International Journal for Housing Science and Its Applications
Publish August 4, 2025. Volume 46, Issue 3 Pages 1961-1970

https://doi.org/10.70517/ijhsa463155

Research on Computerized Detection Algorithm of Intelligent Fraudulent Information for Network Text Frauds

Xue Wang[1,*] and Zhengdao Li[1]
[1] Anhui Police College, Hefei, Anhui, 238076, China
Corresponding authors: (e-mail: 754970696@qq.com).

Abstract In recent years, network fraud as a new form of fraud, to the social and economic development and the safety of citizens' property has caused a greater threat. This paper proposes a computerized detection technique for network text fraud by design, so as to reduce the success rate of network fraud. This paper classifies the text information of network fraud, and improves the text recognition method through the method of knowledge distillation, and constructs a lightweight fraud text recognition model based on just distillation. Through performance test experiments, detect the utility of this paper's method to detect fraudulent text. The F0.5 mean values of this paper's lightweight fraudulent text recognition based on knowledge distillation are 0.72, 0.67, 0.73 on the online fraudulent text training set, validation set and test set, respectively, which are significantly better than other detection models. The accuracy of this paper's method on fraudulent text classification is greater than 0.8, which clearly outperforms other text classification models. All in all, the method in this paper comes out on top in both classification and detection of online fraudulent text, with better results.

Index Terms online text fraud, text classification, text recognition, knowledge distillation, fraudulent information detection

I. Introduction

With the rapid development of the Internet and the widespread popularization of smartphones, the ways and types of information people obtain have become more and more diversified [1]. The Internet brings convenience to people, but also creates opportunities for lawbreakers, who use the Internet to mix harmful content in the massive information, bringing great harm to people's property safety and social stability and harmony [2]-[4]. Due to the covert and variable nature of fraudulent activities, criminals often change their tactics and adopt different camouflage methods and decoys, which makes it very difficult to identify and combat fraudulent behavior [5], [6]. For the majority of Internet users, timely and accurate identification and prevention of fraudulent information has become a key measure to protect personal and social security [7], [8]. Automated fraudulent information recognition technology can quickly and accurately find fraudulent information and issue timely warnings, effectively stopping fraud, ensuring people's property and personal safety, maintaining economic and social harmony and stability, is of great significance, is an indispensable part of China's anti-fraud action [9]-[12].

In the field of telecommunication network fraud governance, artificial intelligence technology has continued to promote the improvement of governance capabilities, especially the emergence of large-scale pre-trained language models, which have an increasing accuracy in recognizing conventional fraudulent information, thus enabling it to be intercepted in the transmission process [13]-[15]. Some organizations and enterprises actively use big data analysis, machine learning and other artificial intelligence technologies to carry out fraud governance and risk prevention and control, with higher technical recognition accuracy, stronger monitoring and interception in real time, and greater protection coverage, effectively reducing fraud risks and hazards [16]-[18]. Many scholars have conducted profound and extensive research on how to quickly and accurately identify the fraudulent information contained in the text. Literature [19] developed an enhanced convolutional neural network-based SMS phishing detection framework, which extracts relevant features from telecom fraud datasets so as to accurately and efficiently classify fraudulent messages in telecom networks. Literature [20] addresses the unstructured nature of phishing text data as well as the nonlinear complex correlation characteristics of phishing text data, and proposes to automate the detection of robust features of phishing text messages by utilizing a hybrid deep learning framework that incorporates multiple detection models in order to improve the detection of unstructured phishing text messages with complex patterns. Literature [21] establishes a two-tier architecture for web-based text message classification based on deep learning techniques, which is capable of detecting and classifying spam and phishing emails received by users with minimal errors, greatly improving the security of email communication

systems. Literature [22] uses natural processing techniques to extract relevant features in email phishing messages and feeds them into the constructed detection model based on deep learning techniques for training and testing, which shows high performance in terms of recognition accuracy.

In addition, with the continuous confrontation and upgrading of scam texts and interception systems, some researchers have already analyzed and processed the deliberately created misspellings in scam texts in order to improve the recognition accuracy. Literature [23] shows that the large number of misspellings, acronyms, and slang contained in SMS texts can have an impact on the recognition effectiveness of traditional classifiers, for which a combination of augmented and stacked Plain Bayesian Classification Migration Learning method is designed to achieve accurate text message detection by migrating the knowledge of the source domain. Literature [24] designed a cueing and spelling based detection and recognition model (PSC-BERT) to extend the BERT model by integrating semantic, phonetic, and graphemic information of fraudulent messages, and subsequently introduced cueing and spelling learning methods to improve the accuracy of text classification and detection tasks. Based on this, it is of great significance to automate the recognition of fraudulent messages, as well as to automate the detection and correction of the phenomena such as the substitution of misspellings in fraudulent messages.

In order to realize the accurate detection of network text fraudulent information, this paper first classifies the network fraudulent text, and forms the network fraudulent text classification model by prompting the method of comparing small sample text classification for task definition, comparing and fine-tuning sentence encoder and classification head training and other operations. On the basis of the classification of network text fraudulent information, the gating network and backbone network of the PSC-BERT model are knowledge distillation, and a lightweight fraudulent text recognition model based on just distillation is constructed, which is put into the detection of intelligent fraudulent information. In order to test the actual efficacy of the network text fraudulent information detection model in this paper, comparative experiments are conducted on the fraudulent information detection performance of this paper's method and the classification effect of fraudulent text to verify the superiority of this paper's method.

II. Classification of Internet fraud texts

II. A. Definition of tasks

The web fraud text classification task can be defined as follows: given a fraud text consisting of n characters $x = \{x_1, x_2, ..., x_n\}$ (where x_i denotes the i-th character in the text), and Eq. (1) is used to determine the category of fraud to which the text belongs to $\hat{y}$ (the set of categories is Y):

$$\hat{y} = \arg\max_{y \in Y} P(y \mid x) \tag{1}$$

In this study, we propose a small-sample text classification method based on cue template and contrast learning, which is composed of three main parts: cue template insertion, contrast fine-tuning sentence encoder, and classification head training. The overall structure of the method is shown in Fig. 1.

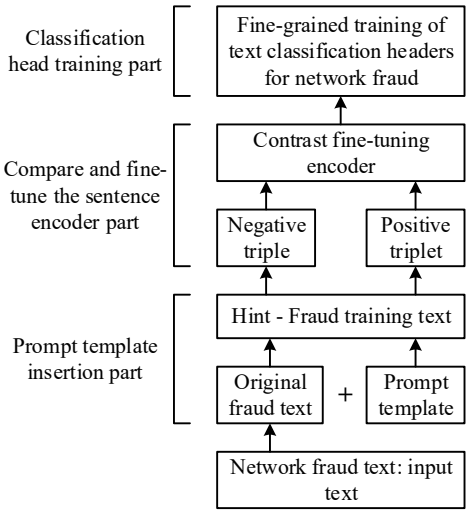

Figure 1: The flowchart of the text classification method of prompt contrast small sample

II. B. Cue Template Insertion

In domain-specific small-sample tasks, manually designing fixed templates for hard template cueing learning usually achieves good cueing results. Since the categorization dataset in this study is extremely purposeful and the

samples after adding the cueing templates need to be adapted to the subsequent comparison fine-tuning of the sentence encoder part, the hard template cueing method is used in this study. By constructing a cue template, the cue part is spliced with the sentence to be categorized so that the final input can form a more semantically complete sentence.

In this study, in order to enhance the model's understanding of online fraud text classification, a cue template was carefully designed based on the task purpose and the characteristics of the original sentence sample: 'This is - a fraud-related text of type [MASK]'. The purpose of this template is to direct the model's attention to the categorization attributes of the scam text, making it easier to identify which type of scam the text belongs to. This form was chosen because it is succinct and fits naturally into the original text while retaining enough contextual information for the model to understand. The original Internet scam text was spliced with the hint template to form the form “Internet scam text + this is - a [MASK] type of scam text” to enhance the model's ability to understand the text's features.

II. C. Contrastive fine-tuning sentence encoder

The core task of the contrast learning fine-tuned sentence encoder part is to construct positive and negative ternary sample pairs based on the internet fraud text categories [25], so that the sentence encoder can fully learn the features among the samples and encode the input text efficiently.

After the raw data samples are completed with cue splicing, in order to utilize the limited samples as efficiently as possible, this study adopts a comparison training method commonly used for image similarity. Formally, suppose a set of samples D where there are a total of K sentence samples spliced by cue templates is given, which can be formalized as:

$$D = \{(x_i, y_i)\} \tag{2}$$

$$C = \{(y_i)\} \tag{3}$$

where x_i and y_i are the sentences and their category labels after splicing by the cued template, respectively.

For each category label $c \in C$ in D, two spliced sentence samples x_i and x_j are randomly selected from sentences belonging to the same category c, where $y_i = y_j = c$, and x_i and x_j form a pair of positive ternary sample pairs and are labeled 1. Each category label c generates R pairs of positive ternary samples, denoted as T_p^c, and this process can be formalized as:

$$T_p^c = \{(x_i, x_j, 1)\} \tag{4}$$

Similarly, for each category label $c \in C$ in D, a spliced sentence sample x_i from category c and a spliced sentence sample x_j that does not belong to that category are selected, where ($y_i = c, y_j \neq c$), x_i and x_j form a pair of negative triad sample pairs and are labeled 0. Each category label c generates R pairs of negative triad samples, denoted as T_n^c, and this process can be formalized as:

$$T_n^c = \{(x_i, x_j, 0)\} \tag{5}$$

Finally, the positive and negative triples generated from all the category labels are combined to produce the final dataset for comparison fine-tuning of the sentence encoder, T. For each category c, there is a set of positive examples T_p^c and a set of negative examples T_n^c. T can be formalized as:

$$T = \{(T_p^0, T_n^0), (T_p^1, T_n^1), \cdots, (T_p^{|C|}, T_n^{|C|})\} \tag{6}$$

where $|C|$ is the number of category labels, $|T| = 2R|C|$ is the number of paired samples in T, and R is a hyperparameter. The number of positive and negative sample pairs can be controlled by the R value, and a reasonable R value can effectively improve the performance of the encoder.

The original small-sample dataset can be compared with the fine-tuning method to expand the size of training data in a few-sample scenario. For example, for a binary categorization sentiment analysis task, assuming that Q labeled samples are given in the original dataset, the possible size of the dataset T used for sentence encoder fine-tuning after constructing positive and negative ternary sample pairs by comparison is:

$$T = Q(Q-1)/2 \tag{7}$$

The final fine-tuned dataset T clearly has a much larger sample size than Q. This allows the sentence encoder to be fine-tuned as efficiently as possible on a small sample dataset.

Three pre-trained models, bert-base-uncased, bert-base-chinese, and chinese-roberta-wwmext, which are trained on large-scale corpus and are able to capture rich linguistic features, are selected for this study.

II. D.Classification head training

The main task of classification head training is to map the encoded sentence embedding representations to the corresponding categories based on them. In this study, logistic regression model is chosen as the classification head for training. This is because the logistic regression model can provide explanatory decision boundaries [26] and has good generalization ability in small sample situations. The first step of training is to encode the training data $\{x_i\}$ using a fine-tuned sentence encoder to generate a sentence embedding for each training sample:

$$Emb^{xi} = MT(x_i) \tag{8}$$

where $MT()$ denotes the function of the fine-tuned sentence encoder.

The generated embeddings with their corresponding category labels form the training set of the classification head T^{CH}, which can be formalized as:

$$T^{CH} = \left\{(Emb^{xi}, y_i)\right\} \tag{9}$$

where $|T^{CH}| = |D|$.

Ultimately, it is assumed that the comparison-fine-tuned sentence encoder encodes and produces a sentence embedding for a piece of telecommunication fraud text (x_i) in a network. Subsequently, a classification head trained by T^{CH} will make category predictions for the input sentences based on this sentence embedding. Formally, this can be expressed as:

$$x_i^{pred} = CH(MT(x_i)) \tag{10}$$

where CH denotes the classified head after being trained by the training set.

III. Lightweight fraudulent text recognition based on knowledge distillation

The knowledge distillation method is shown in Figure 2 [27]. The design of this chapter mainly consists of a teacher model PSC-BERT with a large number of parameters and relative complexity and a student model PSC-BERT4 with a relatively small number of parameters and relative simplicity. The distillation methods for each component and the training methods for the model in general are described in detail below.

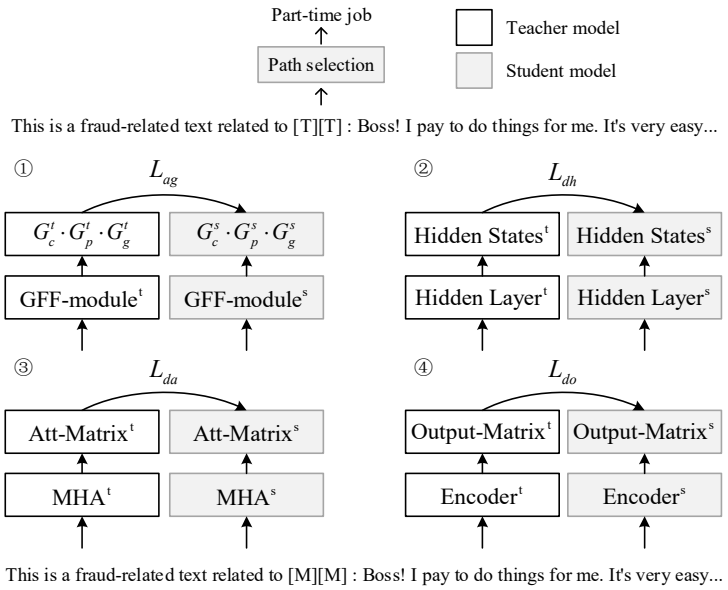

Figure 2: The knowledge distillation method schematic

III. A. Distillation of output gating vectors of gated networks

The output value of the gated network is the extent to which semantic, pronunciation, and glyph information is retained at each location [28], and is a one-dimensional vector of length $N - l_p$, as shown in Equation (11):

$$G = Sigmoid(W \cdot Encoder(E) + b) \tag{11}$$

Each position in the output gating value is a gated value between 0 and 1. The presence of the gating network has a positive effect on both model performance and convergence speed. Distillation of its output values allows the student model to learn what the teacher model knows about feature fusion, thus ensuring as much as possible that the student model learns as consistently as possible about the embedding matrix of the teacher model.

For the training objective, this paper uses the MSE loss to measure the degree of agreement between the teacher model and the student model in terms of gating values, as shown in Equation (12):

$$L_{dg} = MSE(G_c^s, G_c^t) + MSE(G_p^s, G_p^t) + MSE(G_g^s, G_g^t) \tag{12}$$

where G_C^S, G_p^S, G_g^S are the gating values for the student model, and G_C^t, G_p^t, G_g^t are the gating values for the teacher model.

III. B. Distillation of the output states of the hidden layer of the backbone network

The hidden layer output state of the backbone network shows the hidden state of the suspect text output after passing through each layer of the Transformer encoder [29], including the embedding matrix after feature fusion and the hidden state of each layer of the encoder. The actual role of the encoder is to perform layer-by-layer feature extraction of the input sentence as shown in Equation (13):

$$h_i = Transformer_i(h_{i-1}) \tag{13}$$

where $h_i (i = -1, 0, 1, ..., L-1) \in R^{N \times H}$ denotes the output features after passing through the i th layer of Transformer encoder, h_{-1} denotes the embedding matrix, and L is the total number of layers of Transformer encoder.

In this paper, the MSE loss is used to measure the degree of agreement between the teacher model and the student model in the hidden layer output states, as shown in Equation (14):

$$L_{dh} = \frac{1}{L_s + 1} \sum_{i=0}^{L_s} MSE(W_h h_i^s, h_{i'}^t) \tag{14}$$

where L_S is the number of hidden layers of the student model, $h_i^S \in R^{N \times H^S}$ is the first i hidden state of the student model, $h_{i'}^t \in R^{N \times H^t}$ is the teacher model's i'th hidden state, H^S and H^t are the hidden dimensions of the student and teacher models, respectively. $W_h \in R^{H^s \times H^t}$ is a learnable mapping matrix for mapping the hidden states of the student model to the same spatial size as the teacher model. Since the number of layers L of the student model and the teacher model are different, layer mapping is needed to map the student model hidden layers to the hidden layers of the teacher model at equal intervals when performing knowledge distillation, for example, when $L_S = 4$, $L_t = 12$, $i = \{0, 1, 2, 3, 4\}$, $i' = \{0, 1, 4, 7, 11\}$, when $i = 0$ indicates that the embedding matrix of the embedding layer is evaluated for consistency.

III. C. Distillation of the attention matrix of the hidden layer of the backbone network

The attention matrix of BERT contains a lot of linguistic information [30], which can reveal some syntactic and semantic information in the sentence, which is important for the model to understand the whole sentence. Therefore, in this paper, we distill knowledge from the attention matrix of the hidden layer of the network in the hope that the student model can learn the syntactic and semantic information contained in it from the attention matrix of the teacher model.

In this paper, we use the MSE loss to measure the degree of agreement between the teacher model and the student model on the attention matrix, as shown in Equation (15):

$$L_{da} = \frac{1}{L_s} \sum_{i=1}^{L_s} \left(\frac{1}{n_h} \sum_{j=1}^{n_h} MSE(A_{ij}^s, A_{i'j}^t) \right) \tag{15}$$

where n_h denotes the number of heads of multi-head self-attention, $A_i^S \in R^{N \times N}$ is the i th hidden state of the student model, and $A_{i'}^t \in R^{N \times N}$ is the i'th hidden teacher model state. Same as the hidden layer output state, layer mapping is required due to the difference in the number of layers L between the student model and the teacher model, but the embedding layer is not considered here.

III. D. Distillation of the output layer of the backbone network

The ultimate goal of the model is to predict the correct character at each position in the output layer, so the knowledge embedded in the output layer is particularly important. The model obtains the logical output of each category at the output layer by means of a fully connected layer, and then by means of the Softmax function, the output probability of each category is obtained, as shown in Eqs. (16) and (17):

$$z = Wh_{L-1} + b \tag{16}$$

$$P = Softmax(z) \tag{17}$$

where z is the logical vector of the output and P is the probability vector of the output. In addition to the positive labels, some negative labels also contain a certain amount of information, for example, the correct output of a

certain two positions is "gambling", and the word "gambling ball" also has a relatively large probability, which is ignored in the traditional hard label training, and all negative labels are meaningless. Through the distillation of the output layer, the student model can learn not only the information contained in the correct label, but also the information contained in the negative label, so that the amount of information brought by each sample to the student model increases.

In this paper, we use KL scatter with a temperature coefficient to measure the degree of agreement between the teacher model and the student model in terms of output, as shown in Equation (18):

$$L_{do} = \frac{1}{n}\sum_{i=0}^{n-1} D_{kl}(z_i^s / T, z_i^t / T) \tag{18}$$

where z_i^S and z_i^t denote the output logic vectors at the i th position of the student and teacher models, respectively, and the length of their vectors is the same as the length of the word list. T is the temperature coefficient, which is used to control the shape of the logistic distributions, and the larger the value, the smoother the distributions are, and the more the model learns from negative labels, and vice versa.

III. E. Overall model training methodology

The overall training method of the model in the knowledge distillation process is shown in Fig. 3.1) Pre-training of PSC-BERT and PSC-BERT4 on a large-scale generalized corpus.2) Fine-tuning of the pre-trained PSC-BERT in the fraudulent text dataset.3) Use of the fine-tuned PSC-BERT as the instructor model, and the pre-trained PSC-BERT4 as the student model for knowledge distillation in the fraudulent text dataset, combined with fine-tuning using real labels.4) Use the fine-tuned PSC-BERT for subsequent applications.

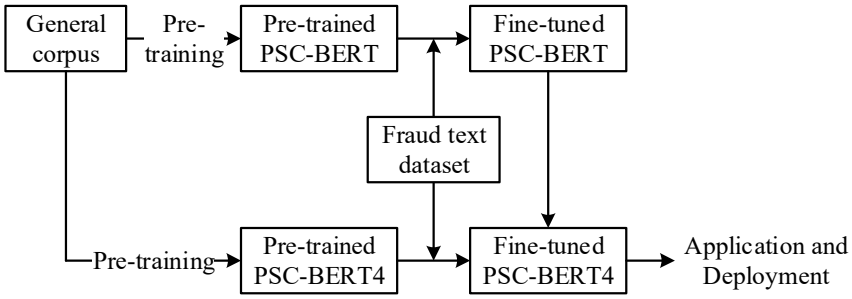

Figure 3: Model general training method

The final loss function L_{fin} in the knowledge distillation process is obtained as shown in Eqs. (19) and (20):

$$L_d = L_{dg} + L_{dh} + L_{da} + L_{do} \tag{19}$$

$$L_{fin} = \mu L_d + (1-\mu)L \tag{20}$$

where L_d is the loss function of the distillation part of the fine-tuning process, L is the loss function of the PSC-BERT, and the constant μ is the distillation loss share of the fine-tuning process.

IV. Experimental results of network text fraud detection

IV. A. Fraud Detection Performance

The experiments will test the performance of SVM, LR, deepwalk, and the PSC-BERT designed in this paper for the fraud detection task on the AML dataset. In the following, deepwalk+directed message passing+neighbor edge numbering will be referred to as the deepwalk+ model. The same will be true for the other models.

The experimental results will be presented in terms of model performance comparisons, and the impact of the test set results presentation.

IV. A. 1) Model performance

The trend of F0.5 values with Epoch on the training set is shown in Fig. 4. By initially analyzing the F0.5 value data on the training set, it is found that the F0.5 values of the PSC-BERT model in this paper are generally higher than those of the other models in all training steps. This finding indicates that the PSC-BERT model exhibits superior performance on the training set. Further observation of the trend of the F0.5 values with the training steps leads to the following conclusion: the overall trend of the F0.5 values of the PSC-BERT model shows an upward trend, which indicates that the model has been continuously optimized during the training process and effectively learned the feature representation of the data. In contrast, the F0.5 values of the deepwalk, node2vec, Line, deepwalk+,

and node2vec+ models have improved, but the magnitude is not as significant as that of the PSC-BERT model, which may not make full use of the features of the different relationship types, which restricts the model's ability to characterize the complex graph structure. The F0.5 values of SVM and LR models, on the other hand, improved in some training steps, but the overall change was small and the improvement slowed down in the later stages, which may have encountered the gradient vanishing or gradient explosion problem during the training process, which would lead to unstable model training, thus affecting the improvement of F0.5 values.

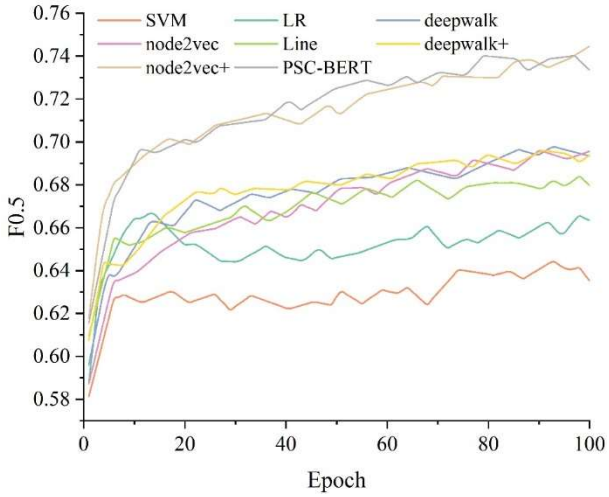

Figure 4: F0.5 with the change of epoch changes in the training set

The trend of F0.5 value with Epoch on the validation set is shown in Fig. 5. And it can be concluded that the F0.5 value of the PSC-BERT model is higher than the other benchmark models combined with the improved model in most steps. And the performance of the PSC-BERT model on the validation set is similar to that on the training set, indicating that the feature representations learned by the PSC-BERT model on the training set can be well generalized to the validation set, and the risk of model overfitting is low.

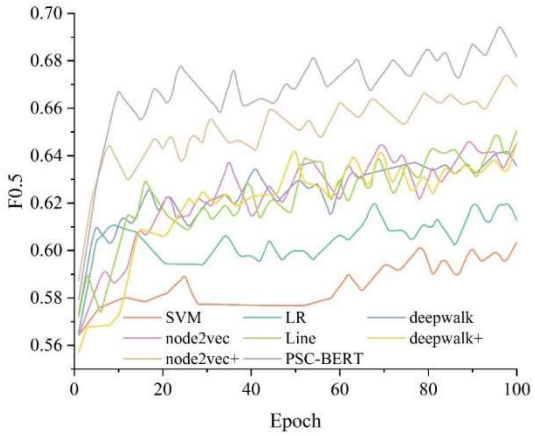

Figure 5: F0.5 with the change of epoch changes in the verification set

IV. A. 2) Presentation of test set results

The trend of F0.5 values with Epoch on the test set is shown in Fig. 6. At the initial step, the F0.5 score of PSC-BERT is 0.687, indicating that the model shows good performance at the early stage of training. As the training steps increase, the F0.5 score of PSC-BERT shows fluctuations, but the overall trend is upward, which indicates that the model gradually learns and improves its performance during the training process. At step 10, the F0.5 score of PSC-BERT reaches a peak of 0.762, which is also the highest F0.5 score among all models, showing the best performance of the model at this step. In the subsequent steps, the F0.5 scores of RGAT+ decreased, but still remained at a high level, indicating that the model has good stability and generalization ability.

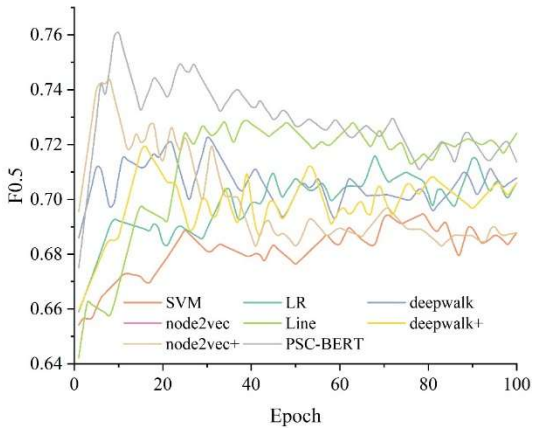

Figure 6: F0.5 with the change of epoch changes in the test set

IV. B. Classification experiments

For the detection model, the classification task is an important factor in measuring the performance of the algorithm. The vectors of nodes are generated by graph embedding methods and are input to the classifier as a collection of attributes of the nodes in the classification task. In order to study the performance of the models under different label completeness, classification experiments are conducted in this paper with different proportions of labeled nodes retained, and in order to minimize the impact of different classifiers, LR is used as the classifier for all models.

For the control experiments with other methods, 10%, 30% and 50% of the dataset are randomly divided into the training set, and the rest is used as the test set to train the network and repeat 10 times to take the average value, the corresponding results are shown in Table 1.

Compared to other methods, the model in this paper has improved results on three different datasets. Observing Table 1, the first two results are obtained by the traditional method, the classification effect on Cora and Citeseer datasets largely depends on the underlying principle of the algorithm, while the anti-fraud effect on Fraud Detection dataset is also closely related to the call feature extraction situation, and both more effective feature extraction and a larger proportion of the training set can improve the classification accuracy. The limited anti-fraud effectiveness of the traditional methods indicates that the current set and scripted fraud methods are not sufficiently distinctive in their features and are difficult to discriminate. The third to fifth results are obtained by the network structure-based method, firstly, comparing with the traditional method, the classification effect of the network structure-based method is substantially ahead of the traditional method with different proportions of training sets, which indicates the effectiveness of this paper's method in network node classification and telecom anti-fraud.

Table 1: Experiment results

	Cora			Citeseer			Fraud Detection		
Rate	10%	30%	50%	10%	30%	50%	10%	30%	50%
SVM	0.627	0.684	0.698	0.777	0.716	0.669	0.724	0.731	0.708
LR	0.792	0.772	0.746	0.688	0.703	0.658	0.714	0.778	0.652
deepwalk	0.652	0.656	0.774	0.706	0.762	0.682	0.706	0.767	0.779
node2vec	0.677	0.655	0.773	0.743	0.818	0.757	0.791	0.724	0.666
Line	0.647	0.718	0.614	0.778	0.685	0.722	0.778	0.729	0.643
deepwalk+	0.754	0.809	0.689	0.657	0.694	0.742	0.644	0.739	0.705
node2vec+	0.764	0.769	0.661	0.653	0.725	0.732	0.766	0.756	0.658
Line+	0.723	0.816	0.723	0.739	0.714	0.726	0.791	0.665	0.709
Ours	0.821	0.825	0.835	0.833	0.821	0.849	0.831	0.803	0.814

Secondly, comparing to the network structure-based methods themselves, the deepwalk and node2vec methods based on random wandering classes can explore the network neighborhood more effectively and learn the network structure more adequately, thus the node classification effect is more prominent than Line. The sixth to eighth results are obtained by the methods based on network structure and splicing attributes, compared with the methods based on network structure, these methods also splice the corresponding attribute features of nodes after obtaining the target vector. Although the node features are not directly involved in the training of the algorithm, after splicing, the target vector is richer in information, and the classification effect is qualitatively improved, which shows

that the incorporation of node attribute features has a positive effect on the learning. The ninth to tenth results are obtained by the method based on the fusion of network structure and node features. The algorithm in this paper considers the problem of weak association between nodes, based on the interaction between nodes, calculates the similarity in structure and attributes respectively, then judges whether it is a positive relationship by thresholding, and finally goes to learn the parameters and thresholds respectively. Therefore, the algorithm in this paper is somewhat ahead of all the previously described methods.

V. Conclusion

In this paper, we first classify the online fraud text, and on the basis of the classified text, we construct a lightweight fraud text recognition model based on knowledge distillation through the method of knowledge distillation to intelligently detect the information in online fraud.

Comparing this paper's PSC-BERT model with other classical content recognition models, it is found that the F0.5 mean value of this paper's lightweight fraud text recognition is 0.72 in the training set, 0.67 in the validation set, and 0.73 in the test set. The F0.5 mean values of the suboptimal methods are 0.71, 0.65, and 0.70, respectively, which is known to be ahead of the other methods in fraud text detection is ahead of other methods. The classification accuracy of this paper's method on the three datasets of fraudulent text is more than 0.8, which is also better than other text classification models, so it can be seen that this paper's method has a greater advantage in detecting online text fraudulent information.

Funding

This work was supported by the Key Scientific Research Project of Anhui Provincial Universities, China: *A Study on the Criminal Record Expungement System from the Perspective of Social Governance* (Project Code: 2023AH053011), and represents an interim research outcome.

References

[1] Koi-Akrofi, G. Y., Koi-Akrofi, J., Odai, D. A., & Twum, E. O. (2019). Global telecommunications fraud trend analysis. International Journal of Innovation and Applied Studies, 25(3), 940-947.

[2] Chen, L. C., Hsu, C. L., Lo, N. W., Yeh, K. H., & Lin, P. H. (2017). Fraud analysis and detection for real-time messaging communications on social networks. IEICE TRANSACTIONS on Information and Systems, 100(10), 2267-2274.

[3] Terzi, D. S., Sağıroğlu, Ş., & Kılınç, H. (2021). Telecom fraud detection with big data analytics. International Journal of Data Science, 6(3), 191-204.

[4] Chu, G., Wang, J., Qi, Q., Sun, H., Tao, S., Yang, H., ... & Han, Z. (2022). Exploiting spatial-temporal behavior patterns for fraud detection in telecom networks. IEEE Transactions on Dependable and Secure Computing, 20(6), 4564-4577.

[5] Ni, P., & Wang, Q. (2022). Internet and telecommunication fraud prevention analysis based on deep learning. Applied Artificial Intelligence, 36(1), 2137630.

[6] Hu, X., Chen, H., Liu, S., Jiang, H., Chu, G., & Li, R. (2022). BTG: A Bridge to Graph machine learning in telecommunications fraud detection. Future Generation Computer Systems, 137, 274-287.

[7] Huang, Z. (2017, November). Causes and prevention of telecommunication network fraud. In 2nd International Conference on Humanities Science and Society Development (ICHSSD 2017) (pp. 164-173). Atlantis Press.

[8] Xu, H., Qadir, A., & Sadiq, S. (2025). Malicious SMS detection using ensemble learning and SMOTE to improve mobile cybersecurity. Computers & Security, 104443.

[9] Shinde, A., Shahra, E. Q., Basurra, S., Saeed, F., AlSewari, A. A., & Jabbar, W. A. (2024). SMS Scam Detection Application Based on Optical Character Recognition for Image Data Using Unsupervised and Deep Semi-Supervised Learning. Sensors, 24(18), 6084.

[10] Al Saidat, M. R., Yerima, S. Y., & Shaalan, K. (2024). Advancements of SMS Spam Detection: A Comprehensive Survey of NLP and ML Techniques. Procedia Computer Science, 244, 248-259.

[11] Mehmood, M. K., Arshad, H., Alawida, M., & Mehmood, A. (2024). Enhancing Smishing Detection: A Deep Learning Approach for Improved Accuracy and Reduced False Positives. IEEE Access.

[12] Sonowal, G. (2020). Detecting phishing SMS based on multiple correlation algorithms. SN computer science, 1(6), 361.

[13] Salman, M., Ikram, M., & Kaafar, M. A. (2024). Investigating evasive techniques in sms spam filtering: A comparative analysis of machine learning models. IEEE Access, 12, 24306-24324.

[14] Mishra, S., & Soni, D. (2023). Dsmishsms-a system to detect smishing sms. Neural Computing and Applications, 35(7), 4975-4992.

[15] Akande, O. N., Gbenle, O., Abikoye, O. C., Jimoh, R. G., Akande, H. B., Balogun, A. O., & Fatokun, A. (2023). SMSPROTECT: An automatic smishing detection mobile application. ICT Express, 9(2), 168-176.

[16] Alabdan, R. (2020). Phishing attacks survey: Types, vectors, and technical approaches. Future internet, 12(10), 168.

[17] Lee, H., Jeong, S., Cho, S., & Choi, E. (2023). Visualization technology and deep-learning for multilingual spam message detection. Electronics, 12(3), 582.

[18] Ahmad, H., & Erdodi, L. (2021). Overview of phishing landscape and homographs in Arabic domain names. Security and Privacy, 4(4), e159.

[19] Hapase, D. S., & Patil, L. V. (2024). Telecommunication fraud resilient framework for efficient and accurate detection of SMS phishing using artificial intelligence techniques. Multimedia Tools and Applications, 1-23.

[20] Ulfath, R. E., Alqahtani, H., Hammoudeh, M., & Sarker, I. H. (2021, December). Hybrid CNN-GRU framework with integrated pre-trained language transformer for SMS phishing detection. In Proceedings of the 5th International Conference on Future Networks and Distributed Systems (pp. 244-251).

[21] Doshi, J., Parmar, K., Sanghavi, R., & Shekokar, N. (2023). A comprehensive dual-layer architecture for phishing and spam email detection. Computers & Security, 133, 103378.
[22] Atawneh, S., & Aljehani, H. (2023). Phishing email detection model using deep learning. Electronics, 12(20), 4261.
[23] Ulus, C., Wang, Z., Iqbal, S. M., Khan, K. M. S., & Zhu, X. (2022, November). Transfer Naïve Bayes Learning using Augmentation and Stacking for SMS Spam Detection. In 2022 IEEE International Conference on Knowledge Graph (ICKG) (pp. 275-282). IEEE.
[24] Gui, J., Zhou, Y., Yu, K., & Wu, X. (2024). PSC-BERT: A spam identification and classification algorithm via prompt learning and spell check. Knowledge-Based Systems, 301, 112266.
[25] Minhao Zou,Zhongxue Gan,Yutong Wang,Junheng Zhang,Chun Guan & Siyang Leng. (2025). Topology-preserving and structure-aware (hyper)graph contrastive learning for node classification. Applied Intelligence,55(7),616-616.
[26] Guanzhao Wang,Tian Yang,Zelong Ouyang,Jinqiong Li,Zhihua Li,Jing Cao... & Qinghua He. (2025). An AHP-multiple logistic regression model for risk assessment of highly pathogenic avian influenza. Journal of Applied Poultry Research,34(2),100523-100523.
[27] Alireza Esmaeilzehi,Hossein Zaredar & M. Omair Ahmad. (2025). DCSR: A deep continual learning-based scheme for image super resolution using knowledge distillation. Applied Intelligence,55(7),627-627.
[28] Linhui Sun,Yunlong Lei,Zixiao Zhang,Yi Tang,Jing Wang,Lei Ye & Pingan Li. (2025). Multi-task coordinate attention gating network for speech emotion recognition under noisy circumstances. Biomedical Signal Processing and Control,107,107811-107811.
[29] Dhirendra Prasad Yadav,Bhisham Sharma,Ajit Noonia & Abolfazl Mehbodniya. (2025). Explainable label guided lightweight network with axial transformer encoder for early detection of oral cancer. Scientific reports,15(1),6391.
[30] Dario Valcamonico,Piero Baraldi,July Bias Macêdo,Márcio Das Chagas Moura,Jonathan Brown,Stéphane Gauthier & Enrico Zio. (2025). A systematic procedure for the analysis of maintenance reports based on a taxonomy and BERT attention mechanism. Reliability Engineering and System Safety,257(PA),110834-110834.