applied sciences

MDPI

Article

SLNER: Chinese Few-Shot Named Entity Recognition with Enhanced Span and Label Semantics

Zhe Ren [1,2], Xizhong Qin [1,2,*] and Wensheng Ran [3]

[1] College of Information Science and Engineering, Xinjiang University, Urumqi 830049, China; 107552103740@stu.xju.edu.cn
[2] Xinjiang Key Laboratory of Signal Detection and Processing, Urumqi 830049, China
[3] Xinjiang Uygur Autonomous Regin Product Quality Supervision and Inspection Institute, Urumqi 830049, China; rws@xju.edu.cn
* Correspondence: qinxz@xju.edu.cn

Abstract: Few-shot named entity recognition requires sufficient prior knowledge to transfer valuable knowledge to the target domain with only a few labeled examples. Existing Chinese few-shot named entity recognition methods suffer from inadequate prior knowledge and limitations in feature representation. In this paper, we utilize enhanced **S**pan and **L**abel semantic representations for Chinese few-shot **N**amed **E**ntity **R**ecognition (**SLNER**) to address the problem. Specifically, SLNER utilizes two encoders. One encoder is used to encode the text and its spans, and we employ the biaffine attention mechanism and self-attention to obtain enhanced span representations. This approach fully leverages the internal composition of entity mentions, leading to more accurate feature representations. The other encoder encodes the full label names to obtain label representations. Label names are broad representations of specific entity categories and share similar semantic meanings with entities. This similarity allows label names to offer valuable prior knowledge in few-shot scenarios. Finally, our model learns to match span representations with label representations. We conducted extensive experiments on three sampling benchmark Chinese datasets and a self-built food safety risk domain dataset. The experimental results show that our model outperforms the F1 scores of 0.20–6.57% of previous state-of-the-art methods in few-shot settings.

Keywords: natural language processing; Chinese named entity recognition; few-shot learning; feature representation; label semantics; neural network; deep learning; attention mechanism; low-resource domain dataset

check for updates

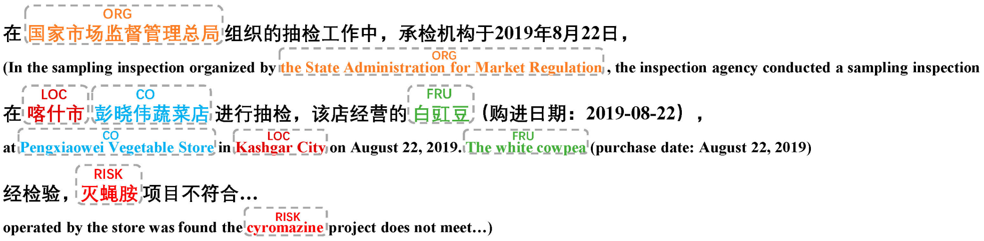
Citation: Ren, Z.; Qin, X.; Ran, W. SLNER: Chinese Few-Shot Named Entity Recognition with Enhanced Span and Label Semantics. *Appl. Sci.* **2023**, *13*, 8609. https://doi.org/10.3390/app13158609

Academic Editor: José Salvador Sánchez Garreta

Received: 8 June 2023
Revised: 12 July 2023
Accepted: 25 July 2023
Published: 26 July 2023

1. Introduction

Named entity recognition (NER) aims to identify specific, meaningful entities from text, such as LOCATION and ORGANIZATION, and classify them into predefined categories, as shown in Figure 1. NER is an essential prerequisite task for many natural language processing task, such as information extraction [1], question-answering systems [2], and machine translation [3].



在国家市场监督管理总局组织的抽检工作中，承检机构于2019年8月22日，

(In the sampling inspection organized by the State Administration for Market Regulation, the inspection agency conducted a sampling inspection

在喀什市彭晓伟蔬菜店进行抽检，该店经营的白豇豆（购进日期：2019-08-22），

at Pengxiaowei Vegetable Store in Kashgar City on August 22, 2019. The white cowpea (purchase date: August 22, 2019)

经检验，灭蝇胺项目不符合...

operated by the store was found the cyromazine project does not meet...)

Figure 1. Example of an NER task. The entities to be recognized are highlighted within dashed boxes, and different colors represent different entity types.

In recent years, neural-network-based techniques have been widely applied in NER [4,5]. However, neural networks are data-driven machine learning methods, and the quantity of training data often limits their performance. Unfortunately, annotated data used for training are often scarce and expensive, especially in specific domains (e.g., the food safety risk domain). Therefore, there has been widespread interest in a challenging yet practical research field: few-shot NER.

One of the challenges of few-shot NER is how to accurately incorporate prior knowledge to effectively classify unseen entity types when confronted with a few examples. Recently, similarity-based methods such as prototype networks have been extensively studied and achieved great success for few-shot learning [6–8]. The core idea is to classify input examples from a new domain based on the similarity between their representations and those of each class in the support set. However, this approach experiences a significant drop in performance in a few-shot setting due to the limited representativeness of the data. The prompt-based approach [9,10], by manually or automatically adding prompt words to sentences, guides the model to learn more quickly and accurately while reducing the gap between pretraining and fine tuning. This approach has shown remarkable performance in few-shot learning. However, these methods do not directly leverage the rich prior knowledge contained in label semantics.

In addition, Chinese NER is more challenging than English NER due to the relatively ambiguous entity mentions in Chinese, which limits feature representation and affects the accuracy of NER. Zhang and Yang [11] addressed this issue by using lattice LSTM to represent the entity in sentences and by incorporating the potential lexical information into a character-based LSTM-CRF model. While this character-based representation effectively solves the segmentation error problem, it requires the introduction of a complex external lexicon. Some more recent attempts have switched to span-based feature representations for Chinese NER [12,13], explicitly utilizing span-level information to address token-wise label dependency and better handle nested entities. However, these span-based feature representations only perform simple concatenation of the start and end positions of the span without fully exploiting the internal information of the span, which limits the feature representation of the named entity. For example, when the named entity "伊河谷食品科技有限公司 (Yihegu Food Technology Limited Company, Urumqi, China)" has the internal information "科技有限公司 (Technology Limited Company)", it becomes easier to classify it as an ORGANIZATION entity.

In response to these challenges, we propose a model called SLNER with enhanced span and label semantic representations to tackle the challenges of Chinese few-shot NER. Specifically, SLNER utilizes two encoders. One encoder is used to encode the text and its spans. This module captures the head and tail information of spans using the biaffine attention mechanism and incorporates self-attention to capture the internal information of spans. Ultimately, these representations are fused to obtain enhanced span representations. This approach fully utilizes the internal composition of entity mentions to achieve more accurate feature representations. In contrast to traditional span-based methods that concatenate the start and end positions of entity mentions, our approach fully exploits the information of each token within entity mentions, providing sufficient and essential clues for entity recognition.

The other encoder is used to encode full label names. Label names are highly generalized specific entity categories and exhibit similar semantics to entities, which can provide additional prior knowledge in few-shot scenarios. Compared to traditional similarity-based methods (e.g., prototype networks), label semantics provide more generic similarity representations, especially in situations where the target domain has a scarcity of samples.

Ultimately, our model learns to match span representations with label representations. We employ a two-stage training strategy using source and target domains, enabling the model to transfer knowledge from the high-resource source domain to the low-resource target domain.

Furthermore, to promote research and applications in low-resource domains, we developed an NER dataset named RISK, which was specifically designed for the food safety risk domain. RISK comprises 5 coarse-grained and 20 fine-grained entity types, each labeled and organized in a hierarchical structure of coarse-grained + fine-grained. We also conducted a performance evaluation of our model on the RISK dataset, and the experimental results demonstrate the challenging nature of the RISK dataset. Constructing an NER dataset in the food safety domain can drive advancements in related research areas such as food traceability and food safety regulation. Additionally, this dataset can serve as a foundation for the development of applications in food safety, including food safety warning systems and food recall management.

We have documented the experimental results on three sampling benchmark Chinese NER datasets and a self-built food safety risk domain dataset. Our contributions can be summarized as follows:

- We propose a simple and effective model named SLNER, which leverages enhanced span representations and label semantics to address the issues of inadequate prior knowledge and limitations in feature representation in Chinese few-shot named entity recognition;
- We created a challenging food safety risk domain dataset, RISK, which is divided into 5 coarse-grained and 20 fine-grained entity categories. This dataset provides data support for the development of named entity recognition applications in the domain of food safety;
- Our proposed model achieved promising performance on the four sampling Chinese NER datasets (including our self-built dataset). Specifically, our model outperformed previous works with F1 scores ranging from 0.20% to 6.57% in different few-shot settings (following the settings of PCBERT) on the Ontonotes, MSRA and Resume datasets. It also achieved promising F1 scores on our self-built RISK dataset.

2. Related Work

2.1. Few-Shot NER

Compared to traditional standard NER [14,15], few-shot NER aims to learn how to recognize named entities with limited data. Currently, research on few-shot named entity recognition based on the traditional standard NER framework primarily focuses on two aspects: incorporating prior knowledge (external information) at the data level and enhancing model generalization at the model level.

In the first aspect, Tong et al. [16] proposed a model, Mining Undefined Classes from Other-class (MUCO), that can automatically induce different undefined classes from the other class to improve few-shot NER. On the other hand, Cui et al. [9], Lee et al. [17], and Chen et al. [18], using a prompt-based approach, treated named entity recognition as a generation task. By manually or automatically adding templates to the training data, the model is trained to predict the [MASK] positions within the templates. This method reduces the gap between pretraining and fine tuning, allowing the model to perform better, even with limited data. Based on multitask instructions, Wang et al. [19] enhanced the knowledge of the training data by adding different auxiliary tasks and corresponding instructions and options; this allowed the model to accurately identify entity information, even with few samples. Additionally, Chen et al. [20] effectively leveraged illustrative instances to precisely transfer knowledge from external resources by describing both entity types and mentions using a universal concept set.

For the second aspect, methods based on meta-learning [21] aim to enable machines to "learning to learn". The essence of these methods is to train the network model to learn a more robust initialization, allowing it to quickly generalize to new tasks with only a few examples. Methods based on contrastive learning also focus on learning shared features among instances of the same class and distinguishing differences between instances of different classes. Das et al. [22] trained an encoder that produces similar encodings for instances of the same class while ensuring that the encodings of different classes are as dissimilar as possible. This approach reduces the limitations of generalization in the target domain, thereby improving performance in the few-shot setting [23], which augments the distribution of entity labels by assigning k-nearest neighbors retrieved from the training set. This strategy makes the model more capable of handling long-tail cases, along with better few-shot learning abilities.

2.2. *Span-Based Method*

Previously, the majority of research work treated NER as a sequence-labeling task, where entity classification is performed at the token level. These methods often employ the BIO tagging scheme, where each character's category is determined to achieve entity recognition for Chinese NER tasks, as shown in Figure 2. As a representative example, Huang et al. [24] utilized BiLSTM as an encoder to learn contextual representations, then employed a conditional random field (CRF) as a decoder to label tokens. Additionally, leveraging the power of pretrained language models such as ELMo [25] and BERT [5] significantly improved the performance of NER.

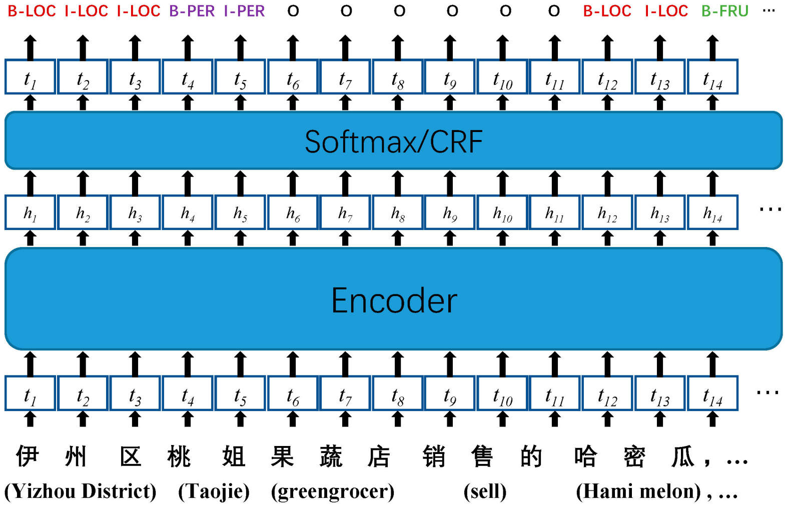

B-LOC I-LOC I-LOC B-PER I-PER O O O O O O B-LOC I-LOC B-FRU ···
t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10} t_{11} t_{12} t_{13} t_{14}
Softmax/CRF
h_1 h_2 h_3 h_4 h_5 h_6 h_7 h_8 h_9 h_{10} h_{11} h_{12} h_{13} h_{14} ···
Encoder
t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10} t_{11} t_{12} t_{13} t_{14} ···
伊 州 区 桃 姐 果 蔬 店 销 售 的 哈 密 瓜，…
(Yizhou District) (Taojie) (greengrocer) (sell) (Hami melon) , …

Figure 2. Traditional sequence-labeling NER method.

While the aforementioned token-based NER methods have achieved considerable success, they suffer from two inherent limitations: token-wise label dependency and difficulty handling nested entities. As shown in Figure 2, this method may fail to correctly classify nested entities such as "Taojie greengrocer" and "Hami Melon". Additionally, if errors occur during the labeling process, incorrect beginning tags (B) or internal tags (I) may affect subsequent labeling, resulting in the erroneous tagging of the entire entity.

Therefore, recent work has increasingly embraced span-based approaches to address NER tasks. This method involves partitioning all possible spans in a sentence into predefined types (e.g., PER or LOC) and determining whether a given text span belongs to a particular category, as shown in Figure 3. Yu et al. [26] adopted a biaffine attention model to assign scores to all potential spans and achieved state-of-the-art performance on both flat and nested English NER datasets. Shen et al. [27] also employed a span-based framework for Chinese NER datasets, effectively addressing the issue of nested entities. Yu et al. [28] and Wang et al. [29] proposed span-level metric learning to bypass the token-wise label dependency problem while still explicitly utilizing phrase representations. Wang et al. [30] introduced a span-based prototype network and a global boundary matrix to learn explicit span boundary information.

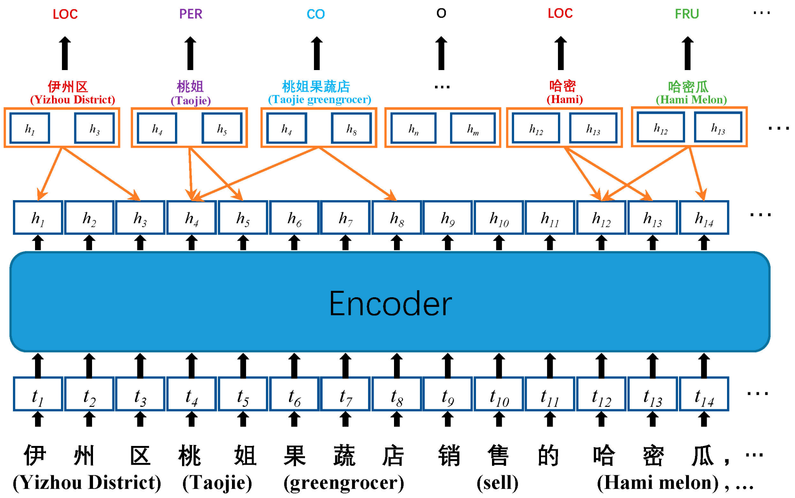

LOC
PER
CO
O
LOC
FRU
…
伊州区
(Yizhou District)
桃姐
(Taojie)
桃姐果蔬店
(Taojie greengrocer)
…
哈密
(Hami)
哈密瓜
(Hami Melon)
h_1 h_3
h_4 h_5
h_4 h_6
h_n h_m
h_{12} h_{13}
h_{12} h_{13}
h_1 h_2 h_3 h_4 h_5 h_6 h_7 h_8 h_9 h_{10} h_{11} h_{12} h_{13} h_{14} …
Encoder
t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10} t_{11} t_{12} t_{13} t_{14} …
伊 州 区 桃 姐 果 蔬 店 销 售 的 哈 密 瓜，…
(Yizhou District) (Taojie) (greengrocer) (sell) (Hami melon) , …

Figure 3. Span-based NER method.

In this study, we also utilize a span-based approach for named entity recognition and extensively explore the internal information of entity word spans to obtain enhanced span feature representations.

2.3. Label Semantics

In the task of few-shot text classification, Luo et al. [31] directly appended the label name to the text input in BERT, obtaining feature vectors with more enriched semantic information, thus demonstrating the effectiveness of label semantics in few-shot scenarios. Cui et al. [9] employed a similar approach by reconstructing the input text template, reframing NER as a cloze task, and using a sequence-to-sequence model to fill the entity label names in predefined templates, achieving few-shot named entity recognition through a prompt-based method. Ma et al. [32] abandoned the template construction process while retaining the word prediction paradigm of pretrained models to predict class-related pivot words (or label words) at entity positions. Inspired by prompt-tuning methods, Zhong et al. and Ye et al. [33,34] initialized markers in the NER task not with random initialization but with meaningful words (such as label names), resulting in a certain degree of improvement in model performance. Our proposed model leverages the connection between label semantics and entity spans (Figure 4) to learn aligning span representations with label representations, exhibiting promising results.

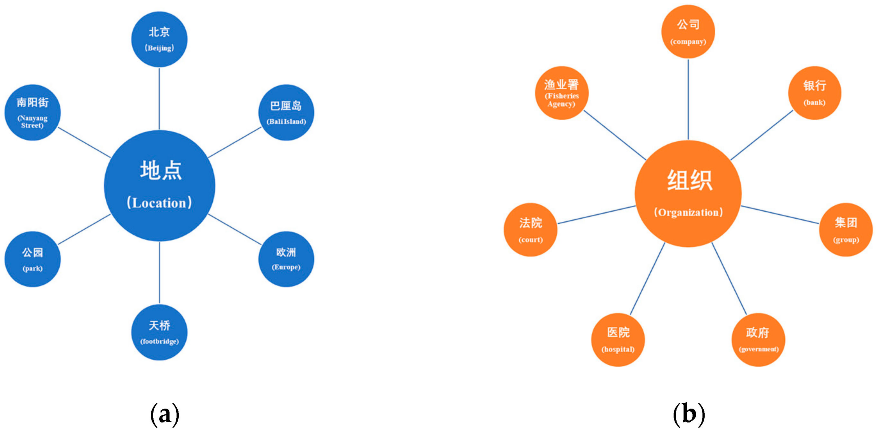


Figure 4. Different entity labels. (**a**) The label 'Location' and its associated entities. (**b**) The label 'Organization' and its associated entities.

3. Method

This chapter first formalizes the problem of few-shot named entity recognition (Section 3.1). Then, we propose our model, SLNER, for Chinese few-shot NER (Section 3.2). The model consists of two encoders: one for encoding the text and its span to obtain better feature representations (Section 3.3.1) and another for encoding the full label name to capture additional prior knowledge (Section 3.3.2). Additionally, we adopt a two-stage training strategy using source and target domains (Section 3.4). The details are outlined as follows.

3.1. Few-Shot NER Task Formalization

For the few-shot NER task, assume that we have a resource-rich source domain NER dataset, $\mathcal{D}^R = \{[T_1, L_1], \cdots, [T_n, L_n]\}$, where $T_i = \{t_1, t_l\}$ represents the i-th text ($i \in [1, n]$), t_i represents the i-th token ($i \in [1, l]$), and L_i represents the labels corresponding to the entity spans in the i-th text ($i \in [1, n]$). We use $\mathbb{C}^R$ to denote the label set of the source domain dataset. Then, given a resource-scarce target domain dataset, $\mathcal{D}^T = \{[T_1, L_1], \cdots, [T_m, L_m]\}$, the number of texts in the target domain dataset is limited (i.e., $m \ll n$), and the label types in the target domain may differ from those in the source domain (i.e., $\mathbb{C}^R \neq \mathbb{C}^T$). We aim to leverage the knowledge from the source domain dataset to improve the model's performance on the target domain dataset.

3.2. Overall Structure

The overall architecture of the SLNER model is illustrated in Figure 5. For span representation, given a sentence ($T = \{t_1, \cdots, t_l\}$) of length l, we use BERT as our encoder, which encodes the context of the i-th token in the sentence as follows:

$$h_i = BERT(t_i) \qquad \forall t_i, t_i \in T \tag{1}$$

where d_h is the hidden dimension of the encoder, and the output dimension after passing the original sentence through the encoder is $\mathbb{R}^{l \times d_h}$.

To further enhance the modelling of the sequential order of the text, the embedding representation obtained from BERT is then passed through a bidirectional LSTM layer. The forward LSTM network captures the hidden forward states (historical features), while the backward LSTM network captures the hidden backward states (future features), resulting in a context-aware encoding representation:

$$x_i = \left[\overrightarrow{LSTM}(h_i, x_{i-1})\, ; \ \overleftarrow{LSTM}(h_i, x_{i+1}) \right] \tag{2}$$

At this stage, the output dimension of the original sentence through the bidirectional LSTM layer is $\mathbb{R}^{l \times d_x}$.

We predefine an n-gram value (w), which represents the maximum length of spans that can be formed in a text. The number of possible spans that can be formed in a sentence of length l is given by:

$$N_s = \begin{cases} w[l - (w - 1)] + \sum_{i=1}^{w-1} i & l \geq w \\ \sum_{i=1}^{w-1} i & l < w \end{cases} \tag{3}$$

Next, the token-embedding sequence obtained from the LSTM layer is used to construct the span-level feature vector representation (h_{span}) through a span extractor (details in Section 3.3.1). The dimension of h_{span} is finally expanded to $\mathbb{R}^{N_s \times d_h}$ through a feed-forward layer.

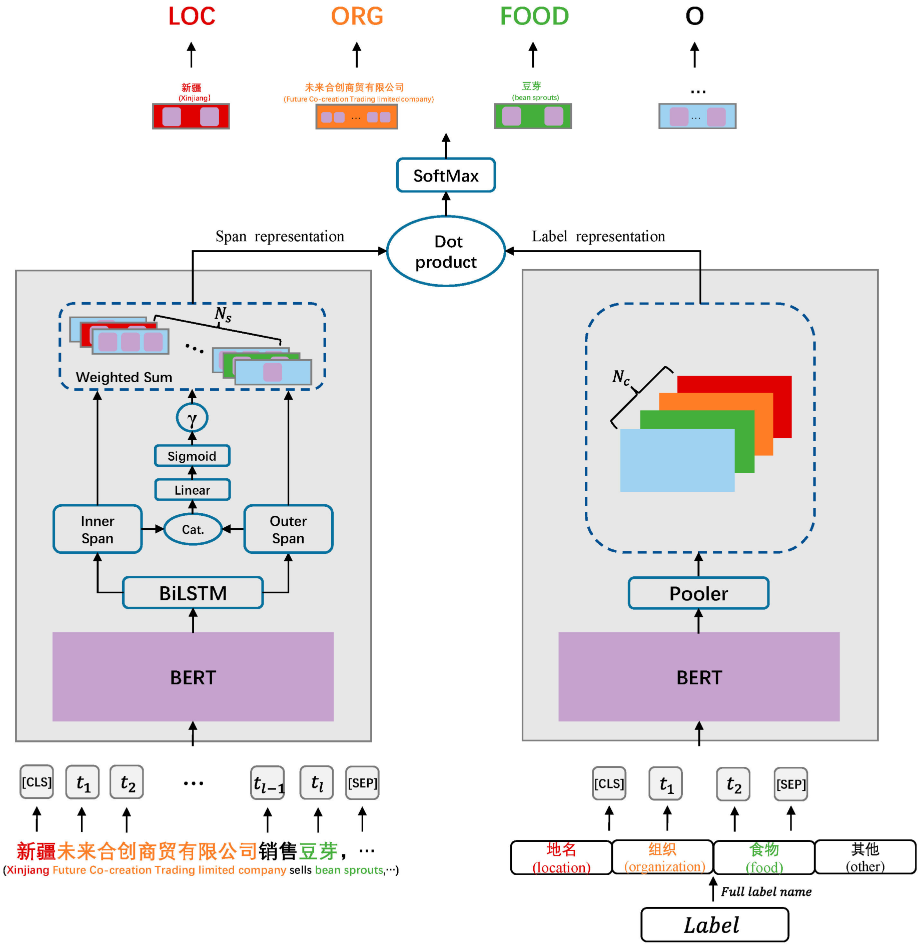


Figure 5. The overall structure of SLNER. The grey module on the left learns span representations, while the grey module on the right learns label representations. The model's final predictions are calculated through distance matching. N_s represents the number of spans, and N_c represents the number of entity categories.

For label representation (details in Section 3.3.2), we manually define the appropriate full label name for each label. Similarly, we use BERT as the encoder and directly encode the label using Equation (1) to obtain the global semantic feature ($h_{[CLS]}$). The difference from span encoding is that we further pass $h_{[CLS]}$ through a pooler layer to obtain the semantic feature vector representation (h_{label}), which serves as the final representation of the label:

$$h_{label} = Tanh\left(W_{(p)} \times h_{[CLS]} + b_{(p)}\right) \tag{4}$$

where $W_{(p)}$ represents the weight parameters in the pooler layer, $b_{(p)}$ represents the bias parameters, and $Tanh$ is the activation function. The dimension of h_{label} needs to be expanded to $\mathbb{R}^{d_h \times N_c}$, where N_c is the number of entity categories.

According to our approach, there is a correlation between labels and spans appearing in the text. Therefore, we capture this correlation through the dot product:

$$\mathrm{H} = h_{LST} \times h_{span} \tag{5}$$

where the similarity matrix (H) has dimensions of $\mathbb{R}^{N_s \times N_c}$. We use a standard linear classifier with a softmax function to predict the entity type for each span, resulting in the final predicted output ($\hat{y}_{span}$):

$$\hat{y}_{span} = Softmax\left(W_{(c)} \times h_{span} + b_{(c)}\right) \tag{6}$$

where $W_{(c)}$ is the trainable parameter of the classifier, and $b_{(c)}$ is the bias. Finally, we use the cross-entropy loss function to compute the loss, which measures the difference between the predicted results and the ground truth labels:

$$\mathcal{L}_{PSNNER} = -\sum y_{span} \log(\hat{y}_{span}) \tag{7}$$

3.3. Specific Structure

3.3.1. Enhanced Span Representation

In previous span-based NER models [35], it was common practice to concatenate the embedding information of the start-position token and the end-position token of the entity (referred to as the "outer span") to represent the span of that entity, which is then used for the final classification decision:

$$h_{span} = [x_{start}; x_{end}] \tag{8}$$

This approach lacks interaction between the start and end tokens and fails to fully utilize the informative content within the span. Moreover, this span representation is coarse-grained. To address these limitations, Yu et al. [26] proposed a biaffine decoder that utilizes two fully connected layers to enable interaction between the start and end tokens while simultaneously predicting the span type. However, in this biaffine method, the information within the span is still ignored.

To fully utilize the informative content within the span, we employ enhanced span to generate the final span representation (as shown in Figure 6). Specifically, we pass the token-embedding information through the outer and inner span modules. The outer span module, similar to the biaffine decoder method, utilizes the biaffine attention mechanism to obtain the outer span representation:

$$h_{outer\ span} = h_{start}^{T} U h_{end} + W(h_{start} \oplus h_{end}) + b \tag{9}$$

where h_{start} and h_{end} represent the start and end token embedding of spans in a text, respectively; U and W are learnable parameters; and b is the bias. The dimension of $h_{outer\ span}$ is expanded to $\mathbb{R}^{N_s \times d_x}$.

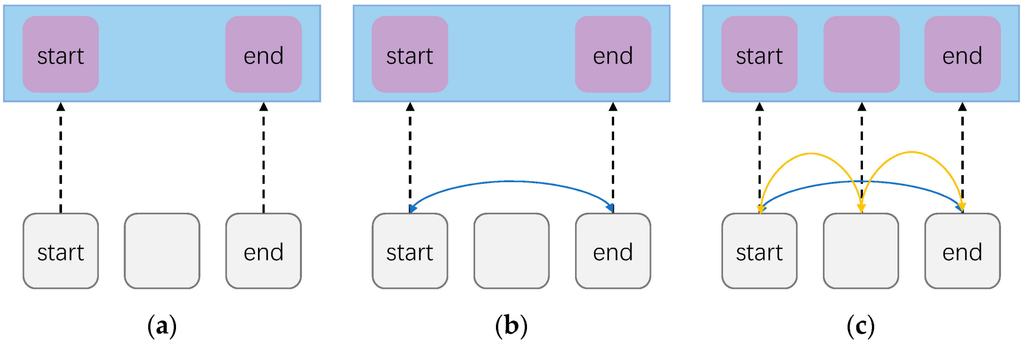


Figure 6. Different ways of span representation. (**a**) Simple start- and end-token concatenation as a span representation. (**b**) Span representation using the biaffine decoder method, which allows for information interaction between the start and end tokens (indicated by blue arrows). (**c**) Span representation method used in our model, which incorporates the interaction between the tokens within the span (indicated by yellow arrows).

We designed the inner span module to capture the token-level information within the span. For this purpose, we use linear attention to generate information interaction for each token. Specifically, this module uses span and the start- and end-position indices of the span as input. It starts by applying a feed-forward neural network (FFNN) to non-linearly transform the input representations, obtaining context-aware representations. Then, it computes normalized scores for each position. Finally, these representations are fed into a self-attention layer, which combines the representations of each position with those of other positions, weighted by the attention scores. This allows the model to capture potential relationships between the tokens within the span. The result is the inner span representation:

$$a_i = W_{(f)} \times x_i + b_{(f)} \tag{10}$$

$$s_i = \frac{exp(a_i)}{\sum_{k=n}^{m} exp(a_k)} \tag{11}$$

$$h_{inner\ span} = \sum_{i=n}^{m} s_i \times x_i \tag{12}$$

where x_i represents the hidden representation from the bidirectional LSTM, and $W_{(f)}$ and $b_{(f)}$ are the learnable weights and biases of the feed-forward neural network, respectively. The indices $i \in \{n, n+1, \cdots, m\}$ correspond to the token indices within the span, where n and m represent the start and end indices of the span, respectively. When $n = m$ (indicating a span of length 1), we do not extract additional features and simply use the hidden representation (x_i).

To predict the entity type, we integrate the outer span representation and inner span representation in a gate network to obtain the weight coefficient (γ) (as shown in Figure 7):

$$\gamma = \sigma\Big(U_{(g)}\big[h_{outer\ span}; h_{inner\ span}\big] + b_{(g)}\Big) \tag{13}$$

where $U_{(g)}$ and $b_{(g)}$ are trainable parameters of the gate network, and σ represents the sigmoid function. The dimension of γ is $\mathbb{R}^{N_s \times 1}$.

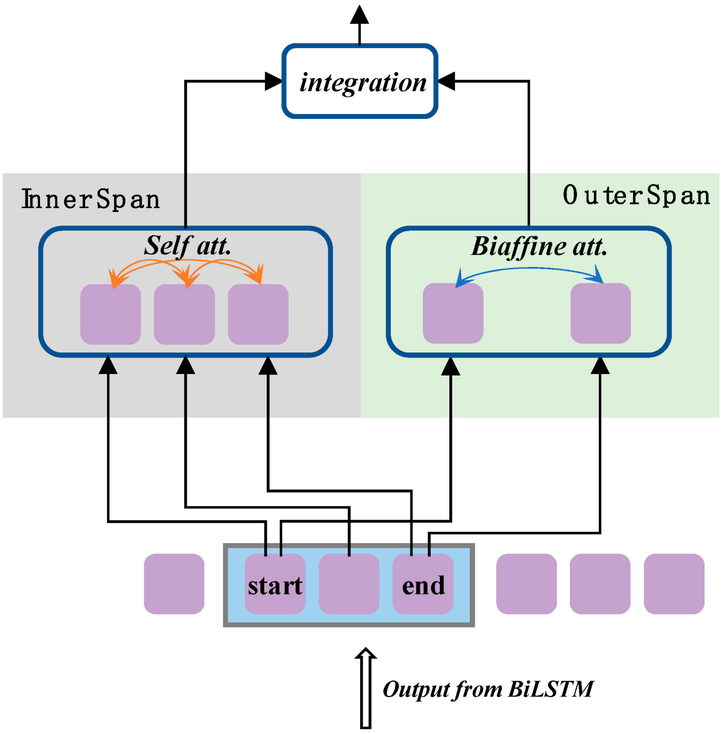


Figure 7. Enhanced span representation.

The final enhanced span representation is obtained by weighting the inner span representation and outer span representation using γ:

$$h_{span} = \gamma \odot h_{outer\ span} + (1 - \gamma) \odot h_{inner\ span} \tag{14}$$

where $\odot$ represents element-wise multiplication, and the resulting h_{span} has dimensions of $\mathbb{R}^{N_s \times d_x}$.

3.3.2. Label Representation

We believe that label semantics can provide additional prior knowledge. Label semantics carry the semantic information of entities in the same category, as this information is manually summarized and induced from a large amount of data. Therefore, when data are limited, especially in small-sample scenarios, we can introduce label semantics to allow our model to make generalizations from the available data. Furthermore, full label names themselves are mentions that appear in various contexts within the text. Their frequencies are synchronized to some extent with the corresponding entity words of their respective categories. Thus, there exists a semantic correlation between label names and the span tokens appearing in the text, and this correlation can be leveraged and utilized.

Considering that our label encoder is based on BERT and incorporates prior knowledge from pretraining, our label representation module allows any form of text to be used as input. This design not only enables easy and rapid expansion to unseen label sets in low-resource domains but also prevents the model from forgetting prior knowledge. We experimented with different label forms and analyzed their effects (Section 5.1). Table 1 presents the final forms of full label names used in this study (for the non-entity type in each dataset, we uniformly use "其他 (other)" as the label name).

3.4. Training Strategy

Compared to the previous work of traditional NER neural architecture, our model does not require a new randomly initialized top-layer classifier for new datasets with new unseen label names. Therefore, our model allows domain transfer for different label categories, which is very beneficial for few-shot learning. On this basis, we adopt a two-stage training program. In the first stage, we pretune our model on the source dataset to obtain a prior knowledge-rich source domain model. In the second stage, we fine tune the source model from the previous stage as the initial model on the target domain dataset. During model training, two encoders are updated at each iteration of the two stages, which helps align the span-embedding space with the label-embedding space.

Table 1. The full label name format of the four datasets we used.

Dataset	Entity Label	Full Label Name
Ontonotes	PER	人名 (person name)
	ORG	组织 (organization)
	LOC	位置 (location)
	GPE	地名 (geographic name)
MSRA	NS	位置 (location)
	NR	姓名、名字 (person name)
	NT	组织 (organization)
Resume	CONT	国籍 (nationality)
	EDU	教育背景、学历 (educational background)
	LOC	位置、地名 (location)
	NAME	姓名、名字 (person name)
	ORG	组织 (organization)
	PRO	专业 (profession)
	RACE	民族 (race)
	TITLE	职称、职业 (title and occupation)

Table 1. *Cont.*

Dataset	Entity Label	Full Label Name
RISK	LOC-PROV	省 (province)
	LOC-PREF	市、区 (city, district)
	LOC-COUNT	县 (county)
	FOOD-VEG	蔬菜 (vegetable)
	FOOD-FRUIT	水果 (fruit)
	FOOD-MEAT	肉 (meat)
	FOOD-GRAIN	粮食 (foodstuff)
	FOOD-DAIRY	奶制品、饮品 (dairy products and beverages)
	CO-PROC	加工生产方式 (processing and production methods)
	CO-PROD	公司、厂 (company, factory)
	CO-TRAD	售卖商店 (sales store)
	CO-CATE	饭店 (hotel)
	CO-MATE	超市 (supermarket)
	ORG-REGU	监督局 (supervisory authority)
	ORG-ADMI	管理部门 (management department)
	ORG-LEGA	法律机构 (legal agency)
	ORG-SOCI	社会组织 (social organization)
	RISK-HIGH	病毒 (virus)
	RISK-MID	化学物质、细菌 (chemicals, bacteria)
	RISK-LOW	添加剂 (additive)

4. Experiments

4.1. Datasets

- **Target Domain Datasets**

To validate the effectiveness of our model, we used three benchmark Chinese datasets and a self-built dataset as target domain datasets:

Ontonotes 4.0 [36]: This dataset comprises corpora from the news and broadcast domains, covering four entity types.

MSRA [37]: This dataset comprises corpora from the news domain and includes three entity types.

Resume [11]: This dataset consists of abstracts from resumes of senior managers in publicly listed companies. It encompasses eight entity types.

RISK: This dataset was created by us and focuses on food safety risk-related corpora. We hierarchically categorized the dataset into 5 coarse-grained and 20 fine-grained entity types, as shown in Figure 8. The entity count for each fine-grained category is illustrated in Figure 9. The distribution of entity categories is not uniform, making RISK challenging at the fine-grained level.

The detailed statistics of the target domain datasets are presented in Table 2. To ensure the comparability of experimental results, we followed the sampling approach of PCBERT [10] to simulate a few-shot scenario (see Section 4.4 for details). Compared to the N-way K-shot setting, we believe that this sampling approach is more representative of realistic few-shot scenarios. We further divided the datasets into scenarios with even fewer examples and tested our model, demonstrating its effectiveness even in scenarios with fewer data.

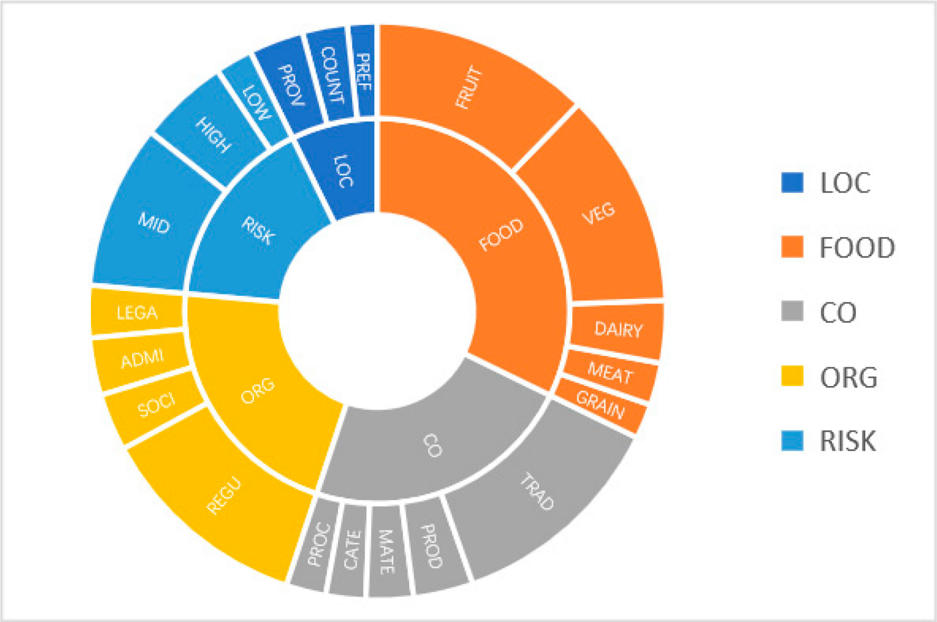


Figure 8. Hierarchy chart of the RISK dataset.

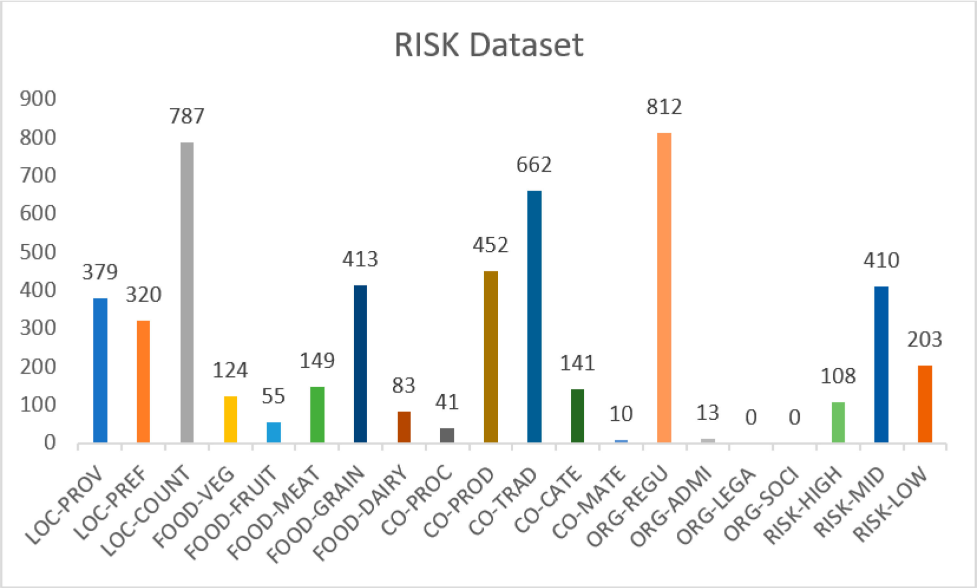


Figure 9. Statistical chart of the number of various entity types in the RISK dataset.

Table 2. Detailed statistical table for the target domain dataset.

Dataset	Train	Dev	Test	Entity Types
Ontonotes	15.7 k	4.3 k	4.3 k	4
MSRA	41.7 k	4.6 k	4.3 k	3
Resume	3.8 k	0.46 k	0.48 k	8
RISK	1.2 k	0.26 k	0.26 k	5 (20) [1]

[1] 5 (20) representing 5 coarse-grained and 20 fine-grained entity types of the RISK dataset.

- **Source Domain Dataset**

We adopted the source domain dataset used in the PCBERT to ensure experimental consistency and reliability. This high-resource dataset is a combination of multiple datasets, including CLUENE [38], CNERTA [39], RenMinRiBao [40], and datasets from unknown sources.

4.2. Implementation Details

We adopted a two-stage training approach, starting with 5 epochs of pretraining on the source domain dataset, followed by 100 epochs of fine tuning using the pretrained model as the initial model in the target domain. For the upstream (BERT) part, we used the standard BERT-wwm-ext pretrained language model with 768-dimensional hidden representations for character embeddings and the learning rate set to 1×10^{-5}. For the downstream model, the learning rate was set to 1×10^{-4}. The hidden state size of the BiLSTM encoder was set to 200. We used the Adam optimizer [41]. We performed three different random samplings for all datasets to reduce randomness and reported the average performance as the final experimental result. All experiments were conducted on a computer with an Intel Core i9 13900K/F CPU®5.8 GHz and a GeForce RTX 3090 GPU with 24 GB of memory. The models were implemented using the PyTorch 1.12.0 framework.

4.3. Baseline

BERT [5] is the fundamental BERT-based NER method that adds a token classifier layer to the downstream of BERT.

BERT-LC is an effective baseline method for handling Chinese NER. It extends the regular BERT by adding a BiLSTM layer and employs a CRF layer as a decoder to predict token types.

Lattice LSTM [11] is a character-based Chinese NER method that introduces lexicon information by using a lattice-structured LSTM model.

FLAT [42] is a lattice-structured NER method based on a transformer. It constructs a flat-structured transformer to fully utilize lattice information and leverage parallel computing on GPUs.

LEBERT [43] addresses Chinese sequence-labeling tasks using lexicon-enhanced BERT. It incorporates lexical information into the encoding process of BERT's underlying layers using lexicon adapter layers.

LEBERT-LC is an extension of LEBERT that further adds a BiLSTM layer after the BERT output layer to facilitate a comparison with our proposed SLNER model.

PCBERT [10] is a prompt-based Chinese few-shot NER model. It consists of the P-BERT component and the C-BERT component, integrating lexical features and implicit label features.

4.4. Experimental Results

Our model was thoroughly evaluated on four sampled Chinese NER datasets. Specifically, we sampled K = 250, K = 500, K = 1000, and K = 1350 examples from the Ontonotes, MSRA, and Resume datasets, respectively, as training sets to cover different levels of data scarcity and comprehensively evaluate the training effectiveness and robustness of the model. The standard F1 score is used as the evaluation metric to measure the final performance of the model.

Table 3 shows the experimental results of our proposed model and the baseline models on three benchmark NER datasets. The results indicate that, except for the 1350-shot setting in the resume dataset, our method outperforms the state-of-the-art models by 0.2–6.57 percent, demonstrating the excellent performance of our method in few-shot settings.

Table 4 presents the experimental results of our proposed model on the RISK dataset. Compared to coarse-grained classification, the model shows a maximum decrease of 3.64–6.11 percent in F1 score under fine-grained classification. This indicates that the dataset is more challenging in fine-grained classification.

Table 3. F1 scores for the four Chinese sampling NER datasets. The best results are highlighted in bold.

Dataset	Method	K = 250	K = 500	K = 1000	K = 1350
Ontonotes	BERT	63.85	69.50	71.33	72.42
	BERT-LC	65.69	73.54	74.97	77.19
	Lattice LSTM	39.71	45.46	54.54	57.48
	FLAT	49.01	46.35	49.34	57.44
	LEBERT	69.48	69.01	73.78	74.84
	LEBERT-LC	70.26	69.89	73.83	76.01
	PCBERT	74.42	75.62	78.33	81.52
	SLNER (ours)	**77.66**	**80.46**	**81.53**	**82.23**
MSRA	BERT	68.44	72.28	81.21	82.28
	BERT-LC	79.01	83.13	87.84	89.32
	Lattice LSTM	54.69	63.61	74.27	76.31
	FLAT	59.62	70.20	80.79	64.95
	LEBERT	79.11	85.18	87.77	89.35
	LEBERT-LC	80.92	86.09	88.11	88.70
	PCBERT	81.08	85.25	87.88	89.72
	SLNER (ours)	**87.65**	**89.30**	**89.67**	**90.08**
Resume	BERT	53.80	62.64	69.36	70.65
	BERT-LC	92.26	94.66	95.16	**96.41**
	Lattice LSTM	85.63	89.60	92.01	93.13
	FLAT	84.62	90.77	92.97	87.79
	LEBERT	89.15	92.56	94.02	95.19
	LEBERT-LC	91.60	93.03	95.40	95.16
	PCBERT	93.42	94.01	94.96	95.97
	SLNER (ours)	**94.02**	**94.86**	**95.99**	96.36

Table 4. The results on the RISK dataset, showing the F1 scores for different sampling sizes in both coarse-grained and fine-grained categories.

Dataset	K-Shot	Coarse-Grained	Fine-Grained
RISK	K = 250	69.67	63.97
	K = 500	71.73	65.62
	K = 1000	72.70	68.06
	Full [1]	72.62	68.98

[1] Full represents K = 1218 (the maximum size of RISK).

In addition, to further validate the feasibility of our model in a low-resource setting, we also sampled data under K = 10, K = 20, and K = 50 settings and evaluated the effectiveness of our model with even fewer examples. The results are shown in Table 5. The experimental results demonstrate that our model can still achieve good results, even with as few as a few dozen samples. We analyze this phenomenon in Section 5.3.

Table 5. Results with fewer samples.

K-Shot	Dataset			
	Ontonotes	MSRA	Resume	RISK
K = 10	65.90	73.18	77.23	59.50 (44.61) [1]
K = 20	70.92	81.19	81.70	67.57 (45.02)
K = 50	74.91	86.34	90.55	68.87 (57.31)

[1] Parentheses represent the fine-grained results of the RISK dataset.

5. Analysis

5.1. Ablation Study

To validate the impact of enhanced span representation and label representation on SLNER, we conducted extensive ablation experiments on the Ontonotes, MSRA, Resume, and RISK datasets under different sampling settings (K = 250, 500, 1000, and 1350) as shown in Table 6. Specifically, we first removed the enhanced span representation module and observed a decrease in performance across all datasets with different sampling sizes. This demonstrates the effectiveness of the module, which we analyze in Section 5.2. Next, we removed the label representation module. It is worth mentioning that, due to the removal of the label representation module, we had to introduce a top-layer classifier, which resulted in the inability to use the source-domain-initialized model. As a result, the performance of our model significantly dropped, confirming our hypothesis that the label semantics carry rich prior knowledge. This also indicates the importance of prior knowledge from the pre-fine-tuning stage in low-resource environments. We provide a detailed analysis of this in Section 5.3.

Table 6. The results of the ablation experiments. ESR, enhanced span representation; LR, label representation; SD, source domain dataset.

Dataset	Method	K = 250	K = 500	K = 1000	K = 1350
Ontonotes	SLNER	77.66	80.46	81.53	82.23
	-ESR w/SD	75.63	80.02	81.04	81.91
	-LR w/o SD	72.90	75.62	77.98	78.09
MSRA	SLNER	87.65	89.30	89.67	90.08
	-ESR w/SD	87.99	89.60	89.42	89.80
	-LR w/o SD	80.72	84.06	84.87	84.54
Resume	SLNER	94.02	94.86	95.99	96.36
	-ESR w/SD	93.52	94.46	95.38	95.95
	-LR w/o SD	87.64	94.17	95.48	95.35
RISK	SLNER	63.97	65.62	68.06	N/A [1]
	-ESR w/SD	63.27	65.26	67.29	N/A
	-LR w/o SD	55.67	58.94	61.23	N/A

[1] Since the maximum size of the RISK dataset is less than 1350, we did not conduct experiments with K = 1350 for this dataset.

To further analyze the impact of label representation, we experimented with different definitions: full label name (the one used in our model), misleading label name, and indistinguishable label name. Misleading label name refers to randomly shuffling the full label name, for example, changing the label name corresponding to "PER" from "人名 (person name)" to "组织 (organization)". Indistinguishable label name refers to unifying all different label names into a single label name, such as using "人名 (person name)" for all labels. We conducted a comparative experiment on the Ontonotes dataset, as shown in Figure 10. The results indicate that a misleading label name slightly negatively affects the model, but an indistinguishable label name has a significant negative impact. This suggests that the selection of label names must be category-related and that a label name carries entity-specific prior knowledge.

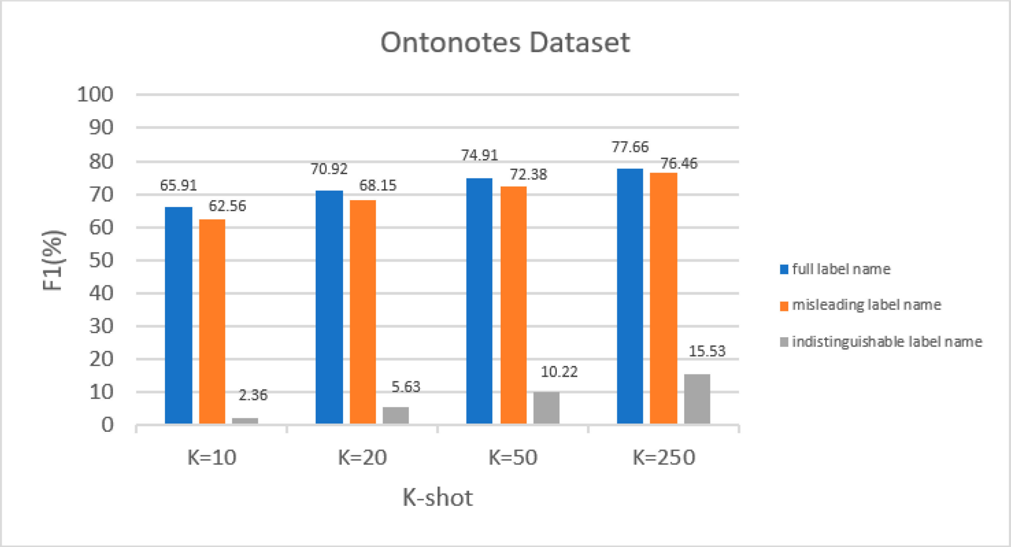


Figure 10. Different definitions of label names.

5.2. *The Impact of Enhanced Span Representation*

From Table 6, it can be observed that removing the inner span module results in a decrease in F1 scores. Particularly, when there are fewer samples (K = 250), the impact of removing the inner span module is more significant. This is because in the two-stage training process, during the source domain training, the lack of accurate span representations leads to a decrease in NER accuracy. However, when transitioning to the target domain, this inaccuracy is amplified, and the effect is more pronounced with fewer samples. Furthermore, during the ablation experiments, it was found that removing the inner span module leads to slower convergence of the model, indicating that without intra-span information, entity span localization becomes more challenging.

5.3. *The Impact of Label Representation*

The experimental results in Table 6 indicate that the impact of label name becomes more pronounced as the number of samples decreases (compared to K = 1350, the model's performance drops more significantly when label name is removed at K = 250). This is because as the number of training samples decreases, the number of words representing the same type of entity also decreases, resulting in less specific entity descriptions. On the other hand, label name itself is a broad concept that can represent entity categories, and its effectiveness increases when the number of samples is reduced. This leads to improved generalization ability.

6. Conclusions

In this study, we propose the SLNER model with enhanced span and label semantics to address the issues of inadequate prior knowledge and feature representation limitations in Chinese few-shot named entity recognition (NER). To tackle the feature representation limitations, we employ the biaffine attention mechanism and self-attention to obtain enhanced span representations, fully leveraging the internal composition of entity mentions. For the problem of inadequate prior knowledge, we introduce label semantics, which provide a highly abstract representation of specific entity categories with similar semantics and provide additional prior knowledge in few-shot scenarios. Our model learns to match span representations and label semantics to achieve entity recognition. Additionally, we constructed the RISK dataset in the food safety risk domain, which consists of 5 fine-grained and 20 coarse-grained entity types, providing a data foundation for NER development in low-resource domains. We extensively validated our proposed model on three benchmark datasets and our self-built dataset, and the results demonstrate the effectiveness of our model in addressing the issues of Chinese few-shot NER.

Author Contributions: Conceptualization, Z.R. and X.Q.; methodology, Z.R.; software, Z.R.; validation, Z.R. and X.Q.; formal analysis, Z.R.; investigation, Z.R. and W.R.; resources, W.R.; data curation, Z.R.; writing—original draft preparation, Z.R.; writing—review and editing, Z.R. and X.Q.; supervision, X.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Major science and technology special projects of Xinjiang Uygur Autonomous Region (2020A03001) and its subprogram, Key technology development and application demonstration of integrated food data supervision platform in Xinjiang region (2020A03001-2).

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lu, Y.; Liu, Q.; Dai, D.; Xiao, X.; Lin, H.; Han, X.; Sun, L.; Wu, H. Unified Structure Generation for Universal Information Extraction. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022; Volume 1, pp. 5755–5772.
2. Mollá, D.; Van Zaanen, M.; Smith, D. Named entity recognition for question answering. In Proceedings of the Australasian Language Technology Workshop 2006, Sydney, Australia, 11 November 2006; pp. 51–58.
3. Stahlberg, F. Neural Machine Translation: A Review. *J. Artif. Intell. Res.* **2020**, *69*, 343–418. [CrossRef]
4. 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, San Diego, CA, USA, 12–17 June 2016; Association for Computational Linguistics: Stroudsburg, PA, USA, 2016; pp. 260–270.
5. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019.
6. Vinyals, O.; Blundell, C.; Lillicrap, T.; Wierstra, D. Matching networks for one shot learning. In Proceedings of the Advances in Neural Information Processing Systems 29 (NIPS 2016), Barcelona, Spain, 5–10 December 2016.
7. Snell, J.; Swersky, K.; Zemel, R. Prototypical networks for few-shot learning. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
8. Hou, Y.; Che, W.; Lai, Y.; Zhou, Z.; Liu, Y.; Liu, H.; Liu, T. Few-shot slot tagging with collapsed dependency transfer and label-enhanced task-adaptive projection network. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020; pp. 1381–1393.
9. Cui, L.; Wu, Y.; Liu, J.; Yang, S.; Zhang, Y. Template-based named entity recognition using BART. In Proceedings of the Findings of the Association for Computational Linguistics, ACL/IJCNLP 2021, Online, 1–6 August 2021; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 1835–1845.
10. Lai, P.; Ye, F.; Zhang, L.; Chen, Z.; Fu, Y.; Wu, Y.; Wang, Y. PCBERT: Parent and Child BERT for Chinese Few-shot NER. In Proceedings of the 29th International Conference on Computational Linguistics, Gyeongju, Republic of Korea, 12–17 October 2022; pp. 2199–2209.
11. Zhang, Y.; Yang, J. Chinese NER using lattice LSTM. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; Volume 1, pp. 1554–1564.
12. Dong, X.; Xin, X.; Guo, P. Chinese NER by Span-Level Self-Attention. In Proceedings of the 2019 15th International Conference on Computational Intelligence and Security (CIS), Macao, China, 13–16 December 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 68–72.
13. Li, X.; Feng, J.; Meng, Y.; Han, Q.; Wu, F.; Li, J. A Unified MRC Framework for Named Entity Recognition. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020.
14. Chiu, J.P.; Nichols, E. Named entity recognition with bidirectional LSTM-CNNs. *Trans. Assoc. Comput. Linguist.* **2016**, *4*, 357–370. [CrossRef]
15. Cui, L.; Zhang, Y. Hierarchically refined label attention network for sequence labeling. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 4115–4128.
16. Tong, M.; Wang, S.; Xu, B.; Cao, Y.; Liu, M.; Hou, L.; Li, J. 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, Virtual, 1–6 August 2021; Volume 1, pp. 6236–6247.

17. Lee, D.H.; Kadakia, A.; Tan, K.; Agarwal, M.; Feng, X.; Shibuya, T.; Mitani, R.; Sekiya, T.; Pujara, J.; Ren, X. Good Examples Make A Faster Learner: Simple Demonstration-based Learning for Low-resource NER. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022; Volume 1, pp. 2687–2700.
18. Chen, X.; Li, L.; Deng, S.; Tan, C.; Xu, C.; Huang, F.; Si, L.; Chen, H.; Zhang, N. LightNER: A Lightweight Tuning Paradigm for Low-resource NER via Pluggable Prompting. In Proceedings of the 29th International Conference on Computational Linguistics, Gyeongju, Republic of Korea, 12–17 October 2022; pp. 2374–2387.
19. Wang, L.; Li, R.; Yan, Y.; Yan, Y.; Wang, S.; Wu, W.; Xu, W. Instructionner: A multi-task instruction-based generative framework for few-shot ner. *arXiv* **2022**, arXiv:2203.03903.
20. Chen, J.; Liu, Q.; Lin, H.; Han, X.; Sun, L. Few-shot Named Entity Recognition with Self-describing Networks. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022; Volume 1, pp. 5711–5722.
21. Ma, T.; Jiang, H.; Wu, Q.; Zhao, T.; Lin, C.Y. Decomposed Meta-Learning for Few-Shot Named Entity Recognition. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, 22–27 May 2022; pp. 1584–1596.
22. Das, S.S.S.; Katiyar, A.; Passonneau, R.J.; Zhang, R. CONTaiNER: Few-Shot Named Entity Recognition via Contrastive Learning. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022; Volume 1.
23. Wang, S.; Li, X.; Meng, Y.; Zhang, T.; Ouyang, R.; Li, J.; Wang, G. *k*NN-NER: Named Entity Recognition with Nearest Neighbor Search. *arXiv* **2022**, arXiv:2203.17103.
24. Huang, Z.; Wei, X.; Kai, Y. Bidirectional lstmcrf models for sequence tagging. *Computer Science arXiv* **2015**, arXiv:1508.01991.
25. Peters, M.E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep contextualized word representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, LA, USA, 1–6 June 2018; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; Volume 1, pp. 2227–2237.
26. Yu, J.; Bohnet, B.; Poesio, M. Named entity recognition as dependency parsing. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020; pp. 6470–6476.
27. Shen, Y.; Ma, X.; Tan, Z.; Zhang, S.; Wang, W.; Lu, W. Locate and label: A two-stage identifier for nested named entity recognition. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics, Virtual, 1–6 August 2021.
28. Yu, D.; He, L.; Zhang, Y.; Du, X.; Pasupat, P.; Li, Q. Few-shot intent classification and slot filling with retrieved examples. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 6–11 June 2021; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 734–749.
29. Wang, P.; Xu, R.; Liu, T.; Zhou, Q.; Cao, Y.; Chang, B.; Sui, Z. An enhanced span-based decomposition method for few-shot sequence labeling. *arXiv* **2021**, arXiv:2109.13023.
30. Wang, J.; Wang, C.; Tan, C.; Qiu, M.; Huang, S.; Huang, J.; Gao, M. SpanProto: A Two-stage Span-based Prototypical Network for Few-shot Named Entity Recognition. *arXiv* **2022**, arXiv:2210.09049.
31. Luo, Q.; Liu, L.; Lin, Y.; Zhang, W. Don't miss the labels: Label-semantic augmented meta-learner for few-shot text classification. In Proceedings of the Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Online, 1–6 August 2021; pp. 2773–2782.
32. Ma, R.; Zhou, X.; Gui, T.; Tan, Y.; Li, L.; Zhang, Q.; Huang, X. Template-free Prompt Tuning for Few-shot NER. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Stroudsburg, PA, USA, 10–15 July 2022; pp. 5721–5732.
33. Zhong, Z.; Chen, D. A Frustratingly Easy Approach for Entity and Relation Extraction. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 6–11 June 2021; pp. 50–61.
34. Ye, D.; Lin, Y.; Li, P.; Sun, M. Packed Levitated Marker for Entity and Relation Extraction. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022; Volume 1, pp. 4904–4917.
35. Bekoulis, G.; Deleu, J.; Demeester, T.; Develder, C. Joint entity recognition and relation extraction as a multi-head selection problem. *Expert Syst. Appl.* **2018**, *114*, 34–45. [CrossRef]
36. Weischedel, R.; Pradhan, S.; Ramshaw, L.; Palmer, M.; Xue, N.; Marcus, M.; Taylor, A.; Greenberg, C.; Hovy, E.; Houston, A.; et al. *Ontonotes Release 4.0. LDC2011T03*; Linguistic Data Consortium: Philadelphia, PA, USA, 2011.
37. Gina-Anne, L. The third international Chinese language processing bakeoff: Word segmentation and named entity recognition. In Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing, Sydney, Australia, 22–23 July 2006; pp. 108–117.
38. Xu, L.; Dong, Q.; Yu, C.; Tian, Y.; Liu, W.; Li, L.; Zhang, X. Cluener2020: Fine-grained name entity recognition for Chinese. *arXiv* **2020**, arXiv:2001.04351.
39. Sui, D.; Tian, Z.; Chen, Y.; Liu, K.; Zhao, J. A large-scale Chinese multimodal NER dataset with speech clues. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Virtual, 1–6 August 2021; Volume 1.

40. Xia, Y.; Yu, H.; Nishino, F. The Chinese named entity categorization based on the people's daily corpus. *Int. J. Comput. Linguist. Chin. Lang. Process.* **2005**, *10*, 533–542.
41. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
42. Li, X.; Yan, H.; Qiu, X.; Huang, X. Flat: Chinese ner using flat-lattice transformer. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 6836–6842.
43. Liu, W.; Fu, X.; Zhang, Y.; Xiao, W. Lexicon enhanced Chinese sequence labelling using bert adapter. *arXiv* **2021**, arXiv:2105.07148.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.