applied sciences

MDPI

Article

A Robust Chinese Named Entity Recognition Method Based on Integrating Dual-Layer Features and CSBERT

Yingjie Xu, Xiaobo Tan *, Xin Tong and Wenbo Zhang

School of Information Science and Engineering, Shenyang Ligong University, Shenyang 110159, China; geoffrey@stu.sylu.edu.cn (Y.X.); tongxin990226@gmail.com (X.T.); zhangwenbo@yeah.net (W.Z.)
* Correspondence: tanxiaobo@sylu.edu.cn

Abstract: In the rapidly evolving field of cybersecurity, the integration of multi-source, heterogeneous, and fragmented data into a coherent knowledge graph has garnered considerable attention. Such a graph elucidates semantic interconnections, thereby facilitating sophisticated analytical decision support. Central to the construction of a cybersecurity knowledge graph is Named Entity Recognition (NER), a critical technology that converts unstructured text into structured data. The efficacy of NER is pivotal, as it directly influences the integrity of the knowledge graph. The task of NER in cybersecurity, particularly within the Chinese linguistic context, presents distinct challenges. Chinese text lacks explicit space delimiters and features complex contextual dependencies, exacerbating the difficulty in discerning and categorizing named entities. These linguistic characteristics contribute to errors in word segmentation and semantic ambiguities, impeding NER accuracy. This paper introduces a novel NER methodology tailored for the Chinese cybersecurity corpus, termed CSBERT-IDCNN-BiLSTM-CRF. This approach harnesses Iterative Dilated Convolutional Neural Networks (IDCNN) for extracting local features, and Bi-directional Long Short-Term Memory networks (BiLSTM) for contextual understanding. It incorporates CSBERT, a pre-trained model adept at processing few-shot data, to derive input feature representations. The process culminates with Conditional Random Fields (CRF) for precise sequence labeling. To compensate for the scarcity of publicly accessible Chinese cybersecurity datasets, this paper synthesizes a bespoke dataset, authenticated by data from the China National Vulnerability Database, processed via the YEDDA annotation tool. Empirical analysis affirms that the proposed CSBERT-IDCNN-BiLSTM-CRF model surpasses existing Chinese NER frameworks, with an F1-score of 87.30% and a precision rate of 85.89%. This marks a significant advancement in the accurate identification of cybersecurity entities in Chinese text, reflecting the model's robust capability to address the unique challenges presented by the language's structural intricacies.

Keywords: cybersecurity knowledge graph; Chinese named entity recognition; CSBERT; few-shot data; cybersecurity dataset

Citation: Xu, Y.; Tan, X.; Tong, X.; Zhang, W. A Robust Chinese Named Entity Recognition Method Based on Integrating Dual-Layer Features and CSBERT. *Appl. Sci.* **2024**, *14*, 1060. https://doi.org/10.3390/app14031060

Academic Editor: Andrea Prati

Received: 19 December 2023
Revised: 25 January 2024
Accepted: 25 January 2024
Published: 26 January 2024

1. Introduction

As network infrastructure continues to improve, traditional industries are moving towards networking, digitalization, and intelligence. This trend is leading to a rapid iteration in communication network technology to meet the increasing industrial capacity demands. However, behind this rapid development in network technology and infrastructure lies significant cybersecurity risks. The "2020 Annual Overview of China's Internet Network Security Situation" [1] focuses on describing potential network risks such as APT attacks, supply chain attacks, illegal collection of personal information, ransomware viruses, targeted delivery, and Wildcard Domain Resolution.

Meanwhile, according to the "2022 Annual Report on Network Security Vulnerability Trends", nearly 25,000 new vulnerabilities were reported in 2022, reaching a historical high and maintaining a trend of annual growth. The overall situation has seen new changes,

characterized by a surge in high-risk vulnerabilities, intensified competition in zero-day exploits, disruptions in international order due to one-sided vulnerability controls, and challenges to cyberspace rights by network hegemony. Consequently, the overall network security situation has become more complex and severe.

Cyberspace is a vast and intricate information environment. In the field of cybersecurity, conventional cybersecurity solutions often rely on predefined rules or signatures to detect and defend against known threats. However, they tend to struggle against newly emerged or unknown types of attack methods. To leverage the scattered and fragmented cybersecurity-related data more effectively within the Internet and enhance capabilities in threat intelligence analysis, risk assessment, and the formulation of protective measures, researchers and engineers are now embarking on the construction of a cybersecurity knowledge graph. A framework constructed based on graph theory and technology can integrate disparate information into a structured knowledge base that is meaningful and easily queryable and analyzable. Named Entity Recognition (NER) technology plays a crucial role in this process. Through Natural Language Processing (NLP) algorithms, NER accurately identifies various entities from unstructured text, such as names of malicious software, identifiers of hacker organizations, system vulnerability codes, and other crucial pieces of information. NER technology aids in rapidly extracting valuable information from extensive textual data, transforming it into a structured format that can be further processed and analyzed.

The existing research and applications of NER technology mostly concentrate on the English language context, where this field has made significant progress. In the English environment, there is a relatively well-established theoretical foundation, technical framework, and abundant data repositories. For instance, within the domain of English cybersecurity, NLP tools can effectively extract critical information such as IP addresses, names of malicious software, vulnerability identifiers, etc., and accurately map them to respective nodes in a knowledge graph.

However, when attempting to transfer these advanced techniques into the Chinese language environment, various challenges arise. Firstly, issues stem from the characteristics of Chinese characters. Unlike English words that are clearly separated by spaces, Chinese characters lack fixed delimiters when written. Secondly, many Chinese characters exhibit polysemy, where a single character can possess different pronunciations and meanings based on different contexts. This complexity makes direct segmentation using spaces or simple character matching to identify entities become intricate and error prone. Moreover, traditional entity recognition methods struggle to adapt to the iterative nature of cybersecurity entities and fail to resolve issues related to semantic overlap among cybersecurity entities. Existing research outcomes are insufficient to support the construction of a Chinese cybersecurity knowledge graph.

This paper proposes a joint model based on CSBERT-IDCNN-BiLSTM-CRT to address the current issues of lack of datasets and low accuracy in Chinese named entity recognition in the field of Chinese network security. This method utilizes the pre-trained model CSBERT to ensure high performance even with small datasets, and integrates IDCNN for extracting local features, BiLSTM for extracting contextual features, and CRF for sequence labeling to further improve the accuracy of named entity recognition in the Chinese network domain, better accomplishing the task of transforming unstructured text into structured text.

The rest of this paper is organized as follows: Section 2 reviews recent domestic and international research achievements in NER problems and cybersecurity knowledge graphs. Section 3 details how to establish the CSBERT-IDCNN-BiLSTM-CRT approach for entity recognition in the Chinese cyber domain. Section 4 presents training results and comparative tests of the model, while Section 5 provides an experimental summary, concluding with references at the end of this paper.

2. Related Work

At present, it is widely recognized in the industry that utilizing entity recognition technology to convert unstructured data into structured data and assist in building knowledge graphs is a direction with prospects and practical significance. However, currently, most named entity recognition is applied in general domains, and there are fewer named entity recognition technologies specifically designed for the field of network security. In recent years, with continuous iterations of software and hardware, deep neural network-based models for named entity recognition have replaced traditional methods based on vocabulary and rules. They have become mainstream approaches in academia and industry with good performance. With the increase in computing power, neural networks have returned to people's attention. In 2003, Hammerton J. [2] first applied Long Short-Term Memory (LSTM) [3] to named entity recognition tasks. Although the performance in entity recognition was not satisfactory, it provided a new research direction. In 2015, Huang Z. et al. [4] designed a neural network model using Bidirectional LSTM (BiLSTM) and Conditional Random Field (CRF) for entity recognition tasks. It achieved excellent results in various English NER tasks and became the state-of-the-art (SOTA) model at that time, as well as the baseline model for many English NER models.

Collobert R. et al. [5] proposed a unified neural network architecture and learning algorithm for handling various natural language processing tasks. They trained a word embedding using a language model, and then applied the word embedding to tasks such as part-of-speech tagging, chunking, named entity recognition, etc. By combining word embeddings, convolutional neural networks (CNN), and conditional random fields (CRF), they achieved better results than previous research. However, CNN are unable to handle long text sequences. To enable the model to process long text data, Lample G. et al. [6] replaced CNN with bidirectional long short-term memory networks (LSTM) as feature extractors to alleviate the problem of long-distance dependencies. However, these methods only consider word-level semantic features and ignore character-level implicit information. Kim Y. et al. [7] proposed a language model that utilizes subword information through character-level CNN and extracts contextual features using LSTM while normalizing them with the SoftMax function. Kuru O. et al. [8], on the other hand, input character sequences into BiLSTM to extract character-level contextual features. They output label probabilities for each character and use Viterbi decoder to convert these probabilities into word-level entity labels. Additionally, this approach also performs well in handling NER tasks in multiple languages.

In the English NER field, named entity boundaries align with word boundaries. However, Chinese does not have clear natural boundaries; therefore, Chinese NER usually involves segmenting input text data before feeding it into models for entity recognition. This segmentation operation can introduce errors that lead to inaccuracies in entity recognition.

Dong C. et al. [9] proposed a Chinese radical-level LSTM to capture the pictographic root features, combined with a character-based BiLSTM-CRF model, achieving better performance on Chinese NER tasks. Zhang Y. et al. [10] used a lattice structure to integrate lexical information and designed a named entity recognition model called Lattice-LSTM based on the LSTM model. In this model, memory units calculate the weighted sum of the character-level input and all potential words, integrating both lexical and character information. Compared to direct-word segmentation methods, it alleviates errors caused by word segmentation operations. In 2018, Google released the natural language processing model BERT [11], introducing the concept of pre-training models into the field of natural language processing. The BERT model is trained on large-scale unlabeled corpora and has strong generalization ability, setting new records for multiple NER tasks. Dai Z. et al. [12] applied the BERT pre-training model to Chinese NER tasks, combining it with the BiLSTM-CRF model for feature extraction and sequence labeling, achieving significantly better results than other contemporary Chinese named entity recognition models.

The construction of a knowledge graph for network security belongs to the problem of constructing a vertical domain knowledge graph [13], which is different from general

domain knowledge graphs such as DBpedia [14], Freebase [15], and Yago [16]. It is constructed from data in the field of network security. In addition to building a network security knowledge graph based on expert systems, researchers both domestically and internationally have also studied other methods for constructing network security knowledge graphs. Fang Y. et al. [17] proposed a network security entity recognition model called CyberEyes, which combines graph convolutional neural networks with BiLSTM models to extract both contextual and graphical-level, non-local dependency relationships simultaneously. The performance of this model on the network security corpus is higher than that of traditional CNN-BiLSTM-CRF models. Yi F. et al. [18] fully utilize the features of security data and the correlations between entities, proposing a named entity recognition model for network security based on regular matching, external dictionaries, CRF, and feature templates. By jointly constraining multiple conditions, more accurate entity recognition results are obtained. Sills M. et al. [19], in order to enhance AI-based network defense systems in capturing, detecting, and preventing known and future attacks, propose a system that generates various medical device vulnerability intelligence and known vulnerability threat intelligence resources through enhanced graph embedding techniques to generate higher quality graphical representations. Tikhomirov M. et al. [20] study BERT models and their variant models' performance on Russian language Network Security Named Entity Recognition tasks, and propose a method of enhancing Network Security Domain Data by adding names after descriptors or replacing descriptors with names.

Jia Y. et al. [21] designed a network security ontology that covers assets, vulnerabilities, and attacks. They used machine learning methods to construct a network security knowledge base based on the five-tuple model deduction rules. Shang Huaijun [22] constructed a vulnerability-based ontology in the field of network security and improved the effectiveness of network security entity recognition by using rule-based and dictionary feature-based methods for specific entities. This ultimately achieved the update and visualization of the network security knowledge base. Wang Tong et al. [23] conducted research on threat intelligence graph construction techniques, proposing a model that can automatically extract entities and relationships from threat intelligence data, and realized visual presentation of threat intelligence knowledge graphs. Peng Jiayi et al. [24] proposed a BiLSTM-CRF model using active learning methods to improve the accuracy of named entity recognition tasks in small-sample information security domains. Zhang Ruobin et al. [25] addressed the problem of identifying security vulnerability entities by proposing a BiLSTM-CRF model that uses dictionary correction to improve identification results while significantly reducing the cost of manually selecting features. Qin Ya and Shen Guowei et al. [26] proposed an improved CRF algorithm based on Hadoop, which effectively partitions datasets to enhance accuracy in recognizing security entities. Qin Ya also proposed a network security identification model that extracts local contextual features using artificial feature templates, character features using CNN, and global text features using BiLSTM; this model outperforms other methods on large-scale network security datasets.

Based on the above analysis, it is found that English-language network security knowledge graph construction techniques are relatively mature, but research on Chinese-language network security knowledge graphs is still in its early stages. Due to significant differences between Chinese and English languages, existing techniques for constructing English-language networks cannot be directly applied to Chinese-language network security knowledge graph construction. There is a lack of Chinese-language named entity recognition methods specifically designed for the Chinese environment in network security NER. To address this, this paper proposes a CSBERT-IDCNN-BiLSTM-CRF method for Chinese-language network security NER, with the following main research work.

(1) From a model perspective, using the CSBERT pre-trained model based on network security can project input representations into the network security semantic space, greatly improving the performance of neural network models on small datasets. By utilizing IDCNN and BiLSTM, the model is endowed with the ability to capture long

dependency features and local features, combining these two types of features to better accomplish entity recognition tasks in the field of network security.

(2) A reliable dataset is constructed. Currently, most knowledge graphs related to network security face issues such as insufficient data or outdated Chinese domain data that lags current developments in network security. To address this dataset problem, this paper used China National Vulnerability Database (CNNVD) as a source for constructing a network security dataset centered around vulnerability data. The text corpus for dataset construction consists of vulnerability reports and vulnerability data from CNNVD. Automated scripts are used to crawl vulnerability report data, download XML format vulnerability data directly, and utilize YEDDA [27] tool for data annotation to complete dataset construction.

(3) This paper has accomplished the training and testing of the model on its own constructed dataset, mitigating issues associated with poor model training due to low-quality datasets. Experimental results indicate that the proposed CSBERT-IDCNN-BiLSTM-CRF model boasts certain advantages in terms of the efficiency and accuracy of entity recognition.

3. Chinese Network Security Named Entity Recognition Method Based on CSBERT-IDCNN-BiLSTM-CRF

Based on the characteristics of the current Chinese network security entity structure, this paper designs a Chinese network security entity recognition model, which is divided into three parts: corpus processing, model training phase, and model testing phase. The overall experimental process of this paper is shown in Figure 1 below.

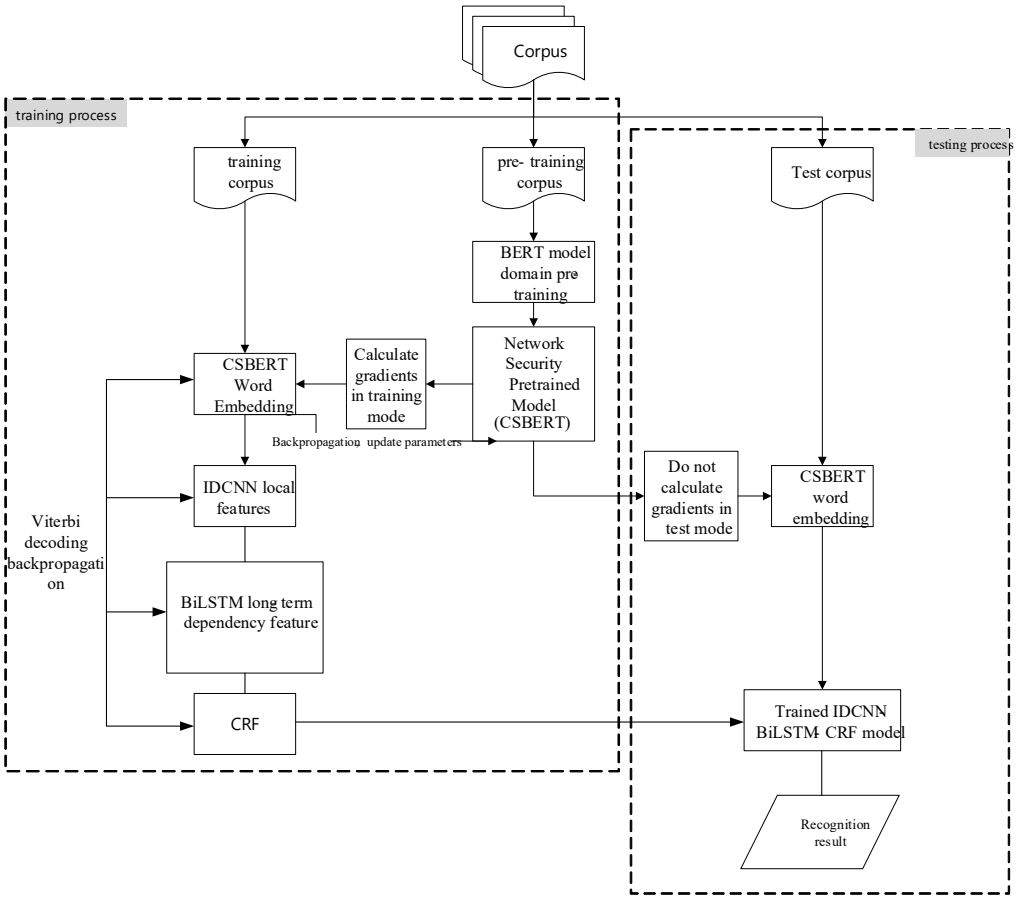

Figure 1. Overall Experimental Process.

This paper starts with the processing of the corpus, accessing China's National Information Security Vulnerability Database (CNNVD), and using vulnerability reports and

data from CNNVD as the core of constructing a text corpus for dataset building. After extracting and cleaning the data, the text is annotated to complete dataset construction. Then, the dataset is used to train and test the accuracy of the proposed model in this paper. After completing testing, importance of proposed modules is demonstrated through ablation experiments. Section 3 mainly introduces theoretical analysis and composition of the model, while Section 4 focuses on experimental configuration and specific experimental process.

3.1. Theoretical Analysis of the CSBERT-IDCNN-BiLSTM-CRF Model

3.1.1. Continue Training CSBERT in the Field of Network Security

Deep learning requires a large amount of annotated data to train models, but it is difficult to obtain such data in professional domains. On the other hand, obtaining unannotated data is relatively easy. How to use unsupervised learning to improve the performance of small sample supervised learning is a research question worth exploring [28]. The BERT model has powerful fitting ability and can enhance the performance of small sample named entity recognition tasks. Additionally, the BERT model is trained using a large amount of unlabeled text, making it highly versatile. By training a professional domain-specific BERT model with unannotated data from that domain, the performance of small-sample named entity recognition models in that domain can be improved.

Compared with fine-tuning on an open-source BERT model using professional domain data and training a professional domain-specific BERT model from scratch with full-scale data, there is little difference in their performances. This is because the professional domain mainly adds numerous proprietary terms compared to the public domain. The publicly available BERT models are trained on open-domain data and have learned general knowledge about language. With this foundation, when trained on professional corpora, pre-trained models can learn specific terms in that field. Training a BERT model essentially involves projecting characters into target semantic space through spatial transformation and representing relationships between characters based on distances between spatial coordinates. Publicly available BERT models have already achieved most character projections in semantic space and cover most content in general semantic space. Continuing training for specific domains aims at projecting general semantic space onto specialized semantic space and reducing distances between specialized vocabulary items within that field. In practical applications, this means that specialized vocabulary items are more likely to be represented as related entities rather than individual characters alone. By leveraging this pre-learned "knowledge" about specific domains, better performance can be achieved in small sample named entity recognition tasks within those domains. Moreover, since fine-tuning occurs on existing models without requiring full-scale data training, it reduces the resource demands of model training and lowers the threshold for individuals to apply BERT models in professional domains.

In this paper, we address the development of a knowledge graph for the cybersecurity domain within a Chinese-language environment. To assess model performance, we input Chinese cybersecurity-specific terminology and general locational nouns into both the BERT and CSBERT models to generate word vectors. We performed a clustering analysis on the output vectors from these models. For visual distinction in the Chinese domain, entities related to cybersecurity are color-coded in purple, while all other entities are marked in yellow. The clustering outcomes for both models are as Figures 2 and 3.

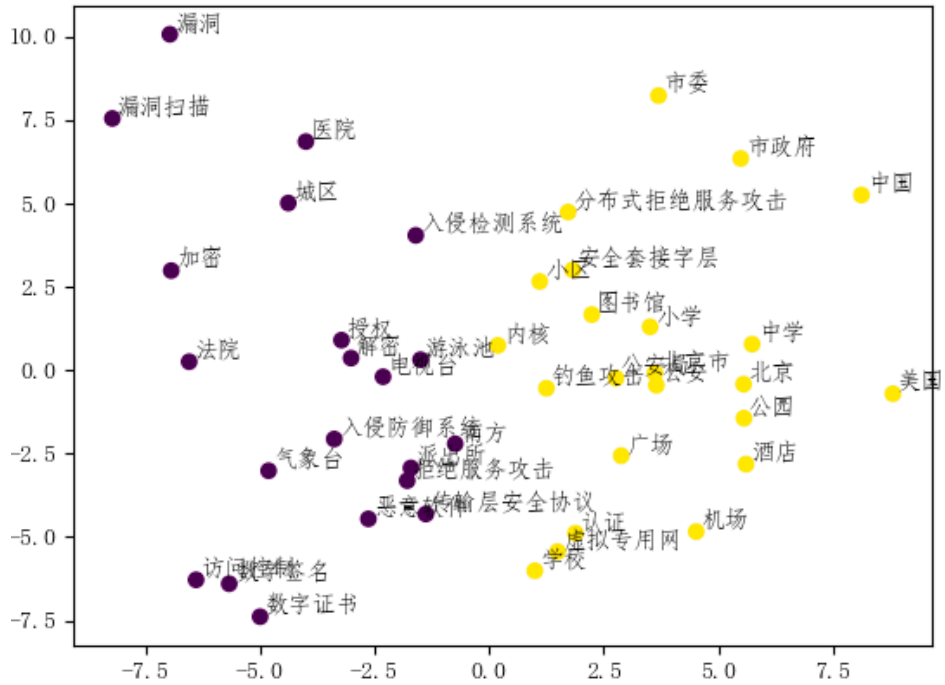

Figure 2. BERT model clustering results in the Chinese environment.

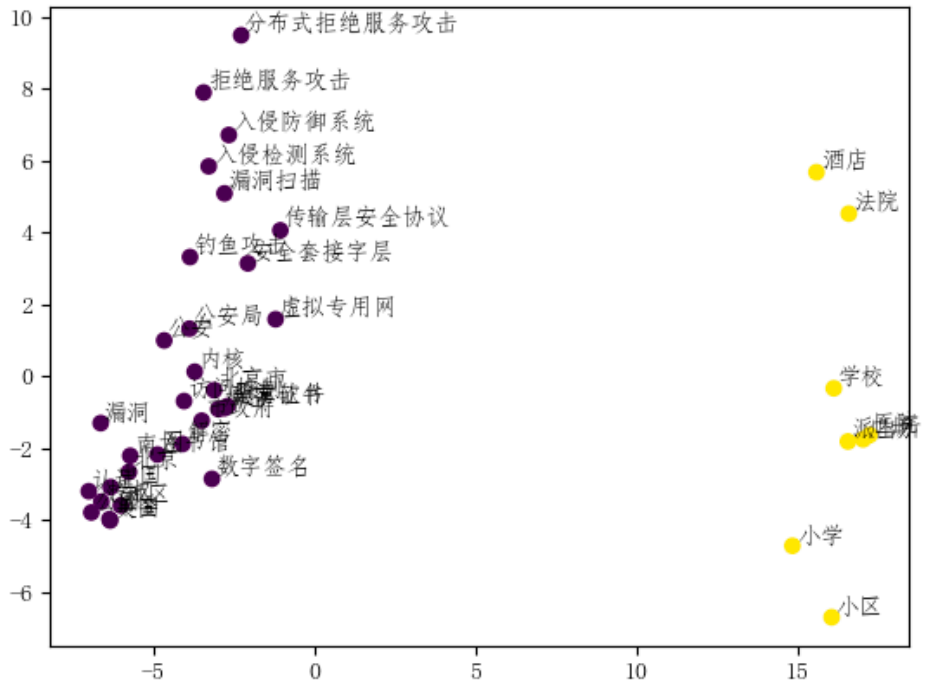

Figure 3. CSBERT model clustering results in the Chinese environment.

From the above two clustering result graphs, the BERT model is trained using publicly available corpora and has learned some prior knowledge. However, the word vectors outputted by the model do not show obvious clusters, indicating a poor ability to recognize network security terms. On the other hand, the CSBERT model is further trained using network security corpora, resulting in distinct distances between word vectors. Network security terms and location names are clustered separately with a significant distance between these two clusters. The CSBERT model can identify more network security terms compared to BERT. Additionally, the distances between network security vocabulary are closer in CSBERT's output word vectors, which contain more network security information

and thus improve the accuracy of named entity recognition for network security entities in this model.

3.1.2. Introduction of BiLSTM-CRF Module Based on CSBERT Model

Due to the temporal nature of natural language, it is crucial to utilize neural network models that can extract sequential features when modeling temporal information [29]. The chain-like structure of Recurrent Neural Network (RNN) naturally fits the sequence modeling problem in natural language processing. Therefore, RNN has achieved good performance in handling natural language tasks. However, with the increase in computing power and input text length, the original RNN model's ability to handle long texts becomes limited. On the other hand, RNN tends to suffer from gradient vanishing problems, leading to poor performance in extracting long-distance dependency features.

To address the issue of long-distance dependencies, Long Short-Term Memory (LSTM) was introduced. LSTM is a special type of RNN that partially solves the problem by incorporating gate mechanisms on top of the basic RNN structure. This greatly improves the upper limit on input text length. However, since LSTM models only capture information from one direction while extracting sequential features, they cannot fully capture contextual information present in each character of natural language text. In different contexts or linguistic environments, a word may have different meanings. To better extract contextual information, this paper adds a backward LSTM on top of forward LSTM and using Bidirectional LSTM (BiLSTM) model instead. The structure of BiLSTM model is as Figure 4:

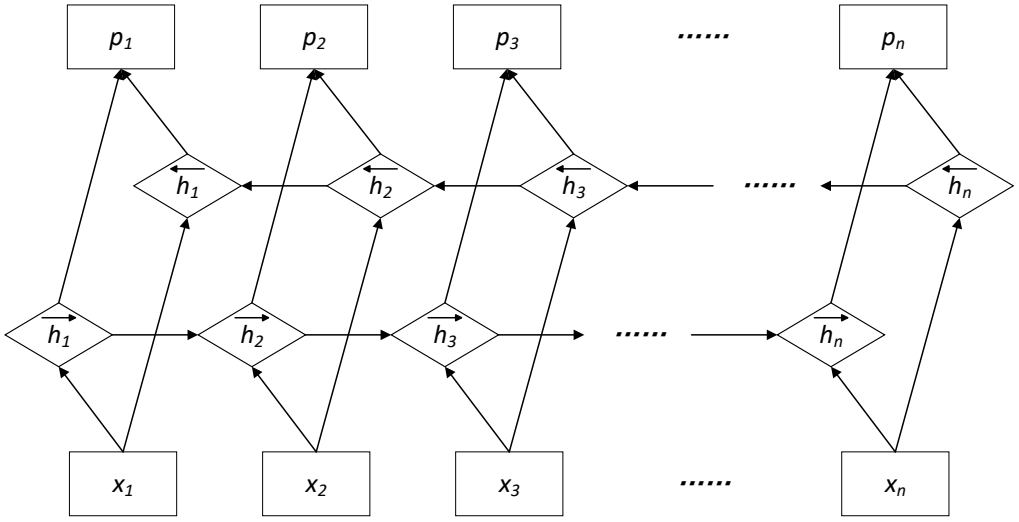

Figure 4. BiLSTM Structure Diagram.

When using a single BiLSTM, the model usually takes the label with the highest probability at each time step as its output result. The output result only depends on the output of that time step and ignores the dependency relationship between the entire label sequence. This labeling method does not consider label regularization issues, which may lead to many illegal label sequences and reduce the performance of the model. To solve this problem, a CRF module is introduced under the premise of BiLSTM. CRF considers dependencies between labels and adds some restrictions to ensure label legality in the model's output. The objective function of CRF consists of two parts and is also constrained by surrounding annotation results. It abstracts sequence labeling problems into dynamic programming problems, seeking for an optimal sequence that maximizes the objective function and reduces the probability of incorrect labels appearing.

The working process of the CRF module is as shown in the formula.

Given a sequence $X = \{c_1, c_2, \ldots\ldots, c_n\}$ composed of n characters and a label sequence $Y = \{y_1, y_2, \ldots\ldots, y_n\}$, calculate the score as shown in Formula (1).

$$score(X, Y) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=0}^{n} P_{i, y_i} \tag{1}$$

where P is the probability matrix output by the upper-level feature extraction model. P_{i,y_i} represents the probability of the i-th character having a label y_i. A is a probability transition matrix generated by a CRF model during training. $A_{y_i,y_{i+1}}$ represents the probability of transitioning from label y_i to label y_{i+1}. Perform SoftMax operation on all label sequences to obtain normalized probabilities, as shown in Formula (2).

$$P(Y|X) = \frac{exp(score(X, Y))}{\sum_{Y'} exp(score(X, Y'))} \tag{2}$$

Maximize its logarithmic likelihood function, as shown in the Formula (3).

$$logP(Y^X|X) = score(X, Y^X) - log(\sum exp(score(X, Y'))) \tag{3}$$

Decode using the Viterbi algorithm and select the label sequence with the highest probability as the output of CRF, as shown in Formula (4).

$$Y^* = arg \max_{Y'} score(X, Y') \tag{4}$$

Using the BiLSTM-CRF module can better handle word sequence labeling tasks output by the CSBERT model and complete NER tasks.

3.1.3. Use DCNN to Stack and FORM IDCNN Module

Although the use of the BiLSTM-CRF model has achieved excellent results in multiple NER tasks and has become one of the baselines for NER tasks, there are some unstructured features in Chinese cybersecurity text data that differ from public domain texts. Specifically, these features include the following two types:

(1) Complexity

Text data in the field of network security is often mixed with Chinese, English, numbers, and special symbols.

(2) Repetition

When describing software and programs, there is often a lot of repetitive content, which increases the difficulty of entity recognition [30].

Due to the inclusion of these two special features, BiLSTM cannot fully extract sequential information from Chinese network security texts. Therefore, this paper proposes a model that incorporates BiLSTM-CRF and introduces CNN. However, when using the original CNN to handle sequence labeling problems, there are two main issues. Firstly, traditional CNN adds pooling operations after convolution operations to reduce data dimensions and minimize model performance requirements. However, adding pooling operations when processing text sequences can lead to loss of sequence information. Secondly, natural language is mostly composed of sentences. Deep learning requires the ability to process long textual data. The receptive field of CNN is small and can only increase by stacking convolutional layers, which easily leads to overfitting problems [31].

To avoid the above-mentioned problems, a different type of convolutional neural network (DCNN) was chosen instead of traditional CNN [32]. DCNN introduces the dilation mechanism in the convolutional kernel, replacing pooling operations to reduce information loss. With the same size of convolutional kernel, it increases receptive field and enhances the model's ability to process long text data.

Calculate the receptive field of dilated convolution, as shown in Formula (5).

$$F_{i+1} = \left(2^{i+w} - 1\right) \times \left(2^{i+w} - 1\right) \tag{5}$$

where them, F represents the size of the receptive field, i is the stride, and w is the dilation width.

Iterated Dilated Convolutional Neural Network (IDCNN) [33] stacks dilated convolution modules on top of DCNN to obtain greater text processing capability. The structure of the dilated convolution module is shown in Figure 5.

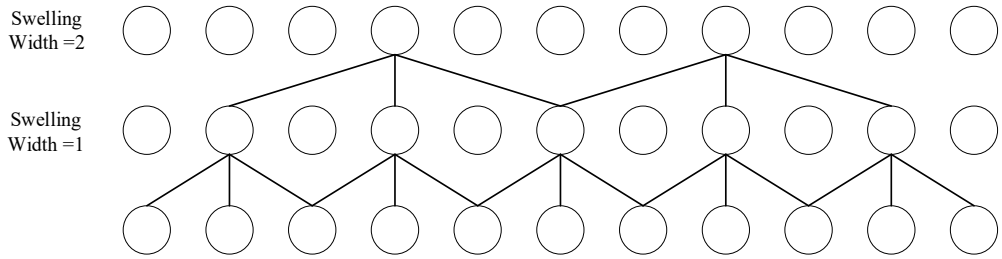

Figure 5. DCNN module structure diagram.

The IDCNN module in this paper is composed of four identical DCNN modules stacked together. Each DCNN module contains three layers of convolution with dilation widths of 1, 1, and 2 respectively. The parameters are shared where the four DCNN modules. The receptive field of the IDCNN model will exponentially increase as the DCNN modules are stacked, and the growth rate of the receptive field is much higher than that of the number of layers to reduce the probability of overfitting issues occurring.

The IDCNN-BiLSTM-CRF feature extraction module combines the local features and sequence features of the text, making more full use of the text information, integrating multiple granularities of text features, and improving the performance of network security entity recognition. Given a sequence $X = \{x_1, x_2, \ldots\ldots, x_n\}$ consisting of n characters and the final encoded sequence $H = \{h_1, h_2, \ldots\ldots, h_n\}$, with dilation width $W_j = \{1, 2, 4\}$ for dilated convolutions where $j = 1, 2, 3$, perform dilated convolution operations on sequence $X = \{x_1, x_2, \ldots\ldots, x_n\}$. Let ${C^j}_k$ represent the j-th layer with dilation width k for dilated convolution layers. After performing dilated convolution operation using the first layer ${C^1}_1$ as an example, the output sequence X_j is obtained.

$$X_j = {C^1}_1 X \tag{6}$$

And X represents the input sequence. Assuming that the dilation width of each subsequent layer's dilated convolution is represented by k_j then the output O_j of each layer is:

$$O_j = relu\left({C^j}_{k_j} O_{j-1}\right) \tag{7}$$

If the three-layer dilated convolution operation of a DCNN module is regarded as a whole D, then the output sequence of the initial sequence X after k rounds of iteration is $D_k = \left\{{d_1}^k, {d_2}^k, {d_3}^k, \ldots\ldots, {d_n}^k\right\}$. Currently, D_k has already integrated the local features of the text. D_k is passed to BiLSTM to extract contextual sequence features. BiLSTM extracts preceding and succeeding information through forward and backward LSTMs and concatenates both directions' sequences together to obtain the final encoding sequence $H = \{h_1, h_2, \ldots\ldots, h_n\}$ as the feature extraction module. The encoding sequence at this point

integrates both local text and contextual sequence features, possessing better text representation capability. Finally, inputting encoding sequence $H = \{h_1, h_2, \ldots\ldots, h_n\}$ into CRF module for sequential labeling yields the model's ultimate output: predicted label sequences. By using IDCNN-BiLSTM-CRF joint model as a feature extractor for structure extraction simultaneously captures both local and overall sequential information in texts which further enhances Chinese network security entity recognition effectiveness.

3.2. CSBERT-IDCNN-BiLSTM-CRF Model Structure and Algorithm Flow

After analyzing and filtering each module in Section 3.1, this paper proposes a joint entity recognition model based on the CSBERT network security pre-training model, called CSBERT-IDCNN-BiLSTM-CRF. The structure diagram is as Figure 6.

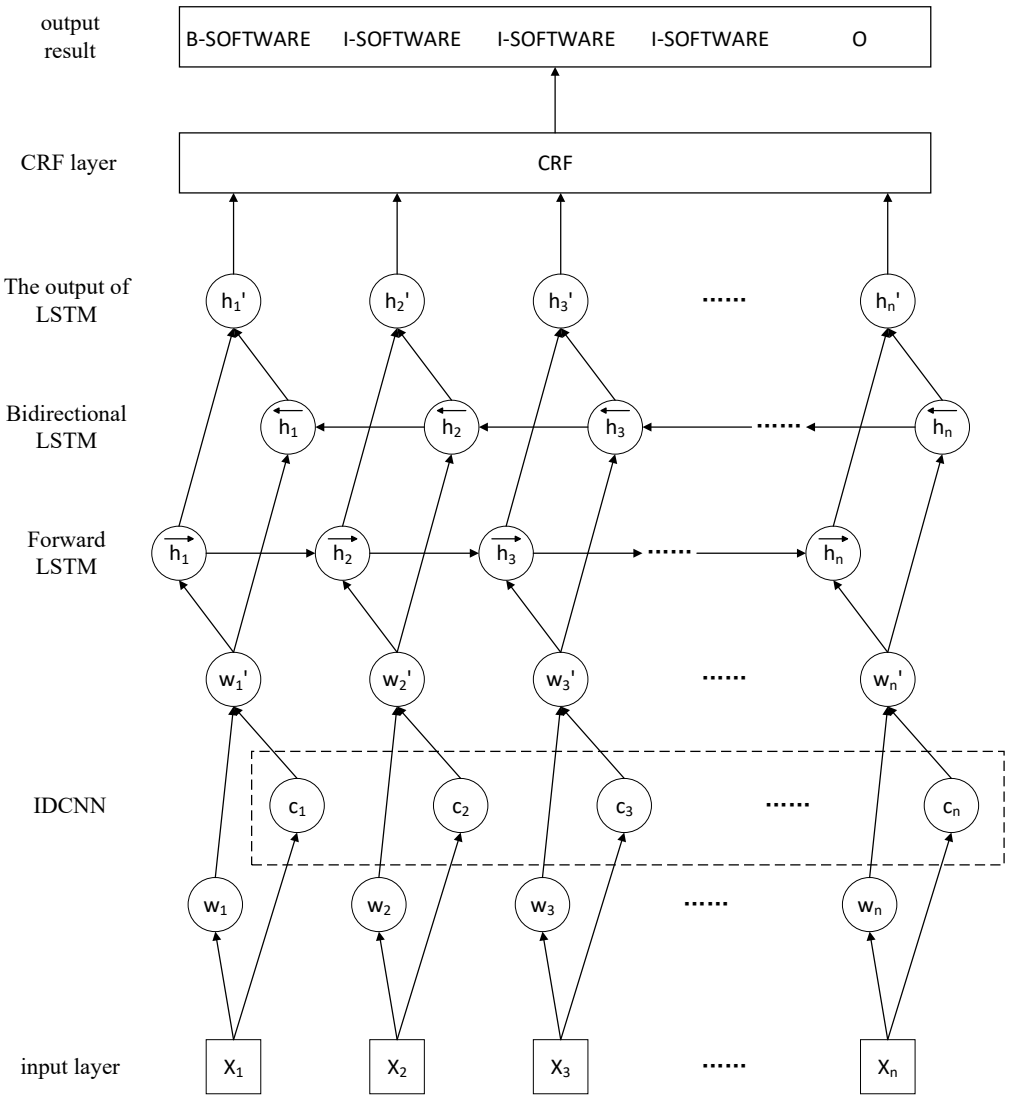

Figure 6. Model flowchart of this paper.

The algorithm flow of the model during its working process is as Algorithm 1:

Algorithm 1 CSBERT-IDCNN-BiLSTM-CRF Algorithm Formula

The input to the model is a text corpus sequence $X = \{c_1, c_2 \ldots, c_n\}$, and the output is a predicted label sequence $\hat{y} = \{\hat{y}_1, \hat{y}_2, \ldots\ldots, \hat{y}_n\}$.

Step 1: Extract a batch size of text data from the dataset for processing. Tokenize the text sequence and perform character-level segmentation on it. Vectorize the text sequence at the character level using a dictionary to generate unique encoding sequences.

Step 2: Input the unique encoding sequence into the BERT model. The model combines context and semantic information to dynamically generate word vectors x_B, which are then passed to downstream task models.

Step 3: IDCNN performs local information modeling for the target task by extracting local features from the text sequence. It outputs a local feature sequence $W = \{w_1, w_2 \ldots, w_n\}$.

Step 4: Bi-LSTM is used to perform context and sequential information modeling for the target task. It extracts contextual features and sequential order features contained in the local feature sequence $W = \{w_1, w_2 \ldots\ldots, w_n\}$, which are used for sequence labeling.

Step 5: Use CRF (Conditional Random Field) to select the predicted label sequence with maximum probability $\hat{y} = \{\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n\}$ as an output of the model.

4. Experimental Design and Analysis

4.1. Data Annotation

In deep learning, named entity recognition problems are often treated as sequence labeling problems. Therefore, the datasets used for NER are usually saved in a form where each character corresponds to an entity label. Common data annotation methods include BIO, BIOES, BMESO, etc. The dataset used by the model in this paper is annotated using the BIO annotation method. Taking the Chinese sentence "API存在跨站脚本漏洞"(API has a cross-site scripting vulnerability) as an example, the annotation result obtained using the BIO annotation method is shown in Figure 7.

	A	P	I	存	在	跨	站	脚	本	漏	洞
BIO	B-NET	I-NET	I-NET	0	0	B-VUL	I-VUL	I-VUL	I-VUL	I-VUL	I-VUL

Figure 7. Entity annotation results.

In the above BIO annotation method, "B" represents the starting character of an entity, "I" in BIO annotation indicates characters other than the starting character within an entity, i.e., middle and ending characters of the entity, and "0" indicates that the word is unrelated to the current named entity under consideration. In the final dataset, we used the BIO annotation method to label a total of 710,000 rows of data for model training.

4.2. Dataset Construction

The data source of the network security dataset used in this paper is the China National Vulnerability Database (CNNVD) at https://www.cnnvd.org.cn (accessed on 9 September 2023). The website provides information related to hot vulnerabilities, vulnerability reports, and other information related to network security vulnerabilities. It has a high-quality network security corpus, which is suitable for constructing the network security dataset in this paper. By extracting and cleaning data from CNNVD's raw corpus, the YEDDA tool is used for data annotation. Based on the network security text corpus, eleven entity types are designed for the network security dataset: person names (PERSON), organization names (ORGANIZATION), company names (COMPANY), location names (LOCATION), software names (SOFTWARE), program names (PROGRAM), hardware names (HARDWARE), vulnerability names (VULNERABILITY), actions (ACTION), network entities (NETWORKENTITY), and version numbers (VERSION). Data annotation using the YEDDA tool is shown in Figure 8.

Figure 8. Use YEDDA for data annotation.

Person (PERSON) includes software developers, company owners, legal representatives, etc.; organization (ORGANIZATION) includes hacker groups, non-profit organizations, open-source software communities, software foundations, etc.; company (COMPANY) specifically refers to for-profit entities such as Apple Inc., Google LLC, Huawei Technologies Co., Ltd., etc.; location (LOCATION) includes countries, regions, specific addresses, etc.; software name (SOFTWARE) includes but is not limited to specific application platforms, operating systems, software plugins; program name (PROGRAM) includes APIs, components within the software system, processes/modules in the software system, variables in the software system, scripts and codes; hardware name (HARDWARE) includes various hardware components such as CPU, GPU, memory, disk drives, routers, and switches; vulnerability name (VULNERABILITY) includes various vulnerabilities such as cross-site scripting vulnerability (SQL injection vulnerability); action (ACTION) includes actual impact of vulnerabilities and specific operations to achieve attack objectives; network entity (NETWORKENTITY) includes various difficult-to-determine network elements that have a greater association with the Internet than other types of entities, such as protocols and protocol implementations, browsers, clients, front-end, C language (programming language), files, and other abstract concepts; version number (VERSION) includes specific version codes for various software and hardware.

4.3. Dataset Entity Statistics

The entity statistics of the network security dataset annotated using the YEDDA annotation tool are shown in Table 1.

Table 1. Dataset entity statistics results.

Entity Category	Entity Quantity
NETWORKENTITY	22,868
SOFTWARE	11,915
VERSION	4987
ACTION	3589
VULNERABILITY	2223
LOCATION	2171
COMPANY	2093
PROGRAM	1914
HARDWARE	1680
ORGANIZATION	688
PERSON	159
Total	54,287

To visually display the distribution of samples, calculate the proportion of each entity in the dataset. The statistical results are shown in the Figure 9.

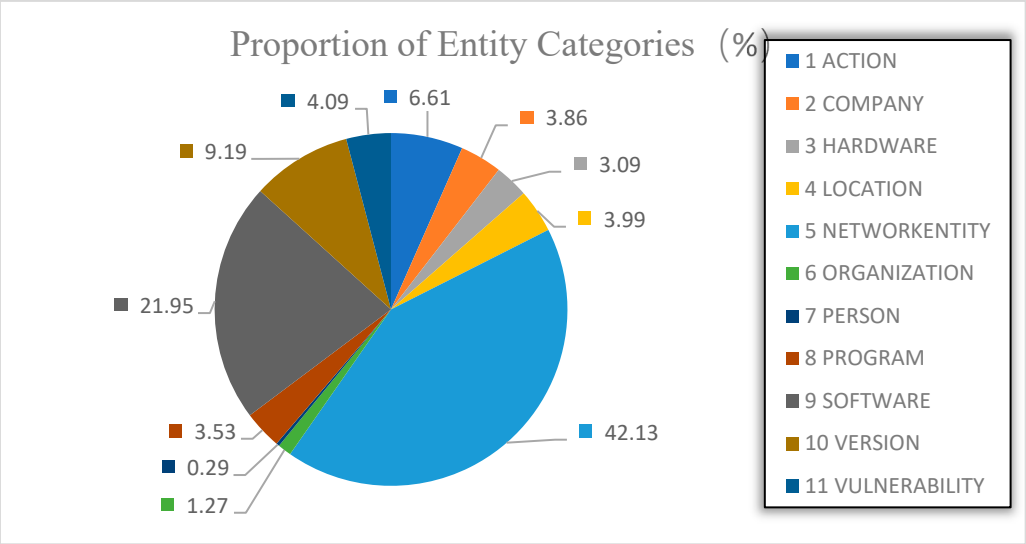

Figure 9. Proportion of each entity category.

4.4. Experimental Hyperparameter Settings

When training the model, the parameter settings in Table 2 were used. Some parameters in the table will be adjusted according to the actual situation of model training to achieve optimal experimental results. The model fine-tunes CSBERT during the training process and separates the learning rates of CSBERT and other parameters to obtain better experimental performance. Additionally, to avoid overfitting during training, a dropout mechanism is introduced to randomly discard some neurons.

Table 2. Experimental hyperparameter settings.

Parameter Name	Numerical Value
Pretrained Model Dropout	0.1
Dropout of LSTM layer	0.5
Dropout of IDCNN layer	0.5
Weight decay	0.01
Word vector	768
LSTM hidden layer size	256
Number of tags	23
Epoch	100
Batch size	32
IDCNN input maximum length	120
BERT learning rate	0.00002
IDCNN learning rate	5 × Bert learning rate
LSTM learning rate	5 × Bert learning rate
Fully connected layer learning rate	5 × Bert learning rate
CRF layer learning rate	100 × Bert learning rate
Warm up	0.1
Optimizer	AdamW

4.5. Entity Recognition Result Analysis

To verify the performance of the CSBERT-IDCNN-BiLSTM-CRF network security named entity recognition model proposed in this paper, different models were selected for experiments on the same dataset. The experimental results are shown in Table 3. The network security entity recognition model proposed by Jia Y. et al. [21] is used as a benchmark model. The remaining models include mainstream sequence labeling models such as the BiLSTM-CRF, CNN-BiLSTM-CRF model with convolutional neural networks introduced, and IDCNN-BiLSTM-CRF proposed in this paper for sequence labeling tasks. Furthermore, by introducing the BERT model and the cybersecurity pre-trained CSBERT model,

a more comprehensive and precise comparative analysis was conducted, thereby accentuating the superior capability of this model in the recognition of Chinese cybersecurity entities. Model performance is evaluated and analyzed based on precision, F1 score, and recall using a classical evaluation system.

Table 3. Experimental results of each model.

Model	P (%)	F1 (%)	R (%)
Jia Y [21]	71.34	73.32	75.37
Peng Jiayi [24]	70.23	75.12	80.13
Zhang Ruobin [25]	70.62	75.71	81.37
BiLSTM-CRF	69.72	74.29	80.11
CNN-BiLSTM-CRF	71.39	75.42	79.31
Qin Ya [26]	71.81	76.02	79.59
IDCNN-BiLSTM-CRF	72.37	76.42	80.33
BERT+IDCNN-BiLSTM-CRF	83.89	85.65	87.7
CSBERT+IDCNN-BiLSTM-CRF	85.89	87.3	88.88

Through the line chart, the changes of various evaluation indicators between different models can be displayed more intuitively. The statistical graph of model performance is shown in Figure 10:

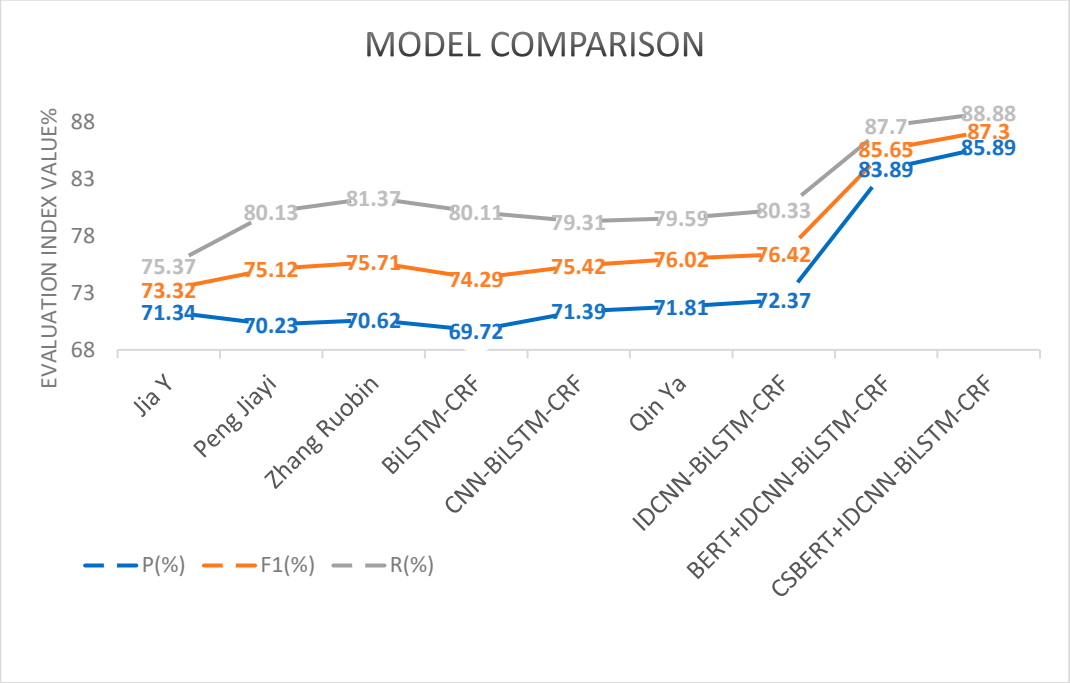

Figure 10. Comparison of multi-model effects.

The experiment was conducted on the nine models mentioned above. As shown in the figure, in the task of network security named entity recognition, the mainstream BiLSTM-CRF model performs better than Jia Y's proposed network security entity recognition model. Peng Jiayi's strategy of improving small-sample information security in the field by introducing active learning did indeed yield results slightly higher than those of the classic BiLSTM-CRF model. Zhang Ruobin used dictionary correction to improve representation outcomes, adding a correction step on top of the classic BiLSTM-CRF model and significantly increasing the model's accuracy. However, due to differences in Chinese and English structures, using BiLSTM-CRF cannot effectively capture Chinese sequence information and local features. From the results of these two models, without addressing the

effective capture of Chinese sequence information and local features, improvements based solely on the classic BiLSTM-CRF model are limited for Chinese entity recognition. After introducing CNN, the CNN-BiLSTM-CRF model has obtained the ability to capture local features and to some extent process Chinese specific structures to obtain sequence information. This can also be seen in Qin Ya's model, where Qin Ya considered using a CNN-based model solution for entity recognition tasks in constructing a network security knowledge graph, and further improved the accuracy of the CNN+BiLSTM+CRF model by combining feature templates on this basis. But compared to IDCNN, traditional CNN has a smaller receptive field and cannot handle longer text sequences. The IDCNN-BiLSTM-CRF model can extract local features from long texts and then fuse them with long-term dependency features extracted by BiLSTM, enabling it to capture more semantic information.

After introducing BERT pre-training models, the performance of the BERT-IDCNN-BiLSTM-CRF model significantly improves compared to models without BERT. This indicates that pre-training models play an important role in improving entity recognition accuracy. When continuing pre-training BERT with network security corpora, a CSBERT pre-trained model more suitable for network security domain is obtained. The F1 scores of the CSBERT-IDCNN-BiLSTM-CRF model reach 87.31%. Compared with BERT-IDCNN-BiLSTM-CRF using general-purpose BERT pre-training models, CSBERT pre-trained models gain abundant prior knowledge in the field of network security through continued training specifically in this domain and ultimately acquire word vectors more suitable for this domain which enhances their performance. This demonstrates the importance of continued training within specific domains.

From the above experimental comparisons, it is necessary to continue training open-source pre-training models with network security corpora for network security named entity recognition tasks. Combining IDCNN as an IDCNN-BiLSTM feature extractor can simultaneously capture both local and long-term dependency features of the text. Compared to traditional BiLSTM, it can capture more semantic features and improve entity recognition accuracy.

4.6. Analysis of Entity Recognition Results by Category

The experimental results of Section 4.5 are the weighted average of evaluation indicators for each entity category, with the evaluation indicators for each entity shown in Table 4.

Table 4. Evaluation index value of each entity.

Entity Number	Entity Type	Precision (%)	Recall (%)	F1(%)
1	ACTION	75.86	78.22	77.02
2	COMPANY	95.76	98.03	96.88
3	HARDWARE	77.33	80.11	78.69
4	LOCATION	98.91	98.55	98.73
5	NETWORKENTITY	63.35	76.57	69.33
6	ORGANIZATION	85.11	89.88	87.43
7	PERSON	100	100	100
8	PROGRAM	75.81	79.21	77.47
9	SOFTWARE	86.06	92.06	88.96
10	VERSION	89.98	89.11	89.55
11	VULNERABILITY	96.65	95.94	96.29

To analyze the experimental results of the CSBERT-IDCNN-BiLSTM-CRF model in detail, the experimental indicators of each entity category are merged with the corresponding quantity in the dataset and analyzed in Figure 11.

Overall, the number of entities in the dataset does not directly correlate with their recognition accuracy. Where "COMPANY", "LOCATION", "ORGANIZATION", "PERSON "COMPANY", "LOCATION", "ORGANIZATION", "PERSON", "SOFTWARE", "VERSION" and "VULNERABILITY" have higher recognition accuracy, with F1 scores ex-

ceeding 85%. However, the recognition accuracy for entities such as "ACTION," "HARDWARE," "NETWORKENTITY," and "PROGRAM" is relatively lower.

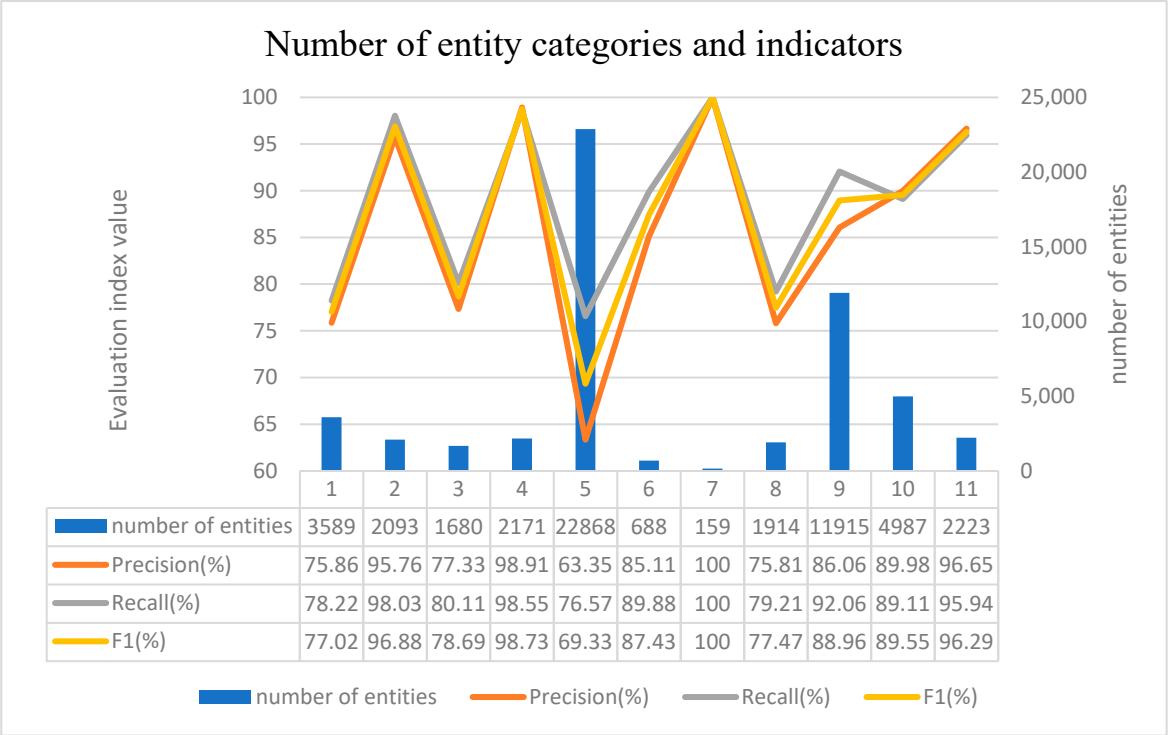

Figure 11. The number of each entity and its evaluation indicators.

The low accuracy in identifying the entities "ACTION" and "PROGRAM" is due to their long lengths, as well as the overlap of some entity content with other entity categories. The model easily confuses the end tags of entities, resulting in incorrect label predictions.

On the other hand, the entity "HARDWARE" has similar features to "SOFTWARE," and both can only be distinguished based on contextual information. However, there are far more instances of "SOFTWARE" entities in the dataset compared to "HARDWARE," which leads to higher accuracy in recognizing "SOFTWARE" entities than "HARDWARE".

As for the entity "NETWORKENTITY," it contains too many concepts and is prone to being predicted as an 'O' (non-entity) by the model compared to other entities. Therefore, its recognition accuracy does not directly correlate with its quantity in the dataset.

Ultimately, we found that the number of entities has a significant impact on prediction accuracy when faced with relatively similar entity types, whereas it is not a major factor in other cases such as entity types with complex concepts.

4.7. Melting Experiment

To verify the effectiveness of the main modules proposed in this paper, a set of comparative experiments is conducted in this section. Based on the IDCNN-BiLSTM-CRF model, two groups are designed by replacing IDCNN with CNN and removing the IDCNN module as controls to validate the role of using the IDCNN module. The results of ablation experiments are shown in Table 5 below.

Table 5. Ablation experiment evaluation index value.

Model	P (%)	F1 (%)	R (%)
IDCNN-BiLSTM-CRF	72.37	76.42	80.33
CNN replaces IDCNN	71.39	75.42	79.31
Remove IDCNN	69.72	74.29	80.11

The performance of the three models in the table decreases sequentially on the network security dataset. This indicates that both CNN and IDCNN can extract local information from text, but IDCNN has a larger receptive field, allowing it to handle long text information and further improve the performance of network security entity recognition models.

5. Conclusions

The Chinese knowledge graph of network security can effectively address the issue of fragmentation and difficulty in integrating information in the field of Chinese network security. Named entity recognition technology plays a pivotal role in the construction of the graph, and significantly enhances the accurate identification and associative analysis of cybersecurity-related entities. Accordingly, this paper introduces a NER method in the Chinese domain called CSBERT-IDCNN-BiLSTM-CRF, based on a cybersecurity pre-trained model. This method initially leverages the cybersecurity pre-trained model for further training, thereby generating more sophisticated word vectors pertinent to the cybersecurity field. Subsequently, considering the temporal nature of natural language, it is crucial to extract sequential features from the model; this is where the chained structure of RNN comes into play for natural language sequence modeling. However, as computational power increases and the length of input text grows, the traditional RNN model's capability to process long texts becomes increasingly inadequate. Additionally, the original RNN models are prone to gradient vanishing, which leads to subpar performance in capturing long-distance dependent features; therefore, BiLSTM are used to resolve the issues with long-distance dependencies and to enhance the capacity to extract contextual information, thus extending the upper limit of natural language processing input texts. Following that, traditional CNN are employed as an auxiliary process to address issues where BiLSTM cannot fully extract the sequential information due to the unique textual structure of Chinese online texts. Nevertheless, the inherent limitation of CNN, after convolution operations followed by pooling, results in the loss of sequence information. Given that natural language consists mostly of sentences and the relatively small receptive field of CNN, which can only be increased by stacking convolution layers, risking overfitting, the paper introduces the DCNN with the atrous mechanism and establishes the IDCNN for processing longer text sequences via stacking techniques. Finally, a CRF is incorporated to consider the interdependency of labels at the sequence level, imposing constraints on the model's output to ensure the legitimacy of the labels. By integrating local and contextual features, the model achieves superior performance. Using the data from the China National Vulnerability Database of Information Security and the YEDDA tool for label annotation, this paper constructs a Chinese cybersecurity dataset for training and testing the proposed model. Experimental results demonstrate that on the Chinese cybersecurity dataset, the CSBERT-IDCNN-BiLSTM-CRF model outperforms other Chinese entity recognition models, leading in terms of accuracy and F1 Score, which sufficiently proves the efficiency of the proposed CSBERT-IDCNN-BiLSTM-CRF model and provides more precise recognition results in the task of named entity recognition in the Chinese cybersecurity domain.

Author Contributions: Conceptualization, Y.X. and X.T. (Xiaobo Tan); Methodology, Y.X.; Software, X.T. (Xin Tong); Validation, Y.X., X.T. (Xiaobo Tan) and W.Z.; Formal analysis, Y.X.; Investigation, W.Z.; Data curation, Y.X.; Writing—original draft, Y.X.; Writing—review & editing, X.T. (Xiaobo Tan) and W.Z.; Visualization, W.Z.; Supervision, Y.X.; Project administration, X.T. (Xin Tong); Funding acquisition, X.T. (Xin Tong). All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Liaoning Provincial Department of Educational Project, China (Grant No. LJKZ0241), Liaoning Provincial Science and Technology Collaborative Innovation Project, China (Grant No. LNKX2023A07) and Liaoning Provincial Department of Science and Technology Project, China (Grant No. 2023JH1/10400093).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Acknowledgments: We thank those anonymous reviewers whose comments/suggestions helped improve and clarify this manuscript.

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

References

1. National Internet Emergency Response Center. Overview of China's Internet Network Security Situation in 2020. [EB/OL]. 2021-5. 2021. Available online: https://www.cert.org.cn (accessed on 1 September 2022).
2. Hammerton, J. Named entity recognition with long short-term memory. In Proceedings of the Seventh Conference on Natural language learning at HLT-NAACL 2003, Edmonton, AB, Canada, 31 May–1 June 2003.
3. Gers, F.A.; Schmidhuber, J.; Cummins, F. Learning to forget: Continual prediction with LSTM. *Neural Comput.* **2000**, *12*, 2451–2471. [CrossRef]
4. Huang, Z.; Xu, W.; Yu, K. Bidirectional LSTM-CRF models for sequence tagging. *arXiv* **2015**, arXiv:1508.01991.
5. Collobert, R.; Weston, J.; Bottou, L.; Karlen, M.; Kavukcuoglu, K.; Kuksa, P. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.* **2011**, *12*, 2493–2537.
6. Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; Dyer, C. Neural architectures for named entity recognition. *arXiv* **2016**, arXiv:1603.01360.
7. Kim, Y.; Jernite, Y.; Sontag, D.; Rush, A. Character-aware neural language models. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016.
8. Kuru, O.; Can, O.A.; Yuret, D. Charner: Character-level named entity recognition. In Proceedings of the COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, Osaka, Japan, 11–16 December 2016.
9. Dong, C.; Zhang, J.; Zong, C.; Hattori, M.; Di, H. Character-based LSTM-CRF with radical-level features for Chinese named entity recognition. In *Natural Language Understanding and Intelligent Applications: 5th CCF Conference on Natural Language Processing and Chinese Computing, NLPCC 2016, and 24th International Conference on Computer Processing of Oriental Languages, ICCPOL 2016, Kunming, China, 2–6 December 2016*; Proceedings 24; Springer: Berlin/Heidelberg, Germany, 2016.
10. Zhang, Y.; Yang, J. Chinese NER using lattice LSTM. *arXiv* **2018**, arXiv:1805.02023.
11. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
12. Dai, Z.; Wang, X.; Ni, P.; Li, Y.; Li, G.; Bai, X. Named entity recognition using BERT BiLSTM CRF for Chinese electronic health records. In Proceedings of the 2019 12th International Congress on Image and Signal Processing, Biomedical Engineering and Informatics (cisp-bmei), Suzhou, China, 19–21 October 2019.
13. Liu, K.; Wang, F.; Ding, Z.; Liang, S.; Yu, Z.; Zhou, Y. Recent Progress of Using Knowledge Graph for Cybersecurity. *Electronics* **2022**, *11*, 2287. [CrossRef]
14. Auer, S.; Bizer, C.; Kobilarov, G.; Lehmann, J.; Cyganiak, R.; Ives, Z. Dbpedia: A nucleus for a web of open data. In *International Semantic Web Conference*; Springer: Berlin/Heidelberg, Germany, 2007.
15. Bollacker, K.; Evans, C.; Paritosh, P.; Sturge, T.; Taylor, J. Freebase: A collaboratively created graph database for structuring human knowledge. In Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, Vancouver, BC, Canada, 10–12 June 2008.
16. Suchanek, F.M.; Kasneci, G.; Weikum, G. Yago: A core of semantic knowledge. In Proceedings of the 16th international conference on World Wide Web, Banff, AB, Canada, 8–12 May 2007.
17. Fang, Y.; Zhang, Y.; Huang, C. CyberEyes: Cybersecurity entity recognition model based on graph convolutional network. *Comput. J.* **2021**, *64*, 1215–1225. [CrossRef]
18. Yi, F.; Jiang, B.; Wang, L.; Wu, J. Cybersecurity named entity recognition using multi-modal ensemble learning. *IEEE Access* **2020**, *8*, 63214–63224. [CrossRef]
19. Sills, M.; Ranade, P.; Mittal, S. Cybersecurity threat intelligence augmentation and embedding improvement-a healthcare use-case. In Proceedings of the 2020 IEEE International Conference on Intelligence and Security Informatics (ISI), Arlington, VA, USA, 9–10 November 2020.
20. Tikhomirov, M.; Loukachevitch, N.; Sirotina, A.; Dobrov, B. Using bert and augmentation in named entity recognition for cybersecurity domain. In Proceedings of the Natural Language Processing and Information Systems: 25th International Conference on Applications of Natural Language to Information Systems, NLDB 2020, Saarbrücken, Germany, 24–26 June 2020; Proceedings 25; Springer: Berlin/Heidelberg, Germany, 2020.
21. Jia, Y.; Qi, Y.; Shang, H.; Jiang, R.; Li, A. A practical approach to constructing a knowledge graph for cybersecurity. *Engineering* **2018**, *4*, 53–60. [CrossRef]

22. Huaijun, S. *Research and Implementation of Web Security Knowledge Base Construction Technology Facing Vulnerability Database*; National University of Defense Technology: Changsha, China, 2018.
23. Wang, T.; Ai, Z.; Zhang, X. Construction technology of threat intelligence knowledge graph based on deep learning. *Comput. Mod.* **2018**, *12*, 21–26.
24. Peng, J.; Fang, Y.; Huang, C.; Liu, L.; Jiang, Z. Research on Named Entity Recognition in the Field of Information Security Based on Deep Active Learning. *J. Sichuan Univ. Nat. Sci. Ed.* **2019**, *56*, 457–462.
25. Zhang, R.; Liu, J.; He, X. Named Entity Recognition in the Security Vulnerability Domain Based on BLSTM-CRF Model. *J. Sichuan Univ. Nat. Sci. Ed.* **2019**, *56*, 469–475.
26. Qin, Y.; Shen, G.; Yu, H. Large-scale network security entity recognition method based on Hadoop. *J. Intell. Syst.* **2019**, *14*, 1017–1025.
27. Yang, J.; Zhang, Y.; Li, L.; Li, X. YEDDA: A lightweight collaborative text span annotation tool. *arXiv* **2017**, arXiv:1711.03759.
28. Zhao, K.; Jin, X.; Wang, Y. A review of small sample learning research. *J. Softw.* **2020**, *32*, 349–369.
29. Basiri, M.E.; Nemati, S.; Abdar, M.; Cambria, E.; Acharya, U.R. ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis. *Future Gener. Comput. Syst.* **2021**, *115*, 279–294. [CrossRef]
30. Li, J.; Sun, A.; Han, J.; Li, C. A survey on deep learning for named entity recognition. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 50–70. [CrossRef]
31. Koutini, K.; Eghbal-zadeh, H.; Widmer, G. Receptive field regularization techniques for audio classification and tagging with deep convolutional neural networks. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2021**, *29*, 1987–2000. [CrossRef]
32. Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv* **2015**, arXiv:1511.07122.
33. Strubell, E.; Verga, P.; Belanger, D.; McCallum, A. Fast and accurate entity recognition with iterated dilated convolutions. *arXiv* **2017**, arXiv:1702.02098.