MTLAT: A Multi-Task Learning Framework Based on Adversarial Training for Chinese Cybersecurity NER

Yaopeng Han, Zhigang Lu, Bo Jiang, Yuling Liu, Chen Zhang, Zhengwei Jiang, Ning Li

MTLAT: A Multi-Task Learning Framework Based on Adversarial Training for Chinese Cybersecurity NER

Yaopeng Han[1,2], Zhigang Lu[1,2], Bo Jiang[1,2], Yuling Liu[1,2], Chen Zhang[1], Zhengwei Jiang[1,2], and Ning Li[1,⋆]

[1] Institute of Information Engineering, Chinese Academy of Sciences
[2] School of Cyber Security, University of Chinese Academy of Sciences
Beijing, China
{hanyaopeng, luzhigang, jiangbo, liuyuling, zchen, jiangzhengwei, lining6}@iie.ac.cn

Abstract. With the continuous development of cybersecurity texts, the importance of Chinese cybersecurity named entity recognition (NER) is increasing. However, Chinese cybersecurity texts contain not only a large number of professional security domain entities but also many English person and organization entities, as well as a large number of Chinese-English mixed entities. Chinese Cybersecurity NER is a domain-specific task, current models rarely focus on the cybersecurity domain and cannot extract these entities well. To tackle these issues, we propose a **M**ulti-**T**ask **L**earning framework based on **A**dversarial **T**raining (MTLAT) to improve the performance of Chinese cybersecurity NER. Extensive experimental results show that our model, which does not use any external resources except static word embedding, outperforms state-of-the-art systems on the Chinese cybersecurity dataset. Moreover, our model outperforms the BiLSTM-CRF method on Weibo, Resume, and MSRA Chinese general NER datasets by 4.1%, 1.04%, 1.79% F1 scores, which proves the universality of our model in different domains.

Keywords: Cybersecurity · Named entity recognition · Adversarial training · Multi-task learning.

1 Introduction

Named entity recognition is the task to identify entity boundaries and the recognition of categories of named entities, which is a fundamental task in the field of natural language processing (NLP).

The NER task in the general domain mainly identifies three types of entities: Person(PER), Organization(ORG), and Location(LOC).

Cybersecurity NER is a domain-specific task, which mainly extracts professional security entities from cybersecurity texts. In the domain of cybersecurity,

⋆ Corresponding author

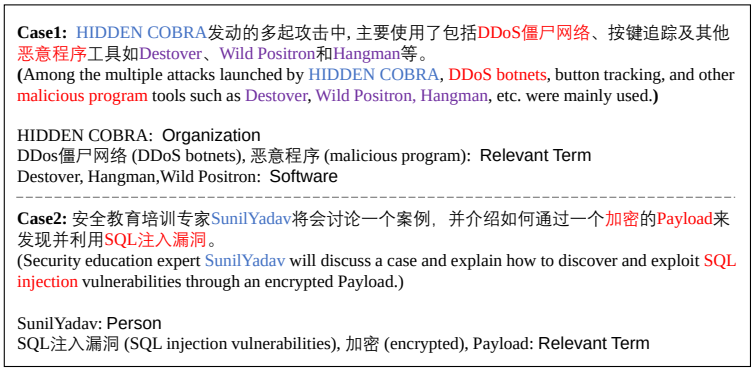

Fig. 1. Examples of the Chinese cybersecurity NER dataset. Organization (ORG), Relevant Term (RT), Software (SW), and Person (PER) are categories of cybersecurity dataset entities.

English NER [5, 18] research is much more than Chinese NER [16]. Compared with English NER, Chinese named entities are more challenging to identify due to their uncertain boundaries and complex composition. In this paper, we focus on Chinese cybersecurity NER. As shown in Fig. 1, compared with Chinese general NER tasks, entity extraction in the cybersecurity domain is a challenging task mainly because Chinese cybersecurity texts are often mixed with English entities, such as person, hacker organizations, and security-related entities (e.g., DDOS僵尸网络(*botnets*), SQL注入漏洞(*injection*)).

In this paper, we propose a novel framework, named multi-task learning based on adversarial training (MTLAT), to tackle the aforementioned challenges in Chinese cybersecurity NER. We design an auxiliary task to predict whether each token is an English entity, a Chinese entity or a non-entity to jointly train with NER task, which helps the model to learn semantic representations of named entities and to distinguish named entities from sequences in the Chinese cybersecurity domain. We also use Convolutional Neural Network (CNN) to enhance the ability of the model to capture local contextual information among characters sentences, which is also helpful for identifying English and security entities. Adversarial training is to enhance the security of machine learning systems [3] by making small perturbations to the input designed to significantly. In the NLP tasks, since the input text is discrete, the perturbation is added to the continuous the embedding layer as a regularization strategy [14] to improve robustness and generalization of the model.

With the above improvements, we can greatly boost the performance of the model in the extraction of name entities in the Chinese cybersecurity dataset. In summary, our main contributions are as follows:

– We propose a multi-task learning framework based on adversarial training (MTLAT) to imporve the performance of Chinese cybersecurity NER, and we use the CNN network to enhance the ability of the model to capture local

contextual information. Our model can well extract cybersecurity entities and English entities from Chinese cybersecurity texts.
– Our model achieves state-of-the-art F1 score on the Chinese cybersecurity NER dataset without using any external resources like lexicon resources and pre-trained models, which make it very practical for real-world NER systems. Furthermore, compared to the BiLSTM-CRF model, our model improves F1 scores on Weibo, Resume, and MSRA datasets in the general domain for 4.1%, 1.04%, 1.79%, which proves the universality of our model.

Our code and data are publicly available[1].

2 Related Work

2.1 NER

Recently, in the NER task, compared to the traditional methods that required hand-crafted features, many NER studies mostly focus on deep learning. [4] firstly proposed the BiLSTM-CRF, which is used by most state-of-the-art models.

Chinese NER is related to word segmentation. Therefore Chinese NER models have two main methods, one based on word-level and the other based on character-level. Recently, many studies [9,10] proved that the model based on the character-level is better than the model based on the word-level. Because word-level models often suffer from data sparsity caused by overly large dictionaries, and it will also cause word segmentation errors and out-of-vocabulary (OOV) problems. Character sequence labeling has been the dominant approach [2, 12] for Chinese NER. [19] proposed a lattice LSTM model, which integrates word-level information to the character-level model, but the lattice LSTM model can not batch-level training samples.

Recently, multi-task learning (MTL) gains significant attention. [15] proposed a model to train NER and word segmentation jointly. [1] proposed a NER model with two additional tasks that predict the named entity (NE) segmentation and NE categorization simultaneously in social media data. [20] proposed a novel deep neural multi-task learning framework to jointly model recognition and normalization on the medical NER task.

Recently, with the increasing number of cyberattacks, cybersecurity texts are also rapidly increasing. How to extract valuable information from cybersecurity texts has gradually become a research hotspot. [5] provided an English cybersecurity dataset that contained several categories of security entities and use CRF to solve this problem. [16] provided the Chinese cybersecurity dataset and use the CNN network to obtain the local feature of each character and use some hand-craft features into BiLSTM-CRF for entity extraction. These methods do not perform well in solving the aforementioned challenges in Chinese cybersecurity NER.

[1] https://github.com/xuanzebi/MTLAT

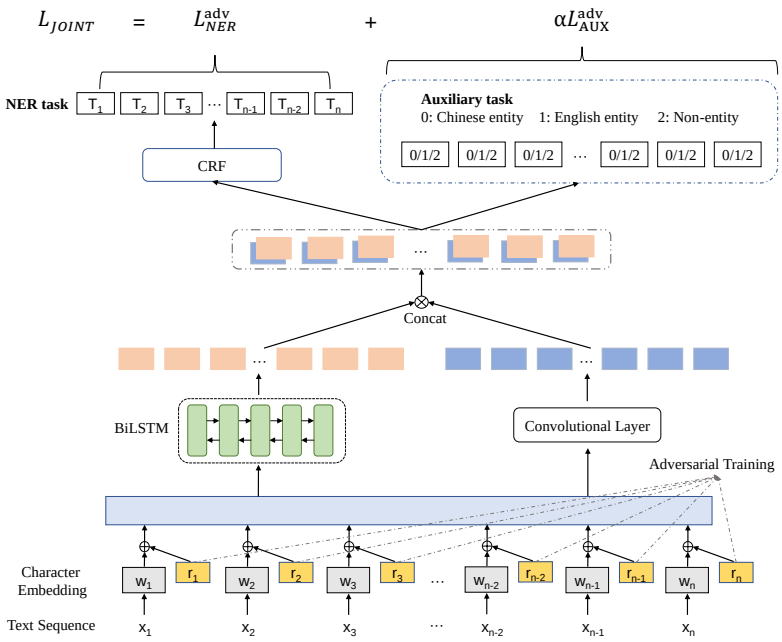

Fig. 2. The main framework of our MTLAT model.

2.2 Adversarial Training

Recently in the field of NLP, there has been a lot of researches [6, 14, 21] on adversarial training, mainly to obtain more accurate perturbations and add it to the embedding layer to improve the robustness and performance of the model. [14] proposed two adversarial training methods, fast gradient method (FGM) and virtual adversarial training (VAT), to enhance generalization of the model by adding perturbation on the embedding layer. To improve the robustness of neural networks against adversarial examples, many researchers pay more attention to propose more effective defense strategies and models. [13] proposed a projected gradient descent (PGD) method, which can be achieved reliably through multiple projected gradient ascent steps followed by a stochastic gradient descent step. [21] proposed a novel adversarial training algorithm Free Large-Batch (FreeLB), based on the Transformer network, which promotes higher invariance in the embedding space by adding adversarial perturbations to the embedding layer and minimizing the resultant adversarial risk inside different regions around input samples.

3 Methodology

In this paper, we propose a novel neural network framework, named multi-task learning based on adversarial training, for Chinese cybersecurity NER. The

structure of the proposed model is shown in Fig. 2. In this section, we first introduce the adversarial training used in MTLAT, and we then introduce the encoding framework of the model and then introduce the decoding and training based on adversarial training and multi-task learning.

Formally, we denote a Chinese sentence as $s = \{c_1, c_2, \ldots, c_n\}$, where c_i denotes the i_{th} character, n is the number of characters in the sentence. By looking up the embedding vector from a static character embedding matrix, we obtain $\mathbf{x}_i = E(c_i)$, where $\mathbf{x}_i \in \mathbb{R}^{d_e}$ and d_e is the dimension of the input character embeddings, E is a character embedding lookup table.

3.1 Adversarial Training

In the field of NLP, since the input is discrete, the perturbations are mostly added to the embedding layer, which can enhance the robustness and performance of the model. Generally, adversarial training can be described by the following formula. Adversarial training seeks to find optimal parameters $\boldsymbol{\theta}$ to minimize the maximum risk for $\mathbf{r}_{adv}$:

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\|\mathbf{r}_{adv}\|\leq\epsilon} L(\mathbf{x} + \mathbf{r}_{adv}, y, \boldsymbol{\theta}) \right] \tag{1}$$

where $\mathbf{x} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, y is the sentence label sequence, $\mathcal{D}$ is data distribution, L is loss function. When get $\mathbf{r}_{adv}$ through the following methods, then add perturbations $\mathbf{r}_{adv}$ to the character embedding $\mathbf{x}$ to get new embeddings $\mathbf{x}^*$, then feed $\mathbf{x}^*$ to the model to calculate the adversarial loss:

$$L^{adv} = L(\mathbf{x}^*, y, \boldsymbol{\theta}) = L(\mathbf{x} + \mathbf{r}_{adv}, y, \boldsymbol{\theta}) \tag{2}$$

In this paper, we use the PGD [13] method to calculate the perturbation $\mathbf{r}_{adv}$. Then we introduce the most effective method of adversarial training, PGD, because it can largely avoid the obfuscated gradient problem.

PGD: [13] proposed to solve the inner maximization of Eq.1 by using PGD (a standard method for large-scale constrained optimization) method. In particular, PGD takes the following step in each iteration:

$$\begin{aligned} \mathbf{g}(\mathbf{r}_t) &= \nabla_{\mathbf{r}} L(\mathbf{x} + \mathbf{r}_t, y) \\ \mathbf{r}_{t+1} &= \Pi_{\|\mathbf{r}\|_F\leq\epsilon}(\mathbf{r}_t + \alpha \mathbf{g}(\mathbf{r}_t) / \|\mathbf{g}(\mathbf{r}_t)\|_F) \end{aligned} \tag{3}$$

where $\mathbf{g}(\mathbf{r}_t)$ is the gradient of the loss with respect to $\mathbf{r}$, and $\Pi_{\|\mathbf{r}\|_F\leq\epsilon}$ performs a projection onto the ϵ-ball. After k-step PGD, add perturbation $\mathbf{r}_k$ to the character embedding:

$$\mathbf{x}^* = \mathbf{x} + \mathbf{r}_k \tag{4}$$

3.2 Encoding Layer

When using adversarial training methods to add perturbation $\mathbf{r}_{adv}$ to the character embedding vector $\mathbf{x}$, the embedding vector input for the current model is $\mathbf{x}^*$, and then feed it to our encoding layer.

BiLSTM: Most studies use BiLSTM to obtain text representations when processing text data, because it is a sequence model that can learn the internal connections of the text well and use context information of the text. In this paper, we use the character-level BiLSTM as the main network structure. We obtain the contextual representation $\mathbf{H} = \{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n\}$, where $\mathbf{H} \in \mathbb{R}^{n \times d_h}$ and d_h is the hidden dimension of BiLSTM output:

$$\begin{aligned} \overrightarrow{\mathbf{h}}_i &= \overrightarrow{LSTM^{(f)}}\left(\mathbf{x}_i^*, \overrightarrow{\mathbf{h}}_{i-1}\right) \\ \overleftarrow{\mathbf{h}}_i &= \overleftarrow{LSTM^{(b)}}\left(\mathbf{x}_i^*, \overleftarrow{\mathbf{h}}_{i+1}\right) \\ \mathbf{h}_i &= \left[\overrightarrow{\mathbf{h}}_i; \overleftarrow{\mathbf{h}}_i\right] \end{aligned} \tag{5}$$

Convolutional Layer: Convolution layers are performed at every window based location to extract local features. We apply a convolutional layer to extract character representation to enhance local feature, which is helpful for extracting English entities and security entities. By the same padding, filters are applied to n possible windows in the sequence and the local contextual representation can be represented as $\mathbf{G} = \{\boldsymbol{g}_1, \boldsymbol{g}_2, \ldots, \boldsymbol{g}_n\}$, where $\mathbf{G} \in \mathbb{R}^{n \times d_c}$ and d_c is the hidden dimension of CNN output:

$$\boldsymbol{g}_i = \mathrm{Re}\,LU\left(\mathbf{W}_1 \mathbf{x}_{i-h+1:i}^* + \mathbf{b}_1\right) \tag{6}$$

where $\mathbf{W}_1 \in \mathbb{R}^{h \times d_e \times d_c}$ and $\mathbf{b}_1 \in \mathbb{R}^{h \times d_c}$ are learnable parameters, $\mathbf{x}_{i-h+1:i}^*$ refers to the concatenation of character $\mathbf{x}_{i-h+1}^*, \mathbf{x}_{i-h+2}^*, \ldots, \mathbf{x}_i^*$ with h window size of filters are applied to the input sequence to generate character embedding representation. Finally, concat the representation obtained by BiLSTM and CNN:

$$\mathbf{R} = [\mathbf{H}; \mathbf{G}] \tag{7}$$

where $\mathbf{R} \in \mathbb{R}^{n \times (d_h + d_c)}$. Then feed $\mathbf{R}$ to CRF and multi-task network to calculate the NER loss and multi-task loss.

3.3 Training

NER task: We use the CRF layer as the decoding layer of the NER task and calculate the probability of the ground-truth tag sequence $p\left(\mathbf{y}_i \mid s_i\right)$, and then we can calculate the NER task loss:

$$L_{\texttt{NER}}^{adv} = -\sum_{i=1}^{N} \log\left(p\left(\mathbf{y}_i \mid s_i\right)\right) \tag{8}$$

$$p(\mathbf{y} \mid s) = \frac{\exp\left(\sum_i \left(A_{y_{i-1}, y_i} + W_{y_i} R_i\right)\right)}{\sum_{y' \in \mathcal{Y}} \exp\left(\sum_i \left(A_{y'_{i-1}, y'_i} + W_{y'_i} R_i\right)\right)} \tag{9}$$

where $\mathcal{Y}$ denotes the set of all possible label sequences, A is the probability of transitioning from one tag to another. W_{y_i} is used for modeling emission potential for the i_{th} character in the sentence.

Table 1. Statistics of the Chinese cybersecurity NER dataset.

Type	Train	Dev	Test
Sentences	38.2k	4.8k	4.8k
Chars	2210.3k	270.1k	278.4k
PER	9944	1355	1291
ORG	14557	1727	1861
LOC	18958	2290	2467
SW	5397	719	647
RT	64471	7753	8263
VUL_ID	265	25	30
Total Entities	113.5k	13.8k	14.6k

Table 2. Statistics of Chinese general NER datasets.

Dataset	Type	Train	Dev	Test
MSRA	Sentences	46.4k	-	4.4k
	Chars	2169.3k	-	172.6k
	Entities	74.8k	-	6.2k
Weibo	Sentences	1.4k	0.27k	0.27k
	Chars	73.8k	14.5k	14.8k
	Entities	1.89k	0.42k	0.39k
Resume	Sentences	3.8k	0.46k	0.48k
	Chars	124.1k	13.9k	15.1k
	Entities	1.34k	0.16k	0.15k

Auxiliary task: Inspired by the [11], to better distinguish between Chinese and English entities, we add an auxiliary task to predict whether the pred tokens are Chinese entities, English entities, or non-entities. Additionally, the auxiliary task acts as a regular method to help the model to learn general representations of named entities. Given a set of training example $\left\{\left(s_i, \hat{\mathbf{y}}_i \in \mathbb{R}^{n\times 3}\right)\right\}\Big|_{i=1}^{N}$ for the auxiliary task, the auxiliary task loss can be defined as follows:

$$p(\hat{\mathbf{y}} \mid s) = \operatorname{softmax}\left(\mathbf{W}_2\mathbf{R} + \mathbf{b}_2\right) \tag{10}$$

$$L_{\texttt{AUX}}^{adv} = -\frac{1}{N}\sum_{i=1}^{N} \tilde{y}_i \log\left(p\left(\hat{\mathbf{y}}_i \mid s_i\right)\right) \tag{11}$$

where $\mathbf{W}_2 \in \mathbb{R}^{(d_h+d_c)\times 3}$ and $\mathbf{b}_2 \in \mathbb{R}^3$ are trainable parameters, $\tilde{y}$ is the auxiliary task gold label of the sentence s.

Through adversarial training, we can get the NER task loss $L_{\texttt{NER}}^{adv}$ and the auxiliary task loss $L_{\texttt{AUX}}^{adv}$, then we add these two losses to update parameters of the model by backpropagation algorithm for jointly training:

$$L_{\texttt{JOINT}} = L_{\texttt{NER}}^{adv} + \alpha L_{\texttt{AUX}}^{adv} \tag{12}$$

where α is the balancing parameter.

4 Experiments

4.1 Datasets

Chinese Cybersecurity NER Dataset: [16] collected and labeled the Chinese cybersecurity NER dataset from the Freebuf website and the Wooyun vulnerability database, mainly including security text data such as technology sharing, network security, vulnerability information, etc. The Chinese cybersecurity dataset includes six types of security entities, including names of the person (PER), location (LOC), organization (ORG), software (SW), relevant term (RT) and vulnerability (VUL_ID). In this paper, we mainly evaluate our model on a larger dataset that they open source[2]. The specific analysis is shown in Table 1.

[2] https://github.com/xiebo123/NER

Table 3. Results with different methods on the Chinese cybersecurity NER test dataset.

Models	Precision(%)	Recall(%)	F1(%)
Baseline	90.74	89.40	90.07
w/ FGM	91.81	90.25	91.03
w/ VAT	91.65	89.53	90.58
w/ PGD	92.37	90.25	91.30
w/ FreeLB	92.60	89.73	91.14
w/ MTL	91.55	90.10	90.82
w/ CNN	92.31	90.17	91.23
Lattice LSTM	91.07	**91.36**	91.21
MTLAT(ours)	**92.90**	90.74	**91.81**

Chinese General NER Datasets: We also evaluate the effect of our model on Chinese general domain NER datasets, Weibo NER [15], MSRA [8], and Resume NER [19]. Their statistics are listed in Table 2. Weibo NER is based on the text in Chinese social media Sina Weibo. MSRA comes mainly from news domain. Resume NER is collected from Sina Finance. These domains are the domains that the public often pays attention to, and can be unified into general domains. On the contrary, except for security personnel, the cybersecurity domain has received little public attention.

4.2 Comparison Methods

Baseline model: In this paper, we use the character-level BiLSTM-CRF [7] model as the comparison baseline method. We also explore the four different methods of adding adversarial training (FGM, VAT, PGD, and FreeLB) based on the baseline model.

Lattice LSTM: Lattice LSTM [19] incorporates word-level information into character-level recurrent units, which can avoid segmentation errors. The Lattice LSTM achieves state-of-the-art performance on the Chinese general domain NER datasets.

4.3 Hyper-Parameter Settings

For hyperparameter configuration, we adjust them according to the performance on the development datasets for all NER datasets. For all of the datasets, we use the Adam optimization to train our networks, and the initial learning rate was set at 0.015 for the cybersecurity NER dataset, 0.005 for other NER datasets. We set α in Eq. 12 to 1. We set the hidden sizes of BiLSTM to 256 dims. We use one layer of CNN with an output channel size of 200 and set the window size as 3. The character embedding used in our all experiments are from [17]. To avoid

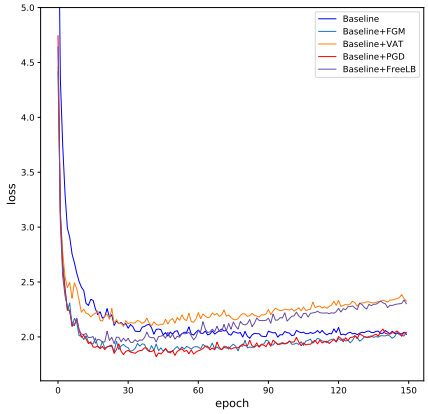

Fig. 3. Loss of the Chinese cybersecurity development dataset with four different adversarial training methods.

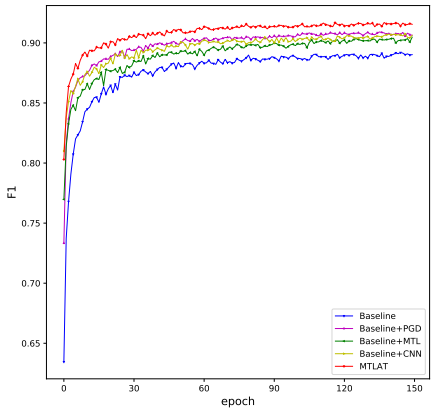

Fig. 4. F1 scores against training iteration number on the Chinese cybersecurity development dataset.

overfitting, we apply dropout (50% dropout rate) on the character embedding and (20% dropout rate) on the output layer. We use "BIEOS" as the decoder tag scheme for all datasets.

5 Results and Analysis

5.1 Results on the Chinese Cybersecurity NER Dataset

We first compare the impact of different adversarial training methods on the Chinese cybersecurity test dataset. As shown in Table 3, adding any kind of adversarial training method to the baseline model can improve the F1 score on the cybersecurity dataset, which proves the effectiveness of adversarial training. PGD and FreeLB use K-step iterations to obtain the optimal perturbations and obtain a higher F1 score. In addition, Fig. 3 compares the loss effect of these four methods (FGM, VAT, PGD, and FreeLB) on the development dataset. Among the four adversarial training methods, PGD can obtain better robustness and generalization on the development dataset. Therefore, the PGD method is used in our MTLAT model. We find that adding the auxiliary task (MTL) to the baseline model is helpful for recalling entities. And using the CNN network can enhance the local feature representation of the text, greatly improving the precision and recall score of entity extraction. Fig. 4 shows the comparison of the effect of adding CNN, PGD and MTL to the baseline model on the development dataset. It shows that the MTLAT achieves the best performance by adding these three methods to the baseline model.

[19] introduce a lattice LSTM to incorporate external lexicon information into the model. Compared with the baseline model, the F1 score of the lattice model using external data is improved by 1.14%. Table 3 shows that our MTLAT

Table 4. F1 scores on Chinese general NER test datasets. 1 represents the word-level LSTM model, 2 indicates the character-level LSTM model, and 3 is the lattice LSTM model.

Models	Weibo	Resume	MSRA
Zhang and Yang(2018) [19][1]	47.33	93.58	86.85
Zhang and Yang(2018) [19][2]	52.77	93.48	88.81
Zhang and Yang(2018) [19][3]	**58.79**	94.46	**93.18**
Baseline†	54.05	93.62	89.45
Baseline-CNN-PGD	**58.15**	**94.66**	**91.24**

Table 5. Case Study. We use red to denote the correct labels, blue to denote the wrong labels and purple to denote entities in the sentence. SW means software and LOC means location.

Case	Sentences	侵入乌克兰工控系统的罪魁祸首可能是 Win32Industroyer. (The culprit of the invasion of the Ukrainian industrial control system may be Win32Industroyer.)
	Gold label	···乌克兰(Ukrainian) (B-LOC, I-LOC, E-LOC)··· ···Win32Industroyer (B-SW, I-SW, E-SW)···
	Baseline predicted label	···乌克兰(Ukrainian) (B-LOC, I-LOC, E-LOC)··· ···Win32Industroyer (O, O, O)···
	MTLAT predicted label	···乌克兰(Ukrainian) (B-LOC, I-LOC, E-LOC)··· ···Win32Industroyer (B-SW, I-SW, E-SW)···

model achieves 91.81% F1 score on the test dataset, which outperforms the lattice LSTM by 0.6%. Overall, our model does not require any external data on the cybersecurity dataset to achieve state-of-the-art performance, which can be more easily applied to real-world systems.

5.2 Results on Chinese General NER Datasets

Because there are few English entities in the general domain of NER datasets, we do not apply the auxiliary task on general domain datasets. We only add the adversarial training method PGD and CNN network to baseline model, namely **Baseline-CNN-PGD**. The results are reported in Table 4. It shows that our character-level baseline model outperforms the same network proposed by [19]. It can see that our model Baseline-CNN-PGD outperforms the best character-level and word-level models on all three datasets. Although the results of our model on Weibo and MSRA datasets are slightly lower than Lattice LSTM, Lattice LSTM leverages external lexicon resources and can not batch training, resulting in highly inefficient. It proves that adversarial training can improve the robustness and generalization of the model, and the CNN network enhance the ability of the model to capture local contextual information, which are of

great help to improve the performance of the model, and further proves the universality of our model in different domains.

5.3 Case Study

To show visually that our model can solve the challenges of identifying English entities, a case study comparing the baseline model and our model in Table 5. In the case, there are two entities, a Chinese location entity "乌克兰(Ukrainian)" and an English software entity "Win32Industroyer". The baseline model can extract Chinese entities well, but it incorrectly recognizes English entities, while our model can extract not only Chinese entities well, but also English professional security entities.

6 Conclusion

In this paper, we propose a multi-task learning framework based on adversarial training (MTLAT) method to enhance the performance of Chinese cybersecurity NER. We incorporate adversarial training into the embedding layer to improve robustness and generalization of the model and use the CNN network to enhance feature local representations. Extensive experiments show that our model does not require any external data on the Chinese cybersecurity dataset to achieve state-of-the-art performance, which can be more easily applied to real-world systems. Moreover, compared with the BiLSTM-CRF method, our model has 4.1%, 1.04%, 1.79% F1 scores improvement on Weibo, Resume, and MSRA datasets, which proves the universality of our model in different domains.

Acknowledgments

This research is supported by National Key Research and Development Program of China (No.2019QY1303, No.2019QY1301, No.2018YFB0803602), and the Strategic Priority Research Program of the Chinese Academy of Sciences (No.XDC02040100), and National Natural Science Foundation of China (No.61702508, No.61802404). This work is also supported by the Program of Key Laboratory of Network Assessment Technology, the Chinese Academy of Sciences; Program of Beijing Key Laboratory of Network Security and Protection Technology.

References

1. Aguilar, G., Maharjan, S., López-Monroy, A.P., Solorio, T.: A multi-task approach for named entity recognition in social media data. CoRR **abs/1906.04135** (2019)
2. 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, pp. 239–250. Springer (2016)
3. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: ICLR (2015)

4. Huang, Z., Xu, W., Yu, K.: Bidirectional lstm-crf models for sequence tagging. CoRR **abs/1508.01991** (2015)
5. Joshi, A., Lal, R., Finin, T., Joshi, A.: Extracting cybersecurity related linked data from text. In: 2013 IEEE Seventh International Conference on Semantic Computing. pp. 252–259. IEEE (2013)
6. Ju, Y., Zhao, F., Chen, S., Zheng, B., Yang, X., Liu, Y.: Technical report on conversational question answering. CoRR **abs/1909.10772** (2019)
7. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. In: NAACL. pp. 260–270 (2016)
8. Levow, G.A.: The third international chinese language processing bakeoff: Word segmentation and named entity recognition. In: ACL. pp. 108–117. Association for Computational Linguistics (2006)
9. Li, H., Hagiwara, M., Li, Q., Ji, H.: Comparison of the impact of word segmentation on name tagging for chinese and japanese. In: LREC. pp. 2532–2536 (2014)
10. Li, X., Meng, Y., Sun, X., Han, Q., Yuan, A., Li, J.: Is word segmentation necessary for deep learning of chinese representations? In: ACL. pp. 3242–3252. Association for Computational Linguistics (2019)
11. Liu, Z., Winata, G.I., Fung, P.: Zero-resource cross-domain named entity recognition. In: ACL. pp. 1–6 (2020)
12. Lu, Y., Zhang, Y., Ji, D.: Multi-prototype chinese character embedding. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16). pp. 855–859 (2016)
13. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. In: ICLR (2018)
14. Miyato, T., Dai, A.M., Goodfellow, I.: Adversarial training methods for semi-supervised text classification. In: ICLR (2017)
15. Peng, N., Dredze, M.: Improving named entity recognition for chinese social media with word segmentation representation learning. In: ACL: Short Papers (2016)
16. Qin, Y., Shen, G.w., Zhao, W.b., Chen, Y.p., Yu, M., Jin, X.: A network security entity recognition method based on feature template and cnn-bilstm-crf. Frontiers of Information Technology & Electronic Engineering **20**(6), 872–884 (2019)
17. Song, Y., Shi, S., Li, J., Zhang, H.: Directional skip-gram: Explicitly distinguishing left and right context for word embeddings. In: NAACL-HLT, Volume 2 (Short Papers). pp. 175–180 (2018)
18. Weerawardhana, S., Mukherjee, S., Ray, I., Howe, A.: Automated extraction of vulnerability information for home computer security. In: International Symposium on Foundations and Practice of Security. pp. 356–366. Springer (2014)
19. Zhang, Y., Yang, J.: Chinese NER using lattice LSTM. In: ACL. pp. 1554–1564 (2018)
20. Zhao, S., Liu, T., Zhao, S., Wang, F.: A neural multi-task learning framework to jointly model medical named entity recognition and normalization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 817–824 (2019)
21. Zhu, C., Cheng, Y., Gan, Z., Sun, S., Goldstein, T., Liu, J.: Freelb: Enhanced adversarial training for language understanding. In: ICLR (2020)