# Review-based Curriculum Learning for Neural Machine Translation

Ziyang Hui[1], Chong Feng[1] and Tianfu Zhang[1]

[1] School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China
{huiziyang, fengchong, tianfuzhang}@bit.edu.cn

**Abstract.** For Neural Machine Translation (NMT) tasks with limited domain resources, curriculum learning provides a way to simulate the human learning process from simple to difficult to adapt the general NMT model to a specific domain. However, previous curriculum learning methods suffer from catastrophic forgetting and learning inefficiency. In this paper, we introduce a review-based curriculum learning method, targetedly selecting curriculum according to long time interval or unskilled mastery. Furthermore, we add general domain data to curriculum learning, using the mixed fine-tuning method, to improve generalization and robustness of translation. Extensive experimental results and analysis show that our method outperforms other curriculum learning baselines across three specific domains.

**Keywords:** Neural Machine Translation, Domain Adaptation, Review-based Curriculum Learning.

## 1 Introduction

Recently, constructing high-quality domain-specific neural machine translation (NMT) models has become a research hotspot. Due to the scarcity of domain-specific parallel corpora, it is currently impossible to train robust domain-specific NMT models from scratch. Domain adaptation uses general domain data and unlabeled-domain data to improve the translation of in-domain models. It focuses on two problems, catastrophic forgetting and overfitting [1]. Common NMT domain adaptation methods can be divided into two categories [2]: data-centric methods, including back translation and data selection; model-centric methods, including training objective-centric methods, architecture-centric methods and decoding-centric methods. These methods can alleviate the catastrophic forgetting and overfitting problems to varying degrees.

Curriculum learning (CL) is also used to solve the above problems. It imitates the way that humans learn curriculum from easier to harder [3], which results in better generalization of the NMT model. Two main questions of CL are how to rank the training examples, and how to modify the sampling procedure based on this ranking [4]. The above questions can be abstracted to difficulty measurer and training scheduler [5]. Usually, difficulty measurers are task-specific, however, the existing prede-

fined training schedulers are data/task agnostic. Training schedulers can be divided into discrete and continuous schedulers, and we focus on the improvement of the discrete schedulers in this paper. One-Pass [3] and Baby Step [6] are two discrete schedulers, which divide the sorted data into shards from easy to hard and then start training with the easiest shard. The difference between two methods is that at each learning phase, One-Pass only uses the current shard but Baby Step merges previously used shards into the current shard. One-Pass may suffer from the problem of catastrophic forgetting, while Baby Step has more generalization but takes longer to train when the number of shards increases.

From practical experience, humans usually review the previous curriculums when they learn. One-Pass can be compared to not reviewing the curriculums they have learned before, and Baby Step is analogous to reviewing all the previous curriculums at each phase. However, it is enough for humans to strengthen their memory by reviewing only some of the previous curriculums at each learning phase. In this paper, we imitate the way humans review curriculums, and propose this review-based CL method. Aiming at the problems of the existing discrete scheduler method, we design two review methods which select the previous curriculums that need to be reviewed and add them to the current training set. The first method calculates time interval of the previous curriculums between their last learning phase and the current phase, and selects curriculums with a larger time interval. The second method is based on the model's mastery of the previous curriculums, calculating the increment of curriculum scores between two close phases to select curriculums which are not proficiently mastered. Fig. 1 shows the difference among the curriculum shards used at each phase for One-Pass, Baby Step and Review. The columns represent the curriculum shards and the rows represent the curriculum shards used at each learning phase. The darker color of each square, the less similar it is to the specific domain.

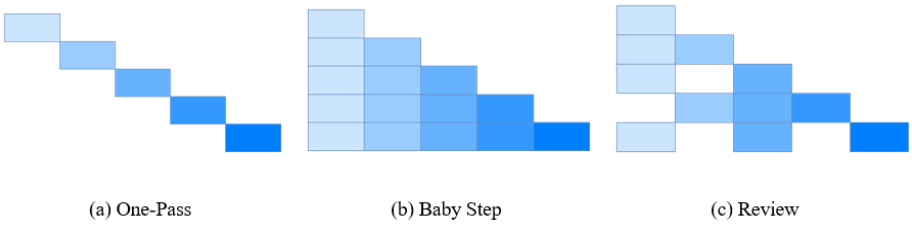


**Fig. 1.** Comparison of shards used at each phase for different curriculum learning methods

With applying to NMT domain adaptation, the above methods still have the problem of forgetting. So we refer to the practice of mixed fine-tuning [7], bringing general domain data into each phase of CL after training the general model. The general domain can be seen as the learning foundation that humans already have when learning curriculums. Although it is not completely consistent with the distribution of specific domains, the knowledge contained in general domain can help NMT model learn common information, enhance the robustness and avoid forgetting happens.

We test our approach on TED talks for German-English and Chinese-English pairs and patent abstracts for German-English pairs. Experimental results show that our approach significantly improves compared to baseline methods, and alleviates the problem of occupying too long training time for Baby Step as well.

## 2 Related Work

From a data-driven perspective, CL is essentially similar to the instance weighting approach in domain adaptation. It makes NMT model pay more attention to the loss of certain training examples, and allows the model to adapt or forget certain pairs. Zhang et al. [8] design different difficulty measurers and training schedulers applying to NMT, and point out that no strategy can perfectly outperform the others, but they did not further analyze the effect of other hyperparameters in CL. Zhang et al. [9] use Baby Step method in NMT domain adaptation for the first time. They take in-domain data as the first curriculum shard, and analyze the effect of two distinct data selection methods and distinct number of shards on NMT model. However, they did not consider the negative impact of slower convergence speed and the problem of forgetting due to fine-tuning with in-domain data and unlabeled-domain data. Xu et al. [10] proposed a dynamic CL method, using training loss decline of two iterations as difficulty measurer and a function of BLEU value on the development set as training scheduler. This method achieves better performance in low-resource scenarios but no improvement when in-domain data is rich.

From a model-driven perspective, CL is also related to training objective-centric methods. Fine-tuning [11] is a classical method which first trains a general domain model and then uses in-domain data to fine-tune it. The fine-tuned model has the problems of catastrophic forgetting and overfitting, so it is difficult to obtain a NMT model with high robustness only by fine-tuning with in-domain data. Thompson et al. [12] use Elastic Weight Consolidation (EWC) method for NMT domain adaptation, reducing the weight of nodes that have too much influence on the general domain to achieve the effect of continuous learning. This method avoids catastrophic forgetting to a certain extent. Chu et al. [7] propose mixed fine-tuning. After training the general NMT model, it uses data mixed with in-domain data and general domain data rather than in-domain data alone, which greatly improves the robustness of the model. We borrow the idea of mixed fine-tuning to add general domain data to CL for solving the problem of catastrophic forgetting.

## 3 Review-based Curriculum Learning

In this paper, we propose review-based curriculum learning for NMT. It focuses on the improvement of discrete training scheduler. We define the number of review curriculums at each phase and how to choose the review curriculum. Also, we introduce general domain data to each phase to solve the forgetting problem for NMT domain adaptation. The overall method is shown in Fig. 2. The solid line pointed out from the curriculum shard represents that it is used at this phase, while the dotted line indicates

that some of the curriculum shards need to be reviewed. These two parts are combined into a review subset for each phase, and then further mixed with the general domain data and in-domain data in a certain proportion to form the whole training set of each phase. At each phase, we continue training the NMT model until it is converged.

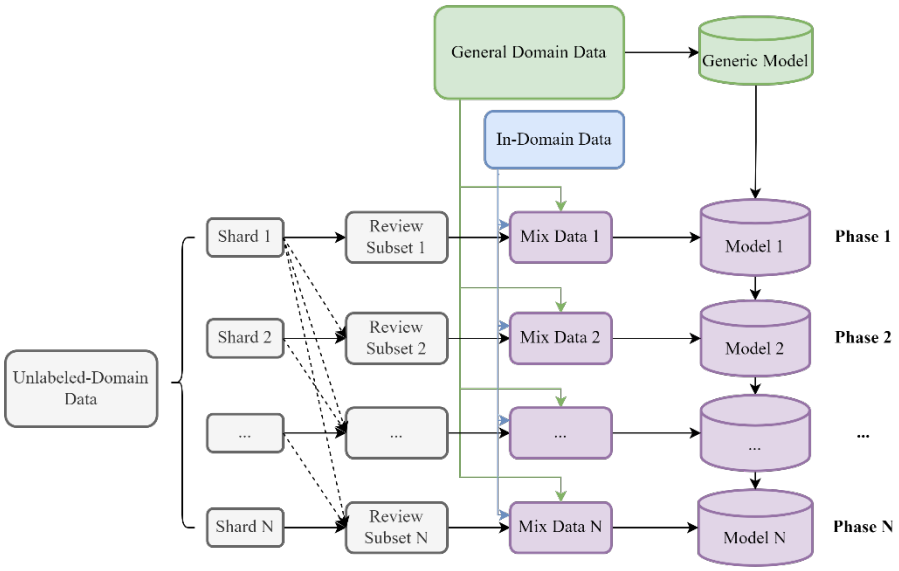


**Fig. 2.** Review-based curriculum learning method enhanced with general domain

### 3.1 Time-Based Review Method

Commonly, humans usually forget curriculums which are learned a long time ago. Inspired by this phenomenon, we believe that the longer a curriculum has been since it was last learned, the more important it is to review it. We assume that the number of CL phases is $T$, and the fixed data shard used at each phase is $C_i (1 \leq i \leq T)$. When reaching a certain phase $i$, in addition to current shard $C_i$, the number of other curriculums to be reviewed is set to $n_i$. Apparently, the range of $n_i$ is between 1 and $i-1$. As Algorithm 1 shows, when reviewing curriculums at phase $i$, we calculate the difference value $\Delta t_{C_i}$ between last used phase $t_{C_j}$ and current phase $i$ for curriculum $C_j$ $(1 \leq j \leq i-1)$. Then we sort the difference values to choose the top $n_i$ curriculum shards from largest to smallest and add them to phase $i$ data $C_i^{re}$. Finally, the last used phase of the top $n_i$ curriculum shards is updated to $i$. It is worth noting that phase 1 does not need to review, so $C_1^{re} = C_1$.

**Algorithm 1** Time-based Review Method

**Input:** Number of curriculum phase $T$, each curriculum shard data $C_i(1 \le i \le T)$.

**Output:** Each curriculum phase data $C_i^{re}(1 \le i \le T)$.

$C_1^{re} \leftarrow C_1$
**for** $i = 2, 3, \dots, T$ **do**
    **for** $j = 1, 2, \dots, i-1$ **do**
        $\Delta t_{C_j} \leftarrow i - t_{C_j}$
    **end for**
    Sort $\Delta t_{C_j}(1 \le j \le i-1)$ from largest to smallest, choose the top $n_i$ curriculum shards $C_{r_1}, C_{r_2}, \dots, C_{r_{n_i}}$.
    $C_i^{re} \leftarrow C_i + C_{r_1} + C_{r_2} + \cdots + C_{r_{n_i}}$
    **for** $k = 1, 2, \dots, n_i$ **do**
        $t_{C_{r_k}} \leftarrow i$
    **end for**
    $t_{C_i} \leftarrow i$
**end for**

### 3.2 Master-Based Review Method

From a different point of view, humans also review the curriculums which are not proficiently mastered. We change this thought into an achievable method. As algorithm 2 shows, first we define the model's mastery of the previous curriculum shards as the BLEU value on them. Considering if we translate all the sentences in the shards, it will cost a long translation time, so we take 1000 sentence pairs from each curriculum shard at equal spacing as a representation of the shard and calculate the BLEU value. We think that compared to the last phase, the less curriculum shard improves, the more it needs to be reviewed. The master score is estimated as:

$$score_{C_j} = \frac{BLEU_{C_j}^{i} - BLEU_{C_j}^{i-1}}{BLEU_{C_j}^{i-1}} \tag{1}$$

where $BLEU_{C_j}^{i}$ represents that the BLEU value of 1000 pairs from curriculum shard $C_j$ $(1 \le j \le i-1)$ at phase $i$ before training. If the score is smaller than others, we think that the NMT model has not learned this shard suffciently, and conversely we consider this shard has improved more and does not need more attention. We select top $n_i$ shards according to the master score from smallest to largest, and add them to phase $i$ data $C_i^{re}$. Finally we calculate $BLEU_{C_i}^{i}$ and train new NMT model.

**Algorithm 2** Master-based Review Method

**Input:** Number of curriculum phase $T$, each curriculum shard data $C_i(1 \le i \le T)$.

**Output:** Each curriculum phase data $C_i^{re}(1 \le i \le T)$.

1: $C_1^{re} \leftarrow C_1$, calculate $BLEU_{C_1}^1$

2: **for** $i = 2, 3, \dots, T$ **do**

3: **for** $j = 1, 2, \dots, i-1$ **do**

4: Use current model to calculate $BLEU_{C_j}^i$.

5: Calculate master score of $C_j$ by Equation 1.

6: **end for**

7: Sort $score_{C_j}(1 \le j < i)$ from smallest to largest, choose top $n_i$ curriculum shards $C_{r_1}, C_{r_2}, \dots, C_{r_{n_i}}$.

8: $C_i^{re} \leftarrow C_i + C_{r_1} + C_{r_2} + \cdots + C_{r_{n_i}}$

9: Use current model to calculate $BLEU_{C_i}^i$.

10: Train the new NMT model.

11: **end for**

### 3.3 General Domain Enhanced Training

General domain can be seen as the inherent memory of humans, so in order to maintain a high level of generalization and robustness of NMT model, we add general domain data to each learning phase. In the experiments of Zhang [9], as training goes on, the weight of in-domain data is decrease due to the increment of unlabeled-domain data. Therefore, we assign weight to in-domain data individually, so that each phase uses a fixed proportion of general domain data, in-domain data and partially unlabeled-domain data:

$$D_{train_t} = w_{GD} * D_{GD} + w_{ID} * D_{ID} + w_{UD} * C_t^{re} \quad (2)$$

where $D_{train_t}$ represents training set at phase $t$, $w_{GD}$, $w_{ID}$ and $w_{UD}$ represent the weight of general domain data $D_{GD}$, in-domain data $D_{ID}$ and review unlabeled-domain data $C_t^{re}$ separately.

## 4 Experiment

### 4.1 Data and Setup

**General Domain Data.** We use two general domain datasets in the experiment, German(de)-English(en) and Chinese(zh)-English. German-English general dataset includes Europarl, news commentary, OpenSubtitles and Rapid corpus, while Chinese-English includes CCMT2017, news commentary, UN Parallel Corpus. After tokeniza-

tion (not to Chinese) and filtering sentence length up to 80 words, we get 19 million sentence pairs for German-English and 20 million sentence pairs for Chinese-English.

**In-domain Data.** Chinese-English and German-English TED domain data are from Duh [13], and German-English patent domain data is from Junczys-Dowmunt et al. [14]. The concrete number of three domain corpora is shown in Table 1.

**Table 1.** Number of sentences in each dataset

| Dataset | Training Set | Development Set | Test Set |
|---|---|---|---|
| TED (zh-en) | 166373 | 1958 | 1982 |
| TED (de-en) | 148460 | 1958 | 1982 |
| Patent (de-en) | 150000 | 2000 | 2000 |

**Unlabeled-domain Data.** For unlabeled-domain data in two language directions, we use web-crawled bitext from the Paracrawl project [15]. After data cleaning and data selection, we get 20 million sentences for German-English and 8.3 million sentences for Chinese-English. For the final corpus size in the experiment, Zhang et al. [9] suggest 1024k pairs, and we follow this setup.

**Curriculum Learning Setup.** We refer to Zhang et al. [9] for some experiment settings. For difficulty measurer we use Moore-Lewis [16] method to build language models trained on in-domain and unlabeled-domain, and calculate the cross-entropy difference of sentence in unlabeled-domain dataset. KenLM [17] is used to build language models on the target side (English). Then, we set $n_i = \lfloor \log_2 i \rfloor$. This setting is designed to review an appropriate number of curriculums to avoid forgetting or inefficient learning problem of not reviewing (like One-Pass) or reviewing all shards (like Baby Step). Finally, we set the number of curriculum phase to 5, which is different to Zhang et al. [9]. It is explained in experiment analysis.

**Subword model.** We use general domain data to train sentencepiece [18] subword segmentation model. The vocab size is set to 32000 both for two languages. Since general domain is large enough to train a robust segmentation model, there is no need to retrain the subword model when we use the in-domain data and unlabeled-domain data.

**NMT Setup.** In all experiments, we use the OpenNMT [19] implementation of the Transformer [20], with 6 layers for both encoder and decoder and 8 attention heads. The word embedding size is set to 512. We use Adam [21] optimizer to adjust the learning rate automatically, with $\beta_1$ = 0.9 and $\beta_2$ = 0.998. We set batch size to 6000, and training stops when the perplexity on the development set has not improved for 5 checkpoints (2000 batches per checkpoint) at each phase. In addition, considering that

the number of general domain data is much larger than the number of in-domain and unlabeled-domain data, we set the weights ($w_{GD}$: $w_{ID}$: $w_{UD}$) to 10:1:1. This is done to oversample in-domain data and maintain high learning ratio on the other two domains, which not only biases the final model distribution towards the specific domain, but also improves the robustness of the NMT model.

**Evaluation Metric.** We use BLEU as the evaluation metric, and calculate with sacreBLEU tool [22].

### 4.2 Main Results

Main experimental results is shown in Table 2. The model trained with large amount of general domain data (GEN) has BLEU scores of 35.98, 18.29 and 26.47. Fine tuning (FT) on in-domain data improves BLEU significantly by 3.25, 3.51 and 23.98. Mixed fine tuning (MFT) brings more robustness to NMT model, with improvement of 2.32, 2.17 and 0.94 BLEU score compared to fine tuning method.

For previous curriculum learning methods, One-Pass suffers from catastrophic forgetting problem apparently, with BLEU scores of 31.09, 15.48 and 34.03. Although Baby Step improves this situation with BLEU scores of 36.97, 22.60 and 50.74, it does not work as well as fine tuning on TED (de-en) domain, and still has the problem of forgetting. Our two methods (T-Review and M-Review) perform better than One-Pass and worse than Baby Step, because the NMT model does not focus on the in-domain data all the time during the training process, and too much attention to the unlabeled-domain data may cause forgetting problem.

After we add general domain data into CL phases, all the CL methods mentioned perform better than original. T-Review+MFT performs best in all the methods with BLEU scores of 42.40, 24.49 and 52.29. Compared to MFT method, it improves BLEU by up to 0.9 score on patent (de-en). Also, compared to Baby Step method, it improves BLEU by up to 5.43 score on TED (de-en). We believe that general domain data enhances the generalization of the NMT model, so that instead of reviewing all the previous curriculum shards, we use only a part of shards that are necessary to be reviewed to improve the effect of NMT model.

As for the comparison of our two methods, T-Review+MFT performs slightly better than M-Review+MFT. Note that T-Review is not related to the NMT model while M-Review is related. The possible reason is that T-Review has a more logical review schedule for the shards and is able to review the curriculum evenly. We also compare the method of randomly selecting shards for review with MFT (Rand-Review+MFT). The result shows that even randomly select curriculums can be better than Baby Step+MFT and One-Pass+MFT, however, designed review curriculum rules are more effective such as T-Review and M-Review.

**Table 2.** Main experiment results

| Method | TED (de-en) | TED (zh-en) | patent (de-en) |
|---|---|---|---|
| GEN | 35.98 | 18.29 | 26.47 |
| FT | 39.23 | 21.80 | 50.45 |
| MFT | 41.55 | 23.97 | 51.39 |
| One-Pass | 31.09 | 15.48 | 34.03 |
| +MFT | 42.24 | 24.08 | 52.19 |
| Baby Step | 36.97 | 22.60 | 50.74 |
| +MFT | 42.05 | 24.06 | 51.97 |
| Rand-Review+MFT | 42.23 | 24.25 | 52.09 |
| T-Review | 35.35 | 21.05 | 47.67 |
| +MFT | **42.40** | **24.49** | **52.29** |
| M-Review | 35.64 | 21.17 | 47.45 |
| +MFT | 42.37 | 24.20 | 52.24 |

# 5 Analysis

## 5.1 Effect of Mixed Fine Tuning

We analyze the effect of MFT for CL. As shown in Table 3, we conduct the ablation studies on whether CL approach incorporate the general domain and whether in-domain weight is fixed, with Baby Step and T-Review method. We can see that when the in-domain weight is fixed, T-Review outperforms original method by up to 3.29 BLEU score on TED (de-en), but Baby Step has an unstable effect as decreasing on TED (zh-en) and patent (de-en). When mixed with general domain only, T-Review increases by up to 4.19 BLEU score on TED (de-en) compared with original method, and this value is 3.81 for Baby Step. However, due to the reason that in-domain weight is unfixed and the Review method is not stable to review the in-domain shard, the effect of T-Review is worse than Baby Step.

When combining the general domain and fixing the in-domain weight, the robustness of NMT model is greatly improved. Relatively increasing the in-domain weight can learn the in-domain knowledge better with the help of general domain and solve the problem of overfitting. So T-Review+MFT performs better than Baby Step+MFT. It is worth noting that One-Pass+MFT is even more effective than Baby Step+MFT, which further proves that MFT does not require multiple repetitions of curriculms when applied to CL. Only the curriculums which need to be reviewed is enough.

**Table 3.** Ablation study results for general domain and in-domain fixed weight

| Method | TED (de-en) | TED (zh-en) | patent (de-en) |
|---|---|---|---|
| Baby Step | 36.97 | 22.60 | 50.74 |
| +Fixed in-domain weight | 39.00 | 21.93 | 50.66 |
| +General domain | 40.78 | 23.49 | 51.42 |
| +MFT (Fixed in-domain weight+ General domain) | 42.05 | 24.06 | 51.97 |
| T-Review | 35.35 | 21.05 | 47.67 |
| +Fixed in-domain weight | 38.64 | 22.44 | 50.28 |
| +General domain | 39.54 | 22.77 | 49.53 |
| +MFT (Fixed in-domain weight+ General domain) | **42.40** | **24.49** | **52.29** |

### 5.2 Low-Resource Scenerio

We also explored the effects of using a review-based CL with MFT in a low-resource scenario. We set the number of patent (de-en) sentence pairs to 15k rather than 150k, in order to simulate the effect of extremely low-resource domain scenario. Table 4 shows that two Review+MFT methods have an average increment of 2.32, 0.55 and 0.53 BLEU score compared to MFT, One-Pass+MFT and Baby Step+MFT methods. This result indicates the effectiveness of using data from other rich resources to increase model robustness and also confirms that CL+MFT, especially review-based CL+MFT, could improve translation abilities of NMT models and avoid the problem of overfitting and catastrophic forgetting.

**Table 4.** Results in low-resource scenario

| Method | patent (de-en) |
|---|---|
| MFT | 45.35 |
| One-Pass+MFT | 47.12 |
| Baby Step+MFT | 47.14 |
| Rand-Review+MFT | 47.39 |
| T-Review+MFT | **47.68** |
| M-Review+MFT | 47.67 |

### 5.3 Data Sharding

We experiment with different number of shards setting and experiment on TED (de-en) domain with Baby Step+MFT and M-Review+MFT. As Fig. 3 shows, the two methods both achieve the best performance at the point of 5 shards. As the number of shards increases, the BLEU scores show a decreasing trend. Although Baby Step

increases when the number of shards is 20, the BLEU score does not change too much. This result differs from the findings of Zhang et al. [9]. The possible reason is that our method is mixed with general domain and fixes weights of three domains, and increasing number of shards with the same number of unlabeled-domain data will reduce the number of data in each shard. This may result in the curriculum being repeated too many times at one phase, which may lead to overfitting. Further, considering the negative effects of too long training time of too many shards, we set the number of shards to 5 better than the number of other shards and Review is better than Baby Step.

### 5.4 Training Efficiency

Table 5 shows the comparison of training steps for three CL methods. We can see that T-Review+MFT reduces training time by average of 18k steps and M-Review+MFT reduces an average of 12k steps both compared to Baby Step+MFT. It proves that review-based method with MFT can accelerate the convergence of NMT model. We

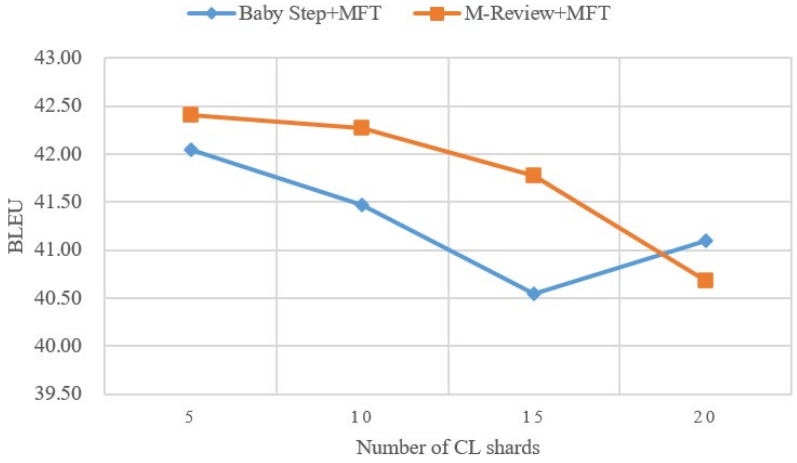


**Fig. 3.** Different number of curriculum learning shards

argue that the number of curriculums learned at each phase has an impact on the convergence speed. More curriculums make the model less easy to converge, however, reviewing appropriate number of courses reduces training time and improves training efficiency.

**Table 5.** Training steps for three curriculum methods

| Method | TED (de-en) | TED (zh-en) | patent (de-en) |
|---|---|---|---|
| Baby Step+MFT | 92k | 168k | 144k |
| T-Review+MFT | **84k** | **146k** | **118k** |
| M-Review+MFT | 86k | 150k | 132k |

## 6 Conclusion

To address the problems of catastrophic forgetting and learning inefficiency of previous curriculum learning methods for NMT domain adaptation, this paper proposes a review-based curriculum learning method. We first select curriculum shards with long time interval or unskilled mastery to review in each learning phase, and add general domain data to improve the robustness of NMT model. The experimental results show that our method improves significantly compared to previous curriculum learning methods and the simulation of low-resource scenario also demonstrate the effectiveness.

For future work, we will explore more effective methods and more applications for review-based curriculum learning. Additionally, it is a meaningful job for adding dynamic weighting method to our approach.

## References

1. Saunders, D.: Domain adaptation and multi-domain adaptation for neural machine translation: A survey. arXiv preprint arXiv:2104.06951 (2021).
2. Chu, C., Wang, R.: A Survey of Domain Adaptation for Neural Machine Translation. In: Proceedings of the 27th International Conference on Computational Linguistics, pp. 1304-1319 (2018).
3. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: Proceedings of the 26th annual international conference on machine learning, pp. 41-48 (2009).
4. Weinshall, D., Cohen, G., Amir, D.: Curriculum learning by transfer learning: Theory and experiments with deep networks. In: International Conference on Machine Learning, pp. 5238-5246. PMLR (2018).
5. Wang, X., Chen, Y., Zhu, W.: A survey on curriculum learning. IEEE Transactions on Pattern Analysis and Machine Intelligence(01), 1-1 (2021).
6. Spitkovsky, V. I., Alshawi, H., Jurafsky, D.: From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 751-759 (2010).
7. Chu, C., Dabre, R., Kurohashi, S.: 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), pp. 385-391 (2017).
8. Zhang, X., Kumar, G., Khayrallah, H., Murray, K., Gwinnup, J., Martindale, M. J., ..., Carpuat, M.: An empirical exploration of curriculum learning for neural machine translation. arXiv preprint arXiv:1811.00739 (2018).
9. Zhang, X., Shapiro, P., Kumar, G., McNamee, P., Carpuat, M., Duh, K.: Curriculum Learning for Domain Adaptation in Neural Machine Translation. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 1903-1915 (2019).
10. Xu, C., Hu, B., Jiang, Y., Feng, K., Wang, Z., Huang, S., ..., Zhu, J.: Dynamic Curriculum Learning for Low-Resource Neural Machine Translation. In: Proceedings of the 28th International Conference on Computational Linguistics, pp. 3977-3989 (2020).

11. Luong, M. T., Manning, C. D.: Stanford neural machine translation systems for spoken language domains. In: Proceedings of the 12th International Workshop on Spoken Language Translation: Evaluation Campaign (2015).
12. Thompson, B., Gwinnup, J., Khayrallah, H., Duh, K., Koehn, P.: 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), pp. 2062-2068 (2019).
13. The multitarget ted talks task, http://www.cs.jhu.edu/~kevinduh/a/multitarget-tedtalks/, last accessed 2018/12.
14. Junczys-Dowmunt, M., Pouliquen, B., Mazenc, C.: Coppa v2. 0: Corpus of parallel patent applications building large parallel corpora with gnu make. In: 4th Workshop on Challenges in the Management of Large Corpora Workshop Programme (2016).
15. Bañón, M., Chen, P., Haddow, B., Heafield, K., Hoang, H., Esplà-Gomis, M., ..., Zaragoza, J.: ParaCrawl: Web-scale acquisition of parallel corpora. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 4555-4567 (2020).
16. Moore, R. C., Lewis, W.: Intelligent selection of language model training data. In: Proceedings of the ACL 2010 conference short papers, pp. 220-224 (2010).
17. Heafield, K.: KenLM: Faster and smaller language model queries. In: Proceedings of the sixth workshop on statistical machine translation, pp. 187-197 (2011).
18. Kudo, T., Richardson, J.: 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, pp. 66-71 (2018).
19. Klein, G., Hernandez, F., Nguyen, V., Senellart, J.: The OpenNMT neural machine translation toolkit: 2020 edition. In: Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track), pp. 102-109 (2020).
20. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ..., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems, 30 (2017).
21. KingaD, A.: A Method For Stochastic Optimization. Anon. International Conference on Learning Representations. SanDego: ICLR (2015).
22. Post, M.: A Call for Clarity in Reporting BLEU Scores. In: Proceedings of the Third Conference on Machine Translation: Research Papers, pp. 186-191 (2018).