Compression Models via Meta-Learning and Structured Distillation for Named Entity Recognition

Qing Zhang[1,2] , Zhan Gao[1,2] , Mei Zhang[1,*], Jianyong Duan[1,2] , Hao Wang[1,2] , Li He[1,2]
[1]*School of Information Science and Technology, North China University of Technology*
[2] *CNONIX National Standard Application and Promotion Lab*
Beijing, China
{zqicl, duanjy, heli}@ncut.edu.cn, 1079332052@qq.com, wanghaomails@gmail.com, zmabcdef@126.com

***Abstract*—This paper addresses the issue of high resource consumption in named entity recognition (NER) under large models by utilizing meta-learning and structured distillation to generate lightweight models. Knowledge distillation from commonly used models in NER tasks poses challenges because of the exponentially large output space. Previous work treated it as a structured prediction task for distillation, but did not consider utilizing the feedback from the student model to optimize the student itself. Therefore, this paper proposes Meta-Structured Distillation (MSD). Specifically, this paper incorporates meta-learning into structured distillation, updating the teacher parameters based on the student's performance feedback on the dataset to obtain a better student model. Experimental results demonstrate the effectiveness of this approach, showing improvement over previous work in structured distillation.**

***Index Terms*—named entity recognition, knowledge distillation, meta-learning**

I. Introduction

Named Entity Recognition (NER) refers to the recognition of entities with specific meaning in text, mainly including person names, place names, organization names, proper nouns, etc. NER is an important basis for many natural language processing (NLP) tasks such as information extraction, question answering systems, and machine translation. Therefore, the accuracy and computational efficiency of NER models are of great importance in many experiments and applications.

In recent years, a lot of research has been done on NER. The main direction is to improve performance by using larger and more complex model architectures, such as BERT [1], ERNIE [8], GPT3 [9] and other large models. Although these models have brought great contributions to the NER task, a major drawback they all share is high computational cost, which limits their application in many sensitive environments and is not suitable for resource-constrained scenarios. Therefore, many current studies hope to reduce these large models to smaller and faster models without significant performance loss.

One feasible approach to address this problem is Knowledge Distillation (KD). In KD, there are two models: the teacher model, which possesses rich knowledge, and the student model, which absorbs knowledge from the pre-trained teacher model. A typical method in KD is to use cross-entropy to make the student model mimic the output probability distribution of the teacher model on training data. However, for NER tasks, the number of label combinations grows exponentially with the sequence length, making the computation and optimization process of cross-entropy complex. This implies that the efficiency of the student model in extracting knowledge from the teacher model is low.On the other hand, in current prevalent KD methods, the parameters of the teacher model are usually fixed after training. If the teacher model's parameters are trained incorrectly, it could lead to errors in the knowledge learned by the student. However, in such cases, the teacher model's parameters do not receive feedback on the student's learning errors and remain unchanged

To address the aforementioned issues, in this paper, we propose an efficient knowledge distillation approach that utilizes KD to train a lightweight model capable of reducing resource consumption costs and improving computational efficiency while maintaining accuracy. Specifically, we employ structured knowledge distillation to distill NER as a structured task and update the teacher model's parameters based on the student's performance feedback on the data to obtain better knowledge.BERT-CRF is one of the most commonly used models for NER tasks. In BERT-CRF, the CRF layer models the relationships between adjacent labels, which yields better results compared to simply predicting each label independently based on BERT's output. The CRF structure globally models the label sequence by considering the correlations between adjacent labels, which increases the difficulty of extracting knowledge from the teacher model. This can be addressed by minimizing the differences between the global sequence structure distributions of the student and teacher through approximation methods or aggregating the global sequence structure into local posterior distributions and minimizing the differences in the aggregated local knowledge.

This paper makes the following contributions:

1. We propose a new method to add the meta-learning method to the structured distillation, and distill the large model to obtain a small model to solve the NER task, which has achieved good performance.

*The corresponding author of this paper is Mei Zhang

979-8-3503-3078-6/23/$31.00

2. Experiments have proved that our scheme can reduce the amount of model parameters and model size, improve the reasoning speed, and have a high accuracy rate.

II. Related Work

A. Named Entity Recognition

The early use of dictionary and rule-based methods in named entity recognition requires huge manpower and material resources, and it is not easy to expand to other entity types or data sets. In machine learning-based approaches, NER is treated as a sequence labeling problem. Compared with the classification problem, the current prediction label in the sequence labeling problem is not only related to the current input feature, but also related to the previous prediction label, that is, there is a strong interdependence between the prediction label sequences, and the main methods used are hidden. Markov model, maximum entropy, conditional random field (CRF) [13], etc. NER tasks often use models such as BiLSTM-CRF, IDCNN-CRF, Lattice LSTM, etc. in the early stages of deep learning. Later, with the birth of Transformer [2], the mainstream direction began to adopt a series of variant models such as BERT, BERT-CRF, etc. Large-scale models with complex structures, these models and their variants have achieved good results in the corresponding NER tasks, but due to the large models and high computing resources, these models may not be suitable for some online systems and low resource scene.

B. Knowledge Distillation

Pruning, distillation, and quantization are commonly used model compression methods. DistilBert [5] proposed in the past are model compression methods for the Bert model, and have achieved good results in classification tasks. Knowledge distillation is a prominent method for training small networks to achieve comparable performance to large networks, and has wider applicability. Hinton et al. [3] first introduced the concept of knowledge distillation to exploit the "dark knowledge" (i.e. soft label distribution) in the large teacher model as additional supervision for training the small student model.

C. Meta-learning Distillation

The core idea of meta-learning is "learning to learn". Meta-learning usually includes a two-layer optimization process, and the inner learner provides feedback for the optimization of the meta-learner. Previous works usually aim at obtaining an optimized meta-learner (i.e., the teacher model), while the optimization of the inner learner (i.e., the student model) is mainly used to provide learning signals for the meta-optimization process. This is different from the goal of knowledge distillation, which is to optimize the student model. Recently, there have been some studies using this dual optimization framework to obtain better internal learners. For example, MetaPL [10] uses meta-learning to optimize pseudo-label generators for better semi-supervised learning and some works on KD employ meta-learning methods. Dualde [11] considered the dual influence between teacher and student, and proposed a soft label evaluation mechanism to distinguish the quality of soft labels of different triplets and a two-stage distillation to improve the adaptability of students to teachers. Metadistill [12] uses meta-learning to learn a better teacher model and transfer knowledge to the student model more efficiently. Instead of fixing the teacher, the method utilizes the feedback from the student model to improve the teacher model.

III. Approach

A. Structured Prediction

Sequence Tagging is one of the common structured prediction tasks, including named entity recognition. The task can be described as for an input sequence $x = \{x_1, x_2, \ldots, x_n\}$ composed of n labels, find the corresponding actual label $y = \{y_1, y_2, \ldots, y_n\}$, $y_n \in \{1, 2, \ldots, N\}$, where N is the size of the label set. The common NER model is BERT-CRF, which uses pre-trained BERT to obtain the semantic features of the text, and obtains the corresponding Logits, also known as Emission scores, after Dropout and Linner changes. The CRF layer provides the transfer score Transition, using the emission score E and the transfer score T, the score of the entire sequence can be obtained:

$$Score(y, x) = Emission + Transition \tag{1}$$

The substructure score of the corresponding sequence is:

$$Score(u_i, x) = E_{u_i} + T_{(u_{i-1}, u_i)} \tag{2}$$

At this point the conditional probability of the output structure y given the input x:

$$\begin{aligned} P(y|x) &= \frac{exp(Score(y, x))}{\sum_{y' \in Y(x)} exp(Score(y', x)} \\ &= \frac{\prod_{u \in y} exp(Score(u, x))}{Z(x)} \end{aligned} \tag{3}$$

The loss function is the negative log-likelihood function of the conditional probability:

$$L_{NLL} = -\log P(y|x) \tag{4}$$

where Y(x) denotes all possible output structures for a given input x, $Score(y, x)$ is the scoring function for evaluating the output y, $Z(x)$ is the partition function, and $u \in y$ denotes that u is a substructure of y . Define the substructure space $U(x) = y \in Y(x)\{u|u \in y\}$as the set of substructures of all possible output structures for a given input x.

B. Knowledge Distillation

Knowledge distillation is a commonly used model compression and acceleration technique. It uses the teacher model to train a small student model, and mimics the output distribution of the large teacher model through cross-entropy:

$$L_{KD} = -\sum_{y \in Y(x)} P_t(y|x) \log P_s(y|x) \tag{5}$$

where P_t and P_s are the distributions of the teacher model and the student model, respectively. The student training objective function loss contains the above distillation loss and the loss of the ground truth label:

$$L_S = \alpha L_{NNL} + (1-\alpha) L_{KD} \tag{6}$$

where α is a dynamic weight, usually set between 0 and 1 during training.

C. Sequence Knowledge Distillation

1) Top-K Distillation: The computation of distillation loss is intractable for sequence tasks because the output space $Y(x)$ grows exponentially with sentence length L. In order to solve this problem, Kim [5] uses the K best sequences predicted by the teacher model to approximate the teacher distribution and uses the Viterbi algorithm to predict the k best label sequences $T = \{y_1, y_2, \ldots, y_k\}$. Then:

$$L_{Top-k} = -\frac{1}{k}\sum_{y\in T} \log P_s(y|x) \tag{7}$$

This can also be viewed as data augmentation by generating k pseudo-label sequences for each input sentence by the teacher. Whereas the Top-K distillation only approximates the teacher's structure distribution, the Top-k method suffers from a large bias because the approximation gets worse as k increases.

2) Posterior Distillation: This method attempts to extract structure-level knowledge based on a local (token) distribution $q(y_k|x)$, which can be computed exactly.

$$\begin{aligned} q(y_k|x) &= \sum_{\{y_1,y_2,\cdots,y_n\}\setminus y_k} p(y_1, y_2, \cdots, y_n|x) \\ &= \frac{\sum_{\{u_1,u_2,\cdots,u_n\}\setminus u_k} \prod_{i=1} Score(u_i, x)}{Z} \\ &\propto \alpha(u_k) \times \beta(u_k) \end{aligned} \tag{8}$$

where Z is the denominator of the equation, often referred to as the partition function, and $\alpha(u_k)$ and $\beta(u_k)$ are computed in both forward and backward passes using a forward-backward algorithm. The distillation loss under this method is:

$$L_{Pos} = -\sum_{i=1}^{n}\sum_{j=1}^{|V|} q_t(y_i = j|x) \log q_s(y_i = j|x) \tag{9}$$

3) Structural Knowledge Distillation: Wang et al. [7] calculated the factorization form of the conditional probability formula (3) as:

$$L_{struct} = -\sum_{u\in U} P_t(u|x)\, Score_s(u, x) + \log Z_s(x) \tag{10}$$

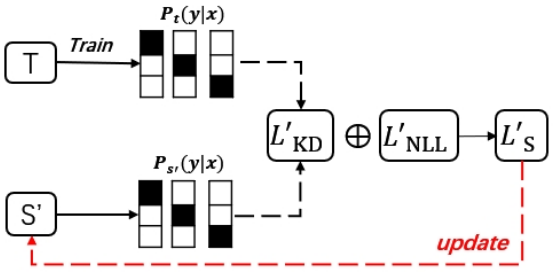
The substructure score is defined as $Score(u, x) = T((u_{i-1}, u_i), x) + E(u_i, x)$.The substructure margin $P_t((u_{i-1}, u_i)|x)$ of the teacher model can be calculated by:

$$\begin{aligned} P_t((u_{i-1}, u_i)|x) &\propto \alpha(u_{i-1}) \times \beta(u_i) \\ &\times exp(Score(((u_{i-1}, u_i), x) \end{aligned} \tag{11}$$

where $\alpha(u_{i-1})$ and $\beta(u_i)$ are forward and backward scores that can be computed using the classic forward-backward algorithm. This method is based on the marginal distribution of two adjacent labels, while the posterior KD is based on the marginal distribution of a single label.

D. Meta-Structured Distillation

Previous work proposes deep mutual learning for switching roles between student and teacher, training the original teacher model with soft labels generated by the student model [15], and recent work proposes to update the teacher model with task-specific losses during the KD process [16]. Different from previous methods, our method utilizes the student's performance feedback to update the teacher model.We hope that teachers can accept and adapt to students' learning styles, and therefore use students' feedback to modify their own knowledge delivery methods or content, so as to improve the knowledge distillation effect on students and better guide students to learn the teacher's knowledge. In general, we hope that teachers and students can adapt and learn from each other during the training process.In our training process, the student S is first copied to S', and S' is trained by knowledge distillation loss. In this way, we can get a student S' that can be used for testing. We then draw samples from the validation set and compute the loss on S' for these samples. We use this loss as feedback to update the teacher by computing the second derivative and performing gradient descent. Finally, we discard the subject S', and use the updated teacher to perform a distillation operation on the same training batch to obtain the student S, as shown in Figure 1 and Figure 2. The meta-



Step 1

Fig. 1. MSD training process step1

structured distillation loss function is as follows:

$$\begin{aligned} L_{MSD}(\theta_S; \theta_T) = \alpha L_{NLL}(y_i, S(x_i; \theta_S)) + \\ (1-\alpha) L_{KD}(T(x_i; \theta_S), S(x_i; \theta_S)) \end{aligned} \tag{12}$$

The student model θ_S is the inner learner, and the teacher model θ_T is the meta-learner. For each training step, we first copy the student model θ_S *to* θ'_S.Then given a batch of training

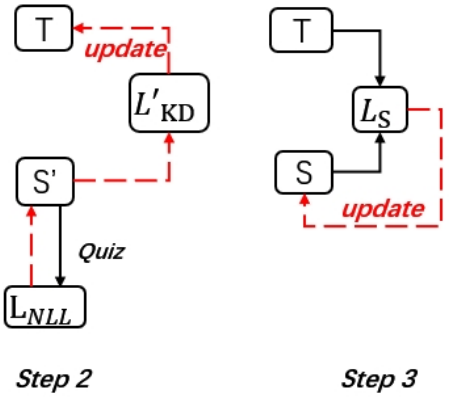


Fig. 2. MSD training process step2 and step3

examples x and learning rate λ, update $\theta_S^{'}$ in the form of structured distillation algorithm:

$$\theta^{'}{}_S\left(\theta_T\right) = \theta_S - \lambda \nabla_{\theta_S} L_{MSD}\left(\theta_S; \theta_T\right) \tag{13}$$

The updated test student parameter $\theta_S^{'}$, and the student test loss $L_q = L_{NLL}(q, \theta_S^{'}(\theta_T))$ on a batch of test samples q sampled from the held-out validation set q, L_q is a function of the teacher parameter θ_T. Therefore, we can optimize the teacher model θ_T by the learning rate μ:

$$\theta_T \leftarrow \theta_T - \mu \nabla_{\theta_T} L_{NLL}\left(q; \theta^{'}{}_S\left(\theta_T\right)\right) \tag{14}$$

Finally, the updated teacher model θ_T performs a structured distillation operation on the student model θ_S using formula (9). MSD is a general framework that can be easily applied to various KD objectives as long as the objective is differentiable with respect to the teacher parameters. We conduct experiments on the dataset to demonstrate the effectiveness of our method.

IV. Experiment

A. Experiment Settings

Use $BERT_{12} - CRF$ as the teacher model and $BERT_6 - CRF$ as the student model. Among them, the pre-trained BERT makes full use of the knowledge from a large unlabeled corpus, and the CRF captures the correlation of predictor variables. For the teacher model BERT-CRF, we first load the pre-trained BERT-base-uncased consisting of 12 Transformer Encoder layers, and then fine-tune with the CRF layer on the target task. We use Adam as the optimizer, and for the training hyperparameters, we fix the maximum sequence length of all tasks to be 128, the temperature to be 1, the learning rate of the Bert layer to be 5e-4, the learning rate of the CRF layer to be 3e-4, and the batch size to be 16. The weight α of KD loss is 0.5, and the training rounds are 10.The undistilled 6-layer BERT-CRF is used as a Baseline reference. We also compared with previous knowledge distillation methods for sequences, including Topk, Posterior, Struct methods [6] [7].For our MSD method, we adopt Posterior Distillation as the loss function of the distillation part.

B. Data set

Using CONLL-2003 [14], WikiAnn as the dataset for the NER task. Use CONLL-2000 as the dataset for the Chunking task.

C. Result

TABLE I
CONLL-2003 NER TASK

Method	#Param.	F1	Recall	Precision
$BERT_{base} - CRF$(teacher)	110M	90.43	91.55	89.33
$BERT_{6L} - CRF$(student)	66M	87.40	88.75	86.09
Topk	66M	87.83	88.93	86.67
Pos	66M	88.05	89.10	86.93
Struct	66M	88.54	88.93	88.36
MSD(ours)	66M	**89.34**	**90.04**	**88.64**

TABLE II
WIKIANN NER TASK

Method	#Param.	F1	Recall	Precision
$BERT_{base} - CRF$(teacher)	110M	82.55	84.37	80.81
$BERT_{6L} - CRF$(student)	66M	80.04	82.55	77.69
Topk	66M	80.10	82.53	77.76
Pos	66M	80.17	82.51	77.96
Struct	66M	80.34	82.68	78.13
MSD(ours)	66M	**80.63**	**82.75**	**78.63**

TABLE III
CONLL-00 CHUNKING TASK

Method	#Param.	F1	Recall	Precision
$BERT_{base} - CRF$(teacher)	110M	96.24	96.28	95.97
$BERT_{6L} - CRF$(student)	66M	95.02	95.26	94.77
Topk	66M	95.72	95.92	95.52
Pos	66M	95.90	96.14	95.66
Struct	66M	95.97	96.20	95.75
MSD(ours)	66M	**96.04**	**96.19**	**95.88**

Among them, teacher is a 12-layer BERT-CRF model, student is a 6-layer BER-CRF model, Topk, Pos, and Struct are the model distillation methods mentioned by Wang [6] [7], and Param is the size of the model parameters. We compare the performance of the model on different datasets respectively. It can be seen that our scheme is better than other baselines while reducing the amount of model parameters and improving the calculation speed, which proves the effectiveness of our method.

On the other hand, our experimental results in the CONLL-00 chunking task demonstrate that our approach achieves favorable performance in other structured tasks as well. Our

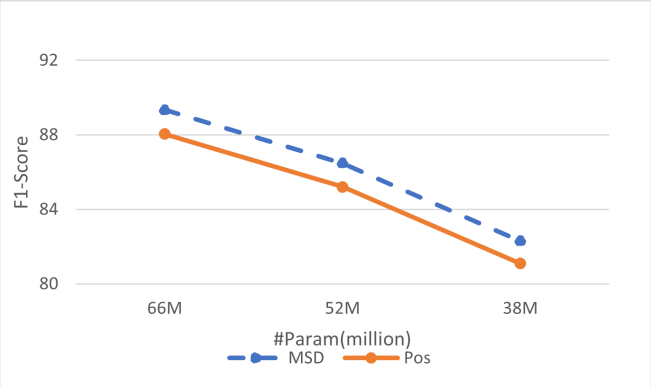


Fig. 3. Results with different student architectures.

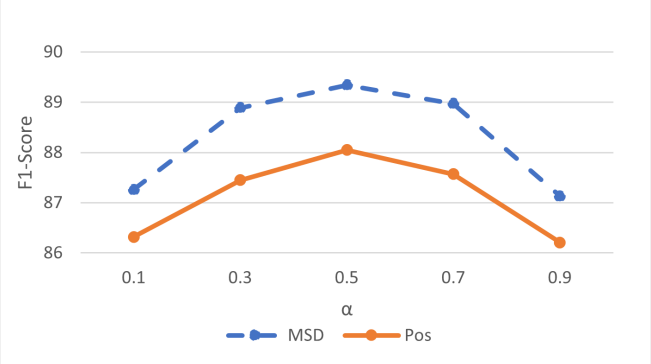


Fig. 4. Results with different loss weight α.

method is a general approach. Additionally, we compared the distillation effects of different student models on the CONLL-03 dataset and the results with different distillation loss weights α, as shown in Figure 3 and Figure 4. It can be observed that the MSD method exhibits low sensitivity to hyperparameters and good robustness.

V. Conclusion

Different from traditional knowledge distillation tasks, structured prediction tasks aim to solve the problem that the output is a complex structure rather than a single variable. Knowledge distillation is difficult for these models because their output space is exponential. Therefore, we decompose the sequence into substructures for knowledge distillation, and we are different from the previous teacher model parameter fixed guidance method, the teacher will update through the students' performance feedback. We have demonstrated the effectiveness of our method on the dataset. The method we propose is superior to previous distillation methods, and this method is general and can achieve a better result for structured prediction tasks.

The disadvantage of this method is that the model training process involves the meta-update to obtain the second order derivative, which consumes a lot of training time and memory. Since our method needs to update the teacher and student models at the same time, the computing resources during training are more expensive than those alone. Training a student model is more expensive and takes longer to train.

VI. Acknowledgments

This work was supported by the Humanities and Social Science Foundation of the Ministry of Education (Grant No.21YJA740052), National Natural Science Foundation of China(61972003,61672040).We would also like to thank the anonymous reviewers for their helpful comments, which helped improve this paper considerably.

References

[1] J. Devlin, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.

[2] Vaswani A,Shazeer N, Parmar N, "Attention is all you need," in Proceedings of the 31st International Conference on Neural Information Processing Systems,pp. 6000–6010, 2017.

[3] Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean, "Distilling the knowledge in a neural network," arXiv preprint arXiv:1503.02531, 2015.

[4] Kim, Yoon, and Alexander M. Rush, "Sequence-level knowledge distillation," arXiv preprint arXiv:1606.07947, 2016.

[5] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," arXiv preprint arXiv:1910.01108.2019.

[6] Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang,Fei Huang, and Kewei Tu, "Structure-level knowledge distillation for multilingual sequence labeling," In Proceedings of the 58th Annual Meeting ofthe Association for Computational Linguistics, pp. 3317–3330, 2020.

[7] Xinyu Wang, Yong Jiang, Zhaohui Yan,and Zixia Jia,"Structural knowledge distillation: Tractably distilling information for structured predictor," 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), pp. 550–564, 2021.

[8] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu," ERNIE: Enhanced Language Representation with Informative Entities," In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 1441–1451,2019.

[9] Brown T, Mann B, Ryder N, "Language models are few-shot learners," Advances in neural information processing systems, pp. 1877-1901, 2020.

[10] Pham H, Dai Z, Xie Q, "Meta pseudo labels," Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11557-11568, 2021.

[11] Zhu Y, Zhang W, Chen M, "Dualde: Dually distilling knowledge graph embedding for faster and cheaper reasoning," Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, pp.1516-1524, 2022.

[12] Wangchunshu Zhou, Canwen Xu, and Julian McAuley,"BERT Learns to Teach: Knowledge Distillation with Meta Learning," In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 7037–7049,2022.

[13] John Lafferty, Andrew McCallum, and Fernando CN Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," Proceedings of the Eighteenth International Conference on Machine Learning , pp. 282–289, 2001.

[14] Erik Tjong Kim Sang and Fien De Meulder,"Introduction to the conll-2003 shared task: Language-independent named entity recognition," In Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, pp. 142–147.

[15] Zhang Y, Xiang T, Hospedales T M, "Deep mutual learning," Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4320-4328, 2018.

[16] Park D Y, Cha M H, Kim D, "Learning student-friendly teacher networks for knowledge distillation," Advances in neural information processing systems, vol.34, pp. 13292-13303, 2021.