Check for updates

RESEARCH PAPER

Few-shot Named Entity Recognition with Joint Token and Sentence Awareness

Wen Wen[1], Yongbin Liu[1,2†], Qiang Lin[1], Chunping Ouyang[1]

[1]Computer School, University of South China, China

[2]Hunan provincial base for scientific and technological innovation cooperation, Hunan, China

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

Keywords: Few-shot Learning; Named Entity Recognition; Prototypical Network

Citation: Wen, W., Liu, Y.B., Lin, Q., Ouyang, C.P.: Few-shot Named Entity Recognition with Joint Token and Sentence Awareness. Data Intelligence 5(3), 767-785 (2023). doi: https://doi.org/10.1162/dint_a_00195

Submitted: May 10, 2022; Received: November, 15, 2022; Accepted: March, 15, 2023

ABSTRACT

Few-shot learning has been proposed and rapidly emerging as a viable means for completing various tasks. Recently, few-shot models have been used for Named Entity Recognition (NER). Prototypical network shows high efficiency on few-shot NER. However, existing prototypical methods only consider the similarity of tokens in query sets and support sets and ignore the semantic similarity among the sentences which contain these entities. We present a novel model, Few-shot Named Entity Recognition with **J**oint **T**oken and **S**entence **A**wareness (**JTSA**), to address the issue. The sentence awareness is introduced to probe the semantic similarity among the sentences. The Token awareness is used to explore the similarity of the tokens. To further improve the robustness and results of the model, we adopt the joint learning scheme on the few-shot NER. Experimental results demonstrate that our model outperforms state-of-the-art models on two standard Few-shot NER datasets.

1. INTRODUCTION

Few-Shot learning (FSL) can reduce the burden of annotated data and quickly generalize to new tasks without training from scratch (usually only one or five per category). The few-shot learning has been made remarkable progress in many areas, such as computer vision (CV) [1, 2, 3] and relation classification (RC) [4, 5, 6, 7, 8]. But the FSL progress is much slower in named entity recognition (NER), mainly because entity recognition which is token-level classification tasks is more fine-grained and complicated than

† Corresponding author: Yongbin Liu (E-mail: yongbinliu03@gmail.com; ORCID: 0000-0002-3369-3101).

© 2023 Chinese Academy of Sciences. Published under a Creative Commons Attribution 4.0 International (CC BY 4.0) license.

sentence-level classification. The fine-grained and complicated expressions also aggravate the negative impacts of other class with abundant semantics and unclear class boundaries.

In the current few-shot models, the prototypical network [4] is a simple and powerful approach for few-shot NER. The basic idea is to learn the prototype of each predefined entity class, then classify the query samples according to their closest prototype [9, 10]. Most existing fewshot NER models mainly focus on the massive semantics hidden in token space [10, 11], such as Tong et al. [10] utilized the clustering method to divide othe classes for learning entity prototype further. However, they ignore the rich semantics in the sentences containing the multiple entity classes. Meanwhile the experiments of these methods were either performed on coarse-grained entity types [10] or on the slot filling of dialogue task [11] which is pretty inefficient for few-shot NER.

Sentence level semantic information can help few-shot NER, mainly because of two aspects: 1) Entity Relation. A large number of sentences contain more than two entities. In fact, the sentences are representations of the relation between entities although this relation does not need to be identified and classified in the NER task. The entity relation in the sentences can be used to improve the entity prototypes in the few-shot NER. 2) O-class Positive and Negative Impact. Sentence-level semantics can leverage rich semantics in other class (O-class) to learn entity prototypes. The sentence semantics are embedded in sentence-level representations, focusing on the contextual information in sentences without other class label impacts. The sentence embedding could represent each predefined entity class. And this way can handle the other class noise issue.

This paper proposes a novel model, few-shot NER with Joint Token and Sentence Awareness (JTSA). The token awareness module aims to learn the association between tokens from support and filter out the tokens that have a more significant impact on recognizing entities. In contrast, the sentence awareness module knows the semantics information from the sentences to improve the few-shot NER. In practice, the sentences often contain rich semantics of the entities and can provide abundant knowledge for discovering the best prototype of each entity class. The prototypes in tokens space are representations by abstracting the essential semantics of words and in the sentences space by embedding the semantic information of the sentences including multiple different entity classes. To improve the few-shot NER further, we joined the token and sentence modules for the final classification, which can learn the prototypes of entities in sentences better. The novel model joints token and sentence modules for deep interaction between token and sentence, capable of adopting their respective useful semantics information. Our model leverages the sentence-level prototype to calibrate the token-level prototype. It can also effectively alleviate the noise impacts of the O-class tokens to improve the few-shot NER.

We conduct a variety of experiments on the FEW-NERD [12] dataset that has just been released. The FEW-NERD is a large-scale human-annotated few-shot NER dataset with 66 finegrained entity types [12]. The experimental results demonstrate that our model outperforms the current SOTA approaches in few-shot NER. The subsequent ablation experiments show the significance of sentences-level awareness. Our contributions can be summarized as follows:

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

- We propose a novel module, sentence awareness, to leverage the entity relation in the sentences to improve few-shot NER. The module can also address O-class issue, and introduce a significant solution of how to adopt the useful semantic information of O-class words and alleviate noise impact.
- To improve few-shot NER further, we also propose token awareness to highlight more helpful tokens for recognizing entities and a novel approach with joint token and sentence awareness. The approach leverage the respective advantages of tokens and sentences to promote the experiment results.
- We conduct the experiments on the large-scale few-shot NER dataset with 66 fine-grained entity types, and compare our approach with multiple state-of-the-art baselines. The overall results strikingly outperforms the SOTA approaches in few-shot NER task. Further ablation studies show the effectiveness of our model and the modules.

2. RELATED WORK

2.1 Named Entity Recognition

In Natural Language Processing (NLP), Named Entity Recognition (NER) aims to identify entities (person, location, organization, drug, time, clinical procedure, biological protein, etc.) from unstructured text, which has been studied and developed widely for decades [13, 14, 15]. NER serves as the fundamental task in NLP, same as question answering, information retrieval, relation extraction, etc. Neural networks have significantly improved the results of the NER task in the last few years [16, 17, 18, 19, 20, 21, 22]. Although neural NER networks have achieved superior performance, these methods need large-scale training data. It is challenging to obtain massive annotated data. Recently, few-shot learning can handle the issue.

2.2 Few-Shot Learning

Few-shot learning has been proposed and rapidly emerging as a viable means for completing various tasks. Many few-shot models have been widely used for classification tasks. Siamese neural network was applied to few-shot classification by Koch et al. [1], and it utilized a convolutional architecture to rank the similarity between inputs naturally. Then, matching network [2] was proposed in 2017. It used some external memories to enhance the neural networks. It added an attention mechanism and a new method named cosine distance as the similarity metric to predict the relations. In 2018, Sung et al. [3] proposed a relation network for few-shot

learning. The relation network learns an embedding and a deep non-linear distance metric for comparing query and sample items. Moreover, the Euclidean distance empirically outperforms the more commonly used cosine similarity on multi-tasks. Thus, a simpler and more efficient model prototypical network was proposed by Snell et al. [4]. The naive approach used a standard Euclidean distance as the distance function. In 2019, Gao et al. [6] introduced a hybrid attention-based prototypical network, which is a more efficient prototypical network, and trained a weight matrix for Euclidean distance.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

2.3 Few-Shot Named Entity Recognition

Few-shot Named Entity Recognition refers to NER task with only one or few examples per category [23, 24, 25, 26, 27]. Hofer et al. [24] studies the named entity recognition of electronic health records that 10 samples are collected from the target dataset for few-shot learning. Yang and Katiyar [25] presents a few-shot NER system based on nearest neighbor learning and structured inference. The approach shows that the nearest neighbor classifier in this feature space is more effective in the few-shot NER task. Hou et al. [11] focuses on the spoken language understanding task and leverages the label semantic to classify the entities. Tong et al. [10] proposes an approach, mining undefined classes from Other-class, adjusting single Other-class prototype to multiple prototypes by clustering method. This way can reduce the O-classes negative impacts on the identification of target entities [10]. These methods are state-of-the-art based on token-level. And most of them aim to recognize coarse-grained entity types [12]. Ding et al. [12] releases a fine-grained dataset, FEW-NERD, which is a large-scale human-annotated few-shot NER dataset with 66 fine-grained entity types [12]. Also, they present superior performance few-shot NER model based on token-level on the FEW-NERD. But the above methods neglected sentence-level semantics. As a contrast, we propose a novel few-shot NER model with token and sentence awareness.

3. METHODOLOGY

In this section, we give a detailed introduction to the implementation of our proposed model JTSA which is shown in Figure 1. JTSA consists of three main parts, including: Token awareness module, Sentence awareness module and Joint learning scheme module.

3.1 Problem Definition

Following Ding et al. [12], we regard NER as a sequence labeling problem. NER aims to label each token x_k in the input sequence $s^q = \{x_1, x_2, ...x_m\}$ with a label $y_k \in Y$, where y_k is one of the pre-defined class set Y or not belong to any entities(Other class) [12], and m is the maximum length of an sentence. In few-shot learning, a system is trained on an annotations of source domains $\{D_1^s, D_2^s, ...\}$, then evaluated on another set of unseen target domains $\{D_1^t, D_2^t, ...\}$ [11, 25]. The target sets only provide few labeled examples, which forms a support set $S = \{(x_i, y_i)\}_{i=1}^{N \times K}$ and a query set $Q = \{(x_j, y_j)\}_{j=1}^{N \times K'}$. In each query sample (x, y), x and y represent the entity and the corresponding class label respectively. N and K come from the definition of N-way K-shot NER task, K' is the number of test entities per class. In this paper, we follow the definition of Ding et al. [12] for few-shot NER. Specifically, this task randomly select N entity classes(N-way) at first, then K samples are randomly chosen(K-shot) from each class. In each instance s_i, a word which not belong to the predefined entity class are regarded as O-class(other class or none-of-the-above), and the O-class is assumed as the $(N + 1)$-th class label. Thus, in the support set $S = \{c_i\{(x_{c_i,1}, y_{c_i}), ..., (x_{c_i,k}, y_{c_i})\}\}_{i=1}^{N+1}$, there are $N \times K$ entity samples with predefined classes. The $x_{c_i,j}$ presents the j-th entity in the c_i-th class and $y_{c_i} \in C$ is the entity class label. The O-class is represented by c_o. Thus, Few-shot NER models are supposed to learn the feature of each class from the few entities in the support set S, and then predict the class label y of an unseen query entity q.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

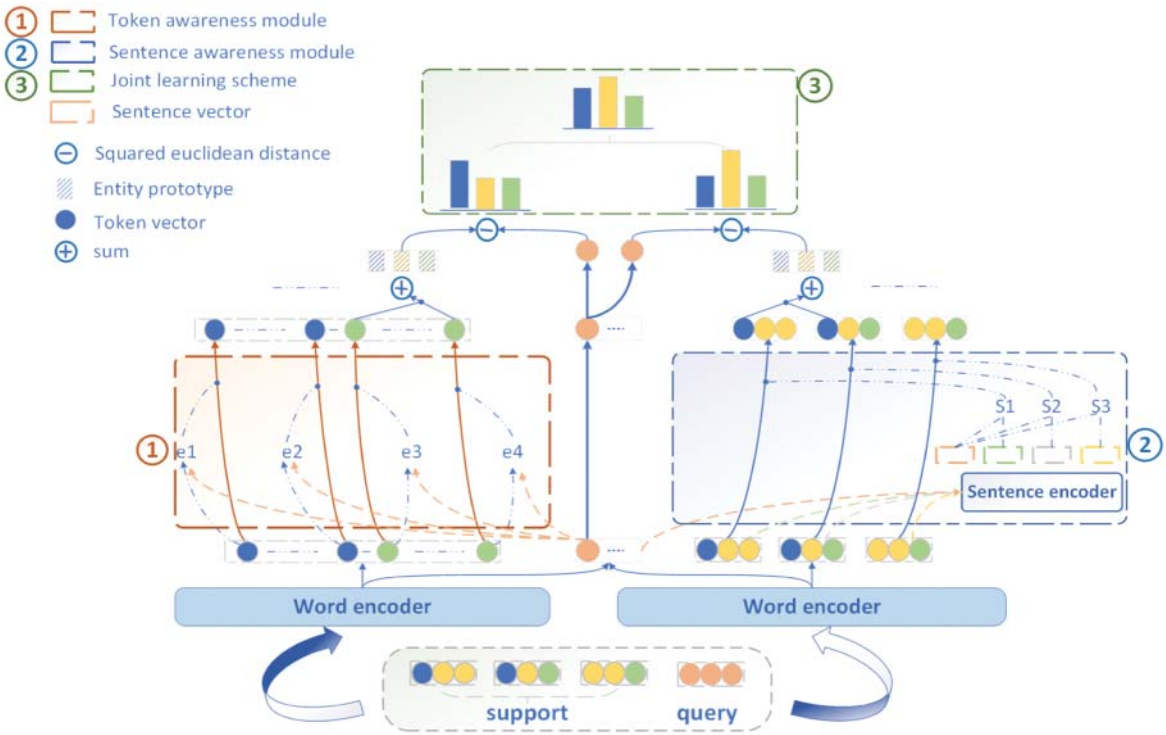


Figure 1. Architecture of our JTSA model. Circles with Blue, green, and yellow indicate different types of entities in support, while orange is the entity in query set that needs to be correctly categorized to the above 3 types by our model. Orange box marked as is our Token Awareness Module, which uses the similarity between the support sample and query sample characters to construct the prototypes in our network. The blue part marked as is our Sentence Awareness Module, which considers the association between sentences, and filters out the support samples that have a greater impact on entity classification. Mark colored green is our Joint Learning Scheme, which joints the token awareness and sentence awareness to enhance the ability to identify entity classes.

3.2 Implementation Details

Our model is based on prototypical network, since it is a simple and effective method for few-shot learning. For few-shot NER task, prototypical network assumes that in each entity class, there is a prototype which is able to represent this class, and each entity clusters around the prototype of the class which they belong to. The purpose of prototypical network is to learn the representation of prototype p_i for each class, and then predict the class label of query entity q. The query entity q is classified in three steps: First, prototypical network gets the representation of prototypes $P = \{p_1, p_2, ..., p_n\}$ for all classes from support set. Second step is to calculate the distance between the query entity q and all the prototypes respectively. Finally, the query entity q is classified to the closest class.

In the first step, we encode each tokens in the support set into a D-dimensional embedding $\mathbf{x}_i = f_\theta(x_i)$, through an embedding function with learnable parameters θ:

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

$$\mathbf{x}_i = f_\theta(x_i), \mathbf{x}_i \in R^D. \tag{1}$$

In our model, the encoder f_θ is the pre-trained language model BERT [28] with transformers. Then, our model calculate the prototype of each class as the following way:

$$p_i = \frac{1}{|c_i|}\sum_{j=1}^{|c_i|}(x_j), \tag{2}$$

where $|c_i|$ is the number of tokens in the class label c_i.

The second step utilize the similarity function shown in Eq.3 to calculate the distance between entity q and each prototype p_i.

$$d(f_\theta(x_q), p_i) = (f_\theta(x_q) - p_i)^2. \tag{3}$$

For the last step, we get the class distribution about the entity q by Eq.4.

$$g_\theta(y = c_i \mid x_q) = \frac{\exp(-d(f_\theta(x_q), p_i))}{\sum_{j=1}^{|C|} exp(-d(f_\theta(x_q), p_j))}, \tag{4}$$

where $|C|$ stands for the number of classes in class set $|C|$. Furthermore, during meta-testing,we add an Viterbi decoder module to get the transfer rules between adjacent entity labels, and then modify the class distribution $g_\theta(y = c_i \mid x_q)$ by the transition distribution $g(y', y)$ as Eq.5.

$$y^* = \text{argmax}_y \prod_{t=1}^{T} g(y_t \mid x_q) * g(y_t \mid y_{t-1}), \tag{5}$$

To achieve better class representation of named entity, we design the sentence awareness module and token awareness module to get a task adaptive class prototype.

3.3 Sentence Awareness

Traditional prototypical network get the prototype of an entity class by simple averaging all the words embedding in it. However, in real-world, the semantic of sentences is different. The primary goal of our sentence awareness is to consider the contribution of each sentence and construct different matrix for each query.

Our sentence awareness module depends on the similarity of sentences. As shown in the sentence awareness module in Figure 1, when predicting a query entity q appeared in query sentence s_q, our sentence awareness module captures the similar sentences from support set S. The entities in these sentences are more interrelated with the query entity q, and the weight of each sentence is updated by the degree of relevance.

First of all, we encode the sentence s_q by 1-Dimension convolutional neural networks, and get a continuous low-dimensional sentence embedding h_q. Then, each sentence s_l in support set is encoded like this to generate the embedding h_s^l. The process is shown in Eq.6.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

$$h_q = conv(s_q < q_1, ..., q_n >)$$
$$h_s^l = conv(s_l < e_1, ..., e_n >). \tag{6}$$

Eq.7 presents the way to calculate the similarity between sentences, where d is the distance function in Eq.3.

$$\alpha_l = \frac{\exp(-d(h_q, h_s^l))}{\sum_{j=1}^{|S|} \exp(-d(h_q, h_s^j))}. \tag{7}$$

Thus, the prototype p_i^{sa} in our model is defined as Eq.8, which is able to pay more attention on the correlation information between query sentence s_q and support set S.

$$p_i^{sa} = \frac{1}{|c_i|} \sum_{j=1}^{|c_i|} (\alpha_l x_j). \tag{8}$$

Furthermore, in few-shot NER task, *O*-class (none-of-the-above) is special and creates great challenges to recognize the query entities, because it contains all the entities which can not be classified into any of the class set and these entities have huge gaps between each other. In the sentence awareness module, we leverage the sentence semantics to alleviate this problem.

3.4 Token awareness module

In the paper, we propose the token awareness mechanism, which focus on the tokens in support set which are more relevant and have more similar feature to the query entity. Our token awareness module is shown in Figure 1. When identifying entities in query sentences, the module captures tokens that are more associated with the query entities from the support set and then update the token weights according to the degree of relevance. Our model calculates the correlation as Eq.9, where the correlation coefficient β_j presents the similarity between the query entity q and the entity sample x_j in c_i class.

$$\beta_j = \frac{\exp(-d(f_\theta(x_q), f_\theta(x_{c_{i,j}})))}{\sum_{k=1}^{|c_i|} \exp(-d(f_\theta(x_q), f_\theta(x_{c_{i,k}})))} \tag{9}$$

Then, for each query sample q, the prototypes of classes is defined as:

$$p_i^{ta} = \frac{1}{|c_i|} \sum_{j=1}^{|c_i|} (\beta_j x_j) \tag{10}$$

For both Sentence Awareness Prototype (SAP) module and Token Awareness Prototype (TAP) module, the optimization goal is to minimize the cross-entropy loss function as Eq.11:

$$L(\theta) = \frac{1}{n} \sum_{i=1}^{n} l(y_i, g_\theta(x_i)) + \lambda ||\theta||_2^2, \tag{11}$$

where g_θ represents our SAP model and TAP model. λ is the weight decay parameter, and l is the cost function to get the distribution between truth-label and predicted-label.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

3.5 Joint learning acheme

In few shot NER task, distinct few-shot settings and distinct granularity have different requirements to the model. In practical, our sentence awareness which pays more attention on the correlation between sentences is superior, when the class label is fine-grained and the number of samples in the support set is extremely few. The token awareness module which focuses on the information shared between entities helps a lot when evaluating on coarse-grained dataset and the support set have a few samples. Therefore, for better coordinate the two modules and make our model achieve a good performance in various scenarios, we explore a simple and verifiable method JTSA. As shown in the Joint learning echeme in Figure 1, we combine the probability distributions predicted by the sentence awareness module and token awareness module to obtain the modified result. If one model considers that the probabilities of two class labels do not differ much when classifying an entity, the model has difficulties identifying the entity. There is a great possibility to be wrong. The introduction of the other model can effectively correct the previous error. When predicting the label of query sample q, our JTSA model gets the class distribution by Eq.12, which joint the distribution $g_\theta^{sa}(y = c_i \mid x)$ of SAP and $g_\theta^{ta}(y = c_i \mid x)$ of TAP.

$$g_\theta^{Jst}(y = c_i \mid x) = \delta g_\theta^{sa}(y = c_i \mid x) + \gamma g_\theta^{ta}(y = c_i \mid x), \tag{12}$$

where δ and γ is the hyper-parameter of model reliability, which is obtained by multiple episodes. For the final results, we utilize the Viterbi decoder as Eq.5.

4. EXPERIMENTS

In this section, we demonstrate the experiments and implementations in detail to show that our model is effective and superior. Firstly, we present the hyper-parameters and the datasets FEW-NERD which we used in our proposed model. Then, the results and the comparisons with existing state-of-the-art models are provided by evaluating our model on the datasets with different granularities. Last, we respectively study the validity of each component in our JTSA model, including sentence awareness module, token awareness module, and joint learning schema.

4.1 Datasets

For N-way K-shot NER tasks, we evaluate our proposed model JTSA on two open benchmarks: FEW-NERD(INTRA) and FEW-NERD(INTER), which are presented in Table 1.

FEW-NERD [12] is a large-scale NER dataset annotated based on Wikipedia, which consists 188,200 sentences with 4,601,223 tokens and 491,711 entities. The entities are assigned into 8 coarse-grained types, 66 fine-grained types. After deleting the sentences without entity, the dataset can be divided into two benchmark datasets: FEW-NERD(INTER) and FEW-NERD(INTRA), according different granularities of types. FEW-NERD(INTRA) is a coarse grained dataset. The training set of it consists of four coarsed-trained entity types: People, MISC, Art, Product, and has all the fine-grained types belonging to the four. Then, "Event" and "Building" are assigned to the validation set, while "ORG" and "LOC" are in the test set.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

Table 1. Datasets.

Dataset	Source	Apply	Supervised setting	Sentence
Few-NERD (INTER)	WiKi	Training	70%	130,112
	WiKi	Validation	10%	18,817
	WiKi	Testing	20%	14,007
Few-NERD (INTRA)	WiKi	Training	People, MISC, Product, Art	99,519
	WiKi	Validation	Event, Building	19,358
	WiKi	Testing	ORG, LOC	44,059

FEW-NERD(INTER) randomly splits 60% fine-grained types for training set, 20% for validation, and 20% for test, that means, the coarse-grained types are shared and for one set may have the whole coarse-grained types.

4.2 Experimental Setup

We evaluate our proposed model JTSA on 5-way 1-shot, 5-way 5-shot, 10-way 1-shot, and 10-way 5-shot tasks. The hyper-parameters of our model are reported in Table 2. Pre-trained BERT module is implemented to extract the initial word embedding representation of our model. The batch size is set to 2 and the number of query is 1. Our model has 10,000 iterations for training, 1,000 iterations for validation, and 500 iteration for test. In our model, we use AdamW as the optimizer, and the learning rate is set to 1e-4 with (0.1∗training iteration) warmup step.

Table 2. Hyper-parameters Setting.

Batch Size 2	2	Max_{tokens_num}	60
Query size	1	Learning_rate	1e-4
Training iteration	10000	Optimizer	Adamw
Val iteration	1000	Lr scheduler	warmup
Test iteration	500	Warmup step	1000
Val step	500	weight decay	0.01

4.3 Baselines

Prototypical Network [12] is a simple and efficient few-shot learning model. It assumes that there is a prototype for each class, and the query is classified by the closest prototype. **NNshot** [25] is a metric based few-shot model. NNshot believes that support entities closest to the query entity have the highest credibility. **Structshot** [25] has the same structure as NNshot, only adding the Viterbi decoder in the meta-testing phase. **ESD** [29] is an enhanced span based decomposition model for NER, which decomposes the span matching problem into a series of span-level procedures. **CONTaiNER** [30] is a new fewshot NER method which uses contrast learning. It attempts to reduce the distance of token embedding for similar entities while increasing the distance of token embedding for different entities.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

4.4 Overall Performance

In this part, we assess our proposed model JTSA from different perspectives based on the two benchmark datasets FEW-NERD(INTER) and FEW-NERD(INTRA), then compare our method with existing state-of-the-art approaches.

For the FEW-NERD(INTER), we first adopt our SAP model and TAP model. As represented in Table 3, the SAP achieves higher performance compared with existing state-of-the-art models for 1-shot tasks. On 5-way 1-shot task, the F1 score of our SAP has 5.78% improvement, and on 10-way 1-shot task, the improvement is around 4%. This sufficiently demonstrates that the sentence awareness module in SAP is effectively to aid entity types identification by integrating the structure information of sentences, when there are few support samples. On multiple-shot tasks, our TAP model achieves a significant improvement due to the advantages of the token awareness module. Since the specificity of NER task, the type of entity is regarded as "*O*-class" when it is not included in the predefined types. This introduce a large number of futile samples inevitably, while our sentence awareness is designed to solve the problem. Our sentence awareness module utilizes the semantics in sentences to filter out the samples which may interfere the entity recognition.

Table 3. Overall Performance on FEW-NERD(INTER). NNshot, Proto, Struct model are from [12] and add Viterbi decoder to the Proto as my baseline ProtoNet, and we evaluate them on our dataset.

FEW_NERD(INTER)(%)												
Model	**5-way 1~2-shot**			**5-way 5~10-shot**			**10-way 1~2-shot**			**10-way 5~10-shot**		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
NNshot	49.96±0.45	58.14±1.01	53.74±0.67	53.30±0.63	62.29±0.70	57.45±0.34	41.36±1.35	51.00±1.20	45.68±1.29	46.54±0.68	56.35±1.23	50.98±0.92
ProtoNet	49.88±0.96	55.35±1.49	52.45±0.48	59.24±0.38	66.09±1.84	62.47±0.83	42.94±0.57	49.58±1.30	46.01±0.63	54.32±0.49	58.54±2.50	56.33-1.14
Struct	56.71±0.87	58.14±1.02	57.41±0.54	63.29±1.16	56.71±3.84	59.77±2.54	52.36±0.47	47.36±1.69	49.72±0.84	59.53±0.85	45.04±2.63	51.25±1.97
ESD	-	-	59.29±1.25	-	-	**69.06±0.80**	-	-	52.16±0.79	-	-	**64.00±0.43**
CONTaiNER	-	-	56.10	-	-	61.90	-	-	48.36	-	-	57.13
SAP	64.05±1.24	62.36±0.88	63.19±0.74	62.51±4.15	64.99±4.26	63.53±1.68	61.04±1.84	47.96±2.30	53.67±1.26	55.14±0.58	59.13±0.46	57.06±0.11
TAP	62.63±2.71	59.70±2.17	61.08±1.46	69.62±1.39	63.79±1.09	66.57±0.96	58.91±1.06	52.21±2.46	55.33±1.45	**65.76±0.88**	61.13±1.64	63.35±0.91
JTSA	**66.89±1.91**	**62.56±1.34**	**64.63±1.06**	**70.46±1.65**	**67.28±3.42**	68.77±1.60	**63.68±1.35**	**52.35±1.34**	**57.38±0.79**	64.53±0.77	**62.06±0.85**	63.26±0.18

For coarse-grained FEW-NERD(INTRA), empirical results reported in Table 4 suggest that the dataset is challenging for all existing models, since the query samples share little information with the reference. However, for various few-shot settings, the improvement of our TAP model relied on our token awareness module is greater compared with the results on FEW-NERD(INTER). The reason is that there are a huge gap between query entities and entities in the support set, while our token awareness has the significant capability to filter out the entities in the support set that are less correlation to the query sample.

Table 4. Overall Performance on FEW-NERD(INTRA). NNshot, Proto, Struct model are from [12] and add Viterbi decoder to the Proto as my baseline ProtoNet, and we evaluate them on our dataset.

FEW_NERD(INTRA)(%)												
Model	**5-way 1~2-shot**			**5-way 5~10-shot**			**10-way 1~2-shot**			**10-way 5~10-shot**		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
NNshot	30.22±0.78	34.75±0.50	32.33±0.66	36.03±0.443	43.92±0.5	39.59±0.27	19.85±1.39	25.68±1.03	22.38±1.25	25.59±0.39	31.44±0.38	28.22±0.38
ProtoNet	30.18±0.89	33.60±1.63	31.78±0.98	46.34±0.73	53.18±1.88	49.52±0.94	23.18±0.88	24.26±2.04	23.67±1.13	39.32±0.99	39.74±2.66	39.51±1.744
Struct	39.25±0.73	33.79±1.25	36.30±0.86	49.09±1.93	35.82±2.61	41.38±2.05	30.63±1.27	22.83±2.08	26.14±1.75	40.58±1.87	19.20±1.51	26.05±1.64
ESD	-	-	36.08±1.60	-	-	52.14±1.50	-	-	30.00±0.70	-	-	42.15±2.6
CONTaiNER	-	-	40.40	-	-	53.71	-	-	33.82	-	-	**47.51**
SAP	48.15±3.64	32.57±1.85	38.74±1.09	47.12±1.12	**53.46±1.73**	50.09±1.37	53.13±0.78	15.84±1.51	24.38±1.79	39.33±2.35	39.73±0.08	39.52±1.15
TAP	49.82±2.79	**41.46±5.52**	44.95±2.90	60.59±2.26	45.78±4.34	52.01±2.42	42.80±2.94	**31.35±2.05**	**36.10±1.26**	**53.85±2.11**	40.16±1.15	46.00±1.29
JTSA	**54.41±1.58**	39.84±4.07	**45.86±2.55**	**60.94±1.25**	52.26±2.20	**56.22±0.88**	**53.61±1.91**	24.89±1.66	33.97±1.58	53.06±1.37	**41.66±1.22**	46.68±1.30

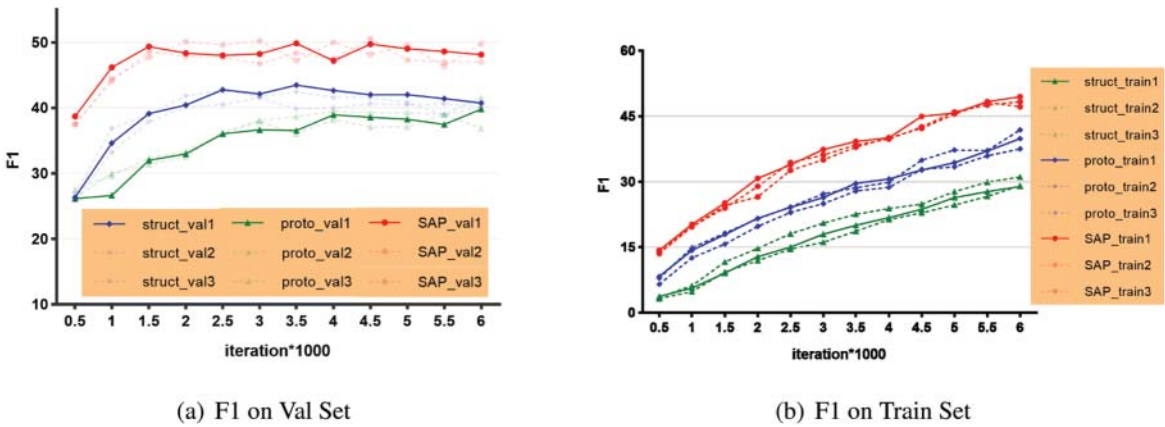
To comprehensively exploit the correlation information of entities and sentences, we construct the joint model JTSA, our mainly proposed method. The token awareness module and the sentence awareness

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

module complement each other and take advantages in different scenarios. As the results in Table 3 and Table 4, our JTSA model achieves superior performance on various few-shot settings; meanwhile, it is better than the structshot model [12] which has strength on the single-shot and the prototypical network which is suitable for multiple-shot setting.

4.5 Convergence Speeds

Firstly, we compare the convergence speed between our proposed model SAP and existing state-of-the-art methods (structshot model [12] and prototypical network [4]) on FEW-NERD(INTER) benchmark on 1-shot tasks. The results are reported in Figure 2. The curve colored red represents our SAP model, while the curve with blue is the structshot model and the curve with green is the prototypical network. Figure 2(a) shows F1 on Val Set, while Figure 2(b) shows the F1 value on Train Set. As shown in the two sub-fig, our SAP model only need half time to arrive at the optimal point, and the performance improves by around 15% compared with the structshot model.



(a) F1 on Val Set (b) F1 on Train Set

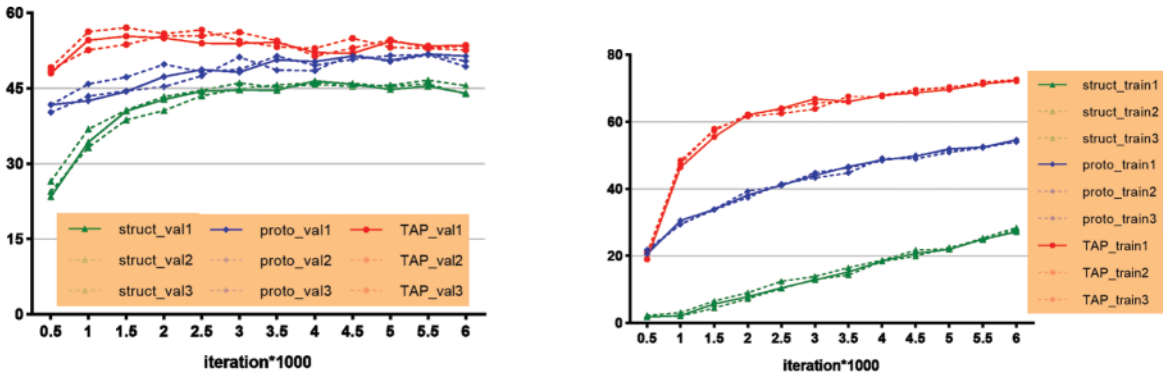
Figure 2. SAP training process.

Secondly, we also compare our TAP model with the two baseline methods on multiple-shot tasks and show the results on the 5-shot task in Figure 3. Although the convergence speed is almost the same initially, our model has better performance on the validation set, which suggests that our model has strong generalization capability. In the second half of time, our TAP model converges at a much faster speed with higher optimal points, and both of the two criteria generally exceed the prototypical network.

4.6 Ablation Study

In this part, we also conduct sets of subsidiary experiments to indicate our main contributions effect, including sentence awareness module, token awareness module, and joint learning scheme.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025



(a) F1 on Val Set (b) F1 on Train Set

Figure 3. TAP training process.

4.6.1 Sentence Awareness Module

To show the effect of our sentence awareness, we take a sample predicted on a 5-way 1-shot task as an example. Given a query sample q and the truth label of "Organization-religion", the NER model needs to distinguish the type label of q. In this experiment, we calculate the distance of each feature dimension between query q and the prototypes respectively, and compare the results of our SAP model with prototypical network. Figure 4 presents the visualization, and the darker the color of the bar, the closer the distance. In this figure, the SAP model provides a higher level of confidence when predicting the label of query q, since most of the prototype feature dimensions gained from our SAP model are more similar to query q. The specific distance over the whole features in our SAP model is 20% lower than prototypical network. To sum up,sentence awareness, which considers the similarity of sentences, is crucial for a prototypical network.

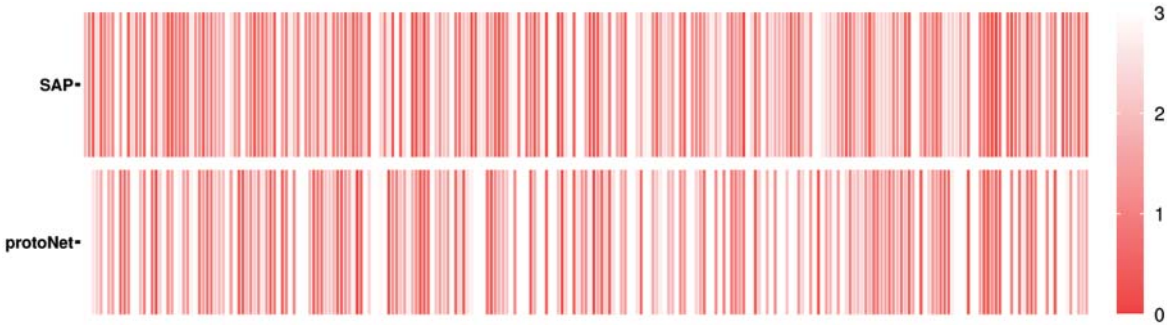


Figure 4. Feature comparsion between query sample q and their corresponding prototypes on SAP and ProtoNet respectively.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

4.6.2 Token Awareness Module

As the experiments show above, we randomly extract samples to evaluate our TAP model on a 5-way 1-shot task, and the query entity comes from the "other-biologything" type. Figure 5 illustrates that the prototypes calculated by our TAP model with token awareness module are more similar to the query q, and the distance of ours is the only 67% of the prototypical network. Thus, the query sample is easier to be classified correctly with the token awareness module.

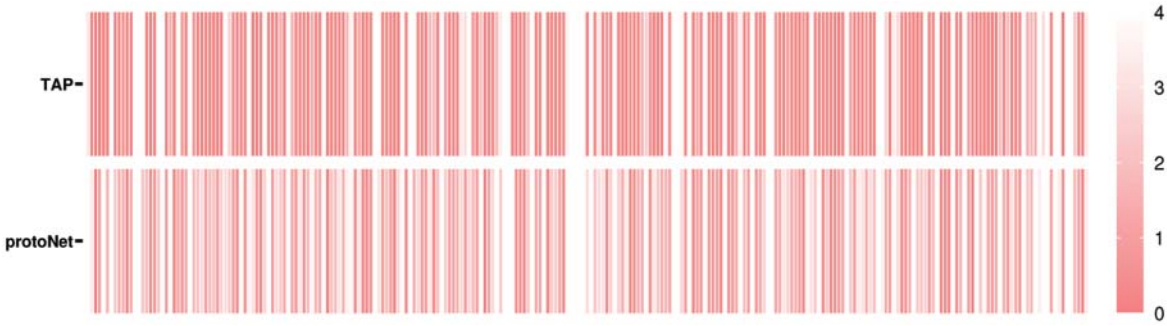


Figure 5. Feature comparsion between query sample q and their corresponding prototypes on TAP and ProtoNet respectively.

4.7 Joint Learning Scheme

To further indicate the effect of our joint learning scheme, we extract two groups of data and evaluate our model on a 5-way task. Figure 6 aims to illustrate how the TAP model with token awareness module corrects the prediction of the SAP model with the sentence awareness module. We show the probability distributions of our SAP, TAP, and JTSA in distinguishing the type label of the entity "stock" that appears in the sentence "It began focusing on foreign exchange transaction in 1976 and listed its shares on the Jakarta stock exchange in 1989". The truth label of the entity "stock" is "organization-government/governmentagency", but the SAP model believes that the entity should be labeled "building-hospital" with over 55% confidence. In this case, our joint model JTSA can take advantage of entity awareness(TAP) and give the correct type label with 80% confidence at last. Figure 7 presents how the sentence awareness module works when

Figure 6. An example of SAP, TAP and JTSA predict the class label of entities "jakarta stock exchange" from the sentences.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

predicting the label of the word "the" in the phrase "republic of the Philippines Commission on elections(Comelec)". We can find that the TAP model is controversial about whether the word "the" belongs to "other-class" or "event-election" and result in misidentification. In contrast, the SAP model gives high confidence to predict the correct type of "event-election". From above, we believe that the joint learning scheme of our JTSA model, which is joint entity awareness and sentence awareness, is meaningful and can achieve the best performance in a variety of scenarios.

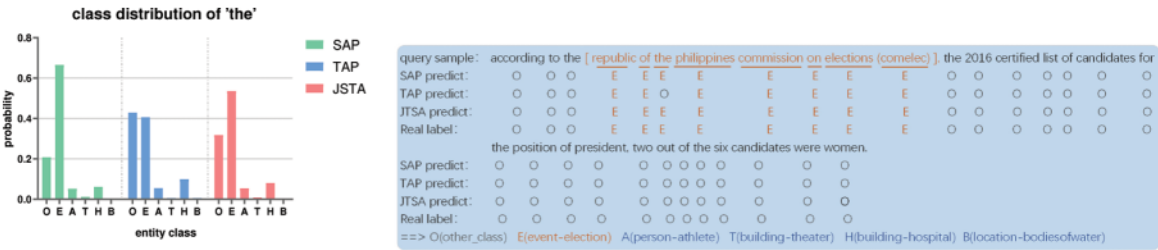


Figure 7. An example of SAP,TAP and JTSA predict the class label of entities "Republic of the philippines commission on elections" from the sentences.

4.8 Error Indicator Analysis

Following Ding et al. [12], we analyze our model from four aspects. Figure 8 presents the comparison results between our proposed model and baselines. All of our model SAP, EAP, and JTSA achieve lower error rates than baselines on most of the situations("FP", "WITHIN", and "OUTER"), for example, the error rate result on "FP" is reduced to 50% compared to the number of traditional prototypical network. The

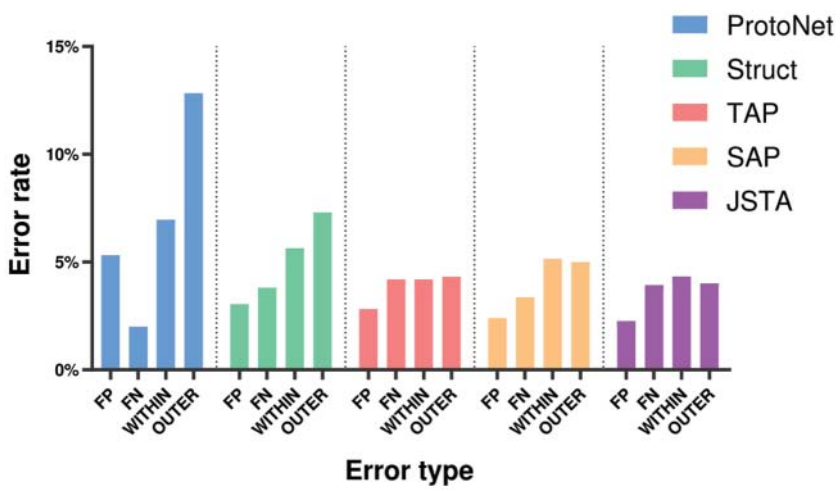


Figure 8. Comparison results on four error indicator. "FP" error stands for an entity with "O" class is predicted to other class. "FN" error indicates an entity is incorrectly predicted to class "O". "WITHIN" error means the coarse-grained class of the query entity is predicted correct, but the fine-grained class label is error. "OUTER" error presents the entityis predicted to a wrong coarse-grained class label.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

results of these experiments sufficiently illustrate that token awareness and sentence awareness effectively recognize the entities with "*O*" class and alleviate the problem in similarity comparison caused by the ambiguity of "*O*" class. On the other hand, our models have the lowest error rate for "WITHIN" and "OUTER" with 10%–24% and 30%–40% reduced respectively, indicating our token awareness mechanism and sentence awareness mechanism is superior on specific classes, especially coarse-grained class with more significant semantic differences. In addition, our joint model JTSA has some reduced on "FP", "WITHIN", and "OUTER" compared with the results of TAP and SAP, which further presents the significance of our joint learning scheme.

5. CONCLUSION

In this paper, we proposed a state-of-the-art named entity recognition FSL model, JTSA. Our model contains 3 modules: a token awareness module, a sentence awareness module, and a joint learning scheme. Token awareness module captures the connections between entities from the token level. Sentence awareness module incorporates sentence information to capture the sentence-level relationships between entities. Then, the joint learning combines these two modules to strengthen the ability to identify entity classes and reduce the error of NER. Experimental results show that our two awareness modules have positive contributions to entity recognition in different contexts, and the joint learning scheme enables our final model to achieve advanced results in both coarse- grained and fine-grained NER.

ACKNOWLEDGEMENTS

The State Key Program of National Natural Science of China, Grant/Award Number:61533018; National Natural Science Foundation of China, Grant/Award Number: 61402220; The Philosophy and Social Science Foundation of Hunan Province, Grant/Award Number: 16YBA323; Natural Science Foundation of Hunan Province, Grant/Award Number: 2020JJ4525,2022JJ30495; Scientific Research Fund of Hunan Provincial Education Department, Grant/Award Number: 18B279,19A439,22A0316.

AUTHOR CONTRIBUTIONS

All authors contributed ideas, text, and review comments in the production of the paper. W. Wen designed the experiment and analyzed the results. W. Wen participated in constructed the model and designed the structure of the model diagram. YB Liu proposed the core idea of the model, and wrote the original draft. Q. Lin constructed and optimized the model, and participated in the discussion of the results. CP Ouyang put forward the research topic and provided important feedback.

REFERENCES

[1] Koch, G., Zemel, R., Salakhutdinov, R.: Siamese neural networks for one-shot image recognition. In ICML Deep Learning Workshop, volume 2. Lille (2015)

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

[2] Vinyals, O., Blundell, C., Lillicrap, T., et al.: Matching networks for one shot learning. In Advances in Neural Information Processing Systems, pp. 3630–3638 (2016)
[3] Sung, F., Yang, Y.X., Zhang, L., et al.: Learning to compare: Relation network for few-shot learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1199–1208 (2018)
[4] Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In Advances in Neural Information Processing Systems, pp. 4077–4087 (2017)
[5] Han, X., Zhu, H., Yu, P.F., et al.: Fewrel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 4803–4809 (2018)
[6] Gao, T.Y., Han, X., Liu, Z.Y., Sun, M.S.: Hybrid attention-based prototypical networks for noisy few-shot relation classification. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pp. 6407–6414 (2019)
[7] Ding, N., Wang, X., Fu, Y., et al.: Prototypical representation learning for relation extraction.(2021)
[8] Yang, S., Liu.: Free lunch for few-shot learning: Distribution calibration (2021)
[9] Kretov Fritzler, M., Logacheva, V.: Few-shot classification in named entity recognition task. Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing (2019)
[10] Tong, M., Wang, S., Xu, B., et al.: Learning from miscellaneous other-class words for few-shot named entity recognition. 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) (2021)
[11] Hou, Y., Mao, J., Lai, Y., et al.: Fewjoint: A few-shot learning benchmark for joint language understanding (2020)
[12] Ding, N., Xu, G., Chen, Y., et al.: Few-nerd: A few-shot named entity recognition dataset (2021)
[13] Sundheim, B., Grishman, R.: Message understanding conference 6: A brief history. In Proceedings of the 16th conference on Computational linguistics - Volume 1 (1996)
[14] Sang, E.F.T.K., De Meulder, F.: Introduction to the conll-2003 shared task: Language-independent named entity recognition. arXiv (2003)
[15] Yadav, V., Bethard, S.: A survey on recent advances in named entity recognition from deep learning models (2019)
[16] Collobert, R., Weston, J.: A unified architecture for natural language processing: deep neural networks with multitask learning. In Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5–9, 2008 (2008)
[17] Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. Computer Science (2013)
[18] Huang, Z., Wei, X., Kai, Y.: Bidirectional lstm-crf models for sequence tagging. Computer Science (2015)
[19] Kim, Y., Jernite, Y., Sontag, D., Rush, A.M.: Character-aware neural language models (2015)
[20] Chiu, J., Nichols, E.: Named entity recognition with bidirectional lstm-cnns. Computer Science (2015)
[21] Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (2016)
[22] Ringland, N., Dai, X., Hachey, B., et al.: Nne: A dataset for nested named entity recognition in english newswire. In Meeting of the Association for Computational Linguistics (2019)
[23] Feng, X.C., Feng, X.C., Qin, B., et al.: Improving low resource named entity recognition using cross-lingual knowledge transfer. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, pages 4071–4077. International Joint Conferences on Artificial Intelligence Organization, 7 (2018). doi: 10.24963/ijcai.2018/566. URL https://doi.org/10.24963/ijcai.2018/566.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

[24] Hofer, M., Kormilitzin, A., Goldberg, P., Nevado-Holgado, A.: Few-shot learning for named entity recognition in medical text (2018)
[25] Yang, Y., Katiyar, A.: Simple and effective few-shot named entity recognition with structured nearest neighbor learning (2020)
[26] Li, J., Chiu, B., Feng, S.S., Wang, H.: Few-shot named entity recognition via meta-learning. IEEE Transactions on Knowledge and Data Engineering (2020)
[27] Huang, J., Li, C., Subudhi, K., et al.: Few-shot named entity recognition: A comprehensive study (2020)
[28] Devlin, J., Chang, M.W., Lee, K., Toutanova K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2018)
[29] Wang, P., Xu, R., Liu, T., et al.: An enhanced span-based decomposition method for few-shot sequence labeling (2021)
[30] Das, S., Katiyar, A., Passonneau, R.J., Zhang, R.: Container: Few-shot named entity recognition via contrastive learning (2021)

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

AUTHOR BIOGRAPHY

Wen Wen received his M.E degree from the University of South China, China, in 2021. Her research interests focus on Relation Extraction and Few-shot learning.

Yongbin Liu received a Ph.D. degree from the University of Science & Technology Beijing, China, in 2013. From 2013 to 2015, he was a post-doc research fellow at Tsinghua University. He is an associate professor at the University of South China. His research interests include natural language processing and knowledge engineering.

Qiang Lin is studying for a master's degree in computer technology at the University of South China. His research interests include Information Extraction and Few-shot learning.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025

Chunping Ouyang received a Ph.D. degree from the University of Science & Technology Beijing, China, in 2011. She is a professor of computer science at the University of South China and supervisor of postgraduate. Her research interests include natural language processing and information retrieval.

Downloaded from http://direct.mit.edu/dint/article-pdf/5/3/767/2158274/dint_a_00195.pdf by guest on 15 January 2025