PEMRC: A Positive Enhanced Machine Reading Comprehension Method for Few-Shot Named Entity Recognition in Biomedical Domain

Yuehu Dong[1], Dongmei Li[1], Jinghang Gu[2], Longhua Qian[1*], and Guodong Zhou[1]

[1] Natural Language Processing Lab, School of Computer Science and Technology, Soochow University

[2] Chinese and Bilingual Studies, The Hong Kong Polytechnic University

20215227045@stu.suda.edu.cn, 20224027005@stu.suda.edu.cn

gujinghangnlp@gmail.com, qianglonghua@suda.edu.cn, gzzhou@suda.edu.cn

Abstract. In this paper, we propose a simple and effective few-shot named entity recognition(NER) method for biomedical domain, called PEMRC(**P**ositive **E**nhanced **M**achine **R**eading **C**omprehension). PEMRC is based on the idea of using machine reading comprehension reading comprehension (MRC) framework to perfome few-shot NER and fully exploit the prior knowledge implied in the label information. On one hand, we design three different query templates to better induce konwledge from pre-trained language models(PLMs). On the other hand, we design a positive enhanced loss function to improve the model's accuracy in identifying the start and end positions of entities under low-resources scenarios. Extensive experimental results on eight benchmark datasets in biomedical domain show that PEMRC significantly improves the performance of few-shot NER.

Keywords: Few-shot Named Entity Recognition · Machine Reading Comprehension · Biomedical Domain

1 Introduction

NER is a fundamental task in information extraction, which aims to identify text segments according to predefined entity categories. Current methods use neural network approaches [3, 14, 25] to solve the NER task. However, neural-based methods require a large amount of annotated data to achieve good performance, and data annotation requires rich domain expertise. Due to the high complexity in the biomedical expertise, which poses challenges for biomedical NER in low-resource scenarios. Recently, few-shot NER [4, 5, 7, 17, 26] has received wide attention.

The current mainstream method for few-shot named entity recognition is metric learning based on Similarity Learning. Similarity-based metric learning methods [5, 24, 26] make the distance between entities of the same class smaller and the distance between entities of different types larger by learning a metric

space. However, the entity and non-entity clustering information learned by this similarity metric function in the source domain cannot be well transferred to the task in the target domain. At the same time, the tokens of other entity types in the source domain is uniformly encoded as non-entities, reducing the expressiveness of the model.

Prompt Learning method can achieve the consistency of upstream and downstream tasks by designing prompt templates and label words. The method proposed by [4] solves this problem by scoring. By incorporating distinct label words in both the source and target domains, the prompt-based approach effectively mitigates discrepancies arising from inconsistent training objectives during pre-training and fine-tuning. Moreover, its well-crafted template design facilitates information induction within the pre-trained language model. However, prompt learning cannot design templates for token-level tasks, and the high complexity caused by enumerating all spans is unacceptable.

In order to deal with the above problems, this paper introduces the method of Machine Reading Comprehension (MRC) [16]. We adopt a span extraction machine reading comprehension method, which can unify upstream and downstream tasks by designing task-specific queries on upstream and downstream tasks. Compared with prompt learning, machine reading comprehension can effectively reduce the complexity of training and inference. In order to further utilize the knowledge in the pre-trained language model, we design three different types of query templates and conduct extensive experiments. To our knowledge, we are the first to introduce the machine reading comprehension method into the few shot named entity recognition in the biomedical domain.

2 Related Work

2.1 Few-shot NER

In this section, we review two types of methods for few-shot NER: similarity-based metric learning and prompt learning.

Similarity-Based Metric Learning Similarity based approach is a common solution in few-shot named entity recognition. The tokens are classfied by assessing the similarity between the entity type representation in the support set and tokens in the test set. The few shot named entity recognition primarily relies on metric learning. Currently, there are two main approaches to metric learning: the prototype network [7, 10, 23] and contrastive learning[5, 11].

The method based on the prototype network learns a metric space that encompasses a class of data around a single type prototype representation, enabling classification into the nearest class by calculating the distance between instance representation and class prototype during inference.

[26] proposed a Nearest Neighbor (NN) classification method which divides the test set token into categories based on comparing distances with support set tokens. Contrastive learning employs Distance metric function(such as Euclidean distance) and relative entropy (Kullback-Leibler Divergence, KLD) to design

various contrastive methods aiming to narrow distances between tokens of the same category while pushing away tokens from different categories for improved token representation.

Prompt Learning Prompt learning originates from GPT [19, 1] (Generative Pre-training Transformer) models and has been widely used in few-shot learning. Prompt learning organizes the downstream task into a cloze task, and with excellent template and label word design, prompt learning effectively bridges the gap between pre-training and fine-tuning. [20, 21] used prompt learning in sentence-level tasks and achieved good results.

The performance of the model can be effectively enhanced by designing prompt templates. [20, 21] employ human-crafted templates for text classification tasks. [22] utilize a gradient-based method to search for discrete templates. [27, 8, 22] generate discrete prompt templates using pre-trained generative models. Meanwhile, [15] adopt continuous prompt templates for classification and generation tasks, thereby avoiding the necessity for intricate template design.Additionally, [9] propose P-Tuning, which involves incorporating learnable continuous prompt into discrete prompt templates.

[4] employed a template-based approach in few-shot Named Entity Recognition. In this methodology, the original sentence is fed into the encoder, while the prompt template and all text spans within the sentence are combined in the decoder. The amalgamated templates are evaluated based on loss. However, this exhaustive enumeration of all spans introduces significant complexity to the method. To address these limitations, [17] proposed an innovative template-free approach that eliminates intricate template design altogether. This alternative method restructures the task as an entity-oriented language model task by predicting label words corresponding to tokens at respective positions.

2.2 Few-shot NER in Biomedical Domain

The study conducted by [18] introduced task hardness information based on [13] to enhance transfer learning in biomedical domain for few-shot named entity recognition tasks. MetaNER[13], which adopts a multi-task learning framework, employs an adversarial training strategy to obtain a more robust, generalizable, and transferable representation method for named entity recognition. Additionally, [13] utilizes a meta-learning training approach that enables it to perform effectively in low resources scenarios.

3 Problem Definition

We adopt the task setup from [5] (as depicted in Figure 1 below). Amongst the four named entity recognition tasks (Disease, Chem/Drug, Gene/Protein, Species), we select three tasks (e.g., Disease, Chem/Drug, Gene/Protein) as source tasks with rich resources. The remaining task served as a low-resource target task (e.g., Species). For this target task, we employ a model pre-trained on the standard training set X_{tr} lines of the source tasks and fine-tune it using

the support set X_{supp} of the target task. The support set is generated by sampling instances from the training set in the target task. Finally, evaluation was conducted on the standard test set X_{test} of that particular target task.

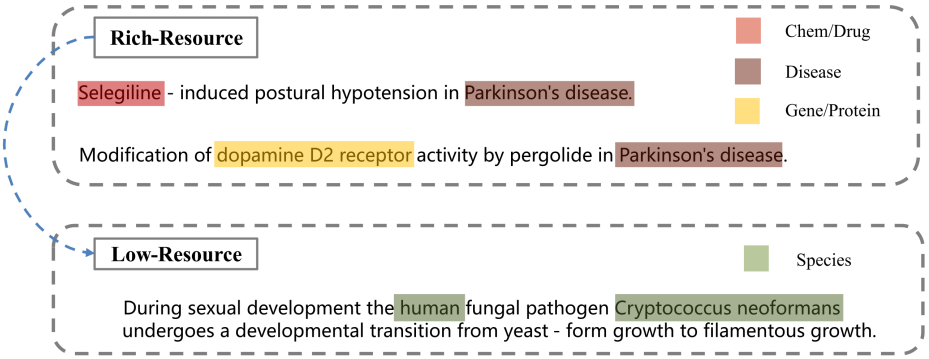

Fig. 1. Task Description.

4 Methodology

The proposed method utilizes a span-extraction approach for machine reading comprehension and incorporates a loss function that focuses on positive tokens. The methodology outlined in this section comprises four components: model architecture, query template design, loss function formulation, and training process implementation. We will present our approach sequentially in the subsequent sections.

4.1 Model Framework

Given the input $X = \{x_1, x_2, x_3, ..., x_n\}$, we concatenate it with the query $Q = \{q_1, q_2, q_3, ..., q_m\}$ to obtain the model input. Then we feed it into the pre-trained model[12] to encode it and obtain the representation **H**, as shown in equation 1.

$$H(e_{cls}, e_1, e_2, ..., e_{m+n}, e_{sep}) = PLM([CLS], q_1, q_2, ..., q_m, x_1, x_2, ..., x_n, [SEP]) \tag{1}$$

We apply a dropout layer to randomly drop the representation **H** twice, obtaining the representation H_{start} for predicting the start position and the representation H_{end} for predicting the end position, as shown in equation 2.

$$H_{start} = Dropout(H), H_{end} = Dropout(H) \tag{2}$$

Start position prediction For the obtained representation H_{start} , we feed it into a classifier FFN to get a score matrix $S \in R^{(m+n)\times 2}$, and then apply

softmax to get a probability matrix $P \in R^{(m+n)\times 2}$. Finally, we select the index with the highest probability as its prediction label $\hat{Y}_{start}$. Regarding the obtained labels, '1' signifies that the current token marks the start of an entity, and '0' indicates that the current token does not mark the start of an entity, as shown in equation 3 and equation 4.

$$P_{start} = Softmax(FFN(H_{start})) \in R^{(m+n)\times 2} \tag{3}$$

$$\hat{Y}_{start} = Argmax(P_{start}) \in (0, 1) \tag{4}$$

End position prediction The prediction process for the end position is the same as that for the start position, except that we use the representation H_{end} to obtain the probability matrix P_{end}.

4.2 Construction of Queries

In prompt learning, designing prompt template can effectively induce prior knowledge in pre-trained language models. Taking inspiration from prompt learning's template design [15,16,20,21], we construct discrete, continuous, and hybrid query templates respectively.

The discrete query template is manually crafted while learnable vectors of varying lengths are employed as continuous query templates without any prior knowledge. In hybrid queries, entity type identifiers, such as disease, are substituted with continuous learnable vectors. The hybrid template incorporates some prior knowledge(discrete query) but excludes entity label information. Examples of these three types of query templates are provided in Table 1.

Table 1. An examples of three query templates.

Query Type	Query Example
Discrete Query	Find disease entities in the next sentence.
Continuous Query	v1 v2 v3 v4 v5 v6 v7.
Hybrid Query	Find v1 entities in the next sentence.

The expression "v1-v7" denotes a learnable vector, akin to the continous prompt template employed in prompt learning. The "[unused]" symbols are utilized as learnable vectors to seamlessly integrate into the input during the implementation.

4.3 Loss Function Formulation

In the context of machine reading comprehension models, it is a common practice to compute the sequence loss ζ_{seq} by applying cross-entropy between the probability matrix P representing start and end positions, and the label Y. The formula is shown in equation 5 below.

$$\zeta_{seq} = CrossEntropy(P_{start}, Y_{start}) + CrossEntropy(P_{end}, Y_{end}) \quad (5)$$

To improve the accuracy of the model in identifying the start and end positions of entities, we augment the loss of gold labeled tokens, namely positive enhanced loss. The objective is for the model to acquire more information about entity head and tail tokens. The loss function can be defined as equation 6: where Y_{start_p} represents a positive token with its starting position, P_{start_p} corresponds to the token probability matrix of that positive token; likewise for P_{end_p} and Y_{end_p}.

$$\zeta_{pos} = CrossEntropy(P_{start\_p}, Y_{start\_p}) + CrossEntropy(P_{end\_p}, Y_{end\_p}) \quad (6)$$

We combine these two functions into ultimate loss function as equation 7:

$$\zeta_{final} = \zeta + \zeta_{pos} \quad (7)$$

4.4 Training Process

The BioBERT model serves as the base model F and is trained on a rich resources training set X_{tr}. At this stage, we do not incorporate a positive enhanced loss L_{tr}. Subsequently, We then fine-tune model with positive enhanced loss L_supp on a few-shot support set X_{supp}. Training on a support set may lead to severe overfitting, we maintain a fixed number of training epochs on the support set throughout the process. The algorithmic details regarding the model's training procedure are elucidated in Algorithm 1.

Algorithm 1: Training and Fine Tuning

Require: Training Data X_{tr}, Support Data X_{supp},
Train loss function L_{tr}, Finetune loss function L_{supp}, Model F

epoch = num_epoches //initialize fixed nums epoch
// training in source domain
for sampled(w/o replacement) minibatch X in X_{tr} do
  $L_{tr} = F(X)$ // L_{tr} without ζ_{pos}
  update F by backpropagation to reduce L_{tr}
end for
$F_{source} \leftarrow F$
// finetuning to target domain
While epoch >0 do
  for sampled(w/o replacement) minibatch x in X_{supp} do
    $L_{supp} = F_{source}(x)$ // L_{supp} with ζ_{pos}
    update F_{source} by backpropagation to reduce L_{supp}
  end for
  epoch $\leftarrow$ epoch-1
end while
$F_{target} \leftarrow F_{source}$
return F_{target}

5 Experiment

5.1 Datasets

The benchmark NER corpora preprocessed by BioBERT[12] are utilized in this study. We have conducted an analysis of entity counts in both the training and test sets within the corpus, with the statistical findings presented in Table 2 below.

Table 2. Corpus Statistic

Task	Corpus	Num of Entities	
		Train	Test
Disease	NCBI	5,145	960
	BC5CDR	9,385	9,809
Drug/Chem	BC5CDR	9,385	9,593
	BC4CHEMD	29,478	25,346
Gene/Protein	JNLPBA	32,178	6,241
	BC2GM	15,197	6,325
Species	LINNAEUS	2,119	1,433
	S800	2,557	767

5.2 Sampling Strategy

The previous sampling has primarily employed two predominant methods, namely the N-way K-shot[6] sampling method and the precision sampling [2, 4, 17] method. Both of these methods are instance-oriented samplings that select a specific number of entities randomly using different strategies. However, in real-world scenarios, inputs do not exist solely as instances. To address this limitation, HGDA[18] proposed a sentence-level oriented sampling method. In the few-shot setting, K sentences containing entities are sampled as the support set. In this paper, we also adopt HGDA's[18] sampling strategy to obtain the support set by performing sampling within the standard train set.

5.3 Experimental Settings

The BioBERT model is used as the base encoder, while the span extraction network is employed for extracting entity spans. The loss calculation involves the utilization of the positive enhanced loss function, and Adam serves as the optimizer. Throughout this study, all experiments are conducted on a 3090Ti, and after employing 5 different seeds for experimentation purposes, an average F1 value is obtained. To provide a clearer details, Table 3 presents all training-related hyperparameters.

Table 3. Hyper-Parameter Settings

Hyper Parameter	Value
Max input lenght	256
Source batch size	8
Target batch size	2
Source task encoder lr	1e-5
Target task encoder lr	1e-4
Source task classifier lr	1e-4
Target task classifier lr	1e-4
Dropout rate	0.1
Number of epoches	10
Number of learnable vectors in continuous query	7
Number of learnable vectors in hybrid query	3

5.4 Experimental Results

In this section, we present the results of performance differences among different query templates, loss function and finally the comparison with SOTA systems.

Impact of Query Template The impact of query templates on performance is investigated in this section, aiming to explore how the form of label information as prior knowledge in the machine reading comprehension framework affects recognition performance. Table 4 compares the F1 values of different query templates on the NCBI corpus under various few-shot settings(K=5, 10, 20, 50), with the highest value for each quantity highlighted in bold.

Table 4. Performance of three different query templates on NCBI dataset.

Query Type	5	10	20	50
Continuous	50.83	58.30	65.58	**69.86**
Hybrid	49.81	54.11	66.94	69.39
Discrete	**57.84**	**61.06**	**67.40**	69.03

The table above clearly demonstrates that the discrete query template outperforms the continuous and hybrid query templates, particularly when K=5, 10, and 20. However, at K=50, all three templates show comparable performance. Notably, the discrete query template exhibits a larger performance gain when the support set is small; however, as the size of the support set increases, this advantage gradually diminishes. Further detailed analysis can be found in Subsection 4.5 of this paper. Consequently, in all subsequent experiments conducted in this study, discrete templates are exclusively employed as query templates for MRC baseline and PEMRC.

The Influence of Positive Enhanced Loss The effectiveness of the positive enhanced loss function is examined by comparing the disparities between two methods, MRC and PEMRC, with F1 value results presented in Table 5. Here, MRC denotes the machine reading comprehension model employing solely the cross-entropy loss function, and PEMRC incorporates the positive enhanced loss. The maximum value for each setting is highlighted in bold.

Table 5. MRC and PEMRC performance comparison

K	Method	Disease		Drug/Chem		Gene/Protein		Species	
		NCBI	BC5	BC5	BC4	JNL	BC2	LINN	S800
5	MRC	**57.84**	64.85	74.78	52.52	46.73	47.20	52.27	52.56
	PEMRC	55.77	**64.87**	**79.07**	**53.71**	**46.87**	**48.75**	**52.89**	**53.26**
10	MRC	61.06	65.50	76.93	53.26	52.63	52.00	60.64	**55.89**
	PEMRC	**62.37**	**66.27**	**79.57**	**55.56**	**52.79**	**52.67**	**62.55**	55.74
20	MRC	67.40	**67.96**	78.93	59.46	**57.33**	54.02	63.10	57.00
	PEMRC	**69.87**	**67.96**	**82.64**	**60.58**	56.61	**55.33**	**67.82**	**57.88**
50	MRC	69.03	**71.61**	81.16	59.78	59.39	56.10	68.28	59.79
	PEMRC	**72.48**	69.36	**84.08**	**61.29**	**60.53**	**57.42**	**71.79**	**59.82**

Table 5 demonstrates that PEMRC outperforms the MRC baseline model on most of the eight datasets. When averaging performance across all datasets, PEMRC achieves a 2.1% improvement over the MRC baseline system, highlighting the effectiveness of the positive enhanced loss function.

Comparison with other SOTA systems In this section, we use PEMRC as a baseline for the methodology. We conduct extensive experiments on 8 datasets and compare them with similar systems. The SOTA systems used for comparison, the experimental results, and the analysis of the results are described below.

SOTA systems

(i) MetaNER [13] is a multi-task learning method for domain adaptation, which combines supervised meta-learning and adversarial training strategies. It can obtain more robust, general and transferable representation methods in named entity recognition tasks.

(ii) HGDA [18] introduces hardness information based on MetaNER and applies it to biomedical domains.

As shown in Table 6, our performance is better than other systems in most cases, and our method achieves significant performance in low-resource situations. This may be due to the fact that our designed query templates can use the information in the pre-trained language model more directly, just like prompt learning. The less annotated data, the more obvious the effect of using the information in the PLM. In addition, the machine reading comprehension method

Table 6. Compared with the performance of SOTA systems, some dataset names are replaced by abbreviations.

K	Method	Disease		Drug/Chem		Gene/Protein		Species		AVG
		NCBI	BC5	BC5	BC4	JNL	BC2	LINN	S800	
5	MetaNER	27.29	21.71	57.84	22.12	21.75	24.43	12.14	15.16	23.87
	HGDA	31.25	26.98	61.02	25.71	37.76	35.73	17.53	28.80	33.10
	PEMRC	**55.77**	**64.87**	**79.07**	**53.71**	**46.86**	**48.75**	**52.89**	**53.26**	**56.90**
10	MetaNER	33.30	36.88	66.59	33.60	33.74	32.65	30.38	31.64	37.35
	HGDA	43.86	42.44	70.97	42.47	47.90	44.89	32.01	37.03	45.14
	PEMRC	**62.37**	**66.27**	**79.57**	**55.56**	**52.79**	**52.67**	**62.55**	**55.74**	**60.94**
20	MetaNER	46.12	47.22	73.01	43.83	41.67	39.26	49.52	29.77	46.30
	HGDA	56.31	55.29	74.72	49.44	54.66	51.24	48.43	52.05	55.26
	PEMRC	**69.87**	**67.96**	**82.64**	**60.58**	**56.61**	**55.33**	**67.82**	**57.88**	**64.84**
50	MetaNER	57.31	61.06	74.78	50.82	53.37	50.58	61.25	36.07	55.65
	HGDA	62.08	61.90	80.23	**62.73**	**61.46**	**60.16**	63.73	58.55	63.90
	PEMRC	**72.48**	**69.36**	**84.08**	61.29	60.53	57.42	**71.79**	**59.82**	**67.10**

has only one classifier for all tasks, while HGDA and MetaNER have multiple classifiers for multi-task learning. We believe that this unified classifier can learn the knowledge transfer between different tasks, while the task-specific classifier will lose some of the knowledge learned on the source tasks to some extent.

5.5 Discussion and Analysis

We offer insightful explanations to analyze the performance disparities resulting from different query template types. There are two potential reasons for this phenomenon:

Firstly, in the low-resource scenario (K=5), natural language text can effectively leverage the knowledge within the pretrained language model, whereas continuous and hybrid query templates constructed from random vectors fail to align with the model's input during pretraining and thus cannot directly harness the knowledge embedded in the pretrained language model.

Secondly, the structure of learnable vectors present in continuous and hybrid templates remains fixed, necessitating more training data to discover an optimal vector. Consequently, they achieve comparable performance to discrete query templates only when greater resources are available (K=50).

To further demonstrate how prior knowledge and label information impact experimental performance in low-resource scenarios, we analyzed error cases generated on the NCBI (disease) test set with K=5.

From Table 7, it is evident that in Case 1, only the discrete query accurately identifies all entities, while the continuous query successfully identifies one entity and the hybrid query fails to identify any entity. In Case 2, sentences without entities are correctly predicted solely by discrete queries, whereas both continuous

Table 7. The following table shows two cases. Gold represents the sentence and the entity that should be predicted, where the entity is marked in red font.

Case 1	The predicted entities
Gold	The risk of cancer, especially lymphoid neoplasias, is substantially elevated in A - T patients and has long been associated with chromosomal instability.
Discrete	cancer — lymphoid neoplasias — A-T
Continuous	lymphoid neoplasias
Hybrid	None
Case 2	**The predicted entities**
Gold	These clustered in the region corresponding to the kinase domain, which is highly conserved in ATM - related proteins in mouse, yeast and Drosophila.
Discrete	None
Continuous	Drosophila
Hybrid	proteins — mouse — yeast — Drosophila

and hybrid queries incorrectly detect false positives. Upon analysis, it becomes apparent that the continuous query possesses limited prior knowledge, resulting in its failure to correctly identify or recognize entities. The hybrid query incorporates some prior knowledge but lacks explicit label information, leading to identification of other types of entities in Case 2 such as proteins/genes (ATM-related proteins) and species (mouse, yeast, Drosophila).

6 Conclusion and Future Work

In this paper, we present a simple yet effective approach to machine reading comprehension. Our query template is designed to better leverage the knowledge in pre-trained language model and facilitate knowledge transfer between source and target tasks. Additionally, our positive enhanced loss function further boosts model performance. This method yields significant improvements in low-resource settings and even outperforms state-of-the-art methods in challenging biomedical domains. Moving forward, we plan to explore machine reading comprehension techniques across various domains with limited resources while also refining our query design.

References

1. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners. In: Proceedings of the 34th International Conference

on Neural Information Processing Systems. NIPS'20, Curran Associates Inc., Red Hook, NY, USA (2020)
2. Chen, X., Li, L., Deng, S., Tan, C., Xu, C., Huang, F., Si, L., Chen, H., Zhang, N.: LightNER: A lightweight tuning paradigm for low-resource NER via pluggable prompting. In: Proceedings of the 29th International Conference on Computational Linguistics. pp. 2374–2387. Gyeongju, Republic of Korea (Oct 2022)
3. Chiu, J.P., Nichols, E.: Named entity recognition with bidirectional lstm-cnns. Transactions of the association for computational linguistics **4**, 357–370 (2016)
4. Cui, L., Wu, Y., Liu, J., Yang, S., Zhang, Y.: Template-based named entity recognition using BART. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. pp. 1835–1845. Association for Computational Linguistics, Online (Aug 2021)
5. Das, S.S.S., Katiyar, A., Passonneau, R., Zhang, R.: CONTaiNER: Few-shot named entity recognition via contrastive learning. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 6338–6353. Association for Computational Linguistics, Dublin, Ireland (May 2022)
6. Ding, N., Xu, G., Chen, Y., Wang, X., Han, X., Xie, P., Zheng, H., Liu, Z.: Few-NERD: A few-shot named entity recognition dataset. 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. 3198–3213. Association for Computational Linguistics, Online (Aug 2021)
7. Fritzler, A., Logacheva, V., Kretov, M.: Few-shot classification in named entity recognition task. In: Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing. pp. 993–1000 (2019)
8. Gao, T., Fisch, A., Chen, D.: Making pre-trained language models better few-shot learners. 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. 3816–3830. Association for Computational Linguistics, Online (Aug 2021). https://doi.org/10.18653/v1/2021.acl-long.295, https://aclanthology.org/2021.acl-long.295
9. Han, X., Zhao, W., Ding, N., Liu, Z., Sun, M.: Ptr: Prompt tuning with rules for text classification. ArXiv **abs/2105.11259** (2021), https://api.semanticscholar.org/CorpusID:235166723
10. Hou, Y., Che, W., Lai, Y., Zhou, Z., Liu, Y., Liu, H., Liu, T.: Few-shot slot tagging with collapsed dependency transfer and label-enhanced task-adaptive projection network. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 1381–1393. Association for Computational Linguistics, Online (Jul 2020)
11. Huang, Y., He, K., Wang, Y., Zhang, X., Gong, T., Mao, R., Li, C.: COPNER: Contrastive learning with prompt guiding for few-shot named entity recognition. In: Proceedings of the 29th International Conference on Computational Linguistics. pp. 2515–2527. International Committee on Computational Linguistics, Gyeongju, Republic of Korea (Oct 2022)
12. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C.H., Kang, J.: BioBERT: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics **36**(4), 1234–1240 (09 2019)
13. Li, J., Shang, S., Shao, L.: Metaner: Named entity recognition with meta-learning. In: Proceedings of The Web Conference 2020. p. 429–440. WWW '20, Association for Computing Machinery, New York, NY, USA (2020)

14. Li, J., Sun, A., Han, J., Li, C.: A survey on deep learning for named entity recognition. IEEE Transactions on Knowledge and Data Engineering **34**(1), 50–70 (2020)
15. Li, X.L., Liang, P.: Prefix-tuning: Optimizing continuous prompts for generation. 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. 4582–4597. Association for Computational Linguistics, Online (Aug 2021)
16. Liu, S., Zhang, X., Zhang, S., Wang, H., Zhang, W.: Neural machine reading comprehension: Methods and trends. ArXiv **abs/1907.01118** (2019)
17. Ma, R., Zhou, X., Gui, T., Tan, Y., Li, L., Zhang, Q., Huang, X.: Template-free prompt tuning for few-shot NER. In: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 5721–5732. Association for Computational Linguistics, Seattle, United States (Jul 2022)
18. Nguyen, N.D., Du, L., Buntine, W., Chen, C., Beare, R.: Hardness-guided domain adaptation to recognise biomedical named entities under low-resource scenarios. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. pp. 4063–4071. Abu Dhabi, United Arab Emirates (Dec 2022)
19. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language models are unsupervised multitask learners (2019)
20. Schick, T., Schütze, H.: Exploiting cloze-questions for few-shot text classification and natural language inference. In: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume. pp. 255–269. Association for Computational Linguistics, Online (Apr 2021)
21. Schick, T., Schütze, H.: It's not just size that matters: Small language models are also few-shot learners. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 2339–2352. Association for Computational Linguistics, Online (Jun 2021). https://doi.org/10.18653/v1/2021.naacl-main.185, https://aclanthology.org/2021.naacl-main.185
22. Shin, T., Razeghi, Y., Logan IV, R.L., Wallace, E., Singh, S.: AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 4222–4235. Association for Computational Linguistics, Online (Nov 2020). https://doi.org/10.18653/v1/2020.emnlp-main.346, https://aclanthology.org/2020.emnlp-main.346
23. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. p. 4080–4090. NIPS'17, Curran Associates Inc., Red Hook, NY, USA (2017)
24. Wiseman, S., Stratos, K.: Label-agnostic sequence labeling by copying nearest neighbors. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 5363–5369. Association for Computational Linguistics, Florence, Italy (Jul 2019)
25. Yadav, V., Bethard, S.: A survey on recent advances in named entity recognition from deep learning models. In: Proceedings of the 27th International Conference on Computational Linguistics. pp. 2145–2158. Association for Computational Linguistics, Santa Fe, New Mexico, USA (Aug 2018)
26. Yang, Y., Katiyar, A.: Simple and effective few-shot named entity recognition with structured nearest neighbor learning. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 6365–6375. Association for Computational Linguistics, Online (Nov 2020)

27. Zhang, Y., Fei, H., Li, D., Li, P.: PromptGen: Automatically generate prompts using generative models. In: Findings of the Association for Computational Linguistics: NAACL 2022. pp. 30–37. Association for Computational Linguistics, Seattle, United States (Jul 2022). https://doi.org/10.18653/v1/2022.findings-naacl.3, https://aclanthology.org/2022.findings-naacl.3