Learning to Adapt to Low-Resource Paraphrase Generation

Zhigen Li[1], **Yanmeng Wang**[1], **Rizhao Fan**[2], **Ye Wang**[1]
Jianfeng Li[1] **and Shaojun Wang**[1]
[1]Ping An Technology, [2]University of Bologna
{lizhigen974,wangyanmeng219,wangye430}@pingan.com.cn
rizhao.fan@unibo.it, {lijianfeng777,wangshaojun851}@pingan.com.cn

Abstract

Paraphrase generation is a longstanding NLP task and achieves great success with the aid of large corpora. However, transferring a paraphrasing model to another domain encounters the problem of domain shifting especially when the data is sparse. At the same time, widely using large pre-trained language models (PLMs) faces the overfitting problem when training on scarce labeled data. To mitigate these two issues, we propose, LAPA, an effective adapter for PLMs optimized by meta-learning. LAPA has three-stage training on three types of related resources to solve this problem: 1. pre-training PLMs on unsupervised corpora, 2. inserting an adapter layer and meta-training on source domain labeled data, and 3. fine-tuning adapters on a small amount of target domain labeled data. This method enables paraphrase generation models to learn basic language knowledge first, then learn the paraphrasing task itself later, and finally adapt to the target task. Our experimental results demonstrate that LAPA achieves state-of-the-art in supervised, unsupervised, and low-resource settings on three benchmark datasets. With only 2% of trainable parameters and 1% labeled data of the target task, our approach can achieve a competitive performance with previous work.

1 Introduction

Paraphrase generation can comprehend a sentence and generate another with the same semantics but with variations in lexicon or syntax, which has various applications on downstream tasks including query rewriting (Dong et al., 2017), data augmentation (Iyyer et al., 2018) and language model pre-training (Lewis et al., 2020a). Conventional approaches (Prakash et al., 2016; Chowdhury et al., 2022) model the paraphrase generation as a supervised encoding-decoding problem, inspired by machine translation systems. However, the success of these methods often relies on a large number of parallel paraphrases, whose collection is time-consuming and requires a lot of domain knowledge. Therefore, in real scenarios with a small amount of parallel data, the model suffers from performance drops facing domain gaps. This phenomenon, known as domain shift problem (Pan and Yang, 2009), comes from the representation gap between training and testing domains with different writing styles or forms.

To tackle this problem, unsupervised methods such as editing-based approaches (Bowman et al., 2016; Miao et al., 2019) or reinforcement learning (Li et al., 2018; Siddique et al., 2020), and weakly-supervised methods such as retrieval-enhanced (Ding et al., 2021; Yin et al., 2022) or prompt-based (Wang et al., 2022) do not introduce or only introduce a small number of supervised signals, which limits their performance such that underperforms supervised methods. In fact, large-scale unlabeled corpus data (UCD) and labeled source domain data (LSDD), as well as a few labeled target domain data (LTDD), can be easily achieved. Therefore, we propose a new three-stage learning paradigm: pre-training, meta-learning, and fine-tuning, aiming to leverage the pre-trained knowledge on UCD, source domain knowledge on LSDD, and adapt to target domain on LSDD to improve the performance of low-resource paraphrase generation. In order to successfully implement this learning paradigm, we propose a simple yet effective model which combined pre-trained language model (PLM) and MAML (Finn et al., 2017), named **L**earning to **A**dapt to low-resource **PA**raphrase generation (**LAPA**). Specifically, before meta-learning, we insert an adapter layer into each transformer layer of PLM. An adapter layer is composed of a few parameters of feed-forward layer and residual connection. During meta-training and fine-tuning, only the adapter layer and normalization layer are trainable. Parameter freezing and residual connection can retain the

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 1014 - 1022

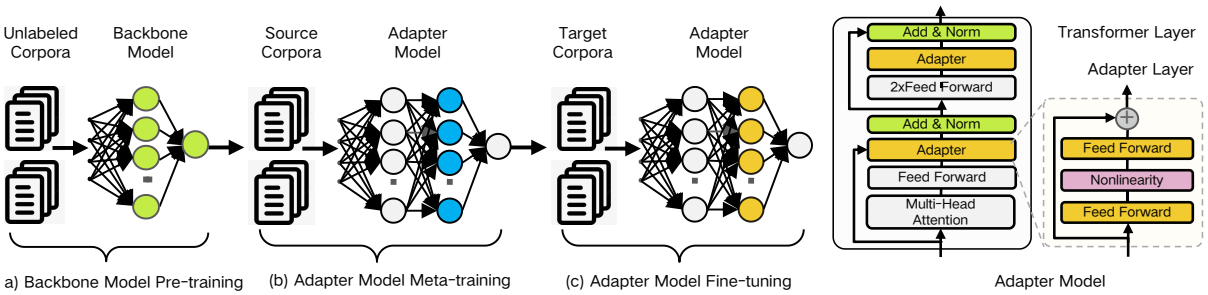

Figure 1: Three training stages of the proposed learning paradigm. Gray represents untrainable parameters, and other bright colors represent parameters that have been trained in different stages.

prior knowledge of PLM to avoid negative transfer effects. Smaller-scale parameter updating can prevent MAML from gradient explosion or diminishing problems when the number of MAML inner loop iterations and model depth increase (Antoniou et al., 2019) or training data is extremely scarce.

Overall, we hold the idea that paraphrasing is a fundamental ability of human beings. The paraphrase model should not rely on domain and seen data. Therefore, we are committed to characterizing the basic ability of the paraphrase model, obtaining gains from each domain, and applying it to a specific domain. Our contributions are summarized as follows:

- We define a novel three stages learning paradigm for low-resource paraphrase generation in data scarcity scenarios.
- We propose that LAPA implement this learning paradigm, which transferred the PLM knowledge and source domain knowledge to complete the low-resource learning in the target domain quickly and with high quality.
- The supervised, unsupervised and weakly supervised experimental results of LAPA on three benchmark datasets achieve state-of-the-art (SOTA). LAPA with only 2% of trainable parameters and 1% target task labeled data can achieve a competitive performance with previous works.

2 Related Work

While the paraphrase generation performance is greatly improved with various supervised techniques (Zhao et al., 2008; Prakash et al., 2016; Egonmwan and Chali, 2019; Cao and Wan, 2020; Hosking and Lapata, 2021; Chowdhury et al., 2022), there are few studies regarding the low-resource setting. West et al. (2021) and Meng et al. (2021) proposed novel unsupervised paraphrasing strategies by data augmentation based on reflective decoding or diverse decoding. Ding et al. (2021) and Yin et al. (2022) achieved improvements on various low-resource datasets with retrieved data and meta reinforcement learning. However, these studies only use a single large corpus for training the full PLM, which suffers from domain-shifting problems (Wang et al., 2019). Besides, under the extreme low-resource setting, directly fine-tuning the full PLM will cause an over-fitting problem (Antoniou et al., 2019).

Meta-learning helps improve low-resource performance in various recent studies, such as image classification (Soh et al., 2020), vehicle tracking (Song et al., 2020) and natural language processing (Park et al., 2021; Chen and Shuai, 2021; Hong and Jang, 2022). Finn et al. (2017) proposed a meta learner named MAML, which uses other example tasks to learn how to effectively initialize a basic learner, which can be quickly generalized to new tasks. Adapter modules have been mainly used for parameter-efficient and quick fine-tuning of a basic PLMs to new tasks (Houlsby et al., 2019; Bapna and Firat, 2019; Pfeiffer et al., 2020, 2021; He et al., 2021). Our paper proposes to incorporate meta-learning approaches to realize multi-domain migration and task adapter to realize parameter effective transfer learning (i.e., limited trainable parameters) to mitigate the above problems of paraphrase generation.

3 The Approach

3.1 Learning Paradigm

As shown in Figure 1, the workflow of our learning paradigm including three stages: 1. Backbone model pre-training on large unlabeled corpora 2. Adapter model meta-training on large source corpora using the meta-learning and 3. Adapter model fine-tuning on target corpora and evaluate model performance. The prior knowledge K_{pri} comes from first two stages: pre-training and meta-learning. We denote our backbone model by $f(\theta)$ with parameters θ. The first stage is pre-training on unlabeled corpora $\mathcal{D}_{pre}$, and we get

$f(\theta_{pre})$. The second stage is meta-training on adapter model $f[\theta_{pre}, \Phi]$ with additional parameters Φ and frozen θ_{pre} on related source corpora $\mathcal{D}_{src}$, and we got $f[\theta_{pre}, \Phi_{src}]$. Finally, we initialize the adapter model with $[\theta_{pre}, \Phi_{src}]$ and fine-tune Φ_{src} on the target corpus $\mathcal{D}_{tgt}$ to obtain a target model $f[\theta_{pre}, \Phi_{tgt}]$ which are model parameters after target adapter, i.e., the posterior knowledge K_{por}.

3.2 Backbone Model

Because PLM is equipped with prior knowledge K_{pri} and exhibits strong capabilities in a range of different generative tasks, we choose the pre-trained BART (Lewis et al., 2020b) as the backbone model for paraphrase generation. Specifically, given a labeled paraphrase pair $i = (\mathbf{x}, \hat{\mathbf{y}})$, where $\mathbf{x} = [x_1, \ldots, x_N]$, $\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_M]$, and inputting $\mathbf{x}$, the model has produced a predicted segment sequence $\mathbf{y}_{<t} = [y_1, \ldots, y_{t-1}]$ before time t, then the probability that the token generated at time t is y_t is $p(y_t|\mathbf{y}_{<t}, \mathbf{x}, \theta)$. The model is optimized by minimizing the negative log-likelihood: $\mathcal{L}_i(f(\theta)) = -\sum_{t=1}^{M} log\, p(\hat{y}_t|\mathbf{y}_{<t}, \mathbf{x}, \theta)$.

3.3 Adapter Model

The adapter model is obtained by inserting the adapter layer into each transformer layer of the backbone model. An adapter layer is a bottlenecked feed-forward network consisting of a down-project layer, a nonlinearity function and an up-project layer. In addition, a skip connection layer from input to output prevents the noised initialization from interference with the training initially. For the adapter in layer l, the function can be formulated as: $Adapter(\mathbf{z}_l) = \mathbf{W}_u^l ReLU(\mathbf{W}_d^l \mathbf{z}_l) + \mathbf{z}_l$ where z_l represents the inputs of the adapter in layer l. Besides, the normalization layers are trainable and initialized from the previous training stage.

3.4 Meta-Learning

The second stage is adapter model meta traning based on MAML (Finn et al., 2017). The learning process is shown in Algorithm 1. First, we freeze the backbone model parameters θ_{pre} that have been pre-trained in the pre-training stage, then, add new adapters with parameters Φ to get adapter model $f[\theta_{pre}, \Phi]$. Based on Algorithm 1, we first complete the meta-learning of the adapter model on the source corpus $\mathcal{D}_{src}$ to help the adapters Φ find the initialization parameters Φ_{src} suitable for paraphrase generation to adapt faster target task. At this time, we obtain the model $f[\theta_{pre}, \phi_{src}]$ with knowledge of the paraphrase generation task. In the third stage, we initialize the parameters of adapter model with $[\theta_{pre}, \phi_{src}]$. Then, based on the Algorithm 1, we fine-tune adapters ϕ on target corpus $\mathcal{D}_{tgt}$ to quickly adapt to the target corpus. Finally, we get the target model $f[\theta_{pre}, \phi_{tgt}]$.

Algorithm 1 Adapter Model Training with Model Agnostic Meta-Learning

Require: $p(\mathcal{T})$: distribution over tasks; stage (b) over $\mathcal{D}_{src}$, and stage (c) over $\mathcal{D}_{tgt}$
Require: $f[\theta, \Phi]$: adapter model
Require: θ_{pre}: pre-trained backbone model parameters
Require: Φ_{init}: initialization parameters of adapters; stage (b) is the zero, and stage (c) is Φ_{src} learned from $\mathcal{D}_{src}$
Require: α, β: step size hyperparameters
Initialize $[\theta, \Phi] \leftarrow [\theta_{pre}, \Phi_{init}]$
Fix θ in the training procedure
while not done **do**
  Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T})$
  for all $\mathcal{T}_i$ **do**
    Evaluate gradient $\nabla_{\Phi}\mathcal{L}_i(f[\theta, \Phi])$ with respect to K examples
    Compute adapted parameters with gradient descent: $[\theta, \hat{\Phi}] = [\theta, \Phi] - \alpha\nabla_{\Phi}\mathcal{L}_i(f[\theta, \Phi])$
  end for
  Update $[\theta, \Phi] \leftarrow [\theta, \Phi] - \beta\nabla_{\Phi}\sum_{\mathcal{T}_i \sim p(\mathcal{T})}\mathcal{L}_i(f[\theta, \hat{\Phi}])$
end while

4 Experimental Settings

4.1 Datasets

We conducted experiments on Quora[1], Twitter (Lan et al., 2017) and MSCOCO (Lin et al., 2014) benchmark datasets, and followed the same setting in previous works (Lin et al., 2014; Liu et al., 2020; Ding et al., 2021). For meta-learning, we choose a different source task's labeld train-set from the target task to randomly construct meta tasks. Appendix Table 4 describes more details.

4.2 Baselines

Supervised methods are trained with all parallel sentences of target task. *Unsupervised* baselines

[1]https://www.kaggle.com/c/quora-question-pairs

	Method	Quora				Twitter			
		BLEU-2	BLEU-4	ROUGE-1	ROUGE-2	BLEU-2	BLEU-4	ROUGE-1	ROUGE-2
Supervised	Res-LSTM	38.52	24.56	59.69	32.71	32.13	25.92	41.77	27.94
	Transformer	42.91	30.38	61.25	34.23	40.34	32.14	44.53	29.55
	RbM	43.54	-	64.39	38.11	44.67	-	41.87	24.23
	RaE	40.35	25.37	62.71	31.77	44.33	34.16	47.55	31.53
	FSET	51.03	33.46	66.17	39.55	46.35	34.62	49.53	32.04
	ConRPG	-	26.81	65.03	38.49	-	-	-	-
	SGCP-R	-	38.00	68.10	45.70	-	-	-	-
	LAPA (ours)	**55.61**	**39.28**	**70.78**	**48.27**	**54.80**	**42.18**	**64.14**	**47.57**
Low-Resource	WS-BART	44.19	31.18	58.69	33.39	45.03	34.00	51.34	35.89
	LTSL	49.18	36.05	64.36	39.71	49.30	37.94	56.02	40.61
	MB-RPG	**54.88**	**41.56**	67.66	43.98	51.65	39.58	61.45	44.19
	LAPA (ours)	54.10	37.51	**70.35**	**47.24**	**52.71**	**40.13**	**63.12**	**46.23**

	Method	Quora				MSCOCO			
		iBLEU	BLEU-4	ROUGE-1	ROUGE-2	iBLEU	BLEU-4	ROUGE-1	ROUGE-2
Unsupervised	VAE	8.16	13.96	44.55	22.64	7.48	11.09	31.78	8.66
	UPSA	12.02	18.18	56.51	30.69	9.26	14.16	37.18	11.21
	PUP	14.91	19.68	59.77	30.47	10.72	15.81	37.38	13.87
	BackTrans	15.51	26.91	52.56	27.85	7.53	10.80	36.12	11.03
	set2seq+RTT	14.66	22.53	59.98	34.09	11.39	17.93	40.28	14.04
	ConRPG	12.68	18.31	59.62	33.10	11.17	16.98	39.42	13.50
	DBlock	20.93	26.76	65.60	42.09	-	-	-	-
	LAPA (ours)	**25.53**	**35.12**	**68.46**	**45.09**	**11.96**	**23.48**	**52.15**	**26.77**
Low-Resource	WS-BART	17.04	31.18	58.69	33.39	10.91	15.90	40.65	15.62
	LTSL	19.20	36.05	64.36	39.71	13.45	18.87	45.18	19.17
	MB-RPG	**33.56**	**41.56**	67.66	43.98	**28.09**	19.39	49.42	25.18
	LAPA (ours)	26.62	37.51	**70.35**	**47.24**	17.79	**23.22**	**54.93**	**28.89**

Table 1: Comparisons of LAPA with baseline methods. We report average scores across five random seeds.

do not use any labels of target task. Results of ConRPG and SGCP-R are from (Meng et al., 2021) and (Kumar et al., 2020). Results for VAE are copied from Meng et al. (2021). For others, we use previously reported results in Ding et al. (2021) and Yin et al. (2022). *Low-resource* methods used a highly small amount training data of target task. The baseline models compared include the recent SOTA model LTSL (Ding et al., 2021) , MB-RPG(contemporaneous with our work) (Yin et al., 2022) and WS-BART with the full parameter fine-tuning based on BART (Lewis et al., 2020b). Like our work, they all used BART as PLM. To compare the performance of our method against the previous works, we use BLEU (Papineni et al., 2002), iBLEU (Sun and Zhou, 2012) and ROUGE (Hovy et al., 2006) metrics. All metrics are computed between the generated and the reference paraphrases in the test set (Kumar et al., 2020).

5 Experimental Results

Table 1 summarizes the performance of different methods on Quora, Twitter and MSCOCO datasets. The best score is shown in bold. Overall, our LAPA method achieves SOTA performance on most metrics across multiple datasets and different scene settings, demonstrating the effectiveness of our proposed framework. At the same time, low-resource LAPA also approaches or even exceeds the supervised SOTA method (i.e SGCP-R) (Quora's BLEU-4 are comparable, and other metrics are exceeded). Compared with supervised methods that need to learn from a large amount of target task labeled parallel paraphrases, our method makes full use of other source domain paraphrases and can achieve comparable results with very low-cost supervision signals in target-domain.

6 Analyses

6.1 Parameter Study

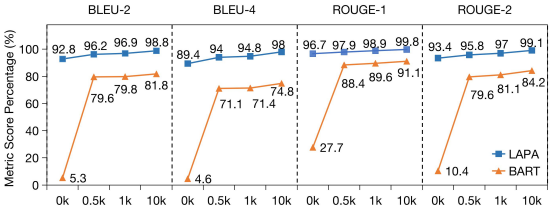

Figure 2: The experimental results of different target data size on Quora for low-resource setting.

We also separately analyze the impact of target task

Example	**Input**	Can we ever store energy produced in lightning?	How does a pencil and a liquid eyeliner differ?	How come there's no physical evidence for sea dragons existing if they're the largestanimal in the sea.
	Reference	Can we store the energy from lightining?	What is the difference between a liquid eyeliner and a pencil eyeliner?	Why is there no evidence of sea dragons existing?
Unsupervised	**BART**	CanCan we ever store energy	How	How.
	LAPA	Can we store energy from lightning?	What is the difference between liquid eyeliner and a pencil?	How come there is no physical evidence of sea dragons?
Low-Resource	**BART**	Is lightning energy storeable?	What is the difference between a liquid eyeliner and a pencil?	What is the physical evidence that sea dragons exist?
	LAPA	Can we store energy produced by lightning?	How does a liquid eyeliner differ from a pencil?	How can sea dragons exist if they're the largest animal in the sea?.
Supervised	**BART**	Is it possible to store the energy of lightning?	How do liquid eyeliners and pencil eyeliners differ from each other?	Why is the sea dragon the largest animal in the world?
	LAPA	Can we store energy from lightning?	What is the difference between liquid eyeliner and a pencil?	Why is there no physical evidence of sea dragons existing?

Table 2: Examples of the generated paraphrases on Quora dataset. We highlight the key phrases in the paraphrases generated and use wavy underline to show the matched parts between LAPA and reference.

labeled data scale under low-resource setting. Figure 2 shows the experimental results on the Quora dataset. It can be conclused that LAPA has a significant effect compared with BART under the same small data size. LAPA can achieve the effect of 89% to 93% of the full amount of data when not using any target task labeled data; when using a very small amount of data such as 0.5k (i.e 0.5% of the full data), it can be improved to 94% to 96%; when the amount of data increases to 10k (i.e 10% of the full data), the performance is almost the same as the full amount of data 100k.

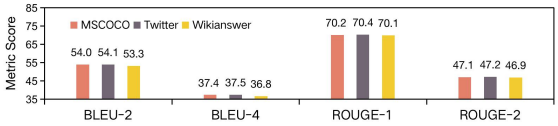

Figure 3: The experimental results of different source corpus for Quora target task under low-resource setting.

It should be pointed out that which dataset is selected as the source data can not have a substantial impact on the migration results, as shown in Figure 3. The results independent of the source dataset prove that LAPA can learn the paraphrasing task itself on any dataset, so it has strong adaptability to the target task.

6.2 Ablation Study

Method	T.P.	BLEU-2	BLEU-4	Rouge-1	Rouge-2
BART	418M	44.19	31.18	58.69	33.39
BART+SD	418M	51.61	35.12	68.46	45.09
BART+SD+ML	12M	54.10	37.51	70.35	47.24

Table 3: Ablation results of pre-trained BART, source data (SD) and meta-learning (ML) variants on Quora dataset. T.P. denote trainable parameters.

We conduct an ablation study with three variants under the low-resource setting of the Quora dataset to investigate the contribution of each component in the proposed method. The experimental results are shown in Table 3. We can get: first, using pre-trained BART can get good results; second, by adding the source task dataset for pre-trained BART, the knowledge of the source domain can be effectively learned, thereby improving the performance of the model in the target domain; third, adding our proposed meta-learning framework can again effectively improve the speed and quality of learning the source domain (LAPA only has 2.8% training parameters compared with BART) and achieve the best performance.

6.3 Case Study

Table 2 lists some paraphrases generated by LAPA and BART with different experimental settings. We can observe that paraphrases produced by LAPA are not only grammatically correct but preserve the semantics of *Input* more completely, and the expression is closer to *Reference* than the other methods. This benefits from the fact that our LAPA approach can make full use of source domain data and task features, and better preserve the prior knowledge of PLM, so as to adapt to new target tasks quickly and efficiently.

7 Conclusion

In this work, we investigate the problem of paraphrase generation under the low-resource setting and propose a simple yet effective approach LAPA. We effectively combine transfer learning and meta-learning by using adapter modules as the bridge. Whether in supervised, unsupervised or low-resource setting, the results that our approach achieves the SOTA results on benchmark datasets. In the future, we plan to explore how to choose a smaller but suitable high-quality source corpus for learning in the source domain to improve the effect of transferring to the target domain, because not all source domain data has a positive effect. Second, we plan to extend this framework to other AI fields to solve low-resource problems in other scenarios and enable more industrial applications.

Limitations

The major limitation of present study is the need for source domain annotated data that can adapt to the target domain. Because this is the source of data for the knowledge of the learning task itself, it cannot be avoided. In the real world, we can find it from public free datasets, exchange it commercially with other institutions, or annotate a batch of raw data ourselves as a cold start to solve this problem. Secondly, this study also has insufficient research on related variables. Due to the limitation of time and article length, we have not been able to study. These findings provide the following insights for future research: What is the lower bound of the amount of source domain data that can be well adapted to the target task? Whether we can apply weak supervision, data augmentation and other methods to create source domain data? How to select high-quality source domain data to get a better adapter model? We leave these questions to future research.

References

Antreas Antoniou, Harrison Edwards, and Amos J. Storkey. 2019. How to train your MAML. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Ankur Bapna and Orhan Firat. 2019. Simple, scalable adaptation for neural machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1538–1548, Hong Kong, China. Association for Computational Linguistics.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21, Berlin, Germany. Association for Computational Linguistics.

Yue Cao and Xiaojun Wan. 2020. DivGAN: Towards diverse paraphrase generation via diversified generative adversarial network. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2411–2421, Online. Association for Computational Linguistics.

Yi-Syuan Chen and Hong-Han Shuai. 2021. Meta-transfer learning for low-resource abstractive summarization. *arXiv preprint arXiv:2102.09397*.

Jishnu Ray Chowdhury, Yong Zhuang, and Shuyi Wang. 2022. Novelty controlled paraphrase generation with retrieval augmented conditional prompt tuning. *arXiv preprint arXiv:2202.00535*.

Kaize Ding, Dingcheng Li, Alexander Hanbo Li, Xing Fan, Chenlei Guo, Yang Liu, and Huan Liu. 2021. Learning to selectively learn for weakly-supervised paraphrase generation. *arXiv preprint arXiv:2109.12457*.

Li Dong, Jonathan Mallinson, Siva Reddy, and Mirella Lapata. 2017. Learning to paraphrase for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 875–886, Copenhagen, Denmark. Association for Computational Linguistics.

Elozino Egonmwan and Yllias Chali. 2019. Transformer and seq2seq model for paraphrase generation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 249–255, Hong Kong. Association for Computational Linguistics.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2013. Paraphrase-driven learning for open question answering. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1608–1618, Sofia, Bulgaria. Association for Computational Linguistics.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR.

Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. 2021. Towards a unified view of parameter-efficient transfer learning. *arXiv preprint arXiv:2110.04366*.

SK Hong and Tae Young Jang. 2022. Lea: Meta knowledge-driven self-attentive document embedding for few-shot text classification. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 99–106.

Tom Hosking and Mirella Lapata. 2021. Factorising meaning and form for intent-preserving paraphrasing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1405–1418, Online. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long*

Beach, California, USA, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Eduard Hovy, Chin-Yew Lin, Liang Zhou, and Junichi Fukumoto. 2006. Automated summarization evaluation with basic elements. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy. European Language Resources Association (ELRA).

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885, New Orleans, Louisiana. Association for Computational Linguistics.

Amirhossein Kazemnejad, Mohammadreza Salehi, and Mahdieh Soleymani Baghshah. 2020. Paraphrase generation by learning how to edit from samples. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6010–6021, Online. Association for Computational Linguistics.

Ashutosh Kumar, Kabir Ahuja, Raghuram Vadapalli, and Partha Talukdar. 2020. Syntax-guided controlled generation of paraphrases. *Transactions of the Association for Computational Linguistics*, 8:329–345.

Wuwei Lan, Siyu Qiu, Hua He, and Wei Xu. 2017. A continuously growing dataset of sentential paraphrases. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1224–1234, Copenhagen, Denmark. Association for Computational Linguistics.

Mike Lewis, Marjan Ghazvininejad, Gargi Ghosh, Armen Aghajanyan, Sida Wang, and Luke Zettlemoyer. 2020a. Pre-training via paraphrasing. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020b. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Zichao Li, Xin Jiang, Lifeng Shang, and Hang Li. 2018. Paraphrase generation with deep reinforcement learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3865–3878, Brussels, Belgium. Association for Computational Linguistics.

Zichao Li, Xin Jiang, Lifeng Shang, and Qun Liu. 2019. Decomposable neural paraphrase generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3403–3414, Florence, Italy. Association for Computational Linguistics.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Xianggen Liu, Lili Mou, Fandong Meng, Hao Zhou, Jie Zhou, and Sen Song. 2020. Unsupervised paraphrasing by simulated annealing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 302–312, Online. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Yuxian Meng, Xiang Ao, Qing He, Xiaofei Sun, Qinghong Han, Fei Wu, Jiwei Li, et al. 2021. Conrpg: Paraphrase generation using contexts as regularizer. *arXiv preprint arXiv:2109.00363*.

Ning Miao, Hao Zhou, Lili Mou, Rui Yan, and Lei Li. 2019. CGMH: constrained sentence generation by metropolis-hastings sampling. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6834–6842. AAAI Press.

Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

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, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Cheonbok Park, Yunwon Tae, TaeHee Kim, Soyoung Yang, Mohammad Azam Khan, Lucy Park, and Jaegul Choo. 2021. Unsupervised neural machine translation for low-resource domains via meta-learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2888–2901, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.

Aaditya Prakash, Sadid A. Hasan, Kathy Lee, Vivek Datla, Ashequl Qadir, Joey Liu, and Oladimeji Farri. 2016. Neural paraphrase generation with stacked residual LSTM networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2923–2934, Osaka, Japan. The COLING 2016 Organizing Committee.

A. B. Siddique, Samet Oymak, and Vagelis Hristidis. 2020. Unsupervised paraphrasing via deep reinforcement learning. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1800–1809. ACM.

Jae Woong Soh, Sunwoo Cho, and Nam Ik Cho. 2020. Meta-transfer learning for zero-shot super-resolution. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 3513–3522. IEEE.

Wenfeng Song, Shuai Li, Yuting Guo, Shaoqi Li, Aimin Hao, Hong Qin, and Qinping Zhao. 2020. Meta transfer learning for adaptive vehicle tracking in uav videos. In *International Conference on Multimedia Modeling*, pages 764–777. Springer.

Hong Sun and Ming Zhou. 2012. Joint learning of a dual SMT system for paraphrase generation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 38–42, Jeju Island, Korea. Association for Computational Linguistics.

Yufei Wang, Can Xu, Qingfeng Sun, Huang Hu, Chongyang Tao, Xiubo Geng, and Daxin Jiang. 2022. Promda: Prompt-based data augmentation for low-resource nlu tasks. *arXiv preprint arXiv:2202.12499*.

Zirui Wang, Zihang Dai, Barnabás Póczos, and Jaime G. Carbonell. 2019. Characterizing and avoiding negative transfer. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 11293–11302. Computer Vision Foundation / IEEE.

Peter West, Ximing Lu, Ari Holtzman, Chandra Bhagavatula, Jena D. Hwang, and Yejin Choi. 2021. Reflective decoding: Beyond unidirectional generation with off-the-shelf language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1435–1450, Online. Association for Computational Linguistics.

Haiyan Yin, Dingcheng Li, and Ping Li. 2022. Learning to selectively learn for weakly supervised paraphrase generation with model-based reinforcement learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1385–1395.

Shiqi Zhao, Cheng Niu, Ming Zhou, Ting Liu, and Sheng Li. 2008. Combining multiple resources to improve SMT-based paraphrasing model. In *Proceedings of ACL-08: HLT*, pages 1021–1029, Columbus, Ohio. Association for Computational Linguistics.

A Appendix

A.1 Datasets Details

Quora Quora includes 260K negative and 140 positive Quora question paraphrase pairs. We only use positive pairs and follow the same setting in Li et al. (2018); Kazemnejad et al. (2020); Ding et al. (2021) and randomly sample 100K, 30K, 3K parallel sentences for training, test, and validation, respectively. Low-resource settings use the same validation and test set, but the training set size is reduced.

MSCOCO MSCOCO (Lin et al., 2014) contains about 500K human annotated captions of over 120K images, i.e. each image contains five captions from five different annotators. We follow the standard data split according to Lin et al. (2014); Liu et al. (2020); Ding et al. (2021).

WikiAnswer WikiAnswer (Fader et al., 2013) contains approximately 18 million paraphrases that are word-aligned question pairs. We only use this dataset as the source task of meta-training, and follow the standard data split according to Li et al. (2019); Liu et al. (2020); Siddique et al. (2020).

Twitter The twitter URL paraphrasing corpus is built by Lan et al. (2017) for paraphrase identification. We follow the setting in Li et al. (2018), Kazemnejad et al. (2020) and Siddique et al. (2020).

The detailed dataset statistics are summarized in Table 4 .

Datasets	Train			Valid	Test	Vocab
	S.	L.R.	U.S.			
Quora	100K	3K	0K	3K	30K	8K
MSCOCO	110K	10K	0K	10K	40K	8K
Twitter	110K	1K	0K	1K	5K	10K
WikiAnswer	500k	20k	0k	20k	6k	8k

Table 4: Statistics of datasets. S./L.R./U.S. denote supervised, low-resource, and unsupervised settings, respectively.

A.2 Evaluation Details

To make a fair and comprehensive assessment, we follow the same experiment setting of each comparison work (Li et al., 2018; Liu et al., 2020; Ding et al., 2021) and conduct the comparison respectively. For data preprocessing, all the sentences are lower cased, and truncate all sentences to up to 20 words. <s> and </s> are spliced to the front and back end of the sentence as start and end markers.

For evaluation metrics, we use BLEU, i-BLEU and ROUGE that have been widely used in the previous work to measure the quality of the paraphrases. The i-BLUE aims to measure the diversity of expression in the generated paraphrases by penalizing copying words from input sentences. Specifically, we follow the unsupervised paraphrase generation baselines and set the balancing parameter $\alpha = 0.9$.

A.3 Implementation

Our experiments were conducted with PyToch on NVIDIA Tesla V100 16GB GPU. Following the comparison methods, we used BART-large as the pre-trained language model and use its pre-trained parameters. For adapter modules, the hidden size is 128. For meta-training, unless otherwise specified, a meta batch includes 3 tasks, and the batch size of each task is 10. Both basic learners and meta learners use the AdamW (Loshchilov and Hutter, 2019) optimizer for optimization, and the learning rate is set by grid search in 1e-5, 5e-5, 1e-6 and 5e-6. The internal gradient step size is 4, and the whole model has enough step size for training. For meta verification, we use a corpus excluded from the source task and the target task. For fine-tuning, we use validation set to select the best model for metrics calculation.