.
- **RLDA-NMT**. The proposed method that fine-tunes the pre-trained NMT model using in-domain source monolingual data through reinforcement learning.
- **RLDA-NMT+BT.** Combination of the proposed method with BT to evaluate the complementarity of RLDA-NMT and data-centric methods. The data synthesized by BT is added as supervised training when fine-tuning NMT using reinforcement learning.

[4] Compared to pre-training, we reduced the batch size to slow down the fitting.

[5] https://github.com/mjpost/sacrebleu

4.2. Main Results

Table 2 shows the performance of the baseline methods and our methods on Zh-En and En-Zh translation tasks on four domains. We have the following observations.

Firstly, compared with *Fine-Tuning*, RLDA-NMT has improved the BLEU scores of most translation tasks, especially in *Education* of the Zh-En translation direction, which has improved by 1.59 compared with the *Fine-Tuning* method. There is also a 0.2~0.7 improvement in most other tasks. For the COMET scores, RLDA-NMT achieves the best performance in all translation tasks, and most of them have observable improvements. It proves that introducing reinforcement learning into the domain adaptation of NMT is effective.

Secondly, the performance of the *Forward-Translation* method has mostly declined compared to the *Fine-Tuning* method, especially in the *Patent* domain, where there has been a significant decline. This phenomenon can be attributed to the reason that large amounts of forward-translated data introduce noise to the decoder in low-resource settings, as Haddow et al. (2022b) have described. In contrast, our method introduces reinforcement learning to automatically score the current translation generation and judge whether it should be strengthened or weakened, which will filter the noisy data. Therefore, the benefits of RLDA-NMT come from RL rather than the expansion of data volume.

Finally, the combination of RLDA-NMT and BT brings further improvements, which validates the complementarity of RLDA-NMT and data-centric approaches. As we analyzed earlier, data-centric methods usually only utilize the target monolingual data, without mining the rich information in the source monolingual data. In contrast, RLDA-NMT+BT utilizes both source and target domain monolingual data at the same time, thus bringing further improvements.

5. Analysis

In this section, we conduct several interesting analyses and controlled experiments to explore the substantial improvement of applying reinforcement learning to the domain adaptation of NMT.

5.1. Alleviation of Overfitting

During the experiment, we found that it is difficult to get a satisfactory model when fine-tuning only using in-domain bilingual data. When we try smaller learning rates, the model hardly converges and the performance is low. When trying a larger learning rate, the model is prone to overfitting. Our expectation is that reinforcement learning can alleviate this problem to some extent.

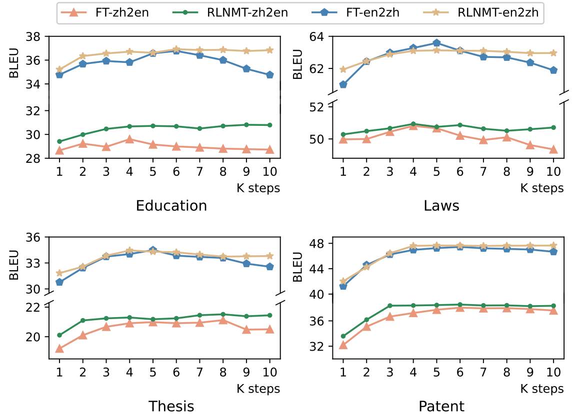

Figure 3: BLEU scores of different methods according to the number of training steps on the test sets of four domains. *-zh2en and *-en2zh represent Zh-En and En-Zh translation tasks respectively.

We save the checkpoint model every 1000 steps when fine-tuning the pre-trained NMT model with or without reinforcement learning. We evaluate the performance of each model on the test set, and the evaluation results are shown in Figure 3. It can be seen that in each translation task, the performance of *Fine-Tuning* (FT) models on the test set tend to increase first and then decrease. In the later stages of training, the model overfits the training set, leading to a decrease in generalization ability. In contrast, RLDA-NMT models do not appear to be overfitting in most cases. Even if there is a slight decrease, it can stabilize at a relatively high performance. It is in line with our expectation that using reinforcement learning to fine-tune the NMT model can alleviate the overfitting phenomenon and improve the generalization ability of the model.

5.2. Effect of the Quality of the QE Model

The scores generated by the QE model for translation pairs guide the process of fine-tuning using reinforcement learning (§ 3.3). Therefore, the quality of the QE model is crucial to the quality of the final NMT model. To explore the relationship between the performance of the two models[6], we used the underperforming QE model to guide the reinforcement learning process and recorded the performance of the final NMT model. Specifically, we conduct experiments in the *Education* domain of the Zh-En translation task. We take the QE models of the 2nd, 4th, 6th, 8th, and 10th epochs in the QE training process to guide the RL fine-tuning

[6]We measure the performance of the QE model using the accuracy of the model on the validation set, which is identically distributed and non-overlapping with the training set.

Epoch	QE (ACC%)	NMT(BLEU)
2	57.39	29.05
4	79.59	30.18
6	87.31	30.53
8	92.17	30.76
10	93.13	30.81

Table 3: Correlation between QE model performance and final NMT performance. ACC% refers to the accuracy of the QE model in scoring the ranking data of the validation dataset.

Data Size	BLEU		
	FT	RLDA-NMT	Δ
10K	29.17	30.74	+1.57
30K	29.22	30.81	+1.59
50K	29.58	30.93	+1.35
100K	29.81	30.98	+1.17

Table 4: Comparison of BLEU scores of *Fine-Tuning* (FT) and RLDA-NMT in Zh-En *Education* translation tasks under scenarios with different amounts of in-domain parallel data. Δ represents the difference between the BLEU score of RLDA-NMT method and *Fine-Tuning* method.

process. The obtained results are shown in Table 3.

It can be seen that when the QE model performance is poor, especially in Epoch 2, the NMT model after fine-tuning using RL drops by 0.2 BLEU, which shows that the poor-performing QE model will introduce noise into the final NMT model. As the performance of the QE model improves, the final NMT performance also gradually improves and tends to saturation. It is in line with our expectations that a better QE model can lead to a better NMT model, but as QE performance gradually improves, the increase in NMT performance gradually decreases and tends to saturation.

5.3. Effect of the In-Domain Parallel Data Size

We explore the utility of RLDA-NMT in scenarios with different sizes of in-domain parallel data to further explore the improvements brought by reinforcement learning. Concretely, we conduct experiments in the *Education* domain of the Zh-En translation task. We compare RLDA-NMT and *Fine-Tuning* in scenarios where the size of in-domain parallel data is [10K, 30K, 50K, 100K]. We conduct experiments in each scenario and evaluate each fine-tuned NMT model with the test set. The experimental results are shown in Table 4.

As we can see, the performance of both *Fine-Tuning* (FT) and RLDA-NMT improves as the size of in-domain parallel data increases, and RLDA-NMT consistently outperforms FT. It proves that RLDA-NMT can bring improvement both in scenarios with little or some in-domain parallel data. Furthermore, RLDA-NMT shows greater improvement over FT in lower-resource scenarios (+1.57 BLEU in the 10K scenario and +1.59 BLEU in the 30K scenario vs. +1.17 BLEU in the 100K scenario), which proves that the application of RLDA-NMT in lower-resource scenarios can obtain greater benefits. This is in line with our expectations. If there are fewer parallel resources in the domain, the overfitting problem is more serious, and RLDA-NMT can alleviate this problem and improve translation performance.

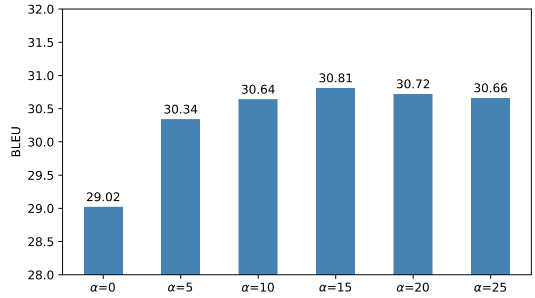

Figure 4: BLEU scores of RLDA-NMT with different weights α of NMT supervised training objective $\mathcal{L}_{nmt}$.

5.4. Effect of the weight of NMT supervised training objective

As shown in Equation 7, the hyperparameter α controls the contribution of the NMT supervised training objective ($\mathcal{L}_{nmt}$) to RL training. For comparison, we set α to be [0, 5, 10, 15, 20, 25] in Zh-En *Education* translation experiments. The results are presented in Figure 4.

The results show that when $\alpha = 0$ (i.e. not combined with the NMT supervised training objective $\mathcal{L}_{nmt}$), RLDA-NMT does not bring improvement, even weaker than the *Fine-Tuning* model. We believe that it is because of the instability of RL, without the participation of $\mathcal{L}_{nmt}$, RL may cause damage to the original training. When $\alpha \leq 15$, the performance of the RLDA-NMT model increases as the weight of the NMT supervised training objective increases, which indicates that the $\mathcal{L}_{nmt}$ can help stabilize the RL training and improve the performance of the RLDA-NMT models. However, when $\alpha > 15$, the performance of the RLDA-NMT model decreases as α increases, we speculate that the overfitting problem of NMT supervised training impairs the improvement brought by RL because of the large weight of $\mathcal{L}_{nmt}$.

6. Conclusion

In this work, we propose a Reinforcement Learning Domain Adaptation method for Neural Machine Translation (RLDA-NMT) in low-resource domains, which leverages domain source monolingual data to fine-tune the pre-trained NMT model through a reward-based mechanism. We use small-scale in-domain parallel data as seed data to train a quality estimation model, and then use the model to automatically score the generated translations, thereby using monolingual data to self-supervised fine-tune the initially trained NMT model and strengthen domain-specific translations. We conduct extensive experiments on four domains of Chinese⇔English translation tasks. The experimental results show that RLDA-NMT can improve the translation quality and alleviate overfitting, and can further supplement back-translation method.

7. Acknowledgements

The research work descried in this paper has been supported by the National Key R&D Program of China (2020AAA0108001) and the National Nature Science Foundation of China (No. 61976016, 61976015, and 61876198). The authors also would like to thank the anonymous reviewers for their valuable comments and suggestions to improve this paper.

8. Bibliographical References

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

Boxing Chen, Colin Cherry, George Foster, and Samuel Larkin. 2017. Cost weighting for neural machine translation domain adaptation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 40–46, Vancouver. Association for Computational Linguistics.

Zhi Chen, Lu Chen, Xiaoyuan Liu, and Kai Yu. 2020. Distributed structured actor-critic reinforcement learning for universal dialogue management. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2400–2411.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Semi-supervised learning for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1965–1974.

Mara Chinea-Rios, Alvaro Peris, and Francisco Casacuberta. 2017. Adapting neural machine translation with parallel synthetic data. In *Proceedings of the Second Conference on Machine Translation*, pages 138–147.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 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 (Volume 2: Short Papers)*, pages 385–391, Vancouver, Canada. Association for Computational Linguistics.

Chenhui Chu and Rui Wang. 2018. A survey of domain adaptation for neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1304–1319, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Paul A. Crook, Simon Keizer, Zhuoran Wang, Wenshuo Tang, and Oliver Lemon. 2014. Real user evaluation of a pomdp spoken dialogue system using automatic belief compression. *Computer Speech & Language*, 28(4):873–887.

Hoang Cuong and Khalil Sima'an. 2014. Latent domain translation models in mix-of-domains haystack. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1928–1939.

Anna Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. 2017. Copied monolingual data improves low-resource neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 148–156, Copenhagen, Denmark. Association for Computational Linguistics.

Hal Daumé Iii and Jagadeesh Jagarlamudi. 2011. Domain adaptation for machine translation by mining unseen words. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 407–412.

Zi-Yi Dou, Junjie Hu, Antonios Anastasopoulos, and Graham Neubig. 2019a. Unsupervised domain adaptation for neural machine translation with domain-aware feature embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1417–1422, Hong Kong, China. Association for Computational Linguistics.

Zi-Yi Dou, Xinyi Wang, Junjie Hu, and Graham Neubig. 2019b. Domain differential adaptation for neural machine translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 59–69, Hong Kong. Association for Computational Linguistics.

Nadir Durrani, Hassan Sajjad, Shafiq Joty, Ahmed Abdelali, and Stephan Vogel. 2015. Using joint models or domain adaptation in statistical machine translation. In *Proceedings of Machine Translation Summit XV: Papers*.

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

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *ArXiv preprint*, abs/1503.03535.

Barry Haddow, Rachel Bawden, Antonio Valerio Miceli Barone, Jindřich Helcl, and Alexandra Birch. 2022a. Survey of low-resource machine translation. *Computational Linguistics*, 48(3):673–732.

Barry Haddow, Rachel Bawden, Antonio Valerio Miceli Barone, Jindřich Helcl, and Alexandra Birch. 2022b. Survey of Low-Resource Machine Translation. *Computational Linguistics*, 48(3):673–732.

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

Vu Cong Duy Hoang, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. 2018. Iterative back-translation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24, Melbourne, Australia. Association for Computational Linguistics.

Junjie Hu, Mengzhou Xia, Graham Neubig, and Jaime Carbonell. 2019. Domain adaptation of neural machine translation by lexicon induction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2989–3001, Florence, Italy. Association for Computational Linguistics.

Yaser Keneshloo, Tian Shi, Naren Ramakrishnan, and Chandan K. Reddy. 2020. Deep reinforcement learning for sequence-to-sequence models. *IEEE Transactions on Neural Networks and Learning Systems*, 31(7):2469–2489.

Huda Khayrallah, Gaurav Kumar, Kevin Duh, Matt Post, and Philipp Koehn. 2017. Neural lattice search for domain adaptation in machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 20–25, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Huda Khayrallah, Brian Thompson, Kevin Duh, and Philipp Koehn. 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*, pages 36–44, Melbourne, Australia. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *Third International Conference on Learning Representations.*

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Philipp Koehn and Josh Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the second workshop on statistical machine translation*, pages 224–227.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language*

Processing: System Demonstrations, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Surabhi Kumari, Nikhil Jaiswal, Mayur Patidar, Manasi Patwardhan, Shirish Karande, Puneet Agarwal, and Lovekesh Vig. 2021. Domain adaptation for NMT via filtered iterative back-translation. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 263–271, Kyiv, Ukraine. Association for Computational Linguistics.

Tsz Kin Lam, Shigehiko Schamoni, and Stefan Riezler. 2019. Interactive-predictive neural machine translation through reinforcement and imitation. In *Proceedings of Machine Translation Summit XVII: Research Track*, pages 96–106.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *Proceedings of the 12th International Workshop on Spoken Language Translation: Evaluation Campaign*, pages 76–79.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Ricardo Rei, Ana C Farinha, José G.C. de Souza, Pedro G. Ramos, André F.T. Martins, Luisa Coheur, and Alon Lavie. 2022. Searching for COMETINHO: The little metric that could. In *Proceedings of the 23rd Annual Conference of the European Association for Machine Translation*, pages 61–70, Ghent, Belgium. European Association for Machine Translation.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Danielle Saunders. 2022. Domain adaptation and multi-domain adaptation for neural machine translation: A survey. *Journal of Artificial Intelligence Research*, 75:351–424.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

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

Chenze Shao and Yang Feng. 2022. Overcoming catastrophic forgetting beyond continual learning: Balanced training for neural machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2023–2036, Dublin, Ireland. Association for Computational Linguistics.

Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum risk training for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1683–1692.

Zhan Shi, Xinchi Chen, Xipeng Qiu, and Xuanjing Huang. 2018. Toward diverse text generation with inverse reinforcement learning. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, page 4361–4367. AAAI Press.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, volume 33, pages 3008–3021. Curran Associates, Inc.

Haipeng Sun, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. 2019. An empirical study of domain adaptation for unsupervised neural machine translation. *arXiv preprint arXiv:1908.09605*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Brian Thompson, Jeremy Gwinnup, Huda Khayrallah, Kevin Duh, and Philipp Koehn. 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, Volume 1 (Long and Short Papers)*, pages 2062–2068, Minneapolis, Minnesota. Association for Computational Linguistics.

Victor Uc-Cetina, Nicolas Navarro-Guerrero, Anabel Martin-Gonzalez, Cornelius Weber, and Stefan Wermter. 2023. Survey on reinforcement learning for language processing. *Artificial Intelligence Review*, 56(2):1543–1575.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

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

Lijun Wu, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2018. A study of reinforcement learning for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3612–3621, Brussels, Belgium. Association for Computational Linguistics.

Lijun Wu, Li Zhao, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2017. Sequence prediction with unlabeled data by reward function learning. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 3098–3104.

Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. 2018. Unsupervised domain adaptation for neural machine translation. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 338–343.

Hongxiao Zhang, Hui Huang, Jiale Gao, Yufeng Chen, Jinan Xu, and Jian Liu. 2022. Iterative constrained back-translation for unsupervised domain adaptation of machine translation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5054–5065, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Tiancheng Zhao, Kaige Xie, and Maxine Eskenazi. 2019. Rethinking action spaces for reinforcement learning in end-to-end dialog agents with latent variable models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1208–1218, Minneapolis, Minnesota. Association for Computational Linguistics.

9. Language Resource References

Felipe Soares, Mark Stevenson, Diego Bartolome, and Anna Zaretskaya. 2020. ParaPat: The multi-million sentences parallel corpus of patents abstracts. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 3769–3774, Marseille, France. European Language Resources Association.

Liang Tian, Derek F. Wong, Lidia S. Chao, Paulo Quaresma, Francisco Oliveira, Yi Lu, Shuo Li, Yiming Wang, and Longyue Wang. 2014. UM-corpus: A large English-Chinese parallel corpus for statistical machine translation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1837–1842, Reykjavik, Iceland. European Language Resources Association (ELRA).

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia. European Language Resources Association (ELRA).