IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 12, 2020, accepted March 28, 2020, date of publication March 31, 2020, date of current version April 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2984582

Cybersecurity Named Entity Recognition Using Multi-Modal Ensemble Learning

FENG YI[1], BO JIANG[2], LU WANG[2], AND JIANJUN WU[3]
[1]School of Computer Science, Zhongshan Institute, University of Electronic Science and Technology of China, Zhongshan 528402, China
[2]Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
[3]Beijing College of Politics and Law, Beijing 100024, China

Corresponding author: Lu Wang (wanglu@iie.ac.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61702508 and Grant 61802404, in part by the National Social Science Foundation of China under Grant 19BSH022, and in part by the National Key Research and Development Program of China under Grant 2019QY1303.

ABSTRACT Cybersecurity named entity recognition is an important part of threat information extraction from large-scale unstructured text collection in many cybersecurity applications. Most existing security entity recognition studies and systems use regular matching strategy or machine learning algorithms. Due to the peculiarity and complexity of security named entity, these models ignore the characteristic of security data and the correlation of entities. Therefore, through the in-depth study of security entity characteristic, we propose a novel security named entity recognition model based on regular expressions and known-entity dictionary as well as conditional random fields (CRF) combined with four feature templates. This model is named RDF-CRF. The rule-based expressions can match security entities with good accuracy in simpler situations, the known-entity dictionary can extract common and specific security entity, and the CRF-based extractor leverages the identified entities by rule-based and dictionary-based extractors to further improve the recognition performance. In order to demonstrate the effectiveness of our proposed model, extensive experiments are performed on a security text dataset collected from public security webs. The experimental results show that can achieve better performance than state-of-the-art methods.

INDEX TERMS Cybersecurity, named entity recognition, regular expression, known-entity dictionary, conditional random fields.

I. INTRODUCTION

Recent years have witnessed the importance of cybersecurity, which is paid attention to more and more importantly such as application attacks, malware, ransomware, phishing and exploit kits. A large amount of cybersecurity data has been published on various network platforms, such as security blogs, forums, software vendors bulletin boards, official news and social networks. These unstructured security texts contain high-value latest security information and events, like software vulnerabilities [1], attack detection [2], and threat action [3]. Nowadays, it becomes a trend to establish a security knowledge graph with open interconnect and semantic processing capabilities which can help security analysts more quickly retrieve and collate large-scale threat data. The basic task of establishing such a knowledge graph is information extraction. Therefore, automatically extracting security knowledge from a collection of unstructured text documents is a critical and fundamental task in the field of cybersecurity.

Named entity recognition (NER) is the most basic step of information extraction that seeks to locate and classify named entities in text into pre-defined categories [4]. NER systems are often used as the first step in question answering, information retrieval, co-reference resolution, topic modeling, etc. The main task is to identify named entities like person, location, organization, time, quantities, monetary values, percentages, etc. from unstructured texts [5]–[7]. In recent years, many named entity recognition models have been proposed to help users to find objects of value information, including recommendation system [8], [9], question answering [10], [11] and biomedical [12], [13]. In the domain of cybersecurity, security information extraction have attracted many research efforts from different perspectives. For example, some researchers have reported the results of security entity recognition from the view of data source, including Twitter [14], National Vulnerability Database [15], hacker

The associate editor coordinating the review of this manuscript and approving it for publication was Ananya Sen Gupta.

forums [16], and technical blogs [17]. On the other hand, there are also a variety of efforts studying different methods for the task, which can be divided into two classes: rule-based and machine learning-based.

The rule-based methods can extract named entity with good accuracy in a simple manner when the to-be-extracted information follows regular speech patterns such as email address, host IP, and Common Vulnerabilities and Exposures (CVE) [17], [18]. However, these methods are not suitable for complex situations while to-be-extracted entity includes many variations or comes from irregular structured text, which is more in line with the actual situation on the network. Meanwhile, these methods are difficult to identify new named entity. Moreover, designing rule-based systems is very time-consuming and requires expert field knowledge. Therefore, the rule-based methods lead to unsatisfactory results for cybersecurity named entity identification in the complex situations. Taking into consideration the good performance and simplicity of rule-based methods and the regular patterns of some security entities such as IP and CVE, in this paper we also introduce the rule-based template to extract cybersecurity named entities.

In these more complex situations, machine learning-based methods outperform rule-based ones by tuning general algorithms with existing data. Meanwhile, they can identify new entities from training corpus and are suitable for widespread applications. Recent years, a lot of approaches for security-relevant named entity recognition (NER) from unstructured text documents have been proposed from different perspectives, including conditional random fields (CRF) [19], [20], support vector machines (SVM) [16], expectation regularization [14], bootstrapping algorithm [21], maximum entropy model (ME) [22], and long short-term memory (LSTM) [23], [24] etc. However, all of the above machine learning methods fail to yield satisfactory results for identifying cybersecurity related concepts and entities from unstructured cybersecurity texts collection. Through analyzing these texts, we find that existing entity recognition techniques is not suitable for the task. Although the named entity recognition technology has gradually matured in the general field, when it is directly applied to the professional zone, it usually fails to produce satisfactory results. For example, in the field of biomedicine, Dongliang *et al.* [25] illustrates, despite the traditional method is easy to use,the assumptions it relies on do not fully reflect the actual situation of a large number of complex biological texts, so the accuracy is relatively poor. The same problem also occurs in the field of cyber security. This is because cybersecurity texts contain a lot of security vocabularies, such as file names, hash value, and even attack tools. On the other hand, these models need to manually explore a wide range of features and ignore the correlation of entities, which is not amenable to large-scale applications.The rules and dictionaries constructed in this paper, as well as the features extracted for training the model, are obtained through observation and training of corpus in the security field, so they are generally applicable to tasks in such field. The experimental results of the article prove that with the expansion and improvement of the corpus in the follow-up work, the accuracy of the recognized professional vocabulary will also be significantly improved.

In this paper, we propose a novel security entity recognition model based on conditional random fields combined with four feature templates and incorporating regular expressions, known-entity dictionary for preprocessing, named RDF-CRF. Specifically, rule-based approach can first extract named entity with good accuracy in simpler situations, then dictionary-based method can match common and specific security entity. After matching by rule-based and dictionary-based methods, the word sequence will be more accurately matched to the feature templates by considering contextual information so that CRF-based model can further improves the recognition performance. To demonstrate the effectiveness of our proposed model, extensive experiments are performed on a security dataset collected from security Webs. The experimental results shows that the proposed method can achieve better performance than state-of-the-art methods.

The contributions of this paper are summarized as follows.

- We propose a novel security named entity recognition model by using a combination of regular expressions, known-entity dictionary and conditional random fields. In the proposed model, the identified entities by rule-based and dictionary-based approaches can further assist CRF-based model in improving the performance of cybersecurity entity recognition.
- We also design four feature templates for unstructured security entity recognition, including atomic features, combination features, maker features, and semantic features, to filter the feature vectors of current word for conditional random fields.
- Various experiments are conducted on real-world cybersecurity dataset, and the results demonstrate that our proposed model can achieve better prediction performance than the state-of-the-art methods.

The remainder of this paper is organized as follows: Section II reviews related work. Section III describes the proposed model and provides an efficient optimization method for the solution. We empirically evaluate our method on real-world dataset in Section IV, including a comparison to competing methods. We conclude the paper in Section V.

II. RELATED WORK

These studies on security named entity recognition can be fallen into two categories: rule-based and machine learning-based approaches. Next, we briefly review these works.

A. RULE-BASED ENTITY EXTRACTION METHODS

The rule-based matching methods to locate and extract information by constructing regular expressions or other heuristic rules. For example, Liao *et al.* [17] propose a fully automated Indicators of Compromise (IOC) [26] extraction, named iACE. iACE uses a set of regular expressions

and common context terms extracted from iocterms to identify the IOC tokens, such as IP and MD5 string. Balduccini *et al.* [18] design a set of regular expressions for matching each entity contained in the file of cyber assets. However, due to the unstructured characteristics and diversity of many security entities, it is very difficult to construct rules for all these types of entity. As a result, the heuristics strategy is expensive and unimplemented in large scale application.

B. MACHINE LEARNING-BASED ENTITY EXTRACTION METHODS

The machine learning-based approaches use training corpus to construct statistical learning models, which can realize automatic information extraction. Many efforts have been made in the task of cybersecurity named entity recognition. For instance, Lal *et al.* [20] utilize conditional random fields algorithm to extract cybersecurity related concepts and entities by using a set of features from manually annotated security texts. Joshi *et al.* [19] use conditional random field to identify cybersecurity-related entities, concepts and relations from the National Vulnerability Database and from text sources. Deliu *et al.* [16] extract cyber threat intelligence from hacker forums based on support vector machines and convolutional neural networks. Jones *et al.* [21] implement a bootstrapping algorithm for extracting security entities and their relationships from security texts. Ritter *et al.* [14] propose a weakly supervised seed-based approach to event extraction from Twitter. Mittal *et al.* [1] analyze tweets about cybersecurity and issue timely threat alerts to security analysts. Weerawardhana *et al.* [27] present machine learning-based and part-of-speech tagging approaches to information extraction from online vulnerability databases. Bridges *et al.* [22] propose a Maximum Entropy Model trained with the many security corpus and achieve a high performance of identification and classification of appropriate entities. Gasmi *et al.* [23] combine the advantage of Long Short-Term Memory (LSTM) and Conditional Random Field (CRF) methods to improve the accuracy of NER extraction compared with traditional pure statistical CRF method. Furthermore, Qin *et al.* [24] propose a combined model of neural networks which is called FT-CNN-BiLSTM-CRF. When training the models,they use feature templates to extract context features as we do and achieve an F-score of 0.86 on their network security dataset.

In conclusion, although the above mentioned methods work well to some extent in incorporating one or two of the three components (i.e., rule-based method, dictionary-based method and machine learning-based method), none of them integrate all the information from these three components into an unified learning framework for cybersecurity named entity recognition, resulting in dissatisfactory results. To the best of our knowledge, there is still a lack of cybersecurity named entity recognition method that extract entities of security texts at high precision level.

TABLE 1. The example of regular expression.

Entity Types	Regular Expression
Filename	[A-Za-z0-9-_\.]+\.(txt|php|exe|dll|bat|sys|htm |$html$|js|jar|jpg|png|vb|scr|pif|chm|zip|rar |cab|pdf|doc|$docx$|ppt|$pptx$|xls|$xlsx$|swf|gif)
Filepath	[a-zA-Z]:(\\([0-9a-zA-Z]+)
Email	[a-z][_a-z0-9-.]+@[a-z0-9-]+[a-z]+
SHA1	[a-f0-9]{40}|[A-F0-9]{40}
SHA256	[a-f0-9]{64}|[A-F0-9]{64}
CVE	CVE−[0-9]{4}−[0-9]{4,6}
URL	(https?|ftp|file)://[-A-Za-z0-9+&@#/%?=∼ _|! : , .;]+[-A-Za-z0-9+&@#/%?=∼ _|]
IPv4	(?:(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0- 9]?)\.)\{3\}(?:25[0-5]|2[0-4][0-9]|[01]?[0- 9][0-9]?)(/([0-2][0-9]|3[0-2]|[0-9]))?

III. THE PROPOSED MODEL

In this section, we present a novel ensemble learning approach for security entity extraction from documents. The proposed model consists of rule-based extractor, dictionary-based extractor and CRF-based extractor. Rule-based extractor is designed based on regular expressions, dictionary-based extractor includes known-entity lists, and CRF-based extractor leverages the identified entities by rule-based and dictionary-based extractors to improve the recognition performance. The overall architecture of the model is illustrated in Figure 1.

A. RULE-BASED EXTRACTOR

A lot of entities have certain rule patterns in the domain of cybersecurity. Through a large number of observations based on unstructured security texts, we find that URL is started with http/https string, Email contains symbol @ in the middle of a string and CVE follows specific named format. Hence, these security entities can be extracted based on regular expression matching. According to the naming rules of specific security entities, we design the template of regular expression rules, as shown in Table 1. The rule-based extractor have the properties of high precision and high recall as well as scalability.

B. DICTIONARY-BASED EXTRACTOR

As far as we know, existing many named entities are well known concepts in the cybersecurity domain, including large security companies (e.g., Cisco, FireEye, and IBM, etc.), software products (e.g., operating systems, firewalls, and anti-virus software, etc.) and hacker groups (e.g., OurMine, Anonymous, and DCLeaks, etc.). Based on these observations, we also design a known-entity dictionary including various entities. The entities can be categorized into the following categories: company, hardware, software, attack means, operating system, protocol, hacker groups and so on.

C. CONDITIONAL RANDOM FIELDS-BASED EXTRACTOR

CRF model can further extract the undiscovered entities on basis of the identified entities by rule-based extractor and dictionary-based extractor. We propose four feature templates to filter the feature vectors of current word for CRF model.

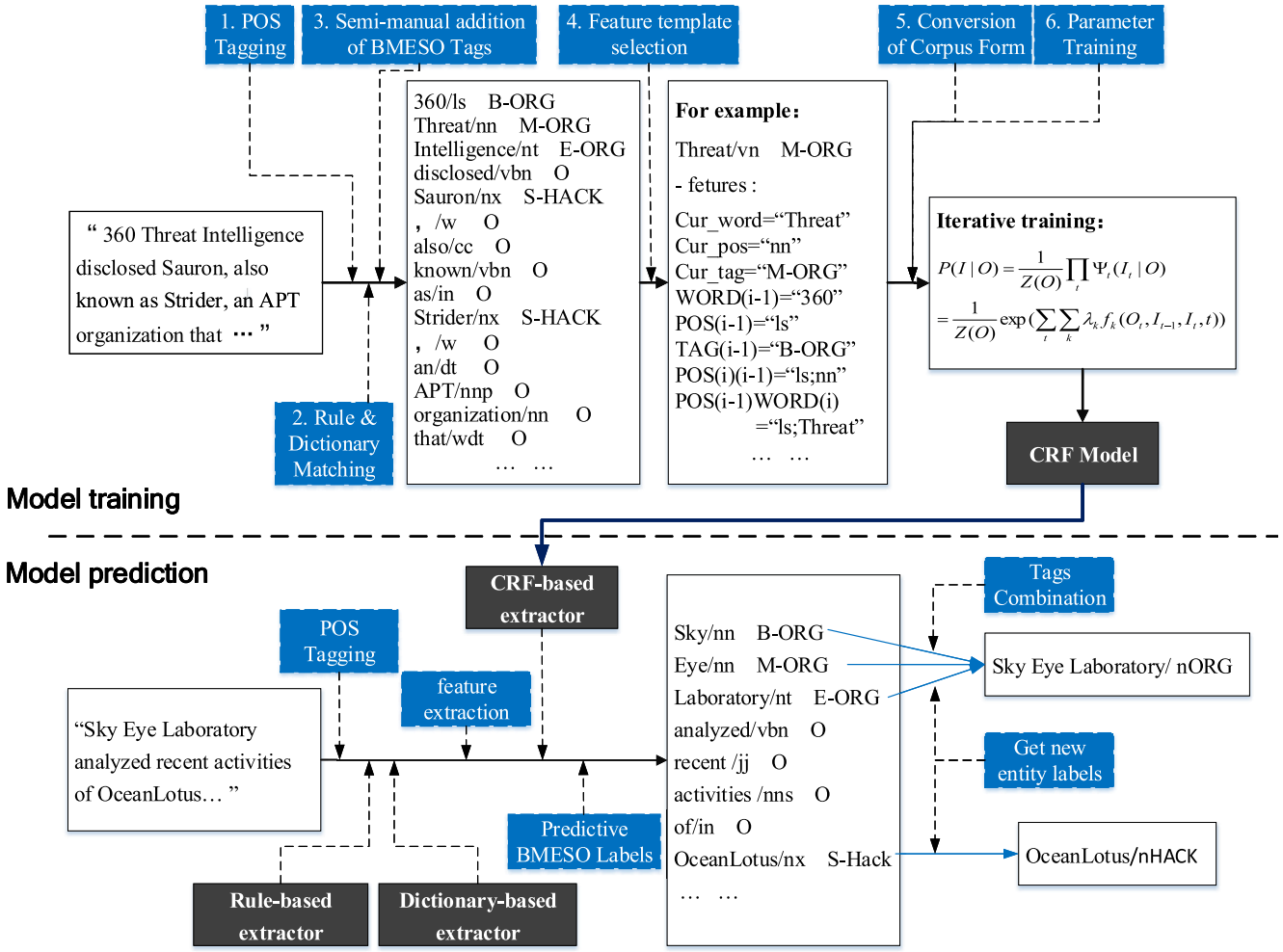


FIGURE 1. Overall architecture of security entity recognition model. Our proposed framework consists of three components: (1) rule-based extractor, (2) dictionary-based extractor and (3) CRF-based extractor.

1) ATOMIC FEATURES TEMPLATE

A simple but powerful method is to use tokenization and Part-Of-Speech (POS) tagger for named entity recognition. Due to not be separable again, we consider the features of part of speech and morphology of words as atomic features. Table 2 summarizes the detailed information of the atomic features.

According to Table 2, when the current word is "Google", which belong to the independent organization word, the corresponding feature functions can be generated as follows:

$$f(x, y) = \begin{cases} 1 & \text{if Word(0)} = \text{"Google" and y} = \text{Org} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where the variable y represents the label of the current word.

The template describes the individual morphology or part of speech of each word in the current word and its context windows, but it can not adequately describe the complex phenomena of language.

2) COMBINATION FEATURES TEMPLATE

In fact, simple morphological and part-of-speech features only contains the limited context information. Combination features can make use of long-distance constraints and rich context information. As shown in Table 3, we construct combination features based on the template of atomic features to form new rule features.

Based on these features, given a sentence "Google Released…", when the current word is "Google", we can

TABLE 2. The template of atomic features.

Atomic Features	Description
Word(0)	Current word
Word(-1)	The first word on the left of current word
Word(-2)	The second word on the left of current word
Word(1)	The first word on the right of current word
Word(2)	The second word on the right of current word
POS(0)	The part of speech of current word
POS(-1)	The part of speech of the first word on the left of current word
POS(-2)	The part of speech of the second word on the left of current word
POS(1)	The part of speech of the first word on the right of current word
POS(2)	The part of speech of the second word on the right of current word

TABLE 3. The template of combination features.

Combination Features	Description
Word(0)+POS(0)	Current word and part of speech
Word(0)+Word(-1)	Current word and the first word on the left of current word
Word(0)+Word(1)	Current word and the first word on the right of current word
Word(-1)+POS(0)	The first word on the left of current word and part of speech of current word
Word(0)+POS(1)	Current word and part of speech of current word
Word(-1)+POS(-1)	The first word and part of speech on the left of current word
Word(-1)+Word(-2)	The first word and the second word on the left of current word
Word(-2)+POS(-2)	The second word and part of speech on the left of current word
Word(1)+Word(2)	The first word and the second word on the right of current word
Word(-1)+Word(1)	The first word on the left of current word and the first word on the right of current word
Word(1)+POS(0)	The first word and part of speech on the right of current word
POS(-2)+POS(-1)	The part of speech of the second word and the first word on the left of current word
POS(-2)+POS(0)	The part of speech of current word and the part of the second word on the left of current word
POS(-1)+POS(0)	The part of the first word on the left of current word and the part of the current word
POS(-1)+POS(1)	The part of the first word on the left of current word and the part of the first word on the right
POS(0)+POS(1)	The part of the word of current word and the part of the word of the first word on the right
POS(0)+POS(2)	The part of speech of current word and the second word on the right of current word
POS(1)+POS(2)	The part of speech of the first word and the second word on the right of current word

define the binary function as follows:

$$f(x,y) = \begin{cases} 1 & \text{if Word(0) = ``Google'' and} \\ & \text{POS(1) = ``verb'' and y = Org} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

With the increase of the size of the combination of atomic template, the complexity of the model will be greatly improved. Meanwhile, related studies show that the combination template with two atomic features can play a better role, but the combination template composed of more than three atom features will cause the the high cost of computation.

3) MARKER FEATURES TEMPLATE

The template of marker features can be inferred the tag of current word by using predicted tag information and be described the mutual constraint information between entities to prevent the appearance of similar situations like ''two adjacent B-tags''. The template is constructed by the rules of the internal indicators and context indicators. The marker feature template is shown in Table 4.

For example, in the phrase ''hacker organization *eqnarray*'', when the current word is ''equation'', we can get the

TABLE 4. The template of marker features.

Marker Features	Description
Tag(-1)	Entity tag of first word on the left of current word
Tag(-2)	Entity tag of second word on the left of current word
Tag(-1)+Tag(-2)	Entity tags of the first word and the second word on the left of current word
POS(0)+Tag(-1)	The part of speech of current word and entity mark of the first word on the left of current word
POS(0)+Tag(-2)	The part of speech of current word and entity mark of second word on the left of current word
POS(0)+Tag(1)	The part of speech of current word and entity mark of first word on the right of current word
Word(0)+Tag(-1)	Current word and entity mark of first word on the left of current word
Word(0)+Tag(-2)	Current word and entity mark of second word on the left of current word
Word(0)+Tag(1)	Current word and entity mark of first word on the right of current word
POS(0)+Tag(-1)+Tag(-2)	The part of speech of current word and entity tags of first word and second word on the left of current word
Tag(-1)+POS(0)+POS(1)	Entity tag of first word on the left of current word and part of speech of current word and part of speech of first word on the right of current word
Tag(-1)+POS(-1)+POS(0)	Entity tag of first word on the left of current word and part of speech of first word on the left of current word and part of speech of current word
Tag(-1)+POS(0)+Word(0)	Entity tag of first word on the left of current word and part of speech of current word and current word
Tag(-2)+Tag(-1)+POS(0)	Entity tags of first word and second word on the left of current word and part of speech of current word

binary function as follows:

$$f(x,y) = \begin{cases} 1 & \text{if Tag(-1) = ``B-nhack'' and Word(0)} \\ & \text{= ``Org'' and y = ``E-nhack''} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

4) SEMANTIC FEATURES TEMPLATE

Many words such as ''teacher'' and ''chairman'' often indicate the appearance of names, and the name recognition is a very important task. It makes up for the inconvenience of expressing the relationship between adjacent words. The basic idea is to recognize demonstrative words and suffixes from dictionaries on the basis of word segmentation. These words need to be filled in manually continuously. Semantic templates are now defined in Table 5.

For example, when identifying the organization name ''sky eye laboratory'', assuming that the current word is ''sky eye'', such a specific feature can be represented by the binary feature function as follows:

$$f(x,y) = \begin{cases} 1 & \text{if Word(1) = ``sky eye'' and} \\ & \text{ORG\_SUFFIX = ``true'' and y = B-norg} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

5) FEATURE SELECTION

The generation of feature sets is accomplished by matching the above feature templates. Next, we perform the process of

TABLE 5. The template of semantic features.

Semantic Features	Description
CUR_PER_FRIST	Whether the current word is name
CUR_ORG_SUF	Whether the current word is an organization name suffix
NEXT_ORG_SUF	Whether the two words on the right side of current word contain organization suffix
LOC_INDICATION	Whether the left or right words of current word contain place indicators
PER_INDICATION	Whether the left or right words of current word contain name indication
ORG_INDICATION	Whether the left or right words of current word contain organization indicator
CUR_LOC	Whether the current word is a common place name
CUR_ORG	Whether the current word is a common organization name
CUR_PER_NAME	Whether the current word is a common name
CUR_LOC +LOC_INDICATION	Whether the current word is a common place name and whether the two words around the current word contain place name indicators
CUR_PER_FRIST +PER_INDICATION	Whether the current word is a Chinese surname and the left and right words contain a person name
Tag(-1)+CUR_ORG_SUF	The first word on the left side of current word is the named entity and the current word is the institutional feature suffix
Tag(-1)+CUR_LOC	The first word on the left side of current word is entity and the current word is the place name.

traversing all words in the corpus in turn to match the words and their contexts with all feature templates. All successfully matched features are added to the feature set. The details of the generation process of feature sets are described in Algorithm 1.

Algorithm 1 The Learning Algorithm for Feature Selection

Require: cybersecurity text corpus $\mathcal{D}$, the library of above four feature templates $\mathcal{T}$

Ensure: feature set $\mathcal{F}$

```
choose a template T from the library of template 𝒯;
read a word w from vocabulary 𝒱 generated by cybersecurity text corpus 𝒟;
while T ∈ 𝒯 do
    while w ∈ 𝒱 do
        match current template T and current word w, and then generate a feature f
        if f ∈ ℱ then
            increment count for f
        else
            add f to ℱ
        end if
    end while
end while
return ℱ
```

Due to a large number of words and the wide variety of feature templates, the number of generated features will be incalculable, and some features have little effect on entity recognition. Instead, these redundant features have seriously affected the efficient of our proposed model, so it is necessary to perform a round of screening of the feature results.

Common feature selection methods are incremental method and threshold method. The former is to calculate the information gain of all features, and retains the features with large information gain of system performance, otherwise deletes. The latter is to count the frequency of each feature. If the frequency of a feature is less than a set threshold, it is deleted, otherwise retained. The incremental method works well but the system performance is expensive. The threshold method is simple to operate, but not intelligent. For simplicity and computational efficiency, we use the threshold method, and the threshold is set to 2.

6) CONDITINAL RANDOM FIELDS MODELING

CRF are a type of discriminative probabilistic graphical model, which often applies in predicting sequences and named entity recognition. It can take into account contextual information from previous labels, thus making a good prediction performance.

In CRF, given the set of input vectors X, y_{i-1} and y_i denote the labels of previous word and current word in X respectively, we define the feature function as $f_i(X, i, y_{i-1}, y_i)$. Each feature function is either 0 or 1 based on the label of previous word and current word. To build the conditional field, we assign each feature function f_i a set of weights λ as follows

$$P(y, X, \lambda) = \frac{1}{Z(X)} exp\{\sum_{i=1}^{n} \sum_{j} \lambda_j f_i(X, i, y_{i-1}, y_i)\} \quad (5)$$

where $Z(X) = \sum_{y' \in y} \sum_{i=1}^{n} \sum_{j} \lambda_j f_i(X, i, y'_{i-1}, y'_i)$. To estimate the parameters λ, we use Maximum Liklihood Estimation to take the negative log of the distribution as

$$\begin{aligned} \mathcal{L} &= -log\{\prod_{k}^{m} P(y^k|x^k, \lambda)\} \\ &= -\sum_{k=1}^{m} log[\frac{exp\{\sum_{i=1}^{n} \sum_{j} \lambda_j f_i(X^m, i, y^k_{i-1}, y^k_i)}{Z(x_m)}\}] \end{aligned} \quad (6)$$

Maximizing log-posterior distribution on Eq. (6) is equivalent to minimizing sum-of-of-squared errors function. The local minimum of the objective function given by Eq.(6) can be found by using gradient descent on parameters λ as

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \lambda} = \frac{-1}{m} &\sum_{k=1}^{m} \sum_{i=1}^{n} f_i(x^k, i, y^k_{i-1}, y^k_i) \\ &+ \sum_{k=1}^{m} p\{y|x^k, \lambda\} f_i(x^k, i, y_{i-1}, y_i) \end{aligned} \quad (7)$$

CRF estimates the global probability, and establishes a unified probability model on all states. Hence, CRF is a relatively good model in named entity recognition.

TABLE 6. Statistics of the constructed dataset.

Class	Number	Class	Number
CVE	68	Product	1402
AS	8	Organization	3047
Cert	10	Person	1372
Host	14	Place	518
Domain	25	Threat	21
Email	17	Hacker_Group	62
MD5	31	Attack	19
Registry	22	Software	427
SHA1	15	Protocol	25
SHA256	18	Conference	14
URL	42	Report	80
IP	24	File_Path	43
File_Name	71	Event	18

IV. EXPERIMENTS

A. DATA PREPARATION

Unlike named entity recognition in the general field, cyber security lacks large-scale publicly available dataset and annotation methods. Therefore, we construct a standard ground truth dataset through the following construction process. First, we collect a large amount of security text corpus from official security forums,[1] software vendors bulletin boards,[2] and various blog articles. Second, we choose a collaborative text annotation system brat,[3] which is an open source Web annotation tool that can annotate a large number of text online. Third, the members of this collaborative annotation using brat tools are domain experts who have rich knowledge of cybersecurity. Each document is annotated by at least three users in turn. The ground-truth class labels are selected based on the majority vote mechanism, Finally, about 14,000 unstructured texts from cyber security domain have been marked, in which the training set consists of 70% of the total documents and the remaining 30% as test set. We use the constructed dataset in the following experiments. The statistics of datasets are summarized in Table 6.

B. BASELINE METHODS

In order to select a model for equilibrium accuracy and performance, we analyze the following models after doing the same rules and dictionary matching preprocessing on our security samples.

- **HMM**: Hidden Markov Model (HMM) is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobservable (i.e. hidden) states. The hidden Markov model can be represented as the simplest dynamic Bayesian network [28].
- **MEMM**: Maximum Entropy Markov Model (MEMM) makes use of both the HMM framework to predict sequence labels given an observation sequence, but incorporating the multinomial Logistic Regression (aka Maximum Entropy), which gives freedom in the type and number of features one can extract from the observation sequence [22].
- **CRF**: Conditional Random Fields (CRF) is a discriminative probabilistic graphical model. It use contextual information from previous labels, thus increasing the amount of information. The model has to make a good prediction [20].

The neural network method has become a major topic in the field of natural language processing (NLP) recently, but its training complexity is often high, generally used to solve complex and high-level tasks, such as machine translation, text understanding and so on. At the expense of certain complexity and computational speed, there are also some researchers use Long short-term memory (LSTM) and their deformation models to extract cybersecurity entities, such as LSTM-CRF [23] and FT-CNN-BiLSTM-CRF [24], and the results proved such models have a certain degree recognition ability on their datasets.

So we also compare the effectiveness of our proposed model with the following state-of-the-art baseline methods on the same dataset.

- **LSTM-CRF**: LSTM is a special recurrent neural network.The advantage of LSTM is to obtain the relationship between the sample and the sample over a long time sequence, and BiLSTM can more effectively acquire the features before and after the input sentence. This model extract features by the LSTM and predict entity types by CRF [23].
- **FT-CNN-BiLSTM-CRF**: In this model, the Convolutional Neural Networks (CNN) is used to extract the character-level feature and the BiLSTM is to capture long-term contextual features. Then CRF is applied for learning and inference. Futhermore, it adds the feature template and extract contextual features of the security entity through feature templates [24].

For HMM, MEMM, and CRF models, we use the default recommended settings. For LSTM-CRF and FT-CNN-BiLSTM-CRF models, we set the word embedding layers to 64, and the word embedding dimensions to 100. Meanwhile, for CNN and LSTM models, we set batch_size to 32, and Dropout to 0.5, and learning rate to 0.01, and gradient to 5 in the following comparison experiments.

C. EVALUATION METRICS

In this paper, we use three representative metrics to evaluate the performance: Precision, Recall, and F1-measure (F1). A greater Precision, Recall, and F1-measure values mean better performance. Without loss of generality, we split randomly with 80% as the training set and 20% as the testing set. We repeat each experiment 5 times and report the average performance.

D. PERFORMANCE AND ANALYSIS

1) THE PERFORMANCE OF CYBERSECURITY ENTITY RECOGNITION

The task of entity recognition is divided into two categories: (1) be or not be a entity, which is a binary classification

[1] http://www.cert.org.cn/
[2] https://www.anquanke.com/
[3] http://brat.nlplab.org/

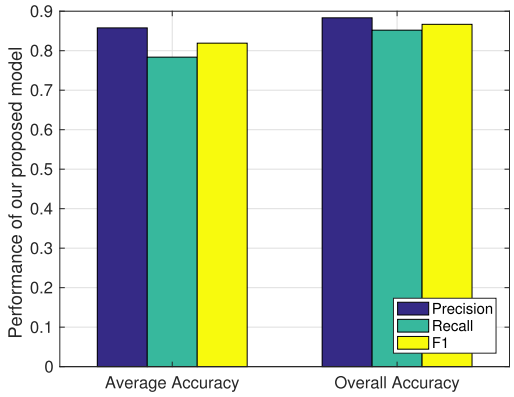


FIGURE 2. Performance of our proposed model on the tasks of average classification and overall classification.

TABLE 7. Performance comparison of different deep recognition models, evaluated by Precision, Recall and F1.

Method	Precision	Recall	F1
LSTM-CRF	0.7945	0.7079	0.7487
FT-CNN-BiLSTM-CRF	0.8157	0.7642	0.7891
RDF-CRF	**0.8578**	**0.7837**	**0.8191**

task; (2) belong to which entity class, which is a multi-classification task. To this end, we conduct extensive experiments with the above two tasks on the cybersecurity dataset. The experimental results are shown in Figure 2 and Table 8.

From the Figure 2, we can see the overall accuracy of whether there is a entity is higher than the average accuracy of entity class recognition. We argue that this phenomenon may be caused by confusion in the process of entity classification, such as Person be classified as Organization, Threat be classified as Hacker_Group, etc. We also can see that the binary classification accuracy is only more 6% than that of multi-classification, which shows our proposed model have good robustness.

On the other hand, from the Table 8, we can also observe the following conclusion that (1) our proposed model has a relatively high performance at most of entity classes; (2) regular-based entities like CVE and Email can be extracted with a highest accuracy, which demonstrates that regular-based extractor is a good strategy; (3) dictionary-based entities such as Product and Organization have a relatively high accuracy, and sometimes the improvements are not statistically significant due to the lack of specific entities; (4) CRF-based extractor obtain poor precision and recall as our dataset contains only a small number of these instances. This problem can be solved given a larger amount of cybersecurity corpus. Hence, by incorporating regular expression, know-entity dictionary and CRF model, our proposed model indeed perform well on the cybersecurity entity recognition task.

2) COMPARISONS WITH THE STATE-OF-THE-ART METHODS

In order to evaluate and compare the effectiveness, we conduct an experiment to compare our method to the latest methods in cybersecurity extraction entities mentioned in the last two years of papers on the same dataset. The first is LSTM-CRF [23], and the second is FT-CNN-BiLSTM-CRF [24]. The comparative experiment results are shown in Table 7.

As we can see, the performance metrics show that the results for RDF-CRF are better than other state-of-the-art methods. Even though the recall score of the FT-CNN-BiLSTM-CRF is close to ours, its precision still have some room for improvement. One of the reasons is that there are a large number of simple but regular entities in cybersecurity texts, such as IP, domain, etc., and the use of complex model methods for these entities will reduce its precision. At the same time, due to the use of neural network for feature extraction, the computational complexity of the model will be greatly increased. The final results prove that in the case of entity pre-matching using rules and dictionaries, the CRF model with feature templates can be used to obtain better recognition results at lower complexity.

3) COMPARISONS OF DIFFERENT RECOGNITION MODELS

In this section, we mainly compares the performance of cybersecurity entity recognition under Hidden Markov Model (HMM), Maximum Entropy Markov Model (MEMM) and Conditional Random Fields (CRF). The main classes of comparison entities are recognized only by statistical model, including Organization, Person, Report, Threat, Event, Conference, Hacker_Group. The experimental results are shown in Figure 3.

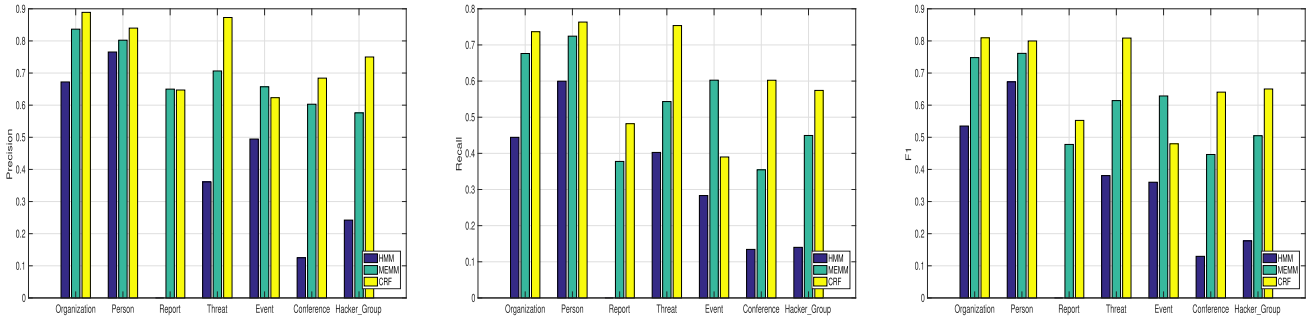
From the figure, the experimental results show that CRF model always outperforms other comparison methods of all metrics. The major reason is that the CRF model can make better use of the sequential state of sentences and its dependence on features, and has the best effect on the named entities recognition in unstructured cybersecurity texts. Through the analysis of the reasons, it is found that for named entity recognition of unstructured cybersecurity texts, each observation value has abundant interacting context features and dependencies. HMM model can choose the best path in the range of its inference sequence, but its independence assumption and no aftereffect restrict the selection of features. MEMM model can improve this problem. However, it only normalizes locally and easily falls into local optimum, which leads to label bias problem. Based on the MEMM, CRF model chooses to normalize all features globally to solve the label bias problem. It also has the ability to express long-distance dependence and overlapping features among elements, and can accommodate arbitrary context information.

4) COMBINATION OF DIFFERENT FEATURE TEMPLATES

The combinations of different feature templates have a great impact on the performance of cybersecurity entity recognition. Therefore, we also implement the different configurations of our proposed model to test the effectiveness of combination of different feature templates. In this paper, we denote atomic features as A, and combination features as C, semantic features as S and marker features as M, respectively. We give the performance of different variants of our

TABLE 8. Performance of our proposed model with Precision, Recall, F1 on different entity classes.

Class	Precision	Recall	F1	Class	Precision	Recall	F1
CVE	1.0000	1.0000	1.0000	Product	0.7579	0.7066	0.7314
AS	1.0000	1.0000	1.0000	Organization	0.8989	0.7366	0.8097
Cert	1.0000	1.0000	1.0000	Person	0.8399	0.7633	0.7998
Host	0.7800	0.8500	0.8135	Place	0.9028	0.8824	0.8925
Domain	0.8225	0.7433	0.7809	Threat	0.8729	0.7536	0.8089
Email	0.8895	0.7965	0.8404	Hacker_Group	0.7500	0.5742	0.6504
MD5	1.0000	1.0000	1.0000	Attack	0.6600	0.5400	0.5940
Registry	0.8901	0.8628	0.8762	Software	0.3396	0.3005	0.3189
SHA1	1.0000	1.0000	1.0000	Protocol	0.8200	0.7800	0.7995
SHA256	1.0000	1.0000	1.0000	Conference	0.6842	0.6023	0.6406
URL	0.9255	0.8700	0.8969	Report	0.6472	0.4821	0.5526
IP	0.9900	0.9900	0.9900	File_Path	0.8936	0.6200	0.7496
File_Name	0.8842	0.8925	0.8883	Event	0.6233	0.3900	0.4798



(a) Precision vs. Entity Classes (b) Recall vs. Entity Classes (c) F1 vs. Entity Classes

FIGURE 3. Precision, Recall and F1 with different entity classes.

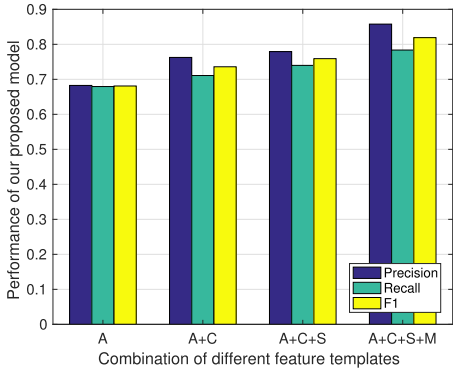


FIGURE 4. Performance of our proposed model with combination of different feature templates.

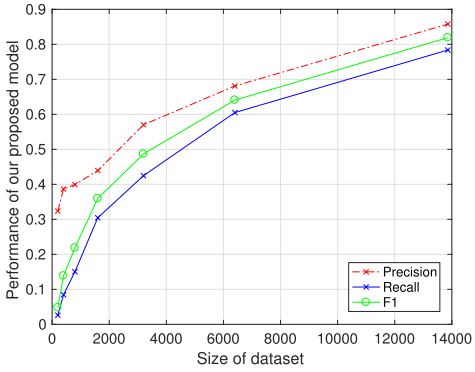


FIGURE 5. Performance of our proposed model under different dataset size.

proposed model in Figure 4. From the results, it is clear that (1) when increasing the amount of combination templates, the performance of our proposed model improves, and the proposed model can achieve best performance by using all feature templates; (2) among variants of our proposed model, the improvements are statistically significant while using marker feature templates; (3) all of these variants have big differences with the degrees of improvements in some cases. From this view, we conclude that our proposed model is a proper choice for improving the performance of cybersecurity entity recognition.

5) IMPACT OF DATASET SIZE

Figure 5 shows the impacts of different dataset sizes on our proposed model. From the figure, we can observe that the size of dataset impacts the results of entity recognition significantly. As the cybersecurity data increases, the recognition accuracy greatly improves, but when the cybersecurity data surpasses a certain threshold, the recognition accuracy become stable with further increase of the size of dataset. This phenomenon coincides with the intuition that our proposed model can efficiently handle different dataset sizes with a significant improvement of recognition power.

V. CONCLUSION

In this paper, we propose a novel security named entity recognition method by incorporating regular expressions, known-entity dictionary and conditional random fields. The proposed model consists of rule-based extractor, dictionary-based extractor and CRF-based extractor. In particular, rule-based extractor is designed to locate specific entities, dictionary-based extractor includes known-entity lists, and CRF-based extractor leverages the identified entities by rule-based and dictionary-based extractors to further improve the recognition performance. In order to verify the effectiveness of our proposed method, we construct a standard ground truth dataset through manually collaborative annotation and perform extensive experiments. The experimental results show that our proposed method can outperform the state-of-the-art baseline methods. In the future work, we will focus on exploring neural network methods to deal with the problem of label imbalance and feature automatic extraction. The results of our work will have a positive effect on the extraction of security knowledge and the construction of knowledge graph.

REFERENCES

[1] S. Mittal, P. K. Das, V. Mulwad, A. Joshi, and T. Finin, "Cyber Twitter: Using Twitter to generate alerts for cybersecurity threats and vulnerabilities," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2016, pp. 860–867.

[2] R. P. Khandpur, T. Ji, S. Jan, G. Wang, C.-T. Lu, and N. Ramakrishnan, "Crowdsourcing cybersecurity: Cyber attack detection using social media," in *Proc. ACM Conf. Inf. Knowl. Manage. (CIKM)*, 2017, pp. 1049–1057.

[3] G. Husari, X. Niu, B. Chu, and E. Al-Shaer, "Using entropy and mutual information to extract threat actions from cyber threat intelligence," in *Proc. IEEE Int. Conf. Intell. Secur. Informat. (ISI)*, Nov. 2018, pp. 1–6.

[4] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition," in *Proc. 7th Conf. Natural Lang. Learn. (HLT-NAACL)*, 2003, pp. 142–147.

[5] A. Ritter, S. Clark, and O. Etzioni, "Named entity recognition in tweets: An experimental study," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2011, pp. 1524–1534.

[6] C. Nogueira dos Santos and V. Guimaraes, "Boosting named entity recognition with neural character embeddings," 2015, *arXiv:1505.05008*. [Online]. Available: http://arxiv.org/abs/1505.05008

[7] T.-H. Pham and P. Le-Hong, "End-to-end recurrent neural network models for vietnamese named entity recognition: Word-level vs. character-level," in *Proc. Int. Conf. Pacific Assoc. Comput. Linguistics*. Springer, 2017, pp. 219–232.

[8] S. M. Yimam, C. Biemann, L. Majnaric, Š. Šabanović, and A. Holzinger, "An adaptive annotation approach for biomedical entity and relation recognition," *Brain Informat.*, vol. 3, no. 3, pp. 157–168, Sep. 2016.

[9] T. Eftimov, B. K. Seljak, and P. Korošec, "A rule-based named-entity recognition method for knowledge extraction of evidence-based dietary recommendations," *PLoS ONE*, vol. 12, no. 6, 2017, Art. no. e0179488.

[10] C. Lee, Y.-G. Hwang, H.-J. Oh, S. Lim, J. Heo, C.-H. Lee, H.-J. Kim, J.-H. Wang, and M.-G. Jang, "Fine-grained named entity recognition using conditional random fields for question answering," in *Proc. Asia Inf. Retr. Symp.* Springer, 2006, pp. 581–587.

[11] M. A. Khalid, V. Jijkoun, and M. De Rijke, "The impact of named entity normalization on information retrieval for question answering," in *Proc. Eur. Conf. Inf. Retr.* Springer, 2008, pp. 705–710.

[12] O. Uzuner, Y. Luo, and P. Szolovits, "Evaluating the state-of-the-srt in automatic de-identification," *J. Amer. Med. Inform. Assoc.*, vol. 14, no. 5, pp. 550–563, Sep. 2007.

[13] M. Krallinger, O. Rabal, F. Leitner, M. Vazquez, D. Salgado, Z. Lu, R. Leaman, Y. Lu, D. Ji, and D. M. Lowe, "The chemdner corpus of chemicals and drugs and its annotation principles," *J. Cheminformat.*, vol. 7, no. 1, p. S2, 2015.

[14] A. Ritter, E. Wright, W. Casey, and T. Mitchell, "Weakly supervised extraction of computer security events from Twitter," in *Proc. 24th Int. Conf. World Wide Web (WWW)*, 2015, pp. 896–905.

[15] V. Mulwad, W. Li, A. Joshi, T. Finin, and K. Viswanathan, "Extracting information about security vulnerabilities from Web text," in *Proc. IEEE/WIC/ACM Int. Conferences Web Intell. Intell. Agent Technol.*, Aug. 2011, pp. 257–260.

[16] I. Deliu, C. Leichter, and K. Franke, "Extracting cyber threat intelligence from hacker forums: Support vector machines versus convolutional neural networks," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 3648–3656.

[17] X. Liao, K. Yuan, X. Wang, Z. Li, L. Xing, and R. Beyah, "Acing the IOC game: Toward automatic discovery and analysis of open-source cyber threat intelligence," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS)*, 2016, pp. 755–766.

[18] M. Balduccini, S. Kushner, and J. Speck, "Ontology-driven data semantics discovery for cyber-security," in *Proc. Int. Symp. Practical Aspects Declarative Lang.* Springer, 2015, pp. 1–16.

[19] A. Joshi, R. Lal, T. Finin, and A. Joshi, "Extracting cybersecurity related linked data from text," in *Proc. IEEE 7th Int. Conf. Semantic Comput.*, Sep. 2013, pp. 252–259.

[20] R. Lal, "Information extraction of security related entities and concepts from unstructured text," Tech. Rep., 2013.

[21] C. L. Jones, R. A. Bridges, K. M. T. Huffer, and J. R. Goodall, "Towards a relation extraction framework for cyber-security concepts," in *Proc. 10th Annu. Cyber Inf. Secur. Res. Conf. (CISR)*, 2015, p. 11.

[22] R. A. Bridges, C. L. Jones, M. D. Iannacone, K. M. Testa, and J. R. Goodall, "Automatic labeling for entity extraction in cyber security," 2013, *arXiv:1308.4941*. [Online]. Available: http://arxiv.org/abs/1308.4941

[23] H. Gasmi, A. Bouras, and J. Laval, "Lstm recurrent neural networks for cybersecurity named entity recognition," in *Proc. 13th Int. Conf. Softw. Eng. Adv.*, 2018, pp. 1–6.

[24] Y. Qin, G.-W. Shen, W.-B. Zhao, Y.-P. Chen, M. Yu, and X. Jin, "A network security entity recognition method based on feature template and CNN-BiLSTM-CRF," *Frontiers Inf. Technol. Electron. Eng.*, vol. 20, no. 6, pp. 872–884, Jun. 2019.

[25] X. Dongliang, P. Jingchang, and W. Bailing, "Multiple kernels learning-based biological entity relationship extraction method," *J. Biomed. Semantics*, vol. 8, no. S1, p. 38, Sep. 2017.

[26] L. Obrst, P. Chase, and R. Markeloff, "Developing an ontology of the cyber security domain," in *Proc. 7th Int. Conf. Semantic Technol. Intell., Defense, Secur.*, 2012, pp. 49–56.

[27] S. Weerawardhana, S. Mukherjee, I. Ray, and A. Howe, "Automated extraction of vulnerability information for home computer security," in *Proc. Int. Symp. Found. Pract. Secur.* Springer, 2014, pp. 356–366.

[28] D. R. H. Miller, T. Leek, and R. M. Schwartz, "A hidden Markov model information retrieval system," in *Proc. 22nd Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 1999, pp. 214–221.

FENG YI received the M.S. and Ph.D. degrees from the University of Chinese Academy of Sciences, China. He is currently an Assistant Professor with the School of Computer Science, Zhongshan Institute, University of Electronic Science and Technology of China. His main research interests include data mining and machine learning.

BO JIANG received the Ph.D. degree in computer science from the University of Chinese Academy of Sciences, China. He is currently an Associate Researcher with the Institute of Information Engineering, Chinese Academy of Sciences, China. His main research interests include data mining, knowledge graph, and recommendation systems. He has served as a Reviewer for the IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS and Elsevier.

LU WANG received the B.S. degree in computer science from the Hebei University of Technology. She is currently pursuing the M.S. degree with the Institute of Information Engineering, Chinese Academy of Sciences. Her main research interests include cyber security situational awareness, knowledge graph, and graph database mining.

JIANJUN WU received the Ph.D. degree in computer science from the University of Chinese Academy of Sciences, China. He is currently an Associate Professor with the Computer Science College, Beijing College of Politics and Law. His main research interests include social network analysis, data mining, and machine learning.