A Multi-Task Semantic Decomposition Framework with Task-specific Pre-training for Few-Shot NER

Guanting Dong*	Zechen Wang*	Jinxu Zhao*	Gang Zhao		
dongguanting@bupt.edu.cn	shenshui@bupt.edu.cn	zhaojinxu@bupt.edu.cn	zhaogang@bupt.edu.cn		
Beijing University of Posts and Telecommunication	Beijing University of Posts and Telecommunication	Beijing University of Posts and Telecommunication	Beijing University of Posts and Telecommunication		
Beijing, China	Beijing, China	Beijing, China	Beijing, China		
Daichi Guo	Dayuan Fu	Tingfeng Hui	Chen Zeng		
guodaichi@bupt.edu.cn	fdy@bupt.edu.cn	huitingfeng@bupt.edu.cn	chenzeng@bupt.edu.cn		
Beijing University of Posts and Telecommunication					
Beijing, China	Beijing, China	Beijing, China	Beijing, China		
Keqing He	Xuefeng Li	Liwen Wang	Xinyue Cui		
hekeqing@meituan.com	lixuefeng@bupt.edu.cn	w_liwen@bupt.edu.cn	tracycui@bupt.edu.cn		
Meituan Group, Beijing Beijing, China	Beijing University of Posts and Telecommunication	Beijing University of Posts and Telecommunication	Beijing University of Posts and Telecommunication		
2013116, 011114	Beijing, China	Beijing, China	Beijing, China		

Weiran Xu[†]

xuweiran@bupt.edu.cn

Beijing University of Posts and Telecommunication

Beijing, China

ABSTRACT

The objective of few-shot named entity recognition is to identify named entities with limited labeled instances. Previous works have primarily focused on optimizing the traditional token-wise classification framework, while neglecting the exploration of information based on NER data characteristics. To address this issue, we propose a Multi-Task Semantic Decomposition Framework via Joint Task-specific **P**re-training (**MSDP**) for few-shot NER. Drawing inspiration from demonstration-based and contrastive learning,

CIKM '23, October 21-25, 2023, Birmingham, United Kingdom

we introduce two novel pre-training tasks: Demonstration-based Masked Language Modeling (MLM) and Class Contrastive Discrimination. These tasks effectively incorporate entity boundary information and enhance entity representation in Pre-trained Language Models (PLMs). In the downstream main task, we introduce a multitask joint optimization framework with the semantic decomposing method, which facilitates the model to integrate two different semantic information for entity classification. Experimental results of two few-shot NER benchmarks demonstrate that MSDP consistently outperforms strong baselines by a large margin. Extensive analyses validate the effectiveness and generalization of MSDP.

CCS CONCEPTS

Computing methodologies → Artificial intelligence;
 Natural language processing → Information extraction.

KEYWORDS

Few-shot NER, Multi-Task, Semantic Decomposition, Pre-training

^{*}The first three authors contribute equally.

 $^{^\}dagger$ Weiran Xu is the corresponding author

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

^{© 2023} Copyright held by the owner/author(s). Publication rights licensed to ACM. ACM ISBN 979-8-4007-0124-5/23/10...\$15.00 https://doi.org/10.1145/3583780.3614766

ACM Reference Format:

Guanting Dong, Zechen Wang, Jinxu Zhao, Gang Zhao, Daichi Guo, Dayuan Fu, Tingfeng Hui, Chen Zeng, Keqing He, Xuefeng Li, Liwen Wang, Xinyue Cui, and Weiran Xu. 2023. A Multi-Task Semantic Decomposition Framework with Task-specific Pre-training for Few-Shot NER. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM '23), October 21–25, 2023, Birmingham, United Kingdom.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3583780.3614766

1 INTRODUCTION

Named entity recognition (NER) plays a crucial role in Natural Language Understanding applications by identifying consecutive segments of text and assigning them to predefined categories [25, 49, 50]. Recent advancements in deep neural architectures have demonstrated exceptional performance in fully supervised NER tasks [10, 34, 51]. However, the collection of annotated data for practical applications incurs significant expenses and poses inflexibility challenges. As a result, the research community has increasingly focused on few-shot NER task, which seeks to identify entities with only a few labeled instances, attracting substantial interest in recent years.

Previous few-shot NER methods [11, 18, 28, 46, 70, 81] generally formulate the task as a sequence labeling task based on prototypical networks [59]. These approaches employ prototypes to represent each class based on labeled instances and utilize the nearest neighbor method for NER. However, these models only capture the surface mapping between entity and class, making them vulnerable to disturbances caused by non-entity tokens (i.e. "O" class) [58, 66]. To alleviate this issue, a branch of two-stage methods [47, 58, 66, 69] arise to decouple NER into two separate processes, including span extraction and entity classification. Despite the above achievement, there are still two remaining problems. (1) Span Over-prediction: as shown in Figure 1, previous span-based works suffer from the span over-prediction problem [27, 80]. Specifically, the model will extract redundant candidate spans in addition to predicting the correct spans. The reason for the above phenomenon is that it is difficult for PLMs to learn reliable boundary information between entities and non-entities due to insufficient data. As a result, PLMs tend to give similar candidate spans high probability scores or even be over-confident about their predictions [23]. (2) Prototype Classification Disarray: Previous prototype-based methods directly utilize the mean value of entity representations to compute prototype embedding, leading to the classification accuracy heavily relies on the quality of entity representations. Unfortunately, PLMs often face the issue of semantic space collapse, where different classes of entity representations are closely distributed, especially for entities within the same sentences. Figure 1 illustrates that different classes of entities interfere with each other under the interaction of the selfattention mechanism, causing close or even overlapping prototypes distribution in the semantic space(e.g. "LOC" prototype overlapping with "PER" prototype). The model finally suffers from performance degradation due to class confusion. Therefore, we urgently need to design a method introducing different aspects of information to alleviate the above problems, which facilitates techniques of fewshot NER to be widely applied in realistic task-oriented dialogue scenarios.

Problem in stage 1 : Span Over-prediction



Figure 1: The illustration of the baseline model suffering from span over-prediction (upper) and Prototype Classification Disarray (down) problem in few-shot NER.

To tackle these limitations, we propose a Multi-Task Semantic Decomposition Framework via Joint Task-specific Pre-training (MSDP), which guides PLMs to capture reliable entity boundary information and better entity representations of different classes. For the pre-training stage, inspired by demonstration-based learning [19] and contrastive learning [9], we introduce two novel taskspecific pre-training tasks according to the data characteristics of NER (entity-label pairs): Demonstration-based MLM, in which we design three kinds of demonstrations containing entity boundary information and entity label pair information. PLMs will implicitly learn the above information during predicting label words for [MASK]; Class Contrastive Discrimination, in which we use contrastive learning to better discriminate different classes of entity representations by constructing positive, negative, and hard negative samples. Through the joint optimization of above fine-grained pre-training tasks, PLMs can effectively alleviate the two remaining problems.

For downstream few-shot NER, we follow the two-stage framework [47, 65] including span extraction and entity classification, and initialize them with the pre-trained parameters. Different from previous methods, we employ a multi-task joint optimization framework and utilize different masking strategies to decompose class-oriented prototypes and contextual fusion prototypes. The purpose of our design is to assist the model to integrate different semantic information for classification, which further alleviates the prototype classification disarray problem. We conduct extensive experiments over two widely-used benchmarks, including Few-NERD [14] and CrossNER [28]. Results show that our method consistently outperforms state-of-the-art baselines by a large margin. In addition, we introduce detailed experimental analyses to further verify the effectiveness of our method. Our contributions are three-fold:

1) To the best of our knowledge, we are the first to introduce a multitask joint optimization framework with the semantic decomposing method into Few-Shot NER task.

2) we futher propose two task-specific pre-training tasks via demonstration and contrastive learning, namely demonstration-based MLM and class contrastive discrimination, for effectively injecting entity boundary information and better entity representation into PLMs.



Figure 2: The illustration of two task-specific pre-training tasks.

3) Experiments on two widely-used few-shot NER benchmarks show that our framework achieves superior performance over previous state-of-the-art methods. Extensive analyses further validate the effectiveness and generalization of MSDP. Our source codes and datasets are available at Github¹ for further comparisons.

2 RELATED WORK

2.1 Few-shot NER.

Few-shot NER aims to enhance the performance of model identifying and classifying entities with only little annotated data [7, 21, 30, 31, 38, 42–45, 62, 77]. For few-shot NER, a series of approaches have been proposed to learn the representation of entities in the semantic space, i.e. prototypical learning [59], margin-based learning [36] and contrastive learning [20, 24, 37]. Existing approaches can be divided into two kinds, i.e., one-stage [11, 28, 59, 81] and two-stage [16, 47, 58, 69]. Generally, the methods in the kind of one-stage typically categorize the entity type by token-level metric learning. In contrast, two-stage mainly focuses on two training stages consisting of entity span extraction and mention type classification.

2.2 Task-specific pre-training Models.

Pre-trained language models have been applied as an integral component in modern NLP systems for effectively improving downstream tasks [13, 40, 52, 54, 55, 71, 74, 76]. Due to the underlying discrepancies between the language modeling and downstream tasks, task-specific pre-training methods have been proposed to further boost the task performance, such as SciBERT [5], VideoBERT [61], DialoGPT [75], PLATO [4], Code-BERT [17], ToD-BERT [68] and VL-BERT [60]. However, most studies in the field of few-shot NER use MLM and other approaches for Data Augmentation [15, 29, 79]. Although DictBERT [8], NER-BERT [41] and others have conducted pre-training, their methods are too generalized to adapt to the structured data features of NER or propose optimization for specific problems. Therefore, we designed demonstration-based learning pre-training and contrastive learning pre-training for NER tasks to improve the performance of the model.

2.3 Demonstration-based learning

Demonstrations are first introduced by the GPT series [6, 55], where a few examples are sampled from training data and transformed with templates into appropriately-filled prompts. Based on the task reformulation and whether the parameters are updated, the existing demonstration-based learning research can be broadly divided into three categories: In-context Learning [6, 48, 67, 78], Prompt-based Fine-tuning [39], Classifier-based Fine-tuning [35, 72]. However, these approaches mainly adopt demonstration-based learning in the fine-tuning that cannot make full use of the effect of demonstrationbased learning. Different from them, we use demonstration-based learning in the pre-training stage that can better capture the entity boundary information to solve the multiple-span prediction problem.

3 METHOD

3.1 Task-Specific Pre-training

The performance of few-shot NER depends heavily on the different aspects of information from entity label pairs. as shown in Figure 2, we introduce two novel pre-training tasks: 1) demonstration-based MLM and 2) contrastive entity discrimination, to learn the different aspects of knowledge.

Demonstration-based MLM. We follow the design of masked language modeling (MLM) in BERT [13] and integrate the idea of demonstration-based learning on this basis. In order to prompt PLMs to figure out the boundary between the entity and noneentity, we propose three different demonstrations which are shown in Figure 2(a):

1) Label demonstration (LD): We let D_{train} denote the train dataset. For each input *x* in D_{train} , we extract the entity label pair (e, l) belonging to *x* and then concatenate them behind input *x* in form of the simple template $T = \{e \text{ is } l\}$. Different demonstrations are separated by *[SEP]*.

2) Retrieved demonstration (RD): Given an entity type label set $L = \{l_1, ..., l_{|L|}\}$, we first enumerate all the entities in D_{train} and create a mapping $M = \{l_i : [e_1, ..., e_n] \mid l_i \in L\}$ between l and the

¹https://github.com/dongguanting/MSDP-Fewshot-NER



Figure 3: The overall architecture of our proposed MSDP framework

corresponding list of entities. Further, we randomly select *K* entity label pairs (e, l) from the mapping *M* according to the label set L_x appearing in input *x*, which aims at introducing rich entity label pair information to prompt the model. Furthermore, we concatenate them behind label demonstration with template $T = \{e \text{ is } l\}$.

3) Negative demonstration (ND): We randomly select *K* noneentities that are easily confused by the model from input *x* to construct negative sample pairs (e_{none} , O), and then concatenate them behind retrieved demonstrations with template $T' = \{e_{none} \text{ is } O\}$. Therefore, our training samples can be formulated as:

$$[CLS] x [SEP] LD [SEP] RD [SEP] ND$$
(1)

After constructing the training set, we randomly randomly replace N entities or labels with mask symbols or labels in the demonstration with the special [*MASK*] symbol², and then try to recover them. If entity *e* consists of multiple tokens, all of the component tokens will be masked. Hence, the loss function of the MLM is:

$$L_{mlm} = -\sum_{m=1}^{M} \log P(\boldsymbol{x_m})$$
(2)

where *M* is the total number of masked tokens and $P(x_m)$ is the predicted probability of the token x_m over the vocabulary size.

Class Contrastive Discrimination. To better discriminate different classes of entity representations in semantic space, we introduce class contrastive discrimination. Specifically, we construct positive (negative) samples as follows:

Given an input x that contains K entities, we employ the following procedure to generate positive and negative samples. For positive samples, we replace these K entities with their corresponding label mentions to create K positive samples for each input utterance. For negative samples, we select samples from other classes within

the batch. Additionally, we replace all entities with irrelevant label mentions to construct a hard negative sample for each instance that is easily confused by the model. These hard negative samples are then included in the negative sample set. Figure 2 illustrates the corresponding positive and negative samples as depicted in our experiment.

The representations of the original, positive, and negative samples are denoted by h_o , h_p , and h_n , respectively. To account for multiple positive samples, we adopt the supervised contrastive learning (SCL) objective [32], which aims to minimize the distance between the original samples h_o and their semantically similar positive samples h_p , while maximizing the distance between h_o and 2 samples: the negative samples h_n and the hard negative samples h_{hn} . The formulation of L_{SCL} is as follow:

$$\mathcal{L}_{SCL} = \frac{-1}{N} \sum_{i=1}^{N} \frac{1}{N_{y_i}} \sum_{j=1}^{N_{y_i}} \sum_{k=1}^{N_{y_i}} \mathbb{I}_{y_{ij}=y_{ik}} e^{sim(h_{oij},h_{pik})/\tau}$$
(3)
$$\log \frac{e^{sim(h_{oij},h_{pik})/\tau}}{\sum_{l=1}^{N} (\mathbb{I}_{j\neq l}) e^{sim(h_{oij},h_{nl})/\tau} + e^{sim(h_{oij},h_{nnij})/\tau}}$$

where *N* and N_{y_i} denote the number of total examples in the batch and positive samples. τ is a temperature hyperparameter and $sim(h_1, h_2)$ is cosine similarity. **1** is an indicator function.

We sum the demonstration-base MLM task loss and the class contrastive discrimination task loss, and finally obtain the overall loss function L:

$$L = \alpha L_{mlm} + (1 - \alpha) L_{SCL} \tag{4}$$

where L_{mlm} and L_c denote the loss functions of the two tasks. In our experiments, we set $\alpha = 0.6$.

²N is an empirical hyperparameter, which is set to 4.

A Multi-Task Semantic Decomposition Framework with Task-specific Pre-training for Few-Shot NER

3.2 Downstream Few-shot NER

After the pre-training stage, our model initially learns different aspects of information. In this section, We formally present the notations and the techniques of our proposed MSDP in the finetuning stage. Figure 3 illustrates the overall framework, which is composed of two steps: span extraction and entity classification.

3.2.1 **Notations**. We denote the train and test sets by \mathcal{D}_{train} and \mathcal{D}_{test} , respectively. Both of them have the form of meta-learning datasets. The dataset consists of multiple episodes of data, and each episode of data $\mathcal{E} = (S, Q)$ consists of a support set S and a query set Q. A sample (X, \mathcal{Y}) in the support set or query set consists of the input sentence X and the label set $\mathcal{Y} = \{(s_j, e_j, y_j)\}_{j=1}^N$ (N denoting the number of spans, s_j and e_j denoting the start and end positions of the *j*-th span, y_j denoting the category of the *j*-th span). h denotes the hidden representation obtained by encoding the input text.

3.2.2 **Span Extractor**. The span extractor aims to detect all entity spans. We initialize encoder with pre-trained parameters to encode the input sentence as a hidden representation h, and calculate attention scores between each token representation to judge the start token and end token of the entity span. Following previous works [47, 65], we use span-based cross-entropy as the loss function to optimise our encoder. We first design the weight matrixes $W_q/W_k/W_v$ of values q/k/v and bias b_q/b_k for the attention mechanism, and then compute the attention score of the *i*-th and *j*-th token, using formula: $f(i, j) = q_i^T k_j + W_v(\mathbf{h}_i + \mathbf{h}_j)$. $\Omega_{i,j}$ indicates whether the span bounded by *i*, *j* is an entity. Therefore, the span-based cross-entropy can be expressed as:

$$\mathcal{L}_{span} = \log(1 + \sum_{1 \le i < j \le L} \exp((-1)^{\Omega_{i,j}} f(i,j))$$
(5)

3.2.3 **Entity Classification**. In the second stage, we classify the entity spans extracted in the first stage. Different from the previous methods only computing the original prototype, we further decompose class-oriented prototypes and contextual fusion prototypes by two masking strategies, which introduce different information to assist in the classification task, thus alleviating the prototype classification disarray problem.

1) Semantic masking strategies. Firstly, we introduce two novel semantic masking strategies for the subsequent construction of semantic decomposing prototypes.

- Class-oriented Masking: Given an input sentence $X = \{x_1, x_2, \dots, x_L\}$, we replace all the entity spans in X whose labels that are not y with [MASK] tokens to obtain class y specific input X_{cs}^y , thereby forcing the model to focus on the information of specific class by shielding the interference of other entities. For example, as shown in Figure 3, we replace the "school" entity of the "LOC" class and the "Sunday" entity of the "DAY" class with [MASK] tokens to obtain "PER" class-oriented input *Mike wants to go to [MASK] on [MASK]*.
- **Contextual Fusion Masking**: we replace all the entities in a sentence with [MASK] tokens, thus allowing the model to focus more on contextual fusion information. As the example

sentence in Figure 3, we mask all entities to obtain $X_{\text{ctx}} = [MASK]$ wants to go to [MASK] on [MASK].

2) Prototype Constructing. After decomposing two types of inputs with different information, we construct original prototype and two extra prototypes for each class in entity classification stage (The upper right corner of Figure 3)

For original prototype, we add up the representations of the start token and the end token of an entity span as the span boundary representation:

$$\boldsymbol{u}_j = \boldsymbol{h}_{s_j} + \boldsymbol{h}_{e_j} \tag{6}$$

where u_j is the representation of the *j*-th span in the sentence, h_i denote the representation of the *i*-th token in the sentence. s_j and e_j are the start and end positions of the *j*-th span respectively.

For class-oriented prototype, we perform a class-oriented masking strategy for class t on X to obtain X_{cs}^{t} , and compute a span representation u_{i}^{cs} in X_{cs}^{t} according to equation 6.

For contextual fusion prototype, we perform all the entity-masking strategy on the original sentence X to obtain X_{ctx} and then compute the span representation u_j^{ctx} by averaging the representations of all tokens as follow:

$$\boldsymbol{u}_{j}^{\text{ctx}} = \frac{1}{L} \sum_{i=1}^{L} \boldsymbol{h}_{i}$$
(7)

where *L* denotes the number of tokens in X_{ctx} .

Afterwards, we construct three different prototypes vectors by averaging the representations of all entities of the same class in the support set:

$$c_t = \frac{\sum_{(X, \mathcal{Y}) \in \mathcal{S}} \sum_{(s_j, e_j, y_j) \in \mathcal{Y}} \mathbb{I}(y_j = t) \boldsymbol{u}}{\sum_{(X, \mathcal{Y}) \in \mathcal{S}} \sum_{(s_i, e_i, y_j) \in \mathcal{Y}} \mathbb{I}(y_j = t)}$$
(8)

where $\mathbb{I}(\cdot)$ is the indicator function; u can be replaced with u_j , u_j^{cs} and u_j^{ctx} to calculate three semantic prototypes separately.

After constructing three different semantic prototypes, we use a metric-based approach for classification and optimize the parameters of the model based on the basis of distance between entity representations and class prototypes:

$$\mathcal{L}_{cls} = \sum_{(X,\mathcal{Y})\in\mathcal{S}} \sum_{(s_j,e_j,y_j)\in\mathcal{Y}} -\log p(y_j|s_j,e_j)$$
(9)

where

$$p(y_j|s_j, e_j) = \operatorname{softmax}(-d(u_j, c_{y_j}))$$
(10)

is the probability distribution. We use cosine similarity as the distance function $d(\cdot, \cdot)$.

3.3 Training and Inference of MSDP

We first perform two task-specific pre-trainings to learn reliable entity boundary information and entity representations of different classes. For fine-tuning, we initialize the BERT encoder with pretrained parameters for the few-shot NER task.

In the downstream training phase, given the training set $D_{train} = (S, Q)$, we compute \mathcal{L}_{span} and three types of prototypes on the support set S, \mathcal{L}_{cls} on the query set Q, and train the two optimization objectives jointly. Following SpanProto [65], we only optimize

Dataset	Domain	# Sentences	# Classes
Few-NERD	Wikipedia	188.2k	66
CoNLL03	News	20.7k	4
GUM	Wiki	3.5k	11
WNUT	Social	5.6k	6
OntoNotes	Mixed	159.6k	18

Table 1: Evaluation dataset statistics.

the \mathcal{L}_{span} objective in the first *T* steps, and jointly optimize both \mathcal{L}_{span} and \mathcal{L}_{cls} after *T* steps.

In the testing phase, given an episode $\mathcal{E} = (\mathcal{S}, Q) \in \mathcal{D}_{test}$, we construct prototypes on the support set \mathcal{S} and perform span detection on the query set Q. Then we calculate the distance between the extracted spans and each class prototype for classification. Note that we only utilize original prototypes during inference.

4 EXPERIMENT

4.1 Datasets

Table 1 shows the dataset statistics of original data for constructing few-shot episodes. We evaluate our method on two widely used few-shot benchmarks Few-NERD [14] and CrossNER [45].

Few-NERD: Few-NERD is annotated with 8 coarse-grained and 66 fine-grained entity types, which consists of two few-shot settings (Intra, and Inter). In the Intra setting, all entities in the training set, development set, and testing set belong to different coarse-grained types. In contrast, in the Inter setting, only the fine-grained entity types are mutually disjoint in different datasets. we use episodes released by Ding et al. which contains 20,000 episodes for training, 1,000 episodes for validation, and 5,000 episodes for testing. Each episode is an *N*-way $K \sim 2K$ -shot few-shot task.

CrossNER: CrossNER contains four domains from CoNLL-2003 [57](News), GUM [73] (Wiki), WNUT-2017 [12] (Social), and Ontonotes [53](Mixed). We randomly select two domains for training, one for validation, and the remaining for testing. We use public episodes constructed by Hou et al. .

4.2 Baselines

For the baselines, we choose multiple strong approaches from the paradigms of one-stage and two-stage. 1) One-stage NER paradigms: ProtoBERT [59], StructShot [70], NNShot [70], CONTAINER [11] and LTapNet+CDT [28]. 2) Two-stage paradigm: ESD [66], MAML-ProtoNet [47] and SpanProto [65]. Due to the space limitation, More details of these baselines and implementations are illustrated as follow:

- **SimBERT** [28] applies BERT without any finetuning as the embedding function, then assigns each token's label by retrieving the most similar token in the support set.
- **ProtoBERT** [18] uses a token- level prototypical network [59] which represents each class by averaging token representations with the same label, then the label of each token in the query set is decided by its nearest class prototype.

- MatchingBERT [64] is similar to ProtoBERT except that it calculates the similarity between query instances and support instances instead of class prototypes.
- L-TapNet+CDT [28] enhances TapNet with pair-wise embedding, label semantic, and CDT transition mechanism.
- **NNShot** [70] pretrains BERT for token embedding by conventional classification for training, and a token-level nearest neighbor method is used at testing.
- **StructShot** [70] improves NNshot by using an abstract transition probability for Viterbi decoding at testing.
- ESD [66] is a span-level metric learning-based method. It enhances the prototypical network by using inter- and cross-span attention for better span representation and designs multiple prototypes for O label.
- **TransferBERT** [28] trains a token- level BERT classifier, then finetunes task-specific linear classifier on the support set at test time.
- **CONTAINER** [11] uses token- level contrastive learning for training BERT as a token embedding function, then finetunes the BERT on the support set and applys the nearest neighbor method at inference time.
- **DecomMeta** [47] trains the span detector by introducing the model-agnostic meta-learning (MAML) algorithm and uses MAML-enhanced prototypical networks to find a good embedding space.
- **SpanProto** [65] transforms the sequential tags into a global boundary matrix in the span extraction stage and performs prototypical learning with a margin-based loss in the mentioned classification stage.

4.3 Implementation Detail

For the upstream work, we use BERT-base-uncased [13] from HuggingFace as the backbone. In two pre-training settings, we set the batch size of BERT to 8 and the pre-training takes an average of 12 hours for 5 epochs. The corresponding learning rates are set to 1e-5. We set the number (K) of the retrieved demonstrations to 5 and the negative demonstration to 3. We set temperature hyperparameter τ_1 to 0.5. In addition, our upstream pre-training corpus is aligned with the downstream task, which means no additional data will be introduced in the pre-training stage. For instance, if the downstream few-shot NER experiment is conducted on the Inter 5 way 1-2 shot of Few-NERD, the pre-training data is the training set of Inter 5 way 1-2 shot.

For the downstream work, we adopt the standard N-way K-shot setting [14] and align the task definition with previous work [47]. We choose Adam [33] as the optimizer with a learning rate of 3e-5. The warm-up rate is set to 0.1. The max sequence length we set is 64 and the batch size is set to 4. The training steps T and T' are set as 2000 and 200, respectively. For all the experiments, we train and test our model on the 3090Ti GPU. It takes an average of 5 hours to run with 3 epochs on the training dataset.

A Multi-Task Semantic Decomposition Framework with Task-specific Pre-training for Few-Shot NER

CIKM '23, October 21-25, 2023, Birmingham, United Kingdom

Paradigms Models		Intra				Inter					
		1~2-shot		5~10-shot			1~2-shot		5~10-shot		
		5 way	10 way	5 way	10 way	Avg.	5 way	10 way	5 way	10 way	Avg.
	ProtoBERT	$23.45_{\pm 0.92}$	$19.76_{\pm 0.59}$	$41.93_{\pm0.55}$	$34.61_{\pm 0.59}$	29.94	$44.44_{\pm 0.11}$	$39.09_{\pm 0.87}$	$58.80_{\pm 1.42}$	$53.97_{\pm 0.38}$	49.08
	NNShot	$31.01_{\pm 1.21}$	21.88 ± 0.23	$35.74_{\pm 2.36}$	$27.67_{\pm 1.06}$	29.08	$54.29_{\pm 0.40}$	$46.98_{\pm 1.96}$	$50.56_{\pm 3.33}$	$50.00_{\pm 0.36}$	50.46
One-stage	StructShot	$35.92_{\pm 0.69}$	$25.38_{\pm 0.84}$	$38.83_{\pm 1.72}$	$26.39_{\pm 2.59}$	31.63	$57.33_{\pm 0.53}$	$49.46_{\pm 0.53}$	$57.16_{\pm 2.09}$	$49.39_{\pm 1.77}$	53.34
	CONTaiNER	40.43	33.84	53.70	47.49	43.87	55.95	48.35	61.83	57.12	55.81
	ESD	$41.44_{\pm 1.16}$	$32.29_{\pm 1.10}$	$50.68_{\pm 0.94}$	$42.92_{\pm 0.75}$	41.83	$66.46_{\pm 0.49}$	$59.95_{\pm 0.69}$	$74.14_{\pm 0.80}$	$67.91_{\pm 1.41}$	67.12
— .	DecomMeta	$52.04_{\pm 0.44}$	$43.50_{\pm 0.59}$	$63.23_{\pm 0.45}$	$56.84_{\pm 0.14}$	53.9	$68.77_{\pm 0.24}$	$63.26_{\pm 0.40}$	$71.62_{\pm 0.16}$	$68.32_{\pm 0.10}$	67.99
1wo-stage	SpanProto	$54.49_{\pm 0.39}$	$45.39_{\pm 0.72}$	$65.89_{\pm 0.82}$	$59.37_{\pm 0.47}$	56.29	$73.36_{\pm 0.18}$	$66.26_{\pm 0.33}$	$75.19_{\pm 0.77}$	$70.39_{\pm 0.63}$	71.3
	MSDP	$56.35_{\pm0.28}$	$47.13_{\pm 0.69}$	$66.80_{\pm0.78}$	$64.69_{\pm0.51}$	58.74	76.86 _{±0.22}	$69.78_{\pm 0.31}$	$84.78_{\pm0.69}$	$81.50_{\pm 0.71}$	78.23

Table 2: F1 scores with standard deviations on Few-NERD for both inter and intra settings.

		1		-			1				
Paradigms Models		1-shot				5-shot					
		CONLL-03	GUM	WNUT-17	OntoNotes	Avg.	CONLL-03	GUM	WNUT-17	OntoNotes	Avg.
One-stage	Matching Network ProtoBERT L-TapNet+CDT	$ \begin{vmatrix} 19.50_{\pm 0.35} \\ 32.49_{\pm 2.01} \\ 44.30_{\pm 3.15} \end{vmatrix} $	$\begin{array}{c} 4.73_{\pm 0.16} \\ 3.89_{\pm 0.24} \\ 12.04_{\pm 0.65} \end{array}$	$\begin{array}{c} 17.23_{\pm 2.75} \\ 10.68_{\pm 1.40} \\ 20.80_{\pm 1.06} \end{array}$	$\begin{array}{c} 15.06_{\pm 1.61} \\ 6.67_{\pm 0.46} \\ 15.17_{\pm 1.25} \end{array}$	14.13 13.43 23.08	$\begin{array}{c} 19.85_{\pm 0.74} \\ 50.06_{\pm 1.57} \\ 45.35_{\pm 2.67} \end{array}$	$\begin{array}{c} 5.58_{\pm 0.23} \\ 9.54_{\pm 0.44} \\ 11.65_{\pm 2.34} \end{array}$	$\begin{array}{c} 6.61_{\pm 1.75} \\ 17.26_{\pm 2.65} \\ 23.30_{\pm 2.80} \end{array}$	$\begin{array}{c} 8.08_{\pm 0.47} \\ 13.59_{\pm 1.61} \\ 20.95_{\pm 2.81} \end{array}$	10.03 22.61 25.31
Two-stage	DecomMeta SpanProto MSDP	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 17.54_{\pm 0.98} \\ 19.92_{\pm 0.53} \\ \textbf{21.88}_{\pm 0.29} \end{array}$	$\begin{array}{c} 25.14_{\pm 0.24} \\ 28.31_{\pm 0.61} \\ \textbf{30.10}_{\pm 0.56} \end{array}$	$\begin{array}{c} 34.13_{\pm 0.92} \\ 36.41_{\pm 0.73} \\ \textbf{38.05}_{\pm 0.88} \end{array}$	30.73 33.09 34.79	$\begin{array}{c} 58.18_{\pm 0.87} \\ 61.88_{\pm 0.83} \\ \textbf{63.98_{\pm 0.80}} \end{array}$	$\begin{array}{c} 31.36_{\pm 0.91} \\ 35.12_{\pm 0.88} \\ \textbf{36.53}_{\pm 0.81} \end{array}$	$\begin{array}{c} 31.02_{\pm 1.28} \\ 33.94_{\pm 0.50} \\ \textbf{35.61}_{\pm 0.72} \end{array}$	$\begin{array}{c} 45.55_{\pm 0.90} \\ 48.21_{\pm 0.89} \\ \textbf{49.99}_{\pm 0.95} \end{array}$	41.53 44.79 46.53

Table 3: F1 scores with standard deviations under 1 shot and 5 shot setting on CrossNER.

All experiments are repeated three times with different random seeds under the same settings. All the models are implemented with PyTorch. We will release our code after blind review.

4.4 Main Results

Table 2 and Table 3 report the main results compared with other baselines. We conduct the following comparison and analysis: 1) Our proposed method significantly outperforms all the previous methods in different settings. Specifically, compared with Span-Proto, MSDP achieves a performance improvement on the overall averaged results over Few-NERD Intra by 4.3% and Inter by 9.7%. Meanwhile, MSDP shows a 3.9% increase on CrossNER. Both results demonstrate the effectiveness of MSDP. 2) All methods in the two-stage paradigm perform better than those one-stage methods, which demonstrates the framework advantages of the span-based approach. 3) The overall performance of the Inter scenario is higher than Intra, since all entities in the training set/development set/test set belong to different coarse-grained types in the Intra setting. We still obtain extraordinary improvement in this challenging situation. All the results show that MSDP can adapt to a new domain in which the coarse-grained and fine-grained entity types are both unseen, which highlights the strong transferring ability of our approach.

4.5 Ablation Studies.

We conduct an ablation study to investigate the characteristics of the main components in MSDP. Table 4 shows the ablation results, and "w/o" denotes the model performance without a specific module. We have following observations: 1) The performance of MSDP drops when removing any one component, which suggests

Methods	Few- Intra	NERD Inter	Cross 1-shot	sNER 5-shot	
MSDP	58.49	78.23	34.79	46.53	
w/o Demonstration-based MLM	54.84	75.25	32.57	44.77	
w/o Class Contrastive Discrimination	56.97	74.87	31.66	43.54	
w/o class-oriented Prototype	55.04	74.58	33.14	44.48	
w/o contextual fusion Prototype	56.53	76.15	32.98	45.08	
w/o Joint pre-training tasks	53.58	73.03	30.30	41.51	
w/o Both two semantic prototypes	54.32	73.32	31.74	43.22	

Table 4: The ablation study results (average F1 score %) for Few-NERD and CrossNER.

that every part of the design is necessary 2) Removing any one semantic prototype results in great performance degradation. This is consistent with our conjecture since class-oriented prototypes and contextual fusion prototypes provide relatively orthogonal semantic information from two perspectives. Missing each part will make the semantic space more chaotic and make the classification effect worse. 3) Removing joint pre-training tasks causes obvious performance degradation compared with removing one of them, which indicates that jointly pre-training objectives have a mutually reinforcing effect.

4.6 Effectiveness on Span Over-prediction

Qualitative analysis. Span over-prediction causes the model to extract redundant candidate spans in addition to predicting the correct spans. This phenomenon can be reflected in high recall rate and low precision rate of the span extractor. As shown in

	1	(nter(1	-2shot)	Intra(1-2shot)			
Method	5 v	vay	10 1	way	5 w	/ay	10 ง	way
	Pre.	Rec.	Pre	Rec.	Pre.	Rec.	Pre.	Rec.
MSDP (Base)	71.6	100	75.5	100	72.8	100	73.7	100
+ Pre-training	74.3	100	77.5	100	73.9	100	74.8	100
MSDP (Full)	75.2	100	78.1	100	74.3	100	75.3	100

Table 5: The span extractor performance (average Precision and Recall) on Few-NERD 1-2 shot. MSDP(Base) denotes MSDP without two pre-training and prototypes.

Input	He played one game for the dallas burn of major league soccer.						
Baseline	dallas burn major league soccer dallas 🗱 major league 🗮						
USDP	dallas burn major league soccer						
Input	She was runner-up to tirunesh dibaba at the great manchester run.						
Baseline	tirunesh dibaba the great manchester 🗱 manchester 🗰 run 🗮						
USDP	tirunesh dibaba the great manchester run						

Figure 4: The cases of Few-NERD. Both wrong and correct labels are marked in red and green, respectively.

Table 5, compared with the MSDP(Base), joint pre-training tasks improve the prediction accuracy(2.7% for 5 way and 2.0% for 10 way) while maintaining a high recall rate, which proves that joint pre-training tasks can bring entity boundary information and better representation into PLMs. For MSDP(Full), we unexpectedly find that the semantic decomposing method also improves the precision rate slightly. Since the joint training of span extractor and entity classification, both contextual fusion and class-oriented information also have a positive effect on distinguishing entity boundaries.

Case Study for Span Extractor To further verify the effect of our MSDP on Span Over-prediction, we randomly sample 100 instances from outputs and select two representative cases in figure 4. The baseline model even generates some wrong spans in order to predict all spans while our method does not require such a cost. These cases suggest that MSDP captures more reliable entity-boundary information. In summary, we demonstrate that MSDP can better solve the over-prediction problem from both statistical and sample aspects.

4.7 Performance on Classification Disarray

Error Analysis We follow [65] to conduct error analysis in Table 6. Results show that MSDP outperforms other strong baselines with fewer false positive prediction errors. Specifically, we achieve 9.22% of "FP-Type" when getting 76.86 F1 scores. Meanwhile, this suggests our MSDP obtains the lowest error rate and effectively solves the problem of prototype classification disarray.

Visualization To further explore the effectiveness of MSDP on prototype classification disarray problems, we investigate how our MSDP adjusts the representations in the semantic space. We use 500 5-way 1-shot episodes data from Few-NERD Inter for training,





Figure 5: The t-SNE visualization of the span representations with 500 5-way 1-shot data from Few-NERD Inter for both SpanProto and MSDP. The points with different colors denote the entity span with different types.

and visualize the span representations of 4 types of entity by t-SNE toolkit [63] in three different settings: MSDP(Base), MSDP(with Pre-training) and MSDP(Full). As shown in Figure 5, the span representations of each class are gathered around the corresponding type prototype region. Compared with MSDP(Base), joint pre-training tasks help the model increases the distance between the representations of different classes. For the MSDP(full), the intra-class distance is further compressed due to the optimization of the semantic decomposing method. In this way, both pre-training and decomposing methods improve the quality of entity representations from different aspects. Thus, MSDP effectively alleviates the prototype classification disarray problem in the entity classification stage.

4.8 Influence of Data Size

To find out the influence of data size, we conduct a comparison experiment between SpanProto and MSDP under different few-shot settings of Few-NERD. As shown in figure 6, the performance of MSDP still has a steady improvement compared with SpanProto with the increase of data in inter setting while the improvement is not obvious in intra setting. We think the possible reasons are as follows: since fine-grained entity types are separated in inter setting, compared with baseline, MSDP can better assist the model to capture the fine-grained entity type information, and the effect is more obvious with the increase of data size. For intra settings which are separated fine-grained entity types, the MSDP has a slight increase, but there is still much room for improvement. So improving the ability of model to capture coarse-grained entity types is a great challenge for future research.

Guanting Dong et al.

A Multi-Task Semantic Decomposition Framework with Task-specific Pre-training for Few-Shot NER

Methods	F1	FP-Type
ProtoBERT	44.44	13.30
NNShot	54.29	15.30
StructShot	57.33	20.00
ESD	66.46	27.20
DecomMeta	76.11	46.53
SpanProto	73.36	10.90
MSDP	76.86	9.22

Table 6: Error analysis (%) of 5-way 1-shot on Few-NERD Inter. "FP-Type" represents extracted entities with the right span boundary but the wrong entity type.



Figure 6: The influence of data size under Inter(left) and Intra(right) setting of Few-NERD

5 HYPER-PARAMETER ANALYSIS

5.1 The Effect of Temperature Parameter.

Table 7 shows the effect of different τ values of SCL in class contrastive discrimination task. We find that when the temperature $\tau = 0.5$, MSDP achieves the best performance in the Inter and Intra setting of Few-NERD. Our method within τ [0.1, 0.6] outperforms sota baselines (SpanProto), and τ in [0.4, 0.6] brings larger improvements (above 2% in Inter and 6% in Intra). This experiment demonstrates the robustness of MSDP, as changes in temperature τ do not affect its performance.

5.2 The Effect of Number of Demonstrations.

We further examine whether the performance of MSDP changes over the number(K) of retrieved demonstration and negative demonstration in the pre-training stage. As shown in Figure 7, the performance of MSDP improves from 75.76 to 76.86 on Few-NERD Inter5-1 with the number of retrieved demonstrations from 1 to 10. However, the performance of negative demonstrations increases first(75.72 to 76.09) and then decreases(76.09 to 75.58) due to the increase in the number of retrieved demonstrations can provide rich entity-label pairs and factual information, which can assist the model to learn good representations. A small amount of negative demonstrations can help the model distinguish the boundary between entities and non-entities, but too many negative demonstrations will introduce a large number of non-entities which brings the noise. CIKM '23, October 21-25, 2023, Birmingham, United Kingdom

	Few-NERD			
Temperature τ	Intra	Inter		
$\tau = 0.1$	56.84	76.33		
$\tau = 0.2$	58.03	77.25		
$\tau = 0.3$	57.44	76.62		
$\tau = 0.4$	58.53	77.89		
$\tau = 0.5$	58.74	78.23		
$\tau = 0.6$	58.60	78.03		





Figure 7: The performance of MSDP changes over the number(K) of retrieved demonstration and negative demonstration

6 CONCLUSION

In this paper, we propose a Multi-Task Semantic Decomposition Framework via Joint Task-specific Pre-training (MSDP) for fewshot NER. Specifically, We introduce two novel pre-training tasks, Demonstration-based MLM and Class Contrastive Discrimination, to solve the span over-prediction and prototype classification disarray problem. Further, We design a multi-task joint optimization framework, and decompose class-oriented prototypes and contextual fusion prototypes to integrate two different semantic information for entity classification. Experimental results demonstrate that MSDP outperforms the previous SOTA methods in terms of overall performance. Extensive analysis further validates the effectiveness and generalization of our approach.

REFERENCES

- Rie Kubota Ando and Tong Zhang. 2005. A Framework for Learning Predictive Structures from Multiple Tasks and Unlabeled Data. *Journal of Machine Learning Research* 6 (Dec. 2005), 1817–1853. https://www.jmlr.org/papers/volume6/ ando05a/ando05a.pdf
- [2] Galen Andrew and Jianfeng Gao. 2007. Scalable training of L₁-regularized loglinear models. In Proceedings of the 24th International Conference on Machine Learning. 33–40. https://dl.acm.org/doi/abs/10.1145/1273496.1273501
- [3] Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016. Stance Detection with Bidirectional Conditional Encoding. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Austin, Texas, 876–885. https://doi. org/10.18653/r1/D16-1084
- [4] Siqi Bao, Huang He, Fan Wang, Hua Wu, and Haifeng Wang. 2019. Plato: Pretrained dialogue generation model with discrete latent variable. arXiv preprint arXiv:1910.07931 (2019).
- [5] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. arXiv preprint arXiv:1903.10676 (2019).
- [6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. Advances in neural information processing systems 33 (2020), 1877–1901.
- [7] Yixin Cao, Zikun Hu, Tat-seng Chua, Zhiyuan Liu, and Heng Ji. 2019. Lowresource name tagging learned with weakly labeled data. arXiv preprint arXiv:1908.09659 (2019).
- [8] Qianglong Chen, Feng-Lin Li, Guohai Xu, Ming Yan, Ji Zhang, and Yin Zhang. 2022. Dictbert: Dictionary description knowledge enhanced language model pre-training via contrastive learning. arXiv preprint arXiv:2208.00635 (2022).
- [9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 1597–1607.
 [10] Jason PC Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional
- [10] Jason PC Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional LSTM-CNNs. Transactions of the association for computational linguistics 4 (2016), 357–370.
- [11] Sarkar Snigdha Sarathi Das, Arzoo Katiyar, Rebecca J Passonneau, and Rui Zhang. 2021. Container: Few-shot named entity recognition via contrastive learning. arXiv preprint arXiv:2109.07589 (2021).
- [12] Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In Proceedings of the 3rd Workshop on Noisy User-generated Text. 140–147.
- [13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018).
- [14] Ning Ding, Guangwei Xu, Yulin Chen, Xiaobin Wang, Xu Han, Pengjun Xie, Hai-Tao Zheng, and Zhiyuan Liu. 2021. Few-nerd: A few-shot named entity recognition dataset. arXiv preprint arXiv:2105.07464 (2021).
- [15] Guanting Dong, Daichi Guo, Liwen Wang, Xuefeng Li, Zechen Wang, Chen Zeng, Keqing He, Jinzheng Zhao, Hao Lei, Xinyue Cui, et al. 2022. PSSAT: A Perturbed Semantic Structure Awareness Transferring Method for Perturbation-Robust Slot Filling. arXiv preprint arXiv:2208.11508 (2022).
- [16] Guanting Dong, Zechen Wang, Liwen Wang, Daichi Guo, Dayuan Fu, Yuxiang Wu, Chen Zeng, Xuefeng Li, Tingfeng Hui, Keqing He, Xinyue Cui, Qixiang Gao, and Weiran Xu. 2023. A Prototypical Semantic Decoupling Method via Joint Contrastive Learning for Few-Shot Named Entity Recognition. In *ICASSP 2023 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 1–5. https://doi.org/10.1109/ICASSP49357.2023.10095149
- [17] Zhangyin Feng, Daya Guo, Duyu Tang, Nan Duan, Xiaocheng Feng, Ming Gong, Linjun Shou, Bing Qin, Ting Liu, Daxin Jiang, et al. 2020. Codebert: A pre-trained model for programming and natural languages. arXiv preprint arXiv:2002.08155 (2020).
- [18] Alexander Fritzler, Varvara Logacheva, and Maksim Kretov. 2019. Few-shot classification in named entity recognition task. In Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing. 993–1000.
- [19] Tianyu Gao, Adam Fisch, and Danqi Chen. 2020. Making pre-trained language models better few-shot learners. arXiv preprint arXiv:2012.15723 (2020).
- [20] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. arXiv preprint arXiv:2104.08821 (2021).
- [21] Abbas Ghaddar and Philippe Langlais. 2018. Transforming Wikipedia into a largescale fine-grained entity type corpus. In Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018).
- [22] James Goodman, Andreas Vlachos, and Jason Naradowsky. 2016. Noise reduction and targeted exploration in imitation learning for Abstract Meaning Representation parsing. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Berlin, Germany, 1–11. https://doi.org/10.18653/v1/P16-1001
- [23] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In International conference on machine learning. PMLR, 1321–1330.

- [24] Daichi Guo, Guanting Dong, Dayuan Fu, Yuxiang Wu, Chen Zeng, Tingfeng Hui, Liwen Wang, Xuefeng Li, Zechen Wang, Keqing He, Xinyue Cui, and Weiran Xu. 2023. Revisit Out-Of-Vocabulary Problem For Slot Filling: A Unified Contrastive Framework With Multi-Level Data Augmentations. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 1–5. https://doi.org/10.1109/ICASSP49357.2023.10094766
- [25] Jiafeng Guo, Gu Xu, Xueqi Cheng, and Hang Li. 2009. Named entity recognition in query. In Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. 267–274.
- [26] Mary Harper. 2014. Learning from 26 Languages: Program Management and Science in the Babel Program. In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. Dublin City University and Association for Computational Linguistics, Dublin, Ireland, 1. https://aclanthology.org/C14-1001
- [27] Keqing He, Jinchao Zhang, Yuanmeng Yan, Weiran Xu, Cheng Niu, and Jie Zhou. 2020. Contrastive zero-shot learning for cross-domain slot filling with adversarial attack. In Proceedings of the 28th International Conference on Computational Linguistics. 1461–1467.
- [28] Yutai Hou, Wanxiang Che, Yongkui Lai, Zhihan Zhou, Yijia Liu, Han Liu, and Ting Liu. 2020. Few-shot slot tagging with collapsed dependency transfer and label-enhanced task-adaptive projection network. arXiv preprint arXiv:2006.05702 (2020).
- [29] Yutai Hou, Yijia Liu, Wanxiang Che, and Ting Liu. 2018. Sequence-to-sequence data augmentation for dialogue language understanding. arXiv preprint arXiv:1807.01554 (2018).
- [30] Chen Jia, Xiaobo Liang, and Yue Zhang. 2019. Cross-domain NER using crossdomain language modeling. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2464–2474.
- [31] Chen Jia and Yue Zhang. 2020. Multi-cell compositional LSTM for NER domain adaptation. In Proceedings of the 58th annual meeting of the association for computational linguistics. 5906–5917.
- [32] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. Advances in Neural Information Processing Systems 33 (2020), 18661– 18673.
- [33] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).
- [34] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. arXiv preprint arXiv:1603.01360 (2016).
- [35] Dong-Ho Lee, Mahak Agarwal, Akshen Kadakia, Jay Pujara, and Xiang Ren. 2021. Good examples make A faster learner: Simple demonstration-based learning for low-resource NER. arXiv preprint arXiv:2110.08454 (2021).
- [36] Elad Levi, Tete Xiao, Xiaolong Wang, and Trevor Darrell. 2021. Rethinking preventing class-collapsing in metric learning with margin-based losses. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 10316–10325.
- [37] Xuefeng Li, Hao Lei, Liwen Wang, Guanting Dong, Jinzheng Zhao, Jiachi Liu, Weiran Xu, and Chunyun Zhang. 2022. A Robust Contrastive Alignment Method for Multi-Domain Text Classification. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 7827–7831. https: //doi.org/10.1109/ICASSP43922.2022.9747192
- [38] Xuefeng Li, Liwen Wang, Guanting Dong, Keqing He, Jinzheng Zhao, Hao Lei, Jiachi Liu, and Weiran Xu. 2023. Generative Zero-Shot Prompt Learning for Cross-Domain Slot Filling with Inverse Prompting. In *Findings of the Association for Computational Linguistics: ACL 2023*. Association for Computational Linguistics, Toronto, Canada, 825–834. https://aclanthology.org/2023.findings-acl.52
- [39] Xiaozhuan Liang, Ningyu Zhang, Siyuan Cheng, Zhen Bi, Zhenru Zhang, Chuanqi Tan, Songfang Huang, Fei Huang, and Huajun Chen. 2022. Contrastive Demonstration Tuning for Pre-trained Language Models. arXiv preprint arXiv:2204.04392 (2022).
- [40] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019).
- [41] Zihan Liu, Feijun Jiang, Yuxiang Hu, Chen Shi, and Pascale Fung. 2021. NER-BERT: a pre-trained model for low-resource entity tagging. arXiv preprint arXiv:2112.00405 (2021).
- [42] Zihan Liu, Genta I Winata, Samuel Cahyawijaya, Andrea Madotto, Zhaojiang Lin, and Pascale Fung. 2021. On the importance of word order information in crosslingual sequence labeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 13461–13469.
- [43] Zihan Liu, Genta Indra Winata, and Pascale Fung. 2020. Zero-resource crossdomain named entity recognition. arXiv preprint arXiv:2002.05923 (2020).
- [44] Zihan Liu, Genta Indra Winata, Peng Xu, and Pascale Fung. 2020. Coach: A coarseto-fine approach for cross-domain slot filling. arXiv preprint arXiv:2004.11727 (2020).
- [45] Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2021. Crossner: Evaluating cross-domain

named entity recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35. 13452–13460.

- [46] Jie Ma, Miguel Ballesteros, Srikanth Doss, Rishita Anubhai, Sunil Mallya, Yaser Al-Onaizan, and Dan Roth. 2022. Label Semantics for Few Shot Named Entity Recognition. arXiv preprint arXiv:2203.08985 (2022).
- [47] Tingting Ma, Huiqiang Jiang, Qianhui Wu, Tiejun Zhao, and Chin-Yew Lin. 2022. Decomposed Meta-Learning for Few-Shot Named Entity Recognition. arXiv preprint arXiv:2204.05751 (2022).
- [48] Sewon Min, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2021. Noisy channel language model prompting for few-shot text classification. arXiv preprint arXiv:2108.04106 (2021).
- [49] Diego Mollá, Menno Van Zaanen, and Daniel Smith. 2006. Named entity recognition for question answering. In Proceedings of the Australasian language technology workshop 2006. 51–58.
- [50] David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes* 30, 1 (2007), 3-26.
- [51] Matthew E Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-supervised sequence tagging with bidirectional language models. arXiv preprint arXiv:1705.00108 (2017).
- [52] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 1802. Deep contextualized word representations. CoRR abs/1802.05365 (2018). arXiv preprint arXiv:1802.05365 (1802).
- [53] Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards robust linguistic analysis using ontonotes. In Proceedings of the Seventeenth Conference on Computational Natural Language Learning. 143–152.
- [54] Gao Qixiang, Guanting Dong, Yutao Mou, Liwen Wang, Chen Zeng, Daichi Guo, Mingyang Sun, and Weiran Xu. 2022. Exploiting domain-slot related keywords description for Few-Shot Cross-Domain Dialogue State Tracking. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2460–2465. https://aclanthology.org/2022.emplp-main.157
- [55] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.
- [56] Mohammad Sadegh Rasooli and Joel R. Tetreault. 2015. Yara Parser: A Fast and Accurate Dependency Parser. *Computing Research Repository* arXiv:1503.06733 (2015). http://arxiv.org/abs/1503.06733 version 2.
- [57] Erik F Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. arXiv preprint cs/0306050 (2003).
- [58] Yongliang Shen, Xinyin Ma, Zeqi Tan, Shuai Zhang, Wen Wang, and Weiming Lu. 2021. Locate and label: A two-stage identifier for nested named entity recognition. arXiv preprint arXiv:2105.06804 (2021).
- [59] Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. Advances in neural information processing systems 30 (2017).
- [60] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2019. Vl-bert: Pre-training of generic visual-linguistic representations. arXiv preprint arXiv:1908.08530 (2019).
- [61] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019. Videobert: A joint model for video and language representation learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 7464– 7473.
- [62] Mingyang Sun, Qixiang Gao, Yutao Mou, Guanting Dong, Ruifang Liu, and Wenbin Guo. 2023. Improving Few-Shot Performance of DST Model Through Multitask to Better Serve Language-Impaired People. In 2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW). 1–5. https://doi.org/10.1109/ICASSPW59220.2023.10193387
- [63] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. Journal of machine learning research 9, 11 (2008).

- [64] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. 2016. Matching Networks for One Shot Learning. In Advances in Neural Information Processing Systems, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (Eds.), Vol. 29. Curran Associates, Inc. https://proceedings.neurips.cc/ paper/2016/file/90e1357833654983612fb05e3ec9148c-Paper.pdf
- [65] Jianing Wang, Chengyu Wang, Chuanqi Tan, Minghui Qiu, Songfang Huang, Jun Huang, and Ming Gao. 2022. SpanProto: A Two-stage Span-based Prototypical Network for Few-shot Named Entity Recognition. arXiv preprint arXiv:2210.09049 (2022).
- [66] Peiyi Wang, Runxin Xu, Tianyu Liu, Qingyu Zhou, Yunbo Cao, Baobao Chang, and Zhifang Sui. 2021. An enhanced span-based decomposition method for few-shot sequence labeling. arXiv preprint arXiv:2109.13023 (2021).
- [67] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. arXiv preprint arXiv:2201.11903 (2022).
- [68] Chien-Sheng Wu, Steven Hoi, Richard Socher, and Caiming Xiong. 2020. TOD-BERT: Pre-trained natural language understanding for task-oriented dialogue. arXiv preprint arXiv:2004.06871 (2020).
- [69] Shuhui Wu, Yongliang Shen, Zeqi Tan, and Weiming Lu. 2022. Propose-and-Refine: A Two-Stage Set Prediction Network for Nested Named Entity Recognition. arXiv preprint arXiv:2204.12732 (2022).
- [70] Yi Yang and Arzoo Katiyar. 2020. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. arXiv preprint arXiv:2010.02405 (2020).
- [71] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. Advances in neural information processing systems 32 (2019).
- [72] Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Chuanqi Tan, and Chang Zhou. 2023. Scaling Relationship on Learning Mathematical Reasoning with Large Language Models. arXiv:2308.01825 [cs.CL]
- [73] Amir Zeldes. 2017. The GUM corpus: Creating multilayer resources in the classroom. Language Resources and Evaluation 51, 3 (2017), 581–612.
- [74] Weihao Zeng, Keqing He, Zechen Wang, Dayuan Fu, Guanting Dong, Ruotong Geng, Pei Wang, Jingang Wang, Chaobo Sun, Wei Wu, and Weiran Xu. 2022. Semi-Supervised Knowledge-Grounded Pre-training for Task-Oriented Dialog Systems. arXiv:2210.08873 [cs.CL]
- [75] Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2019. Dialogpt: Large-scale generative pre-training for conversational response generation. arXiv preprint arXiv:1911.00536 (2019).
- [76] Yupeng Zhang, Shensi Wang, Peiguang Li, Guanting Dong, Sirui Wang, Yunsen Xian, Zhoujun Li, and Hongzhi Zhang. 2023. Pay Attention to Implicit Attribute Values: A Multi-modal Generative Framework for AVE Task. In *Findings of the Association for Computational Linguistics: ACL 2023*. Association for Computational Linguistics, Toronto, Canada, 13139–13151. https://aclanthology.org/2023.findings-acl.831
- [77] Gang Zhao, Guanting Dong, Yidong Shi, Haolong Yan, Weiran Xu, and Si Li. 2022. Entity-level Interaction via Heterogeneous Graph for Multimodal Named Entity Recognition. In *Findings of the Association for Computational Linguistics: EMNLP 2022*. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 6345–6350. https://aclanthology.org/2022.findings-emnlp.473
- [78] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*. PMLR, 12697–12706.
- [79] Ran Zhou, Xin Li, Ruidan He, Lidong Bing, Erik Cambria, Luo Si, and Chunyan Miao. 2022. MELM: Data Augmentation with Masked Entity Language Modeling for Low-Resource NER. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2251–2262.
- [80] Enwei Zhu and Jinpeng Li. 2022. Boundary Smoothing for Named Entity Recognition. arXiv preprint arXiv:2204.12031 (2022).
- [81] Morteza Ziyadi, Yuting Sun, Abhishek Goswami, Jade Huang, and Weizhu Chen. 2020. Example-based named entity recognition. arXiv preprint arXiv:2008.10570 (2020).