Received: 25 September 2024 | Revised: 4 November 2024 | Accepted: 28 November 2024 | Published online: 17 December 2024

Journal of Data Science and Intelligent Systems
2024, Vol. 00(00) 1–14
DOI: 10.47852/bonviewJDSIS42024432
BON VIEW PUBLISHING

RESEARCH ARTICLE

Low-Resource Chinese Named Entity Recognition via CNN-based Multitask Learning

Tao Wu[1], Xinwen Cao[1], Feng Jiang[1], Canyixing Cui[1], Xuehao Li[1] and Xingping Xian[1],*

[1]*School of Cybersecurity and Information Law, Chongqing University of Posts and Telecommunications, China*

Abstract: Named entity recognition (NER) is a fundamental subtask for information extraction that aims to locate and classify named entities in unstructured text into predefined categories. Recently, large-scale language models (LLMs) have achieved SOTA performance on a variety of natural language processing tasks. However, because NER is a sequence labeling task in nature while LLMs is a text-generation model, the performance of LLMs on NER is still significantly below supervised baselines, and NER remains a difficult task. Meanwhile, the word boundary and semantic information of Chinese words are usually quite vague, as words contained in Chinese texts are not separated by spaces. Thus, the NER task still requires supervised learning paradigm and heavily relies on large amounts of labeled data, such as entity type and boundary information. However, the cost of labeling data can be prohibitively large, and the purely supervised approaches usually suffer from poor generalization capability. In this article, we propose a multitask learning-based bidirectional iterated dilated convolution model, BCNN-CWS, for low-resource NER via leveraging word boundary information of Chinese word segmentation (CWS) task. Specifically, to efficiently recognize named entities, an iterated dilated convolutional model with a limited number of layers is implemented. In addition, a bidirectional causal convolution mechanism is presented for contextual information extraction. Results of extensive experiments on public Chinese datasets demonstrate that BCNN-CWS achieves superior performance over state-of-the-art models, and it yields up to about 50% speed improvement over existing methods. It is worth noting that BCNN-CWS can be further improved by combining with a pretrained model.

Keywords: named entity recognition, low resource, iterated dilated convolution, causal convolution, multitask learning

1. Introduction

Named entity recognition (NER), also known as entity recognition or entity extraction, aims to identify the boundaries of entity names in unstructured texts and classify their types, such as names of people, places, and organizations [1, 2]. Typically, NER is considered an essential prerequisite for many downstream tasks in natural language processing (NLP), such as relation extraction [3], entity annotation [4], and entity linking [5, 6].

Conventionally, dictionary matching and the handcrafted rule-based methods have been proposed for NER tasks. However, building dictionaries and rules is time- and resource-consuming, and it is often difficult to obtain good coverage for many named entity types [7]. Subsequently, statistical learning-based methods have been proposed to obtain ground-breaking results, such as hidden Markov models (HMMs) [8], maximum entropy (ME) models [9], and conditional random field (CRF) methods [10]. However, the methods always use one-hot vectors for word representation, which loses the semantic information of sentences and increases computational difficulty. To learn intricate features from the context and generate useful representations, deep learning-based entity recognition models have been proposed, including CNN- and BiLSTM-based methods [11–13]. With the proposal of BERT pre-trained language model [14], the deep learning-based NER methods are enhanced by BERT model, so the Bert-BilSTM-CRF [15] and other methods are proposed. Recently, large-scale language models (LLMs) have achieved SOTA performance on a variety of NLP tasks. However, because NER is a sequence labeling task in nature while LLMs is a text-generation model, the performance of LLMs on NER is still significantly below supervised baselines [16].

Problem setting. Although effective in entity recognition tasks, the success of deep-learning-based supervised methods depends heavily on large-scale training instances with labels. However, the cost of labeling data can be prohibitively large, and in annotated sentences, only common entity types have relatively sufficient labeled samples [17]. Meanwhile, the supervised methods usually suffer from poor generalization capability. In addition, Chinese entities lack strong indications compared with English text, such as capitalization, and they are highly context-dependent in which the same words can be used as names of different kind entities. Moreover, there is no explicit delimiter to separate words, such as whitespace [18], and identifying the boundary of entities is more difficult in Chinese than in English texts. An example of Chinese named entity recognition (CNER) is given in Table 1. According to the different place, the word “阿里” can be the name of

***Corresponding author:** Xingping Xian, School of Cybersecurity and Information Law, Chongqing University of Posts and Telecommunications, China. Email: xianxp@cqupt.edu.cn

Table 1
Examples of context dependence and entity boundary

Categories	Sentence
Context-dependence 1	在杭州，阿里 [Organization] 的发展十分重要。
Context-dependence 2	在西藏，阿里 [Location] 的发展十分重要。
Entity boundary 1	落雨天/留客天/留我不留。
Entity boundary 2	落雨天留客/天留我不留。

organization entity (“在杭州，阿里的发展十分重要”), and location entity (“在西藏，阿里的发展十分重要”). Due to differences in segmentation boundary, “留客天” will or will not be recognized as named entity. As a result, the low-resource CNER has become a major challenge.

Prior work and challenges. In this paper, we propose to learn the representation of Chinese named entities adequately given limited (low-resource) training samples. Currently, although various methods have been proposed, most of them are designed for NER of English text, and low-resource Chinese NER still faces the following challenges: (1) Because of the discrepancy between languages, the existing cross-language NER methods [19, 20] tend to introduce noise from the source language into the training samples of the target languages; (2) Chinese NER is essentially a character-level sequence labeling problem, and the conventional word-level NER methods proposed for English texts are not suitable. Meanwhile, the auxiliary tasks in many NER methods learn sentence- or word-level features and cannot offer rich information for entity boundary prediction and entity type identification at the character-level [21]; (3) Most existing NER models are based on a recurrent neural network (RNN) architecture or its variants like LSTM to obtain sequence information and cannot be computed in parallel, thereby always requiring considerable computation time.

Main contributions. To solve the CNER task under low-resource conditions, we propose a bidirectional iterated dilated convolution-based model, BCNN-CWS, via multitask learning with Chinese word segmentation (CWS). Given an input sentence, CWS is used to predict the boundaries of words in raw texts, which is highly related to extracting entity names from texts in the CNER task. Meanwhile, CNER and CWS all aim to learn the feature representation of input sentence at the character level, so that they can provide each other with sufficient feature information. Thus, the ability of CNER model to identify entity boundaries can be enhanced via joint training with a CWS task. Here, we assume that the upper hidden layers are responsible for high-level processing and the lower layers perform basic feature representations. Since the goals of CNER and CWS are different, we propose to share only part of the hidden layers in the model. Moreover, the use of CWS as an auxiliary task for CNER can avoid the noise introduced by the cross-lingual transfer approaches. Then, because CNN-based models have the capability of parallel computation, a novel bidirectional iterated dilated convolution model, called BCNN, is proposed to replace the traditional RNN-based model by incorporating evidence from the entire input sentence, where the effective input width can grow exponentially with the depth. To learn contextual information and ensure the model cannot violate the ordering in which we model the sentence, the BCNN defines a bidirectional causal convolution for the feature extraction of input sentences. The contributions of our work can be summarized as follows:

1) A novel low-resource Chinese NER model based on multitask learning that combines NER with the word segmentation task, BCNN-CWS, is proposed. The model optimizes the hidden representations of Chinese characters for entity boundary prediction by sharing the embedding layer between CNER and CWS tasks.
2) To extract contextual information from the entire sentence while reducing time consumption of CNER, a novel convolutional neural network model BCNN is proposed by stacking a limited number of iterated dilated convolution layers.
3) To characterize the dependency between Chinese characters and model the Markov property of sentences, a bidirectional causal convolution mechanism is adopted, in which the information from both directions is utilized to learn past and future input features for a given time.
4) Extensive experiments on three representative Chinese NER datasets verify the feasibility of the proposed methods. The effectiveness of the proposed mechanisms is illustrated by ablation experiments and model analysis.

The remainder of this paper is organized as follows. Section 2 discusses related studies. Section 3 presents the problem definitions and preliminaries. In Section 4, the proposed method is introduced. Section 5 presents the experimental results. Finally, conclusions and discussion are presented in Section 6.

2. Related Work

NER has been extensively investigated in the NLP field. A literature overview of recent advances in entity recognition is provided below.

2.1. Named entity recognition

Studies on NER can be broadly classified into three categories: rule-, statistical learning-, and deep learning-based approaches. Specifically, rule-based NER methods rely on handcrafted rules, such as domain-specific gazetteers and syntactic-lexical patterns. Well-known rule-based NER systems include Brill [22], ProMiner [23], and NetOwl [24]. However, rule-based methods work well only when the lexicon is exhaustive and always requires considerable labor [25]. Statistical learning-based methods depend on feature engineering, which represents annotated training samples as vectors. A straightforward option for vector representation is one-hot encoding. Based on feature vectors, many NER methods have been proposed using statistical learning algorithms, including hidden Markov models, ME models, support vector machines, and conditional random fields [26–28]. However, these methods require feature engineering to represent texts, and the generated high-dimensional sparse vectors are difficult to compute and lack contextual information. Deep-learning-based NER methods have become dominant and have achieved excellent results recently. A BiLSTM-CRF model for NER is proposed [29], which can efficiently make use of past and future information. The score of the label sequence $\mathbf{y}$ for the input sentence $\mathbf{x}$ is defined as the sum of the transition matrix $\mathbf{A}$ and output matrix $\mathbf{F}$ of BiLSTM:

$$score(\mathbf{x},\mathbf{y}) = \sum_{i=0}^{n} \mathbf{A}_{y_i,y_{i+1}} + \sum_{i=0}^{n} \mathbf{F}_{i,y_i} \tag{1}$$

Inspired by the outstanding performance of gated recurrent units (GRU) in sequence modeling, a neural network architecture based on a bidirectional GRU combined with CRF for Arabic NER is proposed [30, 31]. Recently, a body of works [32, 33] using BiLSTM as basic architecture have been proposed for NER. Meanwhile, an increasing number of entity extraction methods based on transformer and pretrained models, as well as multimodal information extraction methods, have been proposed [34–37]. Compared with the feature engineering-based methods, deep learning-based methods are useful for discovering hidden feature representations automatically.

Inspired by iterated dilated convolution [38], this study proposes a bidirectional convolutional neural network (BCNN). In contrast to traditional methods, our model can be trained using an end-to-end paradigm. Moreover, compared with RNN-based methods, our model is capable of learning contextual information and effectively reducing the time cost.

2.2. Data augmentation

Data augmentation techniques, as a general approach for generating additional training samples, are commonly used in the low-resource domain to alleviate the data-scarcity problem. Data augmentation methods in NLP can be generally divided into three categories: paraphrasing-based, noising-based, and sampling-based methods [39]. Specifically, paraphrasing-based methods rephrase the original text with proper and restrained changes, while maintaining the same semantics. The mainstream techniques for paraphrasing-based data augmentation include thesaurus, semantic embeddings, language models, heuristics rules, and machine translation [39]. Noising-based methods add noise to the original text to generate effective augmented samples, which involves more changes than paraphrasing-based methods. The specific operations of noising-based methods include swapping, deletion, insertion, and substitution [40, 41]. Sampling-based methods master the distribution of the original text to sample new data as augmented samples, and the methods for sampling-based data augmentation include the Seq2Seq model [42] and self-training [43].

Data augmentation methods add slightly modified copies of existing data or create synthetic data to increase the diversity of training samples. In contrast to these methods, our work adopts a multitask learning framework to solve the low-resource NER tasks, which learn from multiple related tasks simultaneously using a shared model to produce a better hidden representation.

2.3. Multitask learning

Multitask learning aims to train multiple different yet related tasks and optimize more than one loss function simultaneously to improve the capability of models. For NER task, [44] proposed a multitask learning model to improve the performance of NER using sentence classification as auxiliary task, in which the hidden representation $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_T]$, $\mathbf{h}_t = concat\left(\tilde{\mathbf{h}}_t, \ \leftarrow \ \vec{\mathbf{h}}_t\right)$, is shared between the two tasks. In Ref. [45] proposed a stack-LSTM-based model that jointly performed NER and entity linking to improve the performance of both tasks. In Ref. [46] developed a multitask learning framework that utilizes comparable corpora to jointly train the bilingual word embedding and the downstream NER task. In Ref. [47] proposed a multitask learning architecture that predicts the labels for both full sentences and individual tokens. The model aims to learn better language representations and composition functions by combining the objectives at different granularities.

In contrast to the above approaches, word segmentation is used as the auxiliary task in this study. Similarly, [48] proposed a CNN-LSTM-CRF architecture to jointly train Chinese NER and word segmentation models, in which the CNN and Bi-LSTM layers are used to learn hidden representations from local and long-distance contexts, respectively. Unlike CNN-LSTM-CRF model, we propose a bidirectional dilated causal convolutional neural network (BCNN) to learn hidden representations from both local and long-distance contexts. The proposed method shares only the character embedding layer between the Chinese NER and word segmentation tasks to improve the ability of the NER model to predict entity boundaries.

3. Problem Definition and Preliminaries

3.1. Problem definition

Definition 1 (Named Entity Recognition). Given a sentence $\mathbf{E} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_\mathrm{N}\}$, where $\mathbf{w}_i$ is a vector representation of the *i*th word in a sentence, NER aims to identify the boundaries and types of entities with specific meanings from the sentence and outputs the token label $\dagger_i$ of the word $\mathbf{w}_i$. Thus, the sentence $\mathbf{E}$ has a sequence of labels $\mathcal{Y} = \{\dagger_1, \dagger_2, \ldots, \dagger_N\}$. Given M sentences and their label sequences, denoted as $\{(\mathbf{E}_\mathrm{m}, \mathcal{Y}_m)\}_{m=1}^{M}$, the training process of NER model minimizes the following loss function:

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{|M|} \sum_{i=1}^{M} \ell(\mathcal{Y}_m, f(\mathbf{E}_\mathrm{m}; \theta)) \tag{2}$$

where $\ell(\cdot)$ is the loss function, and $f(\mathbf{E}_\mathrm{m}; \theta)$ is the NER model.

Definition 2 (Chinese Named Entity Recognition). For Chinese NER, the task aims to separate Chinese characters to extract entities, that is, a span of tokens $\{\mathbf{c}_i, \ldots, \mathbf{c}_j\}$, $(0 \leq i \leq j \leq N)$, and obtain their type labels from the Chinese sentence $\mathbf{S} = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_\mathrm{N}\}$, where $\mathbf{c}_i$ denotes the vector representation of the *i*th Chinese character in a sentence.

Definition 3 (Chinese Word Segmentation). CWS is typically modeled as a sequence-labeling problem at the character level. Given a sequence of tokens $\mathbf{S} = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_\mathrm{N}\}$, the purpose of Chinese-word segmentation is to divide Chinese characters into words. Suppose that $\mathcal{L}$ is the possible syncopated path on sentence $\mathbf{S}$, the purpose of CWS is to find the cut path $\mathcal{L}^*$ that maximizes the conditional probability P; that is,

$$\mathcal{L}^* = \arg\max_{\mathcal{L}} \ P(\mathcal{L}|\mathcal{S}) \tag{3}$$

where $P(\mathcal{L}|\mathbf{S})$ is the likelihood that cut path $\mathcal{L}$ is the true output of sequence $\mathbf{S}$.

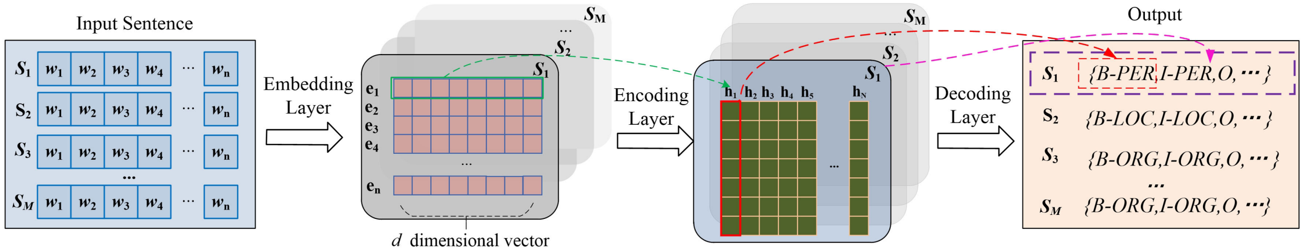
Definition 4. (Low-Resource Chinese NER). Given a corpus set $\mathcal{T}$ that includes a limited number of sentences $\{\mathbf{S}_1, \ldots, \mathbf{S}_\mathrm{n}\}$, the purpose of low-resource Chinese NER is to generate entity labels for character $\mathbf{c}_\mathrm{i}$ in the sentence $\mathbf{S}$ based on $\mathcal{T}$, thereby identifying entities $\{\mathbf{c}_\mathrm{i}, \ldots, \mathbf{c}_\mathrm{j}\}$ with specific meanings.

3.2. Entity recognition process

The NER model often comprises of the embedding, encoding, and decoding layers, as shown in Figure 1.

Embedding layer. The input to the embedding layer is a sentence $\mathbf{S} = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_\mathrm{N}\}$, where $\mathbf{c}_\mathrm{i} \in \mathbb{R}^v$ is the one-hot vector representation of *i*th character, the v is vocabulary size. The output of this layer is a real-valued dense vector, $\mathbf{e}_i$, where each dimension represents a latent feature. Thus, the embedding representation of the

Figure 1
Entity recognition process: input M sentences; the embedding layer represents each word as a 1* d vector; hidden feature representations are generated in encoding layer; finally, the corresponding tags are the output in decoding layer


Input Sentence
S_1 w_1 w_2 w_3 w_4 ··· w_n
S_2 w_1 w_2 w_3 w_4 ··· w_n
S_3 w_1 w_2 w_3 w_4 ··· w_n
S_M w_1 w_2 w_3 w_4 ··· w_n
Embedding Layer
S_M
S_2
S_1
e_1 e_2 e_3 e_4 e_n
d dimensional vector
Encoding Layer
h_1 h_2 h_3 h_4 h_5 h_N
Decoding Layer
Output
S_1 {B-PER,I-PER,O, ··· }
S_2 {B-LOC,I-LOC,O, ··· }
S_3 {B-ORG,I-ORG,O, ··· }
S_M {B-ORG,I-ORG,O, ···}

sentence $\mathbf{S}$ is $\mathbf{E} = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_N\}$. The embedding representation $\mathbf{e}_i$ captures the semantic and syntactic properties of sentences automatically, which are not explicitly present in the one-hot vector representation $\mathbf{c}_i$. Subsequently, the resulting embedding representation $\mathcal{E}$ is transmitted to the encoding layer as an input.

Encoding layer. In this layer, the input vector representation $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_N\}$ is processed by encoding models, such as CNN and RNN. The model captures the local and global contextual information in the sentence and generates a hidden representation $\mathbf{H} = \{\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_n\}$. For example, in the sentence "乔布斯是苹果公司的创始人," the hidden representation of characters in "苹果" contains the contextual information "公司" via the encoding model. Thus, the characters would be inferred as an organization entity rather than a fruit.

Decoding layer. In the decoding layer, the decoding model uses context-dependent hidden representation $\mathbf{H}$ as input and produces tags for tokens (words in English or characters in Chinese). Thus, the model outputs a tag sequence for the input sentence, and entities with boundaries and types can be extracted from the sentence accordingly.

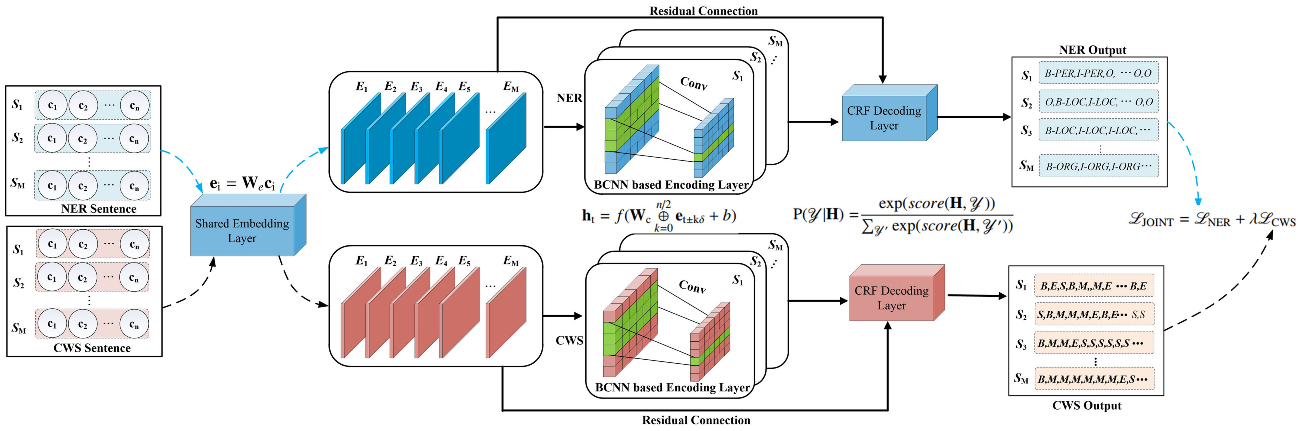
3.3. Tag schemes

The NER task, as a type of sequence labeling task, requires labeling each token in sentences using a tagging scheme to determine the type and position of entities. Representative tag schemes include BIO, IOB, and BIOES. Specifically, the BIO scheme uses B (begin) to indicate the beginning character of an entity, I (inside) to indicate the other characters of an entity, and O (outside) to indicate non-entity characters. The IOB scheme represents all entity characters using I; however, when two consecutive entities of the same type exist, the beginning character of the latter is represented by B. The BIOES scheme adopts E (end) to indicate the last character of an entity and represents the single-character entity using S (single); the other tokens are labeled as in the BIO tag scheme. In this study, the BIO tag scheme is adopted to label tokens.

4. Methodology

In this section, we describe the proposed multitask learning-based BCNN-CWS model in detail. Deep learning methods can automatically learn contextual information and thus perform well in NER tasks. However, a deep-learning-based NER system may not achieve a satisfactory performance if its hidden representations cannot be learned adequately, which frequently occurs in low-resource scenarios. To this end, the proposed BCNN-CWS model adopts a multitask learning framework that allows the embedding representation to acquire more feature information from the auxiliary task. Here, we first introduce the proposed learning model, which unites CNER and CWS by sharing a multilayer perception (MLP)-based embedding layer. Then, we describe our bidirectional iterated dilated convolution network. Figure 2 illustrates the overall architecture of the BCNN-CWS.

Figure 2
Model architecture of BCNN-CWS. The input sentences for the NER and CWS tasks are represented in the shared embedding layer. The embedded representations of the two tasks are fed to different encoding layers. The results are output in the decoding layer and the loss of the two tasks are combined as the loss of the joint model


S_1 c_1 c_2 ··· c_n
S_2 c_1 c_2 ··· c_n
S_M c_1 c_2 ··· c_n
NER Sentence
S_1 c_1 c_2 ··· c_n
S_2 c_1 c_2 ··· c_n
S_M c_1 c_2 ··· c_n
CWS Sentence
$\mathbf{e}_i = \mathbf{W}_e \mathbf{c}_i$
Shared Embedding Layer
E_1 E_2 E_3 E_4 E_5 ··· E_M
NER
CWS
Residual Connection
S_M S_2 S_1
Conv
BCNN based Encoding Layer
$\mathbf{h}_t = f(\mathbf{W}_c \bigoplus_{k=0}^{n/2} \mathbf{e}_{t \pm k\delta} + b)$
$P(\mathcal{Y}|\mathbf{H}) = \frac{\exp(score(\mathbf{H}, \mathcal{Y}))}{\sum_{\mathcal{Y}'} \exp(score(\mathbf{H}, \mathcal{Y}'))}$
CRF Decoding Layer
NER Output
S_1 B-PER,I-PER,O, ··· O,O
S_2 O,B-LOC,I-LOC, ··· O,O
S_3 B-LOC,I-LOC,I-LOC, ···
S_M B-ORG,I-ORG,I-ORG···
$\mathcal{L}_{JOINT} = \mathcal{L}_{NER} + \lambda \mathcal{L}_{CWS}$
S_1 B,E,S,B,M,,M,E ··· B,E
S_2 S,B,M,M,M,E,B,E··· S,S
S_3 B,M,M,E,S,S,S,S,S,S ···
S_M B,M,M,M,M,M,M,E,S···
CWS Output

4.1. Multitask learning framework

Essentially, CWS is a part of the CNER process, in which the characters in the same group are represented with a higher transitional probability between them. A better hidden representation improves the performance of word segmentation models. For example, in the sentence “乔布斯是苹果公司的创始人,” the characters will be grouped into “乔布斯/是/苹果/公司/的/创始人,” and the hidden representations of the tokens in a same group, such as “苹” and “果,” will be closer together in the probability space compared with the other tokens. Thus, the hidden representations learned from the CWS can reduce the difficulty of CNER task. In low-resource scenarios, the combination of the CWS can improve the performance of CNER.

However, they have differences between CNER and CWS that prevent them from completely sharing the underlying model architecture. Specifically, compared with CWS, CNER tasks should determine the categorical tags of named entities (e.g., names of people, places, and organizations) based on contextual information, in addition to their boundaries. For example, in the sentence “谷歌和苹果是业内备受赞誉的两家企业,” the result of CWS is “谷歌/和/苹果/是/业内/备受/赞誉/的/两家/企业”. CNER aims to discover the named entity “谷歌” and “苹果” and determine their type “organization” based on contextual information. Thus, sharing all hidden layers [21, 44, 48] may negatively affect the performance of CNER task. Consequently, we propose a multitask learning framework that shares only the embedding layer of both tasks and combines their losses to jointly optimize the model.

Shared embedding layer. The shared embedding layer uses sentences for CNER and CWS as input so that the layer can be optimized jointly for both tasks. Let $\mathbf{S} = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_\mathrm{N}\}$ be the input sequence, where $\mathbf{c}_i$ denotes the ith token in the sequence. Using a MLP-based embedding layer, the input sequence can be represented as $\mathbf{E} = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_\mathrm{N}\}$, where $\mathbf{e}_\mathrm{i} \in \mathbb{R}^d$ is the embedding representation vector of the ith token, d is the embedding dimension. That is, each token in the input sentence can be embedded as

$$\mathbf{e}_\mathrm{i} = \mathbf{W}_e \mathbf{c}_\mathrm{i} \tag{4}$$

where $\mathbf{W}_e \in \mathbb{R}^{d \times v}$ denotes the trainable weight matrix. The output is subsequently transmitted to the encoding layer.

NER. The NER model comprises an encoding and a decoding layer. In the encoding layer, the embedding representation vectors are fed into the proposed bidirectional convolution network (BCNN) model to generate the hidden feature representation $\mathbf{H} = \{\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_n\}$, where $\mathbf{h}_\mathrm{i}$ denotes the encoded representation of the token $\mathbf{c}_\mathrm{i}$. Then, following [29, 44], we feed $\mathbf{H} \in \mathbb{R}^{\mathbf{n} \times \mathbf{d}}$ from the encoding layer to a CRF-based decoding layer to obtain the probability of a label sequence $\mathcal{Y}$. Typically, the characters in a sentence are not independent and have strong dependency relationships among them. Thus, a desirable NER model should be able to capture the dependency among characters, thereby deciding the current label based on past and future labels. The CRF [49] layer defines the probability of the label sequence $\mathcal{Y}$ given $\mathbf{H}$ as

$$\mathrm{P}(\mathcal{Y}|\mathbf{H}) = \frac{\exp(score(\mathbf{H}, \mathcal{Y}))}{\sum_{\mathcal{Y}'} \exp(score(\mathbf{H}, \mathcal{Y}'))} \tag{5}$$

where the score $score(\mathbf{H}, \mathcal{Y})$ is defined as

$$score(\mathbf{H}, \mathcal{Y}) = \sum_{i=0}^{N} \mathbf{A}_{\dagger_i, \dagger_{i+1}} + \sum_{i=1}^{N} \mathbf{F}_{\mathbf{H}, \dagger_i} \tag{6}$$

where $\mathbf{A}$ is the transition score matrix, $\mathbf{A}_{\dagger_i, \dagger_{i+1}}$ represents the score of a transition from the label $\dagger_i$ to label $\dagger_{i+1}$. Further, $\mathbf{F}_\mathbf{H}$ is the emission score matrix, $\mathbf{F}_{\mathbf{H}, \dagger_\mathbf{i}}$ represents the score of label $\dagger_i$. To train the NER model, the negative log-likelihood of the correct label sequences is minimized over the training set:

$$\mathcal{L}_{\mathrm{NER}} = -\frac{1}{|\mathcal{T}|} \sum_{s \in \mathcal{T}} \log(\mathcal{Y}_s^{ner} | \mathbf{H}_s; \theta^{ner}) \tag{7}$$

where $\mathcal{T}$ is the set of sentences in the training data; $\mathbf{H}_s$ and $\mathcal{Y}_s^{ner}$ are the hidden representation and label sequence of the sentence s, respectively; and θ^{ner} is the parameter set of the NER model.

Word segmentation. Because word segmentation task is similar in nature to NER, we treat the word segmentation task the same as NER, and the loss function of word segmentation is defined as

$$\mathcal{L}_{\mathrm{CWS}} = -\frac{1}{|\mathcal{T}|} \sum_{s \in \mathcal{T}} \log(\mathcal{Y}_s^{cws} | \mathbf{H}_s; \theta^{cws}) \tag{8}$$

where $\dagger_s^{cws}$ is the label sequence of the sentence s for word segmentation, and θ^{cws} is the parameter set of the CWS model.

Joint loss function. The final objective loss function of the joint model is defined by combining Equations (8) and (9) as follows:

$$\mathcal{L}_{\mathrm{JOINT}} = \mathcal{L}_{\mathrm{NER}} + \lambda \mathcal{L}_{\mathrm{CWS}} \tag{9}$$

where λ is the balancing parameter used to adjust the influence of word segmentation on entity recognition. To avoid the risk of overfitting, we add a residual connection to the model by summing the output of the embedding layer with the results of the encoding layer. Moreover, we set up a dropout layer between the decoding and encoding layers. The training process of the BCNN-CWS model is presented in Algorithm 1.

Algorithm 1 Training framework of model BCNN-CWS

```
Input: Input sequence $\mathbf{S} = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_\mathrm{N}\}$, and its word
    segmentation label sequence $\dagger_S^{cws}$ and Chinese NER label
    sequence $\mathcal{Y}_S^{ner}$, and maximum training times $T$.
Output: The parameter of shared component $\omega$, and the parameters
    of the unshared part $\vartheta_1$ and $\vartheta_2$.
Embedding characters to generate representation vector
    $\mathbf{E} = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_\mathrm{N}\}$ via Equation (5).
while training times $< T$ do
    Encoding embedding representation $\mathbf{E}$ to generate hidden representation $\mathbf{H}$ with the BCNN model;
    Decoding $\mathbf{H}$ to generate probability labels $\hat{\dagger}^{ner}$, $\hat{\dagger}^{cws}$ with the CRF model for NER and word segmentation;
    Calculating loss via Equations (8), (9), and (10);
    Updating model parameters $\omega$, $\vartheta_1$, and $\vartheta_2$;
end while
Return the optimal model parameters $\omega^*$, $\vartheta_1^*$, and $\vartheta_2^*$.
```

4.2. Bidirectional convolution neural network (BCNN)

As a powerful RNN architecture, the BiLSTM-CRF [29] is the most widely used model for NER. In particular, the BiLSTM architecture has become the de facto standard for encoding sequence contextual information. Although BiLSTM-based models are expressive and accurate, the inherent sequential property, by which the input of the current step requires the output of the previous step, prevents them from fully exploiting GPU parallelism, thus limiting their computational efficiency. Because CNN models have the ability of parallel computation and significantly reduce the time cost, this study used a CNN as the basic architecture for the NER task.

Because a CNN does not have the capacity for long-term memory, stacking multiple convolutional layers is required to capture the contextual information from the entire input sequence. Thus, the number of convolutional layers increases linearly with the length of the sequence, thereby also increasing the time complexity of the NER method. Although the pooling mechanism is always adopted to obtain a large receptive field in CNN models, it reduces the resolution of the representation and is not appropriate for the sequence-labeling tasks. To this end, we utilize an iterated dilated convolutional network [50] to encode the embedding representation of input sentences, which is defined as

$$\mathbf{h}_{\mathrm{t}} = f\left(\mathbf{W}_{\mathrm{c}} \bigoplus_{k=0}^{n/2} \mathbf{e}_{\mathrm{t}\pm\mathrm{k}\delta} + b\right) \tag{10}$$

where the dilated window skips over δ inputs at a time, n is the width of the convolutional sliding window, $\mathbf{W}_{\mathrm{c}}$ is the convolutional kernel, $f(\cdot)$ is the activation function, $\oplus$ is the vector concatenation, δ is the dilation width, and b is the bias. When $\delta > 1$, the dilated convolution incorporates a broader context into the representation of a token than the conventional convolution. Thus, the size of the receptive field increases exponentially with the number of dilated convolution layers, and the contextual information from the entire sequence for the NER task can be captured with a limited number of dilated convolution layers.

In NLP tasks, the words in a sentence are generally assumed to obey the Markov assumption. That is, the probability of the n-th word is based on the $n-1$ previous words in the sentence. However, traditional CNN models do not satisfy the Markov assumption. In this study, we adopt causal convolution to characterize the Markov properties of sentences. The difference between causal and traditional convolutions is that each causal convolution contains current, past, or future information, and no information can be transmitted in the opposite direction. For example, the forward causal convolution considering the current and past information can be defined as

$$\mathbf{h}_{\mathrm{t}} = f\left(\mathbf{W}_{\mathrm{c}} \bigoplus_{k=0}^{n} \mathbf{e}_{\mathrm{t}-\mathrm{k}\delta} + b\right) \tag{11}$$

where t-kδ denotes the past direction of the current position t.

Because each character in a sentence depends on both its past and future contexts, the BCNN model has a bidirectional architecture, which consists of a forward convolution component and a backward convolution component. The forward convolution component consists of three convolutional layers, of which the first two use forward dilated causal convolution, and the last layer is a fully dilated convolution, which can be formulated as

$$\mathbf{h}_{\mathrm{tf}} = f\left(\mathbf{W}''' \bigoplus_{k=0}^{n/2} f\left(\mathbf{W}'' \bigoplus_{k=0}^{n} \left(f\left(\mathbf{W}' \bigoplus_{k=0}^{n} \mathbf{e}_{t-k\delta_1}\right)\right)_{t-k\delta_2}\right)_{t\pm k\delta_3}\right) \tag{12}$$

where the bias term is not explicitly expressed. Similarly, the backward convolution component includes backward dilated causal and full dilation convolutions.

$$\mathbf{h}_{\mathrm{bf}} = f\left(\mathbf{W}''' \bigoplus_{k=0}^{n/2} f\left(\mathbf{W}'' \bigoplus_{k=0}^{n} \left(f\left(\mathbf{W}' \bigoplus_{k=0}^{n} \mathbf{e}_{t+k\delta_1}\right)\right)_{t+k\delta_2}\right)_{t\pm k\delta_3}\right) \tag{13}$$

To ensure the fusion of context information, we concatenate the representations obtained from the forward and backward convolution components.

$$\mathbf{h}_{\mathrm{t}} = \mathbf{h}_{\mathrm{tf}} \oplus \mathbf{h}_{\mathrm{bf}} \tag{14}$$

5. Experiments

In this section, we describe extensive experiments conducted on three public datasets to verify the performance of the proposed methods. We analyze the experimental results to find answers to the following questions:

1) Q1: Does the proposed joint model BCNN-CWS improve the performance of low-resource Chinese NER compared to state-of-the-art methods?
2) Q2: How does the proposed BCNN affect the performance of low-resource Chinese NER?
3) Q3: How does the proposed shared mechanism in the joint training framework improve the performance of low-resource Chinese NER?
4) Q4: What is the effect of the parameters on the performance of the proposed model?

Table 2
Statistic characteristics of the datasets

Dataset	Type	Train	Test
MSRA	Sentence Char	37,092 1,955,826	9,273 172,600
	Entities	62,169	12,534
Ontonote4	Sentence Char	15,724 491,903	4,346 208,066
	Entities	12,581	7,275
PeopleDaily	Sentence Char	1,499,875 18,452	345,715 4,613
	Entities	43,214	10,028

5.1. Experimental settings

Datasets. Experiments are conducted on three benchmark datasets, namely, MSRA [51], PeopleDaily [52], and Ontonotes4 [53], which are representative of Chinese NER. Here, the entity type person, location, organization, and BIO tag scheme are used. Table 2 shows the detailed statistics, including the number of sentences, chars, and entities in the training and test sets.

Evaluation metrics and comparison methods. To evaluate the performance of all NER methods, four evaluation metrics accuracy, precision, recall, and $F1$ score that commonly used in this field are adopted in this study. Moreover, to verify the performance of the proposed models in low-resource CNER tasks, the representative CNER models BiLSTM-CRF, BiGRU-CRF, IDCNN-CRF, and CNN-LSTM-CWS are selected as the comparison methods. These methods are described as follows:

1) BiLSTM-CRF [15]: This is the most commonly used model for entity recognition tasks. It characterizes the past and future input features of input sentence via a BiLSTM layer so that the hidden representation of the sentences can fully obtain the contextual information; it uses a CRF layer to decode the tag information.
2) BiGRU-CRF [30]: This method uses the bidirectional GRU model to represent the contextual information of the input sentence. GRU is an improved method of LSTM, which yields a significant increase in computing speed compared with LSTM.
3) IDCNN-CRF [38]: This method is a sequence labeling model based on dilated convolutions, which extracts the feature information of long sequences by expanding the convolution field of a CNN using a few layers. The purpose of this method is to improve the speed of entity recognition by taking advantage of CNN parallel computing.
4) CNN-LSTM-CWS [48]: This method consists of three layers: CNN, BiLSTM, and CRF. The CNN and BiLSTM layers are used to learn contextual representations from local and long-distance contexts, and the CRF layer is used to decode character labels. In this method, the character embeddings and the CNN network all are shared between NER and word segmentation models.

Implementation details. We use Keras 2.2.4 and Tensorflow 1.14 to implement our model in Windows 10 within 2080Ti GPU. The embedding dimension of each character is 128, and the sentence length is uniform at 100 characters. The redundant parts of sentences with more than 100 characters are removed, and the sentences with less than 100 characters are padded with zeros. There are three convolution layers, and the dilation factor δ_i is set to 1, 2, and 4, respectively. The dropout rate is set to 0.5. The ReLU is used as the activation function, and the learning rate is set to 0.1. The batch size in the implementation is 64, and the number of epochs is 15. Moreover, we adopt the glorot uniform to initialize the weights of our model.

5.2. Performance comparison

Effectiveness of BCNN-CWS. The performance of the proposed method BCNN-CWS is shown in Figure 3. Compared to other NER methods, BCNN-CWS generally achieves superior results over all evaluation metrics on the three datasets, which demonstrates that the multitask learning framework and bidirectional convolution neural network in BCNN-CWS can achieve excellent performance, especially when there are insufficient training samples. From the perspective of model mechanism, the combination with CWS can improve the performance of entity recognition by optimizing entity boundary identification, and the bidirectional causal convolution can effectively characterize the contextual information of sentences.

To further verify the effectiveness of the proposed method in generating hidden feature representations, we denote BCNN-CWS model without the multitask learning mechanism as BCNN and compare it with BiLSTM-CRF, BiGRU-CRF, IDCNN-CRF, and CNN-LSTM-CRF. Here, to compare the effectiveness of the encoding model, the multitask learning mechanism in CNN-LSTM-CWS is also deleted and denoted as CNN-LSTM-CRF. The results of this comparison are illustrated in Figure 4. The experimental results demonstrate that BCNN generally performs better than BiLSTM-CRF, BiGRU-CRF, IDCNN-CRF, and CNN-LSTM-CRF. This implies that the design of iterated dilated convolution and bidirectional causal convolution is effective in capturing contextual information from the entire sentence. In particular, compared with IDCNN-CRF, our BCNN method performs better for the NER task. This demonstrates the effectiveness of forward and backward dilated causal convolutions in the BCNN to characterize the dependency of characters. Moreover, the top half of Table 3 shows the entity tags generated by the NER methods for the test samples selected from the MSRA dataset. For example, in the sentence “周恩来同志批准学院的建设方案。” the BiGRU-CRF model incorrectly identifies “批准学院” as an entity, and the BCNN model correctly identifies them as a non-entity. The results show that the BCNN performs better than the other methods.

Efficiency of BCNN-CWS. Table 4 shows the training times of the methods on the PeopleDaily, MSRA, and Ontonote4 datasets. The proposed BCNN method is more than 50% faster than the BiLSTM-CRF and BiGRU-CRF methods. Similarly, BCNN-CWS performs the best among all the joint models, exhibiting almost 40% improvement over BiLSTM-CWS, BiGRU-CWS, and CNN-LSTM-CWS. The reason behind the improved performance of BCNN-CWS is that BCNN adopts a CNN as the basic architecture and uses dilation convolution operation to limit the number of convolution layers. Moreover, because BCNN and IDCNN are based on the same basic architecture, the running time of them is about the same. At the same time, the above results also show that CNN-based NER methods are significantly more efficient than those based on the RNN model.

Effectiveness of BCNN-CWS in Low-Resource Cases. To evaluate the effectiveness of the proposed model under low-resource conditions, we construct training samples by gradually increasing the amount of data. Table 5 shows the experimental results of the NER methods with various amounts of data. From Table 5, it can be seen that our proposed method BCNN-CWS in general achieves the best results for the experiments in all the

Figure 3
Comparison of BCNN-CWS and state-of-the-art methods on datasets MSRA, PeopleDaily, and Ontonote4

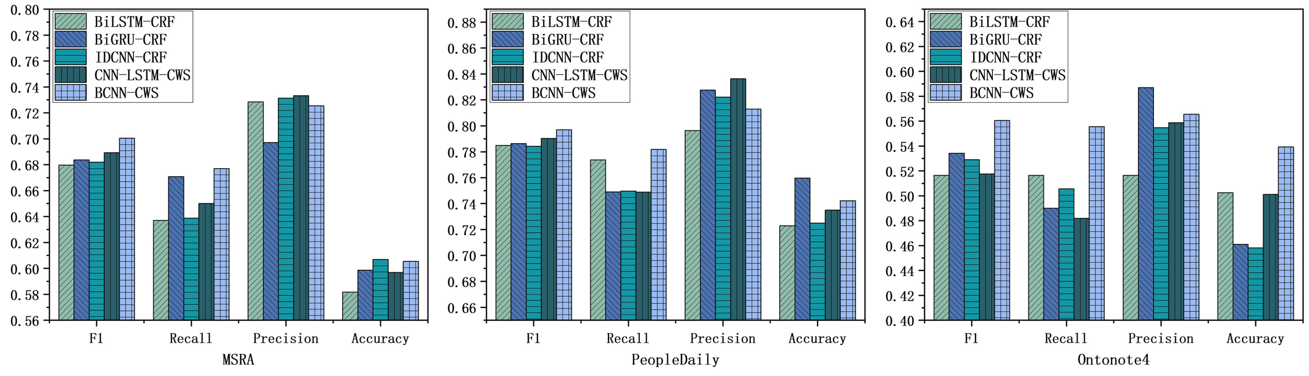


Figure 4
Comparison of BCNN and state-of-the-art methods on datasets MSRA, PeopleDaily, and Ontonote4

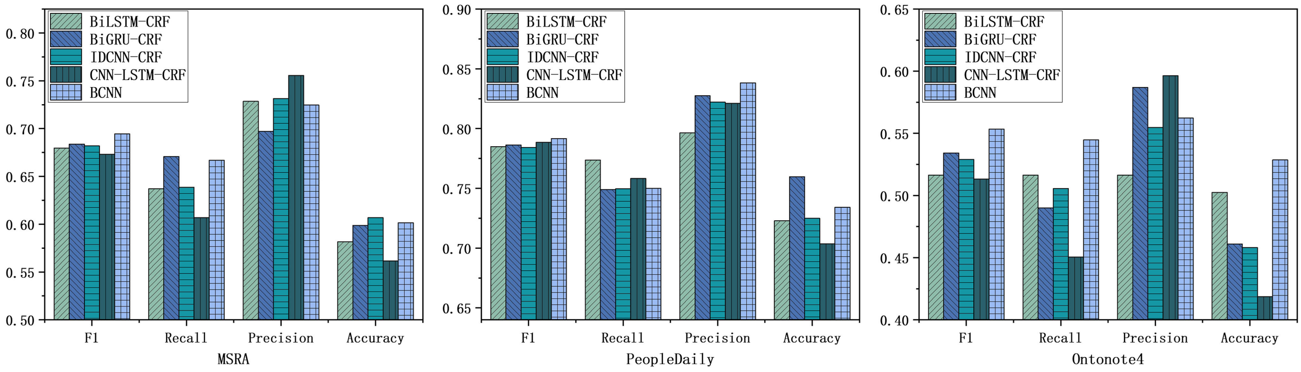


Table 3
Comparison of entity recognition results, where the red tags and the green tags are incorrect and correct results, respectively. The top half presents the results without joint training, and the bottom half illustrates the effectiveness of CWS module

Chinese Text	加速武汉长江三桥和四桥的建设。
BiLSTM-CRF	O O B-LOC I-LOC O B-LOC I-LOC I-LOC O B-LOC I-LOC O O O
BCNN	O O B-LOC I-LOC B-LOC I-LOC I-LOC I-LOC O B-LOC I-LOC O O O
Chinese Text	周恩来同志批准学院的建设方案。
BiGRU-CRF	B-PER I-PER I-PER O O B-ORG I-ORG I-ORG I-ORG O O O O O O
BCNN	B-PER I-PER I-PER O O O O O O O O O O O O
Chinese Text	澳门一定能够实现平稳过度和顺利交接。
IDCNN-CRF	B-LOC O O O O O O O O O O O O O O O O O
BCNN	B-LOC I-LOC O O O O O O O O O O O O O O O O
Chinese Text	国务委员兼国防部长迟浩田。
CNN-LSTM-CRF	O O O O O B-ORG I-ORG I-ORG I-ORG I-ORG I-ORG I-ORG O
BCNN	O O O O O O O O O B-ORG I-ORG I-ORG O
Chinese Text	中华人民共和国驻乌克兰特命全权大使。
BiLSTM-CRF	B-ORG I-ORG I-ORG I-ORG I-ORG I-LOC I-LOC O B-LOC I-LOC I-LOC I-LOC O O O O O O
BiLSTM-CWS	B-ORG I-ORG I-ORG I-ORG I-ORG I-ORG I-ORG O B-LOC I-LOC I-LOC O O O O O O O
Chinese Text	获奖者的建议大部分已付诸实施。
BiGRU-CRF	O O O O B-ORG I-ORG I-ORG I-ORG O O O O O O O
BiGRU-CWS	O O O O O O O O O O O O O O O
Chinese Text	澳门一定能够实现平稳过度和顺利交接。
IDCNN-CRF	B-LOC O O O O O O O O O O O O O O O O O
IDCNN-CWS	B-LOC I-LOC O O O O O O O O O O O O O O O O
Chinese Text	沈阳市政府开始对新开河全面综合治理。
BCNN	B-LOC I-LOC I-LOC O O O O O O O O O O O O O O O
BCNN-CWS	B-LOC I-LOC I-LOC O O O O O B-LOC I-LOC I-LOC O O O O O O O

cases. This indicates that our proposed method can effectively solve the low-resource problem.

5.3. Ablation experiment

Effects of the joint training mechanism with CWS. We conduct an ablation experiment for the CWS module to verify the contributions of the main components of our model to the performance of the entity recognition task. According to the sequence labeling underlying both NER and CWS tasks, our model, BCNN-CWS, uses CWS as the auxiliary task and defines a multitask learning framework to generate hidden representations adequately for entity recognition. Moreover, although CNER and CWS have common characteristics, they have differences that prevent them from completely sharing the underlying model architecture, that is, the embedding and encoding layers. Therefore, we evaluate the proposed multitask learning framework with CWS to verify its effectiveness for the CNER task. We then conduct experiments with various sharing strategies for the embedding and encoding layers between CNER and CWS to explore their effects on entity recognition.

To evaluate the proposed multitask learning framework with CWS for the CNER task, BiLSTM, BiGRU, IDCNN, and BCNN are used as encoding layers in the framework to form the joint CNER models BiLSTM-CWS, BiGRU-CWS, IDCNN-CWS, and BCNN-CWS. Table 6 shows the NER results of the joint models

Table 4
Efficiency of the NER methods on datasets MSRA, PeopleDaily, and Ontonote4. The top and bottom half present the results of the methods without and with joint training with CWS module. Bold numbers denote the best results

Model	Dataset		
	PeopleDaily	MSRA	Ontonote4
BiLSTM-CRF	1186s	1449s	1143s
BiGRU-CRF	1018s	1189s	977s
IDCNN-CRF	556s	703s	539s
CNN-LSTM-CRF	1523s	1865s	1508s
BCNN	**544s**	**686s**	**535s**
BiLSTM-CWS	2524s	3027s	2500s
BiGRU-CWS	2270s	2512s	2267s
IDCNN-CWS	1377s	1798s	1405s
CNN-LSTM-CWS	2340s	2774s	2370s
BCNN-CWS	**1364s**	**1760s**	**1332s**

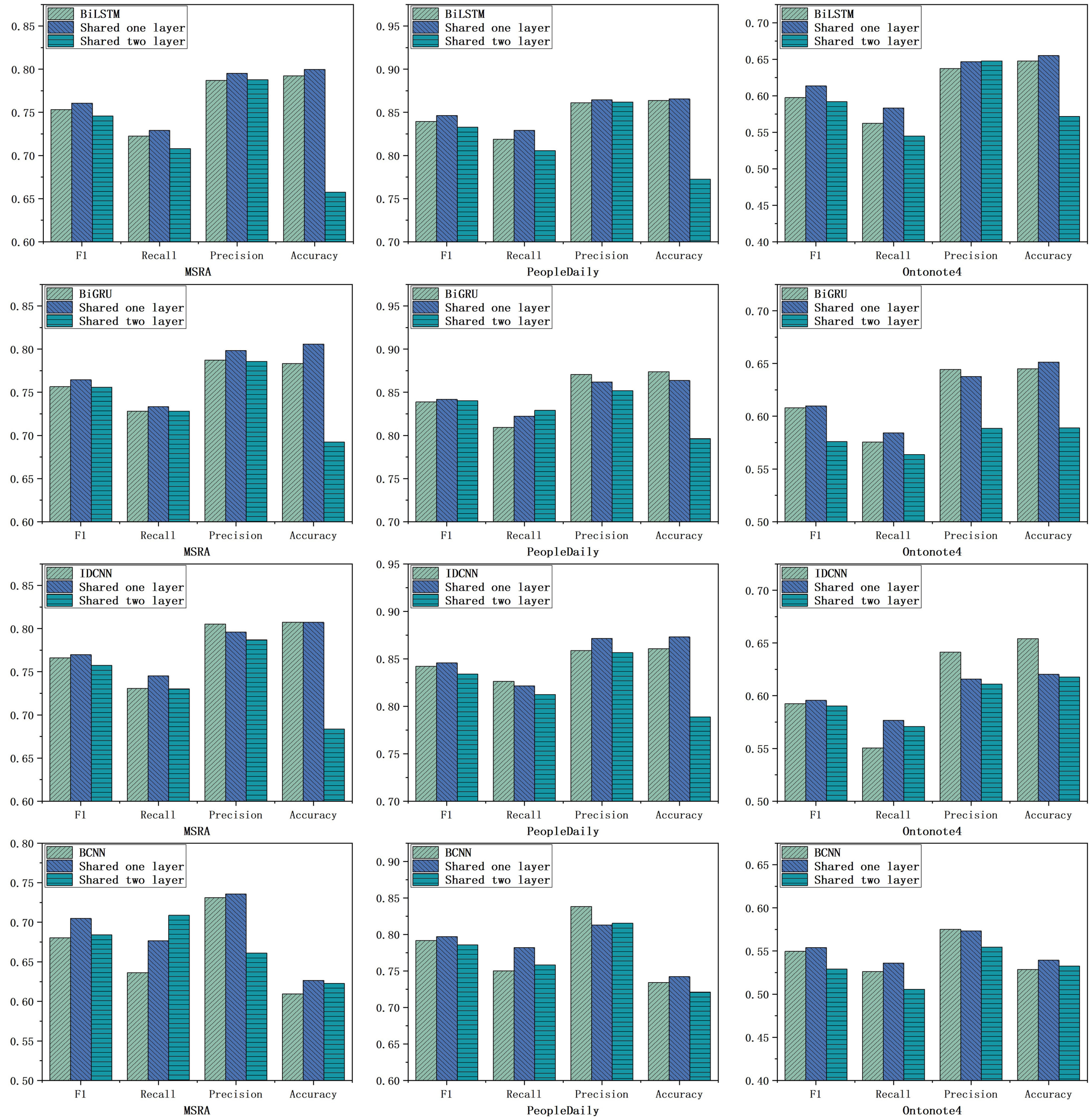
Table 5
Effectiveness of the NER methods with various amounts of training data. Bold numbers denote the best results

Models	MSRA(5k)				PeopleDaliy(5k)				Ontonote4(5k)			
	*F*1 Score	Recall	Precision	Accuracy	*F*1 Score	Recall	Precision	Accuracy	*F*1 Score	Recall	Precision	Accuracy
BiLSTM	0.6797	0.6370	0.7316	0.5817	0.7849	0.7737	0.7973	0.7229	0.5164	0.5164	0.5690	0.5025
BiGRU	0.6837	0.6708	0.7003	0.5987	0.7863	0.7490	0.8328	**0.7597**	0.5342	0.4901	**0.5940**	0.4610
IDCNN	0.6819	0.6387	0.7360	0.6069	0.7842	0.7496	0.8271	0.7249	0.5290	0.5056	0.5581	0.4582
CNN-LSTM-CWS	0.6962	0.6515	**0.7522**	0.5616	0.7914	0.7391	**0.8551**	0.7035	0.5290	0.5031	0.5875	0.5011
BCNN-CWS	**0.7004**	**0.6771**	0.7273	**0.6095**	**0.7942**	**0.7742**	0.8181	0.7342	**0.5605**	**0.5556**	0.5674	**0.5286**
Models	MSRA(10k)				PeopleDaliy(10k)				Ontonote4(10k)			
	F1 Score	Recall	Precision	Accuracy	F1 Score	Recall	Precision	Accuracy	F1 Score	Recall	Precision	Accuracy
BiLSTM	0.7533	0.7224	0.7870	0.7923	0.8395	0.8189	0.8612	0.8637	0.5974	0.5622	0.6373	0.6477
BiGRU	0.7565	0.7280	0.7873	0.7834	0.8388	0.8092	0.8706	**0.8739**	0.6080	0.5755	0.6444	0.6451
IDCNN	0.7662	0.7307	**0.8055**	**0.8074**	0.8422	0.8262	0.8588	0.8731	0.5925	0.5505	0.6414	**0.6541**
CNN-LSTM-CWS	0.7673	0.7398	0.8014	0.7169	0.8523	0.8295	**0.8773**	0.7966	0.5397	0.5031	0.5875	0.6538
BCNN-CWS	**0.7709**	**0.7503**	0.7927	0.7934	**0.8532**	**0.8405**	0.8663	0.8686	**0.6438**	**0.6377**	**0.6500**	0.6521
Models	MSRA(all)				PeopleDaliy(all)				Ontonote4(all)			
	F1 Score	Recall	Precision	Accuracy	F1 Score	Recall	Precision	Accuracy	F1 Score	Recall	Precision	Accuracy
BiLSTM	0.8714	0.8647	0.8794	0.8294	0.8795	0.8696	0.8898	0.8513	0.6178	0.5821	0.6725	0.6151
BiGRU	0.8713	0.8424	0.9027	0.8436	0.8757	0.8683	0.8838	0.8488	0.6187	0.5668	**0.6933**	0.5723
IDCNN	0.8699	0.8545	0.8867	0.8495	0.8790	0.8558	0.9043	0.8191	0.6086	0.5787	0.6499	0.6149
CNN-LSTM-CWS	0.8759	0.8448	**0.9114**	0.8429	0.8843	0.8525	**0.9193**	0.8499	0.6335	0.6112	0.6594	0.5483
BCNN-CWS	**0.8792**	**0.8661**	0.8933	**0.8573**	**0.8896**	**0.8793**	0.9017	**0.8518**	**0.6501**	**0.6373**	0.6649	**0.6875**

Table 6
Ablation experiment about multitask learning with CWS. Bold numbers denote the best results

Models	MSRA				PeopleDaliy				Ontonote4			
	*F*1 Score	Recall	Precision	Accuracy	*F*1 Score	Recall	Precision	Accuracy	*F*1 Score	Recall	Precision	Accuracy
BiLSTM	0.7533	0.7224	**0.7870**	0.7923	0.8395	0.8189	0.8612	0.8637	0.5974	0.5622	0.6373	0.6477
BiLSTM-CWS	**0.7607**	**0.7291**	0.7952	**0.7997**	**0.8464**	**0.8290**	**0.8645**	**0.8655**	**0.6134**	**0.5833**	**0.6468**	**0.6550**
BiGRU	0.7565	0.7280	0.7873	0.7834	0.8388	0.8092	**0.8706**	**0.8739**	0.6080	0.5755	**0.6444**	0.6451
BiGRU-CWS	**0.7645**	**0.7334**	**0.7984**	**0.8059**	**0.8417**	**0.8223**	0.8620	0.8637	**0.6098**	**0.5842**	0.6377	**0.6514**
IDCNN	0.7662	0.7307	**0.8053**	**0.8074**	0.8422	**0.8262**	0.8588	0.8607	0.5925	0.5505	**0.6414**	**0.6541**
IDCNN-CWS	**0.7697**	**0.7452**	0.7959	0.7973	**0.8457**	0.8214	**0.8715**	**0.8731**	**0.5956**	**0.5767**	0.6158	0.6203
BCNN	0.7680	0.7460	0.7913	0.7927	0.8504	0.8376	0.8636	0.8645	0.6187	0.6066	0.6313	0.6362
BCNN-CWS	**0.7709**	**0.7503**	**0.7927**	**0.7934**	**0.8532**	**0.8405**	**0.8663**	**0.8686**	**0.6438**	**0.6377**	**0.6500**	**0.6521**

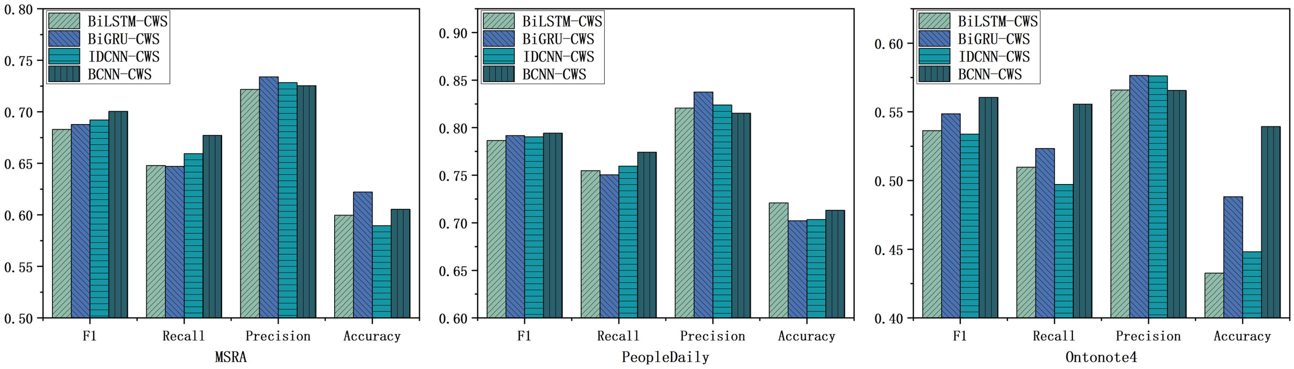
Figure 5
Effects of various sharing mechanisms between NER and CWS in the proposed joint training framework



compared with the original methods. We can find that the joint models BiLSTM-CWS, BiGRU-CWS, IDCNN-CWS, and BCNN-CWS performed better than the non-joint models on all datasets. The average improvements of all the joint models over the non-joint models on the PeopleDaily, MSRA, and PeopleDaily datasets are 0.4%, 0.66%, and 1.4%, respectively. These results demonstrate that multitask learning with CWS task can effectively improve the performance of CNER in low-resource scenarios. Moreover, to evaluate the proposed multitask learning framework, the bottom half of Table 3 also shows the entity tags generated by the CNER methods. The results show that the joint models combining with word segmentation tasks obtain better entity recognition results.

To verify the rationality of the sharing mechanism in BCNN-CWS, we compare the non-joint models, the joint models sharing only the embedding layer with CWS, and the joint models sharing both embedding and encoding layers with CWS. According to the results of this comparison, shown in Figure 5, it can be concluded that the models sharing only the embedding layer with CWS generally perform best on all datasets, which proves the effectiveness of the sharing mechanism proposed in this paper. Moreover, we can observe from the results that the models

Figure 6
Ablation experiment about BCNN model on datasets MSRA, PeopleDaily, and Ontonote4



sharing both layers perform worse than the models sharing only the embedding layer and even worse than the non-joint models in most cases. This observation demonstrates that fully sharing the underlying mechanisms of CNER and CWS negatively affects the performance of the CNER task.

Effects of the BCNN model. The proposed method BCNN-CWS mainly includes the joint training mechanism with CWS and the BCNN model. Besides the joint training mechanism with CWS, here we evaluate the contribution of the BCNN model to the proposed method. To this end, BiLSTM, BiGRU, IDCNN, and BCNN are used as the encoding layers in our joint training framework, denoted by BiLSTM-CWS, BiGRU-CWS, IDCNN-CWS, and BCNN-CWS, respectively. Thus, the encoding models of the methods are fairly compared under the same calculation scenario, and the results on the PeopleDaily, MSRA, and Ontonote4 datasets are displayed in Figure 6. Evidently, the proposed BCNN model performs better than the comparison models. This indicates that BCNN model significantly contributes to the performance advantages of the BCNN-CWS model.

5.4. Model analysis

Impact of the balance coefficient. The balance coefficient λ in Equation (9) is used to adjust the influence of the word segmentation task on the performance of entity recognition methods. That is, the coefficient λ determines the influence of the CWS module on the learned hidden representation for entity recognition. Figure 7 shows the results of BCNN-CWS model under various ratios between $\mathcal{L}_{\text{NER}}$ and $\mathcal{L}_{\text{CWS}}$ in Equation (9). The results show that BCNN-CWS model achieves the best results on the datasets when the weights of the loss functions for entity recognition and word segmentation are similar, i.e., λ approaches 1, which is the choice made in this study.

Effectiveness of the bidirectional layers. To improve the performance of the CNER method, we propose a bidirectional convolutional model to comprehensively learn the contextual information of the sentences. To verify whether the bidirectional convolutional model is effective and outperforms the unidirectional convolutional model for entity recognition, we conduct experiments on the PeopleDaily, MSRA, and Ontonote4 datasets. The experimental results, shown in Figure 8, prove that the bidirectional model outperformed the unidirectional model in all datasets. Thus, it is experimentally demonstrated that the proposed bidirectional convolutional model may compensate for the shortcomings of the unidirectional convolutional model and has a better capability for characterizing contextual information.

Impact of convolutional layer dilation factors. Table 7 shows the results of BCNN-CWS under various dilation factors δ. According to Table 7, the performance of the BCNN-CWS for entity recognition varies as the value of the dilation factors changes. Here, the best performance is achieved when the dilation factors are equal to 1, 2, 4, which is the choice in this study.

Figure 7
Effect of balance coefficient λ. The horizontal axis indicates the ratio between NER and CW loss functions

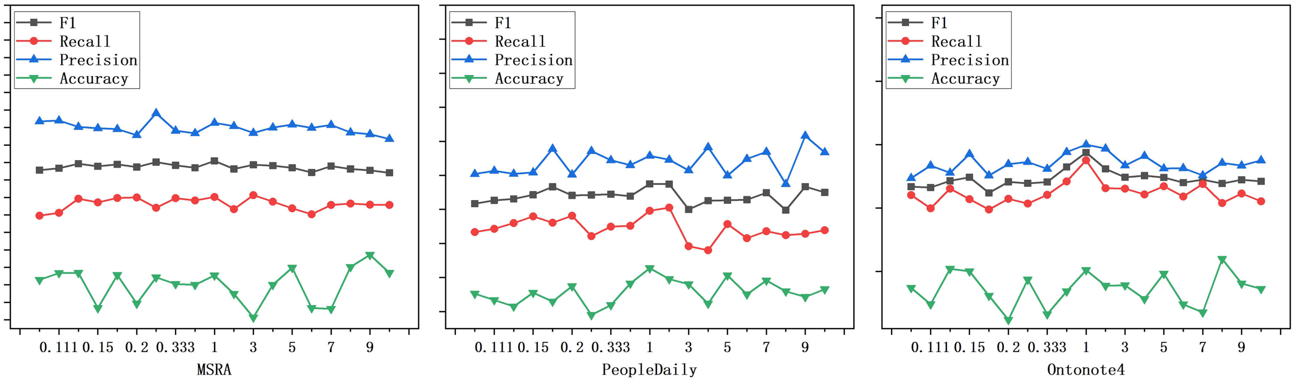


Figure 8
Results of unidirectional and bidirectional convolution models on datasets MSRA, PeopleDaily, and Ontonote4

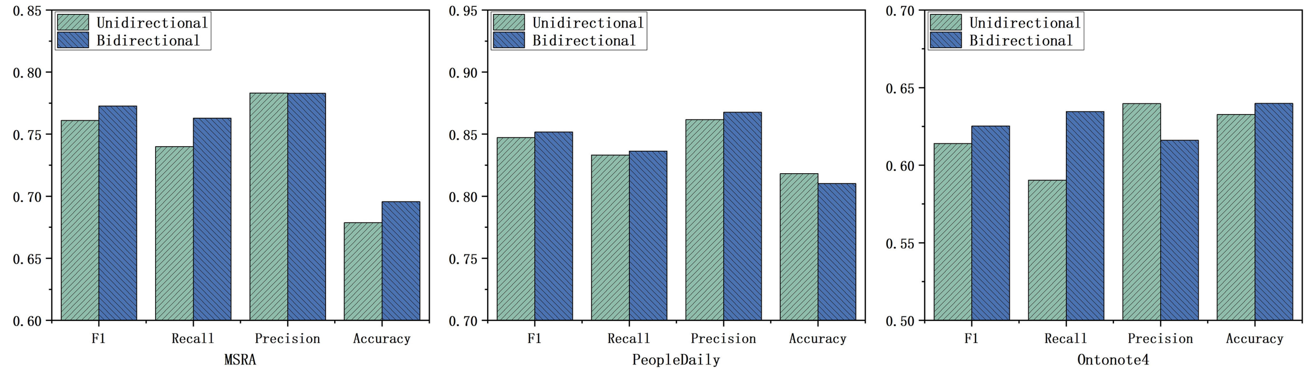


Table 7
Impact of convolutional layer dilation factor

Dilate	MSRA				PeopleDaliy				Ontonote4			
	*F*1 Score	Recall	Precision	Accuracy	*F*1 Score	Recall	Precision	Accuracy	*F*1 Score	Recall	Precision	Accuracy
1,2,4	**0.7080**	**0.6774**	**0.7415**	0.6249	**0.7934**	**0.7738**	**0.8140**	0.7276	**0.5588**	0.5397	**0.5793**	0.5377
1,3,5	0.7036	0.6739	0.7360	0.6174	0.7895	0.7689	0.8112	0.7343	0.5486	0.5366	0.5611	**0.5583**
1,4,6	0.6947	0.6784	0.7118	**0.6271**	0.7872	0.7732	0.8017	**0.7364**	0.5582	**0.5428**	0.5745	0.5407
2,4,6	0.5989	0.5343	0.6813	0.4596	0.7035	0.6773	0.7318	0.6248	0.4721	0.4615	0.4832	0.4449
2,5,7	0.5876	0.5404	0.6438	0.4740	0.7048	0.6794	0.7322	0.6280	0.4670	0.4406	0.4968	0.4237

6. Conclusions and Discussion

In this study, we investigated the low-resource CNER problem and presented a novel bidirectional dilated convolution-based entity recognition model called BCNN-CWS. Based on multitask learning, auxiliary data on CWS were incorporated into the training process to learn high-quality hidden representations, thereby improving the performance of entity recognition. Specifically, considering the relatedness and difference between entity recognition and word segmentation tasks, a joint training framework that shares only the embedding layer was developed to learn the hidden representations. To increase the efficiency of entity recognition methods, we defined a CNN architecture by stacking convolutional layers, in which dilated convolution was used to capture contextual information from the entire input sequence using only a limited number of convolutional layers. Moreover, to characterize the dependency of characters and model the Markov property of sentences, a bidirectional causal convolution mechanism was proposed. Extensive experiments shown that the BCNN-CWS model outperforms state-of-the-art entity recognition methods; it yields up to about 50% speed improvement over existing methods. According to ablation experiments and model analysis, the multitask learning framework and encoding model BCNN of BCNN-CWS exhibited satisfactory performance. This work provides a new reference for the subsequent research of deep learning-based methods and lays a foundation for NER research based on new technologies such as pretraining. In addition, the proposed method can be used in machine translation, question answering, and knowledge graph construction to promote the construction and application of intelligent systems.

This study focused on the problem of non-nested entity recognition. Similar to existing works on this topic, the auxiliary task for joint training in this study is selected manually. Therefore, it is heavily dependent on domain knowledge and is a trial-and-error process. In the future, the proposed entity recognition model can be optimized using a learning mechanism to automatically determine optimal auxiliary tasks. Moreover, the proposed BCNN-CWS model assumes that all input sentences are reliable. However, the vulnerability of entity recognition methods under adversarial attacks has received little attention. Thus, the evaluation and development of robust entity recognition methods should be considered in future research.

Acknowledgment

This work was partially supported by the National Natural Science Foundation of China under Grant Nos. 62376047, 62106030; Key Project of Science and Technology Research Program of Chongqing Education Commission of China under Grant No. KJZD-K202300603; Chongqing Technology Innovation and Application Development Project under Grant No. CSTB2022TIAD-GPX0014.

Ethical Statement

This study does not contain any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

The data that support the findings of this study are openly available in Github at https://github.com/jiangfeng13/BCNN-CWS.

Authors Contribution Statement

Tao Wu: Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Visualization, Supervision. **Xinwen Cao:** Resources, Data curation. **Feng Jiang:** Software, Validation, Formal analysis, Investigation, Writing – original draft. **Canyixing Cui:** Data curation, Writing – review & editing. **Xuehao Li:** Resources. **Xingping Xian:** Supervision, Project administration, Funding acquisition.

References

[1] Li, J., Sun, A., Han, J., & Li, C. (2020). A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, *34*(1), 50–70.

[2] Roy, A. (2021). Recent trends in named entity recognition (NER). *arXiv Preprint:2101.11420*.

[3] Nayak, T., & Ng, H. T. (2020). Effective modeling of encoder-decoder architecture for joint entity and relation extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*, 8528–8535.

[4] Zhukova, A., Hamborg, F., & Gipp, B. (2021). ANEA: Automated (named) entity annotation for German domain-specific texts. In *2nd Workshop on Extraction and Evaluation of Knowledge Entities from Scientific Documents*, 5–14.

[5] Sevgili, Ö., Shelmanov, A., Arkhipov, M., Panchenko, A., & Biemann, C. (2022). Neural entity linking: A survey of models based on deep learning. *Semantic Web*, *13*(3), 527–570.

[6] De Cao, N., Wu, L., Popat, K., Artetxe, M., Goyal, N., Plekhanov, M., . . . , & Petroni, F. (2022). Multilingual autoregressive entity linking. *Transactions of the Association for Computational Linguistics*, *10*, 274–290.

[7] Chen, X., Ouyang, C., Liu, Y., & Bu, Y. (2020). Improving the named entity recognition of Chinese electronic medical records by combining domain dictionary and rules. *International Journal of Environmental Research and Public Health*, *17*(8), 2687.

[8] Perikos, I., Kardakis, S., & Hatzilygeroudis, I. (2021). Sentiment analysis using novel and interpretable architectures of hidden markov models. *Knowledge-Based Systems*, *229*, 107332.

[9] Riaz, F., Anwar, M. W., & Muqades, H. (2020). Maximum entropy based Urdu named entity recognition. In *2020 International Conference on Engineering and Emerging Technologies*, 1–5.

[10] Patil, N., Patil, A., & Pawar, B. (2020). Named entity recognition using conditional random fields. *Procedia Computer Science*, *167*, 1181–1188.

[11] Nasar, Z., Jaffry, S. W., & Malik, M. K. (2021). Named entity recognition and relation extraction: State-of-the-art. *ACM Computing Surveys*, *54*(1), 1–39.

[12] Shang, F., & Ran, C. (2022). An entity recognition model based on deep learning fusion of text feature. *Information Processing & Management*, *59*(2), 102841.

[13] Su, S., Qu, J., Cao, Y., Li, R., & Wang, G. (2022). Adversarial training lattice LSTM for named entity recognition of rail fault texts. *IEEE Transactions on Intelligent Transportation Systems*, *23*(11), 21201–21215.

[14] Min, B., Ross, H., Sulem, E., Veyseh, A. P. B., Nguyen, T. H., Sainz, O., . . . , & Roth, D. (2023). Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Computing Surveys*, *56*(2), 1–40.

[15] Dai, Z., Wang, X., Ni, P., Li, Y., Li, G., & Bai, X. (2019). Named entity recognition using BERT BiLSTM CRF for Chinese electronic health records. In *2019 12th International Congress on Image and Signal Processing, Biomedical Engineering and Informatics*, 1–5.

[16] Wang, S., Sun, X., Li, X., Ouyang, R., Wu, F., Zhang, T., . . . , & Wang, G. (2023). GPT-NER: Named entity recognition via large language models. *arXiv Preprint:2304.10428*.

[17] Lange, L., Hedderich, M. A., & Klakow, D. (2019). Feature-dependent confusion matrices for low-resource NER labeling with noisy labels. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 3554–3559.

[18] Liu, P., Guo, Y., Wang, F., & Li, G. (2022). Chinese named entity recognition: The state of the art. *Neurocomputing*, *473*, 37–53.

[19] Liang, S., Gong, M., Pei, J., Shou, L., Zuo, W., Zuo, X., & Jiang, D. (2021). Reinforced iterative knowledge distillation for cross-lingual named entity recognition. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 3231–3239.

[20] Mo, Y., Yang, J., Liu, J., Wang, Q., Chen, R., Wang, J., & Li, Z. (2024). mCL-NER: Cross-lingual named entity recognition via multi-view contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, *38*, 18789–18797.

[21] Qian, T., Zhang, M., Lou, Y., & Hua, D. (2021). A joint model for named entity recognition with sentence-level entity type attentions. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *29*, 1438–1448.

[22] Kim, J.-H., & Woodland, P. C. (2000). A rule-based named entity recognition system for speech input. In *Sixth International Conference on Spoken Language Processing*, 1–4.

[23] Hanisch, D., Fundel, K., Mevissen, H.-T., Zimmer, R., & Fluck, J. (2005). Prominer: Rule-based protein and gene entity recognition. *BMC Bioinformatics*, *6*(1), 1–9.

[24] Krupka, G., & IsoQuest, K. (2005). Description of the netOwl extractor system as used for MUC-7. In *Proceedings of 7th Message Understanding Conference*, 21–28.

[25] Chang, Y., Kong, L., Jia, K., & Meng, Q. (2021). Chinese named entity recognition method based on BERT. In *2021 IEEE International Conference on Data Science and Computer Application*, 294–299.

[26] Zhou, G., & Su, J. (2002). Named entity recognition using an HMM-based chunk tagger. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 473–480.

[27] Bender, O., Och, F. J., & Ney, H. (2003). Maximum entropy models for named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, 148–151.

[28] Sobhana, N., Mitra, P., & Ghosh, S. (2010). Conditional random field based named entity recognition in geological text. *International Journal of Computer Applications*, *1*(3), 143–147.

[29] Gao, W., Zheng, X., & Zhao, S. (2021). Named entity recognition method of Chinese EMR based on BERT-BiLSTM-CRF. *Journal of Physics: Conference Series*, *1848*(1), 012083.

[30] Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv Preprint:1412.3555*.

[31] Gridach, M., & Haddad, H. (2017). Arabic named entity recognition: A bidirectional GRU-CRF approach. In *International Conference on Computational Linguistics and Intelligent Text Processing*, 264–275.

[32] Yang, G., & Xu, H. (2020). A residual BiLSTM model for named entity recognition. *IEEE Access*, *8*, 227710–227718.

[33] Rei, M., Crichton, G., & Pyysalo, S. (2016). Attending to characters in neural sequence labeling models. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 309–318.

[34] Yuan, L., Wang, J., Yu, L.-C., & Zhang, X. (2024). Encoding syntactic information into transformers for aspect-based sentiment triplet extraction. *IEEE Transactions on Affective Computing*, *15*(2), 722–735.

[35] Yuan, L., Cai, Y., Wang, J., & Li, Q. (2023). Joint multimodal entity-relation extraction based on edge-enhanced graph alignment network and word-pair relation tagging. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 11051–11059.

[36] Xiao, Y., Ji, Z., Li, J., & Zhu, Q. (2025). DualFlat: Dual flat-lattice transformer for domain-specific Chinese named entity recognition. *Information Processing & Management*, *62*(1), 103902.

[37] Zeng, Q., Yuan, M., Wan, J., Wang, K., Shi, N., Che, Q., & Liu, B. (2024). ICKA: An instruction construction and knowledge alignment framework for multimodal named entity recognition. *Expert Systems with Applications*, *255*, 124867.

[38] Strubell, E., Verga, P., Belanger, D., & McCallum, A. (2017). Fast and accurate entity recognition with iterated dilated convolutions. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2670–2680.

[39] Li, B., Hou, Y., & Che, W. (2022). Data augmentation approaches in natural language processing: A survey. *AI Open*, *3*, 71–90.

[40] Dai, X., & Adel, H. (2020). An analysis of simple data augmentation for named entity recognition. In *Proceedings of the 28th International Conference on Computational Linguistics*, 3861–3867.

[41] Chao, G., Liu, J., Wang, M., & Chu, D. (2023). Data augmentation for sentiment classification with semantic preservation and diversity. *Knowledge-Based Systems*, *280*, 111038.

[42] Sennrich, R., Haddow, B., & Birch, A. (2016). Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 86–96.

[43] Thakur, N., Reimers, N., Daxenberger, J., & Gurevych, I. (2021). Augmented SBERT: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 296–310.

[44] Kruengkrai, C., Nguyen, T. H., Aljunied, S. M., & Bing, L. (2020). Improving low-resource named entity recognition using joint sentence and token labeling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 5898–5905.

[45] Martins, P. H., Marinho, Z., & Martins, A. F. (2019). Joint learning of named entity recognition and entity linking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, 190–196.

[46] Wang, D., Peng, N., & Duh, K. (2017). A multi-task learning approach to adapting bilingual word embeddings for cross-lingual named entity recognition. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, 383–388.

[47] Rei, M., & Søgaard, A. (2019). Jointly learning to label sentences and tokens. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 6916–6923.

[48] Wu, F., Liu, J., Wu, C., Huang, Y., & Xie, X. (2019). Neural Chinese named entity recognition via CNN-LSTM-CRF and joint training with word segmentation. In *The World Wide Web Conference*, 3342–3348.

[49] Jie, Z., & Lu, W. (2019). Dependency-guided LSTM-CRF for named entity recognition. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 3862–3872.

[50] Salehi, A., & Balasubramanian, M. (2023). DDCNet: Deep dilated convolutional neural network for dense prediction. *Neurocomputing*, *523*, 116–129.

[51] Zhang, S., Qin, Y., Hou, W.-J., & Wang, X. (2006). Word segmentation and named entity recognition for SIGHAN bakeoff3. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, 158–161.

[52] Yu, S. (2003). Specification for corpus processing at Peking University: Word segmentation, pos tagging and phonetic notation. *Journal of Chinese Language and Computing*, *13*, 121–158.

[53] Pradhan, S. S., Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., & Weischedel, R., (2007). OntoNotes: A unified relational semantic representation. In *International Conference on Semantic Computing*, 517–526.

How to Cite: Wu, T., Cao, X., Jiang, F., Cui, C., Li, X., & Xian, X. (2024). Low-Resource Chinese Named Entity Recognition via CNN-based Multitask Learning. *Journal of Data Science and Intelligent Systems*. https://doi.org/10.47852/bonviewJDSIS42024432