Webpage Tampering Detection Method Based on BiGRU-CRF-RCNN

Xiangyu Fan, Jilong Yang, Wei Zhao, Jincheng Deng(✉), and Fangming Liu

Beijing Knownsec Information Technology Co.,Ltd, Beijing 100097, China
{fanxy,yangjl,zhaow,dengjc,liufm}@knownsec.com

Abstract. With the development of the Internet, cyber security events occur frequently, especially webpage tampering events account for a high proportion. In response to this phenomenon, this paper constructs a webpage tampering detection framework BCR. Based on the webpage to be detected, the webpage text data is segmented and extracted according to the webpage structure, the text features are extracted by using BiGRU model combined with context dependence, and then combined with the CRF to learn sequence state labeling named entities, the word vector is constructed by the extracted named entity and brought into the RCNN model for tampering detection. The experiment results show that the framework has achieved 95.37% precision, 95.35% recall and 95.34% F1-Score in webpage tampering detection, which is better than Textrank RCNN framework in webpage tampering detection. In practical application, it also achieved 95.13% precision and 93.25% recall.

Keywords: Webpage tampering · Named entity recognition · Text classification · Bidirectional gated cyclic unit network · Conditional random field

1 Introduction

With the rapid development of the Internet, various cyber security incidents continue to occur, among which the proportion of webpage tampering events has always been high. How to quickly and accurately locate the tampered content in the webpage and rectify it in time is of great significance to reducing the loss of the site.

At this stage, NLP technology is developing rapidly, text classification technology has a wide range of applications in various fields, and named entity recognition technology is becoming more and more mature. This paper is based on the named entity model to extract the named entities of the text in the webpage segment by segment, and then combined with the text classification model to identify the tampered text.

2 Research Status

At present, the commonly used webpage tampering detection methods are mainly through image recognition and comparison and rule-based detection. Yan Yufeng and

W. Lu et al. (Eds.): CNCERT 2022, CCIS 1699, pp. 113–126, 2022.
https://doi.org/10.1007/978-981-19-8285-9_8

Shen Yong [1] proposed to capture the original image and real-time image of the webpage and detect the feature point information in the before and after images according to the image processing model, and calculate the similarity of webpages according to the feature point information to determine whether the webpage has been tampered with. This method has a good application in the detection of webpage tampering with relatively fixed content or low content update frequency, however, in the case of webpage tampering detection with high content update frequency and rich content, it will affect the model efficiency and detection accuracy. Hongwei R et al. [2] proposed to classify webpage attributes according to principal component analysis, and introduce corresponding rules for each category to realize the judgment of webpage tampering. This method has better effect and efficiency in the scenario of simple webpage structure, but the recognition accuracy will be affected when the web page attributes are complex and the rules cannot cover new objects.

Named entity recognition is a popular research direction of NLP, and named entity recognition models have very good applications in big data research in many fields. The early named entity recognition mainly used the method of building a dictionary, which required a lot of labor costs. After continuous optimization and iteration, today's named entity recognition model mainly relies on various machine learning algorithms to achieve. In the field of named entity recognition in cyber security, Chiu J et al. [3] proposed a method of combining BiLSTM-CNN to build a dictionary in a neural network to encode some words and then match them, this method has better F1-Scrore than other methods on open source datasets. Fan Xiaoxia et al. [4] proposed a method of constructing a named entity recognition system (DNER) for darknet market text based on Branwen's open source darknet market data text using CBOW-CNN-BiLSTM-CRF. Of entity types, the system can significantly improve recognition. Yi F et al. [6] proposed a named entity recognition model based on regular expressions, entity dictionary, CRF combined with feature templates after considering the particularity and complexity of security entities, got good results.

3 Research Content and Methods

It can be seen from the above that most of the detection of webpage tampering, the final data carrier is text data, how to extract effective and well-characterized key words from the text data plays a decisive role in webpage tampering detection. Different webpages have different text complexity, there is often more noise text data in complex text, and the structure of complex text is more complex than simple text, which has a great impact on the extraction of key words with effective features. In view of the interference of complex text data, this paper designs and implements a framework that extracts text data segment by segment according to the structure of webpages, and then uses named entity model to extract named entities to construct text vectors and bring them into the text classification model for webpage tampering detection, including: Data Preprocessing Framework, BiGRU-CRF Named Entity Recognition Model, RCNN text classification model.

3.1 Data Sources

The experimental data in this paper comes from the historical data of webpage tampering monitoring in the threat intelligence data of Knownsec Security Intelligence Brain. The data is HTML text data, involving five types of websites of government, universities, hospitals, transportation, and energy. It contains 20,000 untampered webpage data and 10,000 tampered webpage data. The tampered content involves pornography, gambling, novels, tripartite movie website, tripartite investment website and reactionary information.

3.2 Data Preprocessing

According to the above content and method, the original data is firstly extracted in segments according to the structure of the webpage, and then perform manual labeling and stop word filtering on the extracted data.

Data Extraction. 1) Parse the HTML data. 2) Build a DOM tree. 3) Traverse the DOM tree to find the tag where the required text is located. 4) Extract the text data segmented based on the webpage structure from the returned HTML data according to the tag.

Data Labeling

1) Named Entity Labeling
   This paper uses the word segmentation tool Jieba to perform word segmentation and part-of-speech tagging on the text data. Since named entities are derived from nouns, data labeling is based on the nouns after word segmentation. According to the tampering content of the webpage, a total of 5 types of entity types are labeled, including: PER (person), ORG (company/organization), PLF (platform), OBJ (special noun), 0 (irrelevant word), to ensure that each segment corresponds to one Named Entity Labeling to serve as the data basis for subsequent model building.
2) Text Classification Labeling
   According to whether it has been tampered or not, the text category is labeled as 0 (not tampered) and 1 (tampered).
3) Label the page to which the text belongs
   Use each webpage domain name as the source label of segmented text data to facilitate subsequent positioning.

Stop Word Filtering. Build a stop word database, including: webpage navigation vocabulary, website copyright statement vocabulary, common auxiliary words, special symbols, etc.

3.3 Text Vectorization

Use word2vec to build text vectors. Word2vec has two models of CBOW and SKIP-GRAM in building text vectors. The CBOW model predicts the central word according to the context of the input text, and the SKIP-GRAM model predicts the context according to the central word. Based on the research background, this paper adopts the CBOW model to construct text vectors.

3.4 BiGRU Model

In the field of named entity recognition, the LSTM model has a wide range of applications. In the LSTM model, a single module consists of three gate units: input gate, forget gate, and output gate. The input gate determines the necessary information to retain, the forget gate determines to discard the information, and the output gate shows the final result. In the GRU network, the three gating units of the LSTM model are replaced by the update gate and the reset gate. The update gate determines the amount of attention information, and the reset gate determines the amount of forgotten information. The reduction of gating units also reduces the parameters in the network, making GRU more concise and efficient than LSTM. BiGRU is a neural network model composed of two unidirectional and opposite GRUs, The current hidden layer state of BiGRU is jointly determined by the current input X_t, the forward hidden layer state $h_{t-1}^{\rightarrow}$ at time $t-1$, and the backward hidden layer state $h_{t-1}^{\leftarrow}$ at time $t-1$. The state of the hidden layer at time t:

$$h_t^{\rightarrow} = G\left(X_t, h_{t-1}^{\rightarrow}\right) \tag{1}$$

$$h_t^{\leftarrow} = G\left(X_t, h_{t-1}^{\leftarrow}\right) \tag{2}$$

$$h_t = \omega_t h_t^{\rightarrow} + \vartheta_t h_t^{\leftarrow} + b_t \tag{3}$$

The function $G()$ is a nonlinear transformation of the input word vector, encoding the word vector at this moment into the corresponding hidden layer state, ω_t and ϑ_t respectively represent the weights corresponding to $h_t^{\rightarrow}$ and $h_t^{\leftarrow}$ at time t, and b_t represents the corresponding bias. Its structure diagram is shown in Fig. 1:

3.5 CRF Model

The Conditional Random Field (CRF) model is a special Markov random field. It is assumed that there are only observation values X and state values Y in the model. In the CRF model, each state value Y_n is only related to its adjacent state value, and its observation value X_n is not has Markov properties. The CRF model needs to consider the correlation between the output state values. The feature function ∂ can be used to learn the relationship between states. The CRF will output a sequence score, and normalize all sequence scores to find the path with the highest probability as the prediction sequence. The CRF model includes state feature function ∂ and state transition function μ.

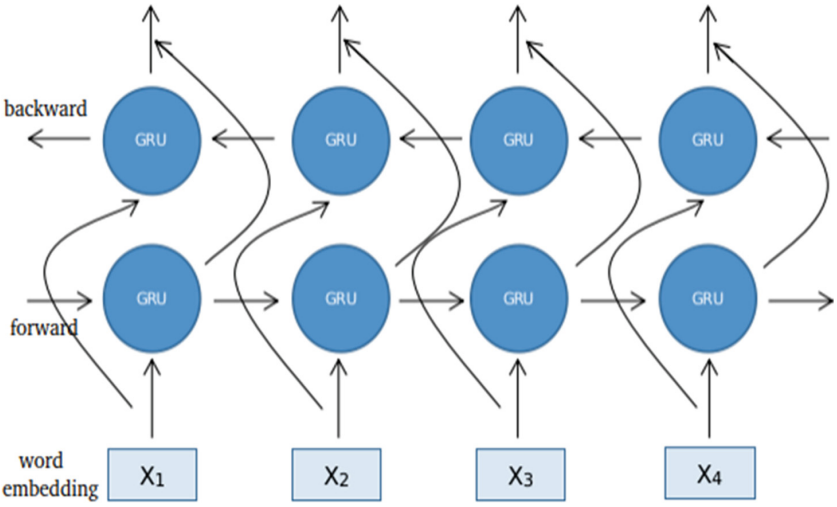

Fig. 1. BiGRU model structure diagram

State Feature Function. Only related to the current node, ϑ represents the current weight of the feature function, that is:$\vartheta\,\partial(Y_i, X_i)$.

State Transition Function. Related to both node $i+1$ and node $i-1$, ω represents the current weight of the transfer function, that is:$\omega\mu(Y_{i+1}, Y_{i-1}, Y_i, X_i)$.

Suppose there are state feature functions $\partial_1, \partial_2,\ldots, \partial_L$ whose weights are $\vartheta_1, \vartheta_2,\ldots, \vartheta_L$, and transition state feature functions $\mu_1, \mu_2,\ldots, \mu_K$, whose weights are $\omega_1, \omega_2,\ldots,\omega_L$, for the sequence $X = \{X_1,X_2,\ldots,X_n\}$, the probability of the output sequence Y can be calculated as:

$$P(Y|X) = \frac{1}{Z(X)} exp\left(\sum \vartheta_L \partial_L(Y_i, X_i) + \sum \omega_K \mu_K(Y_{i+1}, Y_{i-1}, Y_i, X_i)\right) \tag{4}$$

of which:

$$Z(X) = \sum exp\left(\sum \vartheta_L \partial_L(Y_i, X_i) + \sum \omega_K \mu_K(Y_{i+1}, Y_{i-1}, Y_i, X_i)\right) \tag{5}$$

$Z(X)$ is the generalization factor, which can be seen as the sum of the scores of all output sequences.

When the transition feature and state feature are represented by unified functions s and f, the probability of the output sequence Y is:

$$P(Y|X) = \frac{1}{Z(X)} exp \sum s_i f_i(Y, X) \tag{6}$$

of which:

$$Z(X) = \sum exp \sum s_i f_i(Y, X) \tag{7}$$

When the CRF model is used for named entity recognition, its graph structure is shown in Fig. 2:

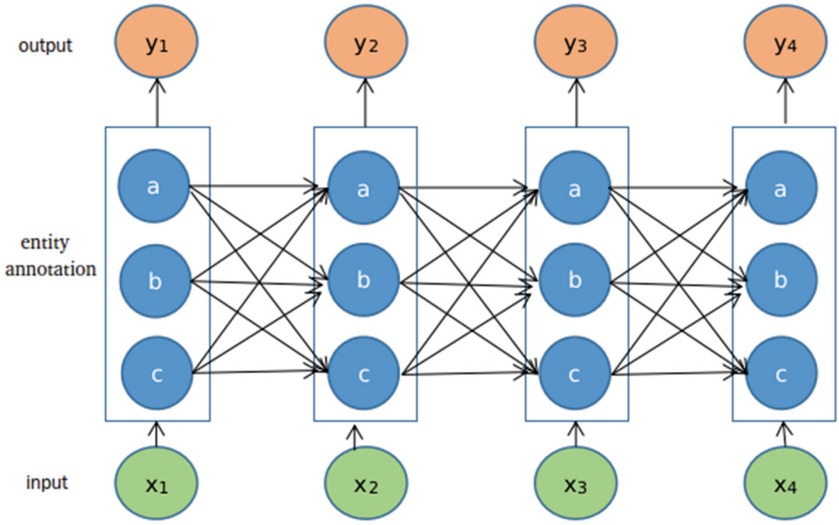

Fig. 2. CRF model structure diagram

3.6 RCNN Model

The RCNN model is a commonly used text classification model, and its structure is divided into three parts.

Region-CNN Model. A bidirectional RNN model is used to obtain the context information of each word embedding, and its expression is:

$$c_l(w_i) = f\left(W_{(l)}c_l(w_{i-1}) + W_{(sl)}e(w_{i-1})\right) \tag{8}$$

$$c_r(w_i) = f\left(W_{(r)}c_r(w_{i+1}) + W_{(sr)}e(w_{i+1})\right) \tag{9}$$

of which:

$c_l(w_i)$ represents the above of the word w_i.

$c_r(w_i)$ represents the context of the word w_i.

$e(w_i)$ represents the embedding vector of word w_i.

$W_{(l)}$ and $W_{(r)}$ are weight matrices, which transfer the above and below of the previous word to the above and below of the next word.

$W_{(sl)}$ and $W_{(sr)}$ are feature matrices, which combine the semantic features of the current word to the upper and lower parts of the next word.

Computing Hidden Semantic Vectors. The context information obtained in the previous step is merged with the expanded word embedding information, and the activation function is used to calculate the hidden semantic feature vector of the word w_i. Expanded word embedding information is:

$$X_i = [c_l(w_i); e(w_i); c_r(w_i)] \tag{10}$$

Hidden semantic vector is:

$$Y_i^{(2)} = tanh\left(W^{(2)}X_i + b_{(2)}\right) \tag{11}$$

Continuous Learning, Output Results. After continuous learning of TextCNN, max-pooling and fully connected layers, the classification result is obtained.

The structure diagram of the RCNN model is shown in Fig. 3:

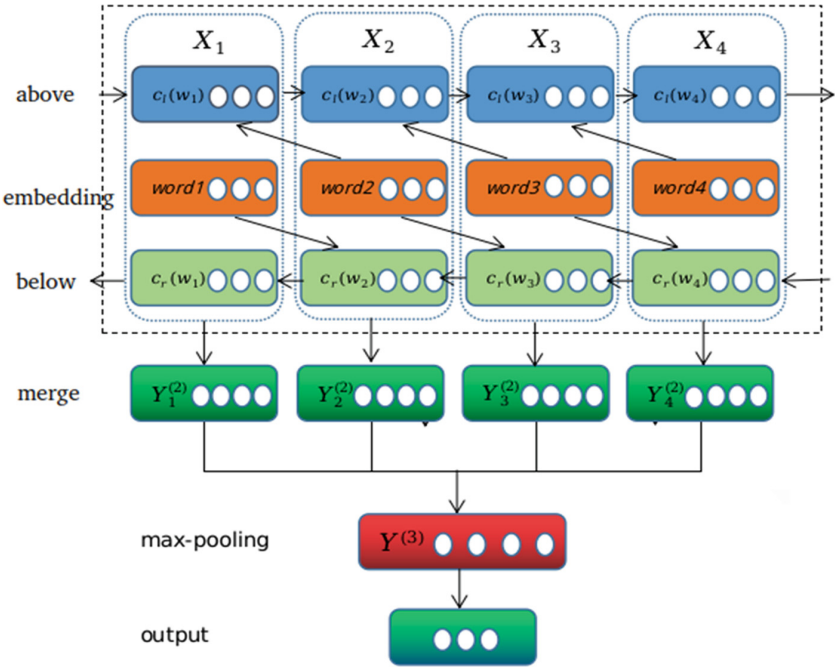

Fig. 3. RCNN model structure diagram

4 Experiment and Result Analysis

4.1 Experimental Environment and Evaluation Indicators

This experiment was performed in the following configuration:

In this experiment, both the named entity model and the text classification model use the precision rate (PRE), the recall rate (REC), and the comprehensive evaluation (F1-Score) as the model's accuracy evaluation indicators.

4.2 Experimental Configuration

Named Entity Recognition. The 30,000 pieces of data after data preprocessing are divided into training set, test set and validation set according to the ratio of 6:2:2. The distribution of the data set is as follows:

In order to verify that the framework proposed in this paper is better, BiGRU-CRF model, BiLSTM-CRF model, and CNN-LSTM model are set up as comparison models. The three comparison model structures are shown in Table 3 (Tables 1 and 2):

Table 1. Configuration table.

Software and hardware	Configuration
CPU	i7-6700HQ @2.6 GHz
GPU	GTX 970 m
Memory	16 GB
Operating System	Deepin 20.5 GNU/Linux

Table 2. Named entity dataset partitioning.

Data set	Quantity (bar)
Training set	18000
Test set	6000
Validation set	6000

Table 3. Named entity vs model structure.

	BiGRU-CRF	BiLSTM-CRF	CNN-LSTM
Layer1	Input	Input	Input
Layer2	Embedding	Embedding	Embedding
Layer3	bgru	blstm	conv
Layer4	dense	dense	lstm
Layer5	crf_dense	crf_dense	dropout
Layer6	crf	crf	time_distributed
Layer7	–	–	activation

The main parameter configuration of each model is shown in Table 4 (Table 5):

Text Categorization. The 30,000 pieces of data after data preprocessing are divided into training set, test set and validation set according to 6:2:2. The distribution of the data set is as follows:

Table 4. Parameter configuration.

Model	Layer	Epoch	Batch_size	Active
BiGRU-CRF	bgru	30	256	tanh
BiLSTM-CRF	blstm	30	256	tanh
CNN-LSTM	lstm	30	256	softmax

Table 5. Text classification dataset partitioning.

Data set	Quantity (bar)
Training set	18000
Test set	6000
Validation set	6000

Use two methods to build word vectors and then bring them into the RCNN model for comparison. They are: Named entities combined with RCNN model for classification, Text summarization combined with RCNN model for classification.The RCNN model epoch is set to 30, batch_size is set to 256, and the training process is shown in Table 6:

Table 6. RCNN model training process

Layer	Output shape	Active
input	(None, 12)	–
layer_embedding	(None, 12, 100)	–
layer_conv1d	(None, 8, 128)	relu
layer_max_pooling	(None, 128)	–
layer_dense	(None, 64)	relu
dense_1	(None, 2)	softmax

4.3 Experimental Results and Analysis

Named Entity Recognition. The accuracy indicators of each model are shown in Fig. 4:

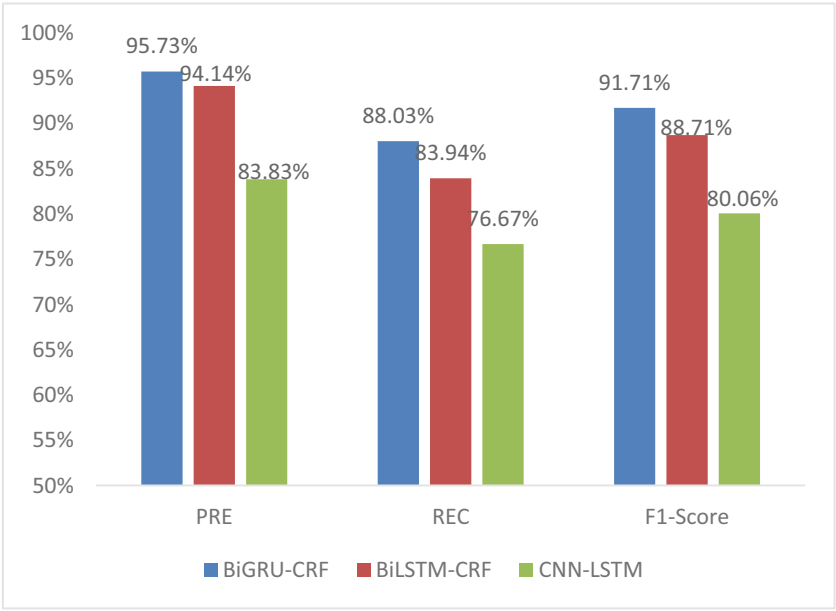

Fig. 4. Accuracy indicators of each named entity model

In terms of recognition accuracy, the PRE, REC, and F1-Score of the BiGRU-CRF model in this scenario are 93.88%, 91.36%, and 92.60% respectively, which is a certain improvement compared to the other two models. The main reason is that the data set is based on segmented text data after webpage structure segmentation, and the BiGRU-CRF model has improved and optimized the gate control unit compared with the BiLSTM-CRF model, and has better applications in simple text data. Both BiGRU-CRF model and BiLSTM-CRF model can encode text information from front to back and from back to front, which can better capture bidirectional text semantic dependencies, while CNN-LSTM model cannot encode text information from back to front, It can only capture one-way text semantic dependencies, so it is lower than the other two models in terms of accuracy.

Figure 5, Fig. 6, and Fig. 7 show the evaluation indicators of each category of named entity recognition accuracy of each model:

Compared with the other two models, the BiGRU-CRF model has obvious advantages in PLF named entity recognition, and is comparable to the BiLSTM-CRF model in other types of named entity recognition. The CNN-LSTM model is far behind the other two models in terms of OBJ and PLF named entity recognition. From the comprehensive view of the above radar charts, BiGRU-CRF is relatively better in named entity recognition in this scenario.

Text Categorization. The accuracy evaluation indicators of each model are shown in Fig. 8:

Compared with TextRank-RCNN, BiGRU-CRF-RCNN has a certain improvement in precision, recall and F1-Score. The main reason is that BCR framework extracts

Fig. 5. The precision of each model for each type of named entity recognition

Fig. 6. The recall of each model for each type of named entity recognition

keywords representing text based on the characteristics of BiGRU-CRF model. Entities can better represent the domain features and context features of the current text. While the TextRank-RCNN framework constructs a network based on the relationship between local adjacent nodes when extracting keywords representing text The mechanism of exclusive nouns, the extracted information features are not comprehensive, so the accuracy of tampering identification is relatively poor.

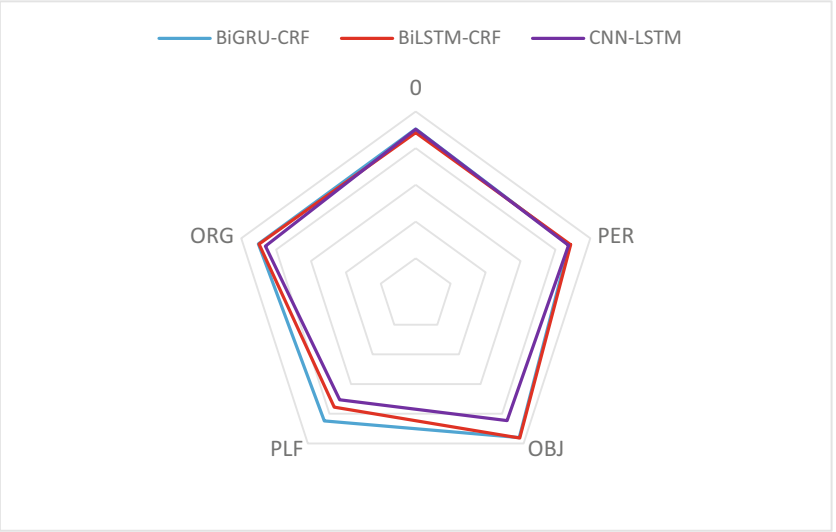

Fig. 7. Each model recognizes the F1-Score for each type of named entity

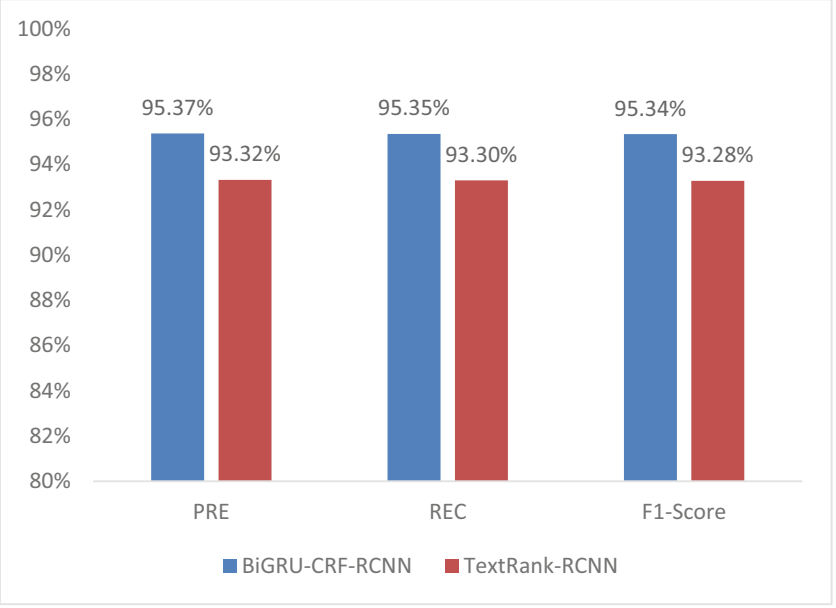

Fig. 8. The accuracy index of each text classification model

4.4 Practical Application

This framework has been applied in Knownsec Security Intelligence Brain. From the test results, an average of 108,326 webpages are detected every day, and an average of

411 tampered webpages are identified every day. After manual sampling by the sampling team, the sampling precision was 95.13%, and the recall was 93.25%.

4.5 Conclusion

At this stage, named entities and text classification technology have been widely used in the field of cyber security, but less in webpage tampering detection. Therefore, the BiGRU-CRF-RCNN framework is proposed for webpage tampering detection. According to the above experimental process and practical application effect, we can get:

Advantages of this Framework. Due to the structural characteristics of the gated unit of the BiGRU-CRF model, it has a better application than other models in this scenario. In terms of text classification, the named entities extracted based on the named entity model can better reflect the characteristics of the current field. Therefore, in the scenario of this paper, using the text vector constructed based on named entities for text classification has a better effect.

Weaknesses of the Framework. The BiGRU-CRF-RCNN model achieves better results because the industry content of the website detected in production and experiments is less related to the tampered content. Considering the problem of model generalization, if the data surface is widened, and the positive samples and negative samples are related, it needs to be improved according to the actual effect.

References

1. Yan, Y., Shen, Y.: Web page tamper detection based on image processing. Comput. Digit. Eng. **48**(6), 5 (2020)
2. Hongwei, R., Liu, T., Hua, L., et al.: Webpage tamper detection based on principal component analysis. China Sci. Pap. (2012)
3. Chiu, J., Nichols, E.: Named entity recognition with bidirectional LSTM-CNNs. Comput. Sci. **4**, 357-370 (2015)
4. Fan, X., Zhou, A., Zheng, R., et al.: Darknet market named entity recognition based on deep learning. J. Inf. Secur. Res. (2021)
5. Qin, Y., Shen, G., Zhao, W., et al.: Research on the method of network security entity recognition based on deep neural network. J. Nanjing Univ. Natl. Sci. **55**(1), 12 (2019)
6. Yi, F., Jiang, B., Wang, L., et al.: Cybersecurity named entity recognition using multi-modal ensemble learning. IEEE Access, (99), 1–1 (2020)
7. Li, K.-Y., Liu, X.-D.: Web tampering detection system based on feature recognition. Electron. Des. Eng. **28**(440)(18), 22–25+30 (2020)