Low-resource Neural Machine Translation based on Improved Reptile Meta-Learning Method

Nier Wu[1], Hongxu Hou[2*], Xiaoning Jia[3], Xin Chang[4], and Haoran Li[5]

College of Computer Science-college of Software, Inner Mongolia University, China
[1]wunier04@126.com, [2*]cshhx@imu.edu.cn, [3]jiaxning@163.com
[4]changxin03@163.com [5]489633848@qq.com

Abstract. Multilingual transfer learning has been proved an effective method to solve the problem of low-resource neural machine translation (NMT). However, the global optimal parameters obtained through transfer learning can not effectively adapt to new tasks, which means the problem of local optimum will be caused when training the new task model. Although this problem can be alleviated by optimization-based meta-learning methods, but meta-parameters are determined by the second-order gradient term corresponding to the model parameters of a specific task, which consumes a lot of computing resources. Therefore, we proposed improved reptile meta-learning method. First, a multilingual unified word embedding method is proposed to represent multilingual knowledge. Secondly, the direction of meta-gradient is guided by calculating cumulative gradients on multiple specific tasks. In addition, the midpoint is taken as the meta-parameter in the space of the initial meta-parameter and the final task-specific model parameter to ensure that the meta-model has better multi-feature generalization ability. We conducted experiments in the CCMT2019 Mongolian-Chinese (Mo-Zh), Uyghur-Chinese (Uy-Zh) and Tibetan-Chinese (Ti-Zh), and the results show that our method has significantly improved the translation quality compared with the traditional methods.

Keywords: Meta-learning · Low-resource · Machine translation.

1 Introduction

Low-resource NMT model is easy to produce over-fitting during model training due to the sparse data. In order to solve the problem of insufficient training sets for low-resource machine translation, there are two common methods: one is unsupervised learning[1], which uses large-scale monolingual corpus as an aid, expands pseudo-corpus by back translation or denoising self-encoding, and trains the model through self-learning or adversarial learning. However, the common pseudo-corpus noise reduction methods (deletion, replacement, addition) and shared word embedding mapping methods cannot fundamentally improve the

* Corresponding Author

noise and word alignment problem, so the current unsupervised machine translation translation effect is still lower than that of supervised model. Another method is transfer learning[2], which applies the model parameters learned from the high-resource language pair to the translation model of the low-resource language, and adapts the model to the low-resource task via fine-tuning. It mainly uses the prior knowledge of the high-resource language to assist the generation of the low-resource translation model[3].

Meta-learning is similar to transfer learning, which is essentially learning to learn. The meta-learning method is a model-independent method, which has better generalization ability and can quickly adapt to new tasks through a few training examples. Recently, there are mainly two methods for machine translation research using meta-learning: optimization-based method and model-based method.[4] proposed an optimization-based machine translation method for low-resource domains. Meta-parameters are iteratively learned through the proposed training strategies on translation tasks in different domains to adapt to translation tasks in new low-resource domains.[5] proposed an optimization-based meta-learning neural machine translation model training method. They used model-agnostic meta-learning (MAML) algorithm[6] to obtain shared initial parameters in multilingual large-scale language pairs, and the model can realize rapid convergence on low-resource translation tasks using initialized meta-parameters. Meanwhile, in order to solve the problem of inconsistency in word embedding space in multilingual translation tasks, the above studies all adopt a similar general word representation method[7] to adapt it to various meta-learning episodic.

Although the optimization-based meta-learning method shows potential in low-resource translation tasks, in the model training stage, the second-order gradient corresponding to the model parameters of a specific task will be repeatedly calculated, while consumes too much computing resources, and the performance of multi-task fitting is not ideal. Therefore, in order to avoid the above problems, we proposed an improved reptile meta-learning method. Specifically, it includes the following aspects.

- We proposed an unified word embedding representation method, which maps multiple languages including the target language into a new word embedding space instead of mapping to the word embedding space of the target language. This method improves the alignment accuracy between arbitrary languages without passing through the "pivot" language.
- We proposed an improved reptile meta-learning method, which can replace the original second-order gradient term to guide the direction of the meta-gradient, so that it has better multilingual knowledge transfer ability, and improves generalization performance while saving computing resources.

2 Background

Neural Machine Translation Given the source language X, the neural machine translation model encodes X into a set of continuous intermediate representations, and the decoder decodes the target language Y from left-to-right

according to the set of intermediate representations, as shown in Equation 1.

$$p(Y|X;\theta) = \prod_{t=1}^{T+1} p(y_t|y_{0:t-1}, x_{1:T'};\theta) \tag{1}$$

In general, recurrent neural network (RNN) is used to build the model. Recently, a decoder model with self-attention model and convolution structure has been proposed. Compared with the traditional model based on RNN method, the structure shows remarkable performance.

Low-Resource Machine Translation Generally, unsupervised methods mainly include back translation[8] and dual learning[9]. While knowledge sharing methods mainly include transfer learning[10] and multi-task learning.

Meta-learning based NMT mainly draws lessons from MAML method, which includes two steps: meta-training and meta-testing. For meta-training, given a set of high-resource meta-translation tasks $(T1, ..., Tk)$, a set of tasks are sampled from the translation task generator each step, and the parameters are updated by MAML method to obtain the corresponding prior knowledge. For meta-testing, the low-resource translation model is initialized by using the learned parameters, so that the low-resource machine translation model can use prior knowledge and train a new translation model with a few number of samples. The learning process is shown in Equation 2.

$$\theta^* = Learn(T^0; MetaLearn(T^1, ..., T^K)) \tag{2}$$

For a specific low-resource language learning task T^0, the initial parameters are obtained from meta-model. It is assumed that the prior parameter distribution of the expected model satisfies isotropic gaussian distribution $N(\theta_i^0, 1/\beta)$. Meanwhile, to prevent the updated parameters from being far away from meta-parameters, the learning process of a specific language can be understood as maximizing logarithmic posteriori of model parameters for a given data set D_T, as shown in Equation 3.

$$Learn(D_T, \theta^0) = \underset{\theta}{argmax} \sum_{(X,Y)\in D_T} logp(Y|X;\theta) - \beta \left\|\theta - \theta^0\right\|^2 \tag{3}$$

Where X and Y represents the source language and target language of the data set, β is model parameter, $\left\|\theta - \theta^0\right\|^2$ indicate modulo. In order to use high-resource language to repeatedly simulate low-resource translation episodic to obtain initialization parameters, the loss function of meta-learning is defined as Equation 4.

$$Loss(\theta) = E_k E_{D_{T^k}, D'_{T^k}} \left[\sum_{(X,Y)\in D_{T^k}} logp(Y|X; Learn(D_{T^k}, \theta)) \right] \tag{4}$$

3 Our approach

We proposed a unified word embedding representation method, and an improved reptile meta-learning NMT method. As shown in Figure 2.

3.1 Unified word embedding representation

The vocabulary of each language only subject to an independent distribution space. To integrate multilingual knowledge, it is necessary to make universal representation of words in different languages. The common method is to map grammatically and semantically equivalent words from different languages to the same position in the vector space of the target language. Therefore, the mapping between other languages can be realized through the target language as a "pivot".

General methods such as cross-domain similarity local scaling (CSLS) optimize this mapping by minimizing the difference of word embedding of the same word in different languages. The optimal mapping matrix Q is constructed based on the loss of two norms, such as Equation 5.

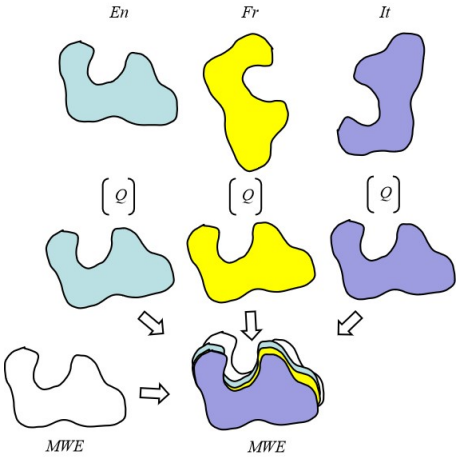

Fig. 1. Multi-aligned multilingual word embedding representation (MWE).

$$\min_{Q \in R^{d \times d}} \|XQ - PY\|_2^2 \tag{5}$$

Where X is the mapped language, Y is the target language, and P represents the allocation matrix. However, this method needs bilingual dictionaries to assist and can only embed words in two languages. If multi-language embedding is done, only one language needs to be used as the transmission language. When there is no bilingual dictionary, $Wasserstein - Procrustes$ constraint is used on the allocation matrix P, so that the sum of each row and column of the allocation matrix is 1, and the matrix elements represent the degree of association of different words. Therefore, the allocation matrix P and the mapping matrix Q are optimized, and the 2-norm objective function embedded in multilingual words is obtained, such as Equation 6.

$$\min_{Q \in Q_d, P \in P_n} \sum_i l(X_i Q_i, P_i X_0) \tag{6}$$

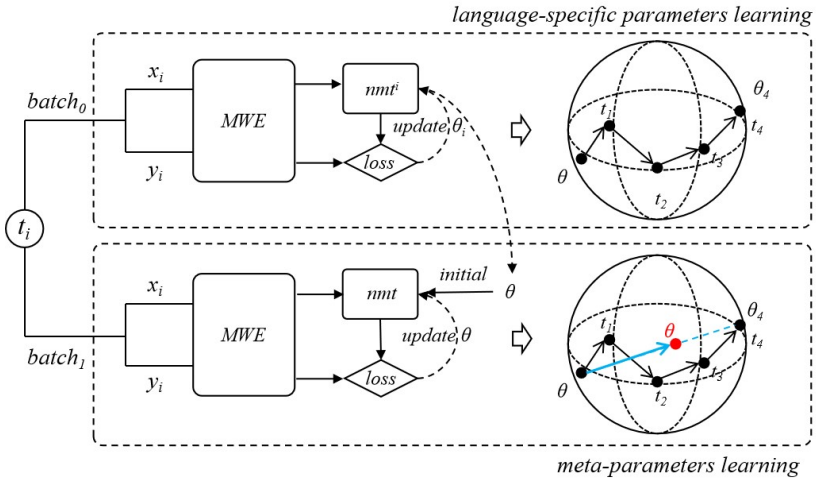

Fig. 2. In batch 0, the parameters of a specific task are learned and used it as initialization parameters for the next task. In batch 1, when the meta parameter is ready to be updated, there are two steps: 1. Utilize the cumulative gradient obtained by K-sampling as the direction of the meta-gradient. 2. The initial meta-parameters advance after K/2 steps and update (about half the distance between the final task-specific model parameter and the meta-parameters).

Among them, X_i, Q_i and P_i respectively represent the word embedding, mapping matrix and allocation matrix of the language mapping to the transfer language X_0. From Equation 6, it can be seen that multilingual word embedding does not get the mapping between any two languages, but the mapping between one language and the target language (transfer language), which cannot guarantee the quality of word embedding except the target language. We use different cross-lingual word embedding methods to observed the quality of the translation. Therefore, we propose a new general vocabulary representation method: multi-aligned multilingual word embedding representation (MWE) As shown in Figure 1. This is specifically shown in Equation 7.

$$\min_{Q \in Q_d, P_{ij} \in P_n} \sum_{i,j} \alpha_{ij} l(X_i Q_i, P_{ij} X_j Q_j) \tag{7}$$

α represents weights, and i and j represent the number of languages. We take advantage of the fact that all word embedding maps to a unified space to realize better alignment.

3.2 NMT method based on improved reptile meta-learning

Parameters of task-specific model Task-specific learning is similar to transfer learning. It mainly learns the model parameters of a specific tasks θ_i from

high-resource translation tasks. Assume that the model corresponding to the i-th task is nmt_θ^i, and model parameter θ is represented. Given the current task t_i and the corresponding data set $(D_{train}^{(i)}, D_{test}^{(i)})$, then the model parameters are updated by using the stochastic gradient descent method (SGD), as shown in Equation 8.

$$Learn(D_{train}^{(i)}; \theta^{'}) = \theta - \alpha \bigtriangledown_\theta Loss_{t_i}^{(0)}(nmt_\theta^i) \tag{8}$$

α represents the learning rate, $Loss_{t_i}^{(0)}$ is the loss calculated from batch data numbered 0 in the task t_i, and is usually expressed by the maximum likelihood estimation (MLE).

Meta-parameter To be able to better extend to a series of tasks. That is, to find the most efficient parameter θ^* in the fine-tuning process after any given task, we need to re-sample a batch of data to update meta-parameters. If the corresponding loss function is set to $Loss_{t_i}^{(1)}$, the most efficient parameter θ^* and meta-parameter θ of the fine-tuning process expressed as shown in Equation 9 and 10.

Algorithm
Require: $p(\tau)$: Distribution over tasks
Require: α,K:step hyper-parameters
Initialisation: Random θ
for $i = 1, 2, ..., n$ **do**
sample tasks $\tau_i \sim p(\tau)$
for all τ_i **do**
Evaluate the update $\theta_i = \theta - \alpha \bigtriangledown_\theta Loss_{\tau_i}(\theta)$ k times
end for
update:$\theta = \theta + \frac{2\alpha}{K} \sum_i^n (\theta_i - \theta)$
end for

Table 1. Improved reptile meta-parameter update algorithm.

$$\theta^* = \underset{\theta}{argmin} \sum_{t_i \sim p(t)} Loss_{t_i}^{(1)}(nmt_{\theta'}^i) \tag{9}$$

$$\theta \leftarrow \theta - \beta \bigtriangledown_\theta \sum_{t_i \sim p(t)} Loss_{t_i}^{(1)}(nmt_{\theta - \alpha \bigtriangledown_\theta Loss_{t_i}^{(0)}(nmt_\theta^i)}) \tag{10}$$

According to equation 10, after specific task parameters are learned in the inner loop, new data will be sampled from the same data set in the outer loop, and calculate a new gradient based on the same loss function, and meta parameters will be updated according to the new gradients. When sampling a new task, the initial meta-parameters are updated to the meta-parameters of the previous iteration, and the meta parameters are iteratively updated by repeatedly executing the previous steps. We proposed an improved reptile meta-learning method, when calculating the gradients of K tasks, we will no longer divide them into K different parameters θ_i^*, as shown in Table 1.

As shown in Table 1, given the initial parameter θ, k-round stochastic gradient descent of $SGD(Loss(nmt_\theta^i), \theta, k)$ is carried out according to $Loss(nmt_\theta^i)$,

and then the parameter vector is returned. The version with batch samples multiple tasks at a single step. The gradient of our method is defined as $(\theta - W)/s$, where s is the step size used by SGD.

4 Experiments

4.1 Datasets

We use transfer learning based NMT (TF-NMT) and MetaNMT[1] as baselines and use 5 European (English (En), French (Fr), German (De), Spanish (Es), Italian (It)) and 3 Asian languages (Korean (Ko), Vietnamese (Vt), Japanese (Ja)) for the meta-training. All European language datasets from Europarl[2], however, all European languages use English as the target language instead of Chinese, so we adopt pivot-based method to construct an NMT model based on pivotal language (English) to obtain parallel sentence pairs from European languages to Chinese. Korean (Ko) corpus obtained from Korean Parallel Dataset,[3]. For Vietnamese (Vt), we use a crawler collect Vietnamese texts[4] from the Internet and then feed the Google translator[5] with the text, so that we obtain a loose parallel corpus between Vietnamese and Chinese. For Japanese (Ja), We conducted experiments with the ASPEC-JC corpus, which was constructed by manually translating Japanese scientific papers into Chinese. During meta-test period, we selected the following three different languages pairs (Mongolian-Chinese (Mo-Zh), Tibetan-Chinese (Ti-Zh), Uyghur-Chinese (Ug-Zh)) from CCMT2019: We use the officially provided train sets, valid sets, test sets for these languages. The size of the training sample is shown in Table 2.

Corpus	sents.	src-tokens	trg-tokens
En-Zh	1.93M	3.22M	33.61
Fr-Zh	2.77M	51.39M	50.2M
De-Zh	1.92M	44.55M	47.81M
Es-Zh	1.96M	51.58M	49.09M
It-Zh	1.91M	47.4M	49.67M
Ko-Zh	0.54M	10.82M	11.11M
Vt-Zh	0.8M	15.95M	16.3M
Ja-Zh	0.68M	10.17M	11.23M
Mo-Zh	0.26M	8.85M	9.39M
Ti-Zh	0.4M	9.12M	8.68M
Ug-Zh	0.46M	10.12M	11.29M

Table 2. The size of the training sample during meta-training and meta-test.

4.2 Setting and Baseline

[1] https://github.com/salesforce/nonauto-nmt
[2] http://www.statmt.org/europarl
[3] https://sites.google.com/site/koreanparalleldata
[4] The Vietnamese corpus has 0.8 million Vietnamese sentences and 10 million Vietnamese monosyllables.
[5] https://translate.google.cn/?sl=vi&tl=zh-CN&op=translate

Setting Our model is implemented using Pytorch[6], a flexible framework for neural networks. We base our model on the Transformer model and the released Pytorch implementation[7]. Parameters are set as follows: word embedding size = 300, hidden size = 512, number of layers=4, number of heads=6, dropout=0.25, batch size=128, and beam size=5. Because our semantic space is obtained by multiple alignment of different languages. Therefore, we need to vectorize multilingual, here, two kinds of vectorization representation strategies are proposed: Static representation and dynamic representation. For static cross-lingual word embedding, we first employed FastText tools[8] to generate static monolingual word vector, and then use MUSE[9] or VECMAP[10] to generate cross-lingual representation. For dynamic cross-lingual word embedding, we obtained contextual dynamic word embedding through the ElMo model[11], and then use our multiple alignment approach[12] to get dynamic cross-lingual word embedding. In test phase, we use beam search to find the best translated sentences. Decoding ends when every beam gives an $\langle EOS \rangle$.

Model	**Mo-Zh**	**Ug-Zh**	**Ti-Zh**
Transformer	28.15	23.42	24.35
TF-NMT	28.58	24.39	25.27
Meta-NMT	29.95	25.52	26.73
IR-Meta-NMT	30.83	26.29	27.18

Table 3. Comparison of experimental results. Our model shows potential advantages in three different target tasks in a fully supervised environment.

Baseline We compared our approach against various baselines:

- Transformer[13]: The mainstream machine translation framework at this stage.
- TF-NMT[14]: A common method based on parameters transfer.
- Meta-NMT[15]: The method proposed by [5].
- IR-Meta-NMT: A model that improved reptile meta-learning methods that we proposed.

4.3 Result and Analysis

To observe the results, we give different experimental choices: First is to compare our model with a variety of experiments, mainly to observe the performance of our method. Second is the influence of different meta-learning datasets on the translation quality of the target tasks. As shown in Table 3 and 4. According

[6] https://pytorch.org/
[7] https://github.com/pytorch/fairseq
[8] https://github.com/facebookresearch/fastText
[9] https://github.com/facebookresearch/MUSE
[10] https://github.com/artetxem/vecmap
[11] https://github.com/DancingSoul/ELMo
[12] https://github.com/PythonOT/POT
[13] https://github.com/tensorflow/tensor2tensor
[14] https://github.com/ashwanitanwar/nmt-transfer-learning-xlm-r
[15] https://github.com/MultiPath/MetaNMT

Meta-Train	Mo-Zh		Ug-Zh		Ti-Zh	
	none	finetune	none	finetune	none	finetune
Es It	9.98	14.61± .18	3.58	5.61± .18	4.41	4.51± .28
En Fr De	11.76	16.92± .3	4.05	7.25± .24	4.29	5.94± .15
European	14.53	19.08± .12	4.46	8.16± .08	5.17	6.91± .35
Ko	11.39	15.97± .25	6.39	10.38± .14	6.53	8.14± .16
Vt Ja	15.55	21.38± .11	7.11	9.57± .31	6.74	7.89± .15
Asia Languages	18.86	23.15± .29	10.76	11.41± .12	10.76	11.57± .10
All Languages	19.49	24.01± .27	11.12	12.56± .08	12.17	12.96± .19
Full Supervised	**31.76**		**27.1**		**28.35**	

Table 4. Low resource translation quality corresponding to various source datasets.

to Table 3, we found that compared with the Transformer, BLEU scores of our method are increased by 2.68, 2.87 and 2.83 respectively in the three target tasks. In addition, compared with TF-NMT, the BLEU scores are also increased by 2.25, 1.9 and 1.91, which fully demonstrates that the global optimal parameters obtained from multilingual translation model training phase can not achieve better performance in low-resource translation tasks, because the gradient corresponding to the optimal parameter is easily introduced into the local minimum problem. Compared with [5], we also get same conclusions. They utilized conventional meta-learning algorithm and take the target language as the "pivot" to realized multilingual unified word representation. They query and locate the position of low-resource languages words in the unified semantic space via the key-value networks, and then integrated the multilingual knowledge. In addition, excessive consumption of computing resources during training phase. Therefore, our method has also been greatly improved in training efficiency, as shown in Table 5.

Model	Time consuming	Speedup
Meta-NMT	≈ 3day	-
IR-Meta-NMT	≈ 1.7day	1.76×

Table 5. Time consumption.

When we select different meta-training data, we found that the results are also different. When we select several large European languages, such as En, Fr, De, the parameters obtained by meta-learning are transferred to the low-resource translation tasks, the BLEU scores is better than that of other European languages. However, the BLEU scores of the model is higher when Asia languages was used. In other words, whether model-dependent or model-independent methods are adopted, the effect will be further improved when there are some internal relations between the high-resource and low-resource languages, such as belonging to the same language family or having the same or similar grammatical structure.

4.4 Ablation experiments

We observed the influence of various modules on the NMT model through ablation experiments, and analyzed the translation quality when using the CSLS, MWE, Meta-learning (ML) and Reptile Meta-learning (RML). In addition, we also evaluated the impact of sentences of different lengths on the quality of the model. As shown in Table 6. We mainly use BPE

Model	**Mo-Zh**		**Ug-Zh**		**Ti-Zh**	
	Dev	Test	Dev	Test	Dev	Test
ML+CSLS	28.78	28.16	24.2	23.35	25.77	25.28
ML+MWE	31.34	29.95	28.36	25.52	28.51	26.73
RML+CSLS	32.48	29.61	27.29	25.8	28.85	27.02
RML+MWE	**33.35**	**30.83**	**28.58**	**26.29**	**30.07**	**27.18**

Table 6. The Ablation Experiment.

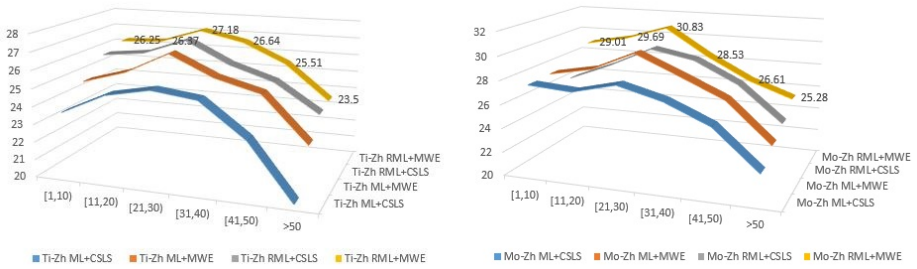

Fig. 3. The BLEU scores in different translation tasks.

to process the data. According Table 6, we found that when using the common meta-learning method, the NMT model represented by MWE word embedding is 1.79, 2.17 and 1.45 higher on the test set than the model represented by CSLS word embedding. It can be inferred that the MWE method has higher alignment accuracy and representation ability. Meanwhile, when using the improved reptile meta-learning method (RML), the translation quality of the MWE method is also better than that of the CSLS method. In addition, under the same conditions, the BLEU score of the test set using the RML method is increased by 0.88, 0.77 and 0.45 respectively compared with the model using the ML method, which is fully demonstrates the remarkable generalization ability of the model in this paper.

According to the experiment shown in Figure 3. The BLEU scores was highest when the sentence length was 20 to 30 words, and significantly decreased when the length was greater than 50 words.

4.5 Case study

Case study include crosslingual word embedding alignment visualization and translation analysis. To observe the word embedding quality of meta training data and meta test data, we map Mongolian and English with the same semantics into Chinese vector space, aligned word pairs have more weight (cyan line), which proved that our method has a significant effect on crosslingual alignment, as shown in Figure 4.

As shown in Figure 5, we observe the translation results of different methods, and find that Transformer model has significant translation generation ability, which alleviates the problem of unknown words (UNK); TF-NMT and Meta-NMT methods ignore the relationship between source languages due to the problem of crosslingual word embedding mapping. Our method not only alleviates the above problems, but also learns more semantic representation including named entity (bold font), which shows remarkable effect.

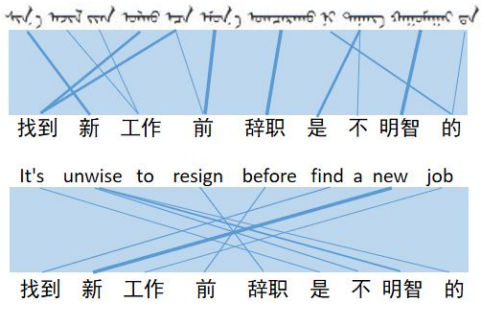

Fig. 4. Unified word embedding alignment visualization.

Source	كۆردى تىزىلمىشنى سەپكە قاراۋۇللىرىنىڭ ھۆرمەت سۆيىسىدا پارات رەھبەرلىرى دۆلەت نىككى.
Ref.	两国 元首 回 到 检阅 台 观看 仪仗 队 分列 式 。
Transformer	两个 国家 的 总统 去往 审查 台子 欣赏 分列 。
TF-NMT	两个 家园 主席 来到 检阅 <unk> <unk> 仪式 。
Meta-NMT	两国 首脑 来到 检阅 舞台 <unk> 队伍 仪式 散开 。
IR-Meta-NMT	**两国 元首** 返回 **检阅 主席台** 查看 队伍 **发散** 式 。

Fig. 5. Translation analysis.

5 Conclusion

In this paper, we proposed an improved reptile meta-learning method, in which the parameters of the previous specific task are taken as the initial parameters of the new specific task, and the final meta-parameter gradient is determined in combination with the first-order calculation method of the meta-gradient. Compared with the traditional method, this method is more efficient and effective. In addition, in order to integrate multi-language knowledge, we propose a

multi-aligned cross-language word embedding, which alleviates the problems of knowledge sharing.

References

1. Di Jin, Zhijing Jin, Joey Tianyi Zhou, Peter Szolovits: Unsupervised Domain Adaptation for Neural Machine Translation with Iterative Back Translation. CoRR, vol. abs/2001.08140 (2020). https://arxiv.org/abs/2001.08140
2. Barret Zoph, Deniz Yuret, Jonathan May, Kevin Knight: Transfer Learning for Low-Resource Neural Machine Translation. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016. pp. 1568-1575. doi:10.18653/v1/d16-1163
3. Alham Fikri Aji, Nikolay Bogoychev, Kenneth Heafield, Rico Sennrich: In Neural Machine Translation, What Does Transfer Learning Transfer?. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020. pp. 7701-7710. doi:10.18653/v1/2020.acl-main.688
4. Rumeng Li, Xun Wang, Hong Yu: MetaMT, a Meta Learning Method Leveraging Multiple Domain Data for Low Resource Machine Translation. The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020. pp. 8245-8252. https://aaai.org/ojs/index.php/AAAI/article/view/6339
5. Jiatao Gu, Yong Wang, Yun Chen, Victor O. K. Li, Kyunghyun Cho: Meta-Learning for Low-Resource Neural Machine Translation. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31-November 4, 2018. pp. 3622-3631. doi:10.18653/v1/d18-1398
6. Chelsea Finn, Pieter Abbeel, Sergey Levine: Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017. vol. 70. pp. 1126-1135. http://proceedings.mlr.press/v70/finn17a.html
7. Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, Hervé Jégou: Word translation without parallel data. 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings (2018). https://openreview.net/forum?id=H196sainb
8. Idris Abdulmumin, Bashir Shehu Galadanci, Abubakar Isa: Iterative Batch Back-Translation for Neural Machine Translation: A Conceptual Model. CoRR. vol. abs/2001.11327 (2020). https://arxiv.org/abs/2001.11327
9. Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, Wei-Ying Ma: Dual Learning for Machine Translation. Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. pp. 820-828. https://proceedings.neurips.cc/paper/2016/hash/5b69b9cb83065d403869739ae7f0995e-Abstract.html
10. Yong Cheng, Yang Liu, Qian Yang, Maosong Sun, Wei Xu: Neural Machine Translation with Pivot Languages. CoRR. vol. abs/1611.04928 (2016). http://arxiv.org/abs/1611.04928