Construction of a Security Defense Model for the University's Cyberspace Based on Machine Learning

Wang Bin
Shanxi Technology and Business College, Computer and Information Engineering College
Network and Information Technology Teaching and Research Office
Taiyuan City, Shanxi Province, 030036

Abstract*—In order to ensure the security of university teachers and students using cyberspace, a machine learning based university cyberspace security defense model is constructed. Adopting a compression perception based data collection method for university cyberspace, the data information collection of university cyberspace is completed through sparse representation, compression measurement, and recovery reconstruction. Combining the advantages of Convolutional Neural Network (CNN) model in spatial feature extraction of data and Long Short Term Memory (LSTM) model in sequential feature extraction of data, extract the features of university network spatial data. After completing the multi feature dimensionality reduction processing of university network data based on the non-negative matrix decomposition algorithm, the feature dimensionality reduction processing results are input into the ConvLSTM-CNN model. After convolution calculation and integration, the security threat detection results of university network space are output. Based on the results of security threat detection, corresponding network attack defense measures are selected to ensure the security of the university's cyberspace. The experimental results show that the average attack interception rate of the model after application can reach 97.6%. It has been proven that building a model can accurately detect security threats to the university's cyberspace and achieve defense against various network attacks in different environments.

***Keywords*—Machine learning; University's cyberspace; security defense; construction of a model; compressed sensing; non-negative matrix**

I. Introduction

The development of computer technology has accelerated the process of information popularisation. By utilising the Internet, various services such as electronic banking, online teaching, and video conferencing can be provided to the public, gradually increasing people's dependence on the Internet in their daily lives, studies, and work. Currently, the Internet is also widely used in various universities. The university network has the characteristics of spatial freedom, large distribution scale, and relatively opens, but at the same time, there are also security risks in the university network. The rapid development of the Internet has led to a large number of intrusion behaviours, posing a huge threat to the network security of universities. Therefore, achieving a comprehensive perception and defense of the university's cyberspace attack events is significant for building a complete network security defense system [1]–[3].

The research on secure and reliable defense methods for network-blocking attacks has become an urgent issue to be solved, and many relevant experts and scholars have achieved fruitful research results on this topic. Wang et al. designed a network active security defense system based on the K-means algorithm [4]. However, this method is greatly affected by attackers during maintenance and is easily exploited by them, and its defense ability needs to be improved. Oliveira et al. proposed an intelligent network attack detection and classification method for network-based intrusion detection systems [5]. However, the effectiveness of this method in network security situational awareness is relatively low, and it cannot solve some low perception network attacks. The research in [6] constructed an integrated learning model for detecting botnet attacks in IoT networks. However, this method does not take into account the changes in feature quantities that exist in university networks. Zhang et al. proposed a network security attack detection method for the network Physical system based on deep learning [7]. However, this method addresses a relatively single type of network attack, and its defense effectiveness against different network attacks is not outstanding. Tonkal et al. proposed a machine learning method with neighborhood component analysis for detecting DDoS attacks in software defined networks [8]. Gurumanapalli et al. [9] proposed a state-of-the-art generalized Feistel network-assisted Shannon condition and dynamic key based SSPN (GFS-SSPN) lightweight encryption system for IoT security to achieve high attack resilience. Majeed et al. [10] used intelligent machine learning methods to design IoT assisted drones. This method will provide an intelligent network security system that helps detect network security threats using blockchain. However, due to the processing of multi-dimensional data for network attacks, this method did not undergo dimensionality reduction processing, and the accuracy of network attack detection needs to be improved. Although the above methods have achieved attack defense, they still have issues such as inadequate defense capabilities, low perceptual efficiency, and lack of consideration for feature changes, single attack type resolution, and insufficient dimensionality reduction processing.

Machine learning aims to train computers to perform certain operations through data to train "learning" in datasets and enable computers to make improved decisions in the future. Machine learning is widely used in data mining, which involves searching for unknown or hidden patterns through a large amount of data. Machine learning can be divided into two categories: supervised and unsupervised. In the supervised learning dataset, all training and test data are assigned a value, that is, a label, which can be a numeric or a character value. Labels of unknown data can be learnt and predicted through machine learning algorithms. In unsupervised learning, data does not have labels and machine learning algorithms are needed to find out the data patterns and predict unknown data. Therefore, in response to the university's network attacks, this paper constructs a security defense model of the university's cyberspace through machine learning, implements effective defense of attack data, and ensures the safe and stable operation of the university's cyberspace. The proposal of this model will provide innovative solutions for university cyberspace security, improving the resilience and accuracy of cyberspace security defense. The main research structure of this article is as follows:

1) Construction of a security defense model for university cyberspace: This section includes several parts: university cyberspace data collection based on compressed sensing, university cyberspace security threat detection based on machine learning, university cyberspace data feature extraction based on ConvLSTM-CNN model, and university cyberspace security defense methods.

2) Experimental analysis: This section verifies the defense effectiveness of the constructed model from different perspectives.

II. Construction of a Cyberspace Security Defense Model for Universities

While operating the security defense model for the university's cyberspace, this paper conducts network security threat defense through three stages: data collection, threat detection, and defense generation. The specific process is as follows:

A. University's Cyberspace Data Collection based on Compressed Sensing

In order to meet the actual application needs of the university's cyberspace [11], this paper implements the university's cyberspace data collection through the principle of compressed sensing. The core goal of the data acquisition method based on the principle of compressed sensing is to ensure the high accuracy of data. Therefore, on the basis of analysing the core theory of compressed sensing and combining it with the specific characteristics of the university's cyberspace monitoring data, it should study practical and feasible data collection methods for the university's cyberspace that meet the needs.

As a whole, the operating status of each node in the university's cyberspace will be interrelated and affect each other [12], which indicates that there must be a strong correlation between the data of each node and data with strong correlation must have a lot of redundancy and data containing redundancy can be expressed from another perspective through simple information. The large quantity and variety of data in universities' online space can be expressed by relatively simple information in a certain space, which shows sparsity in that space. This processing method of transforming data acquisition into information acquisition is one of the core ideas of compressed sensing. Therefore, the collected data from the university's cyberspace can be compressed using the principle of compressed sensing.

The speed and accuracy of monitoring data collection are crucial for the safe operation of a university's cyberspace. The data acquisition method based on the compressed sensing principle includes three processes: sparse representation, compressed measurement and recovery reconstruction, as shown in Fig 1. By studying and improving each process of the collection method, the accuracy of data collection can be further improved.

The starting point of this paper is the co-domain sampling of the university's network spatial data, and its research focuses on the sparse representation of data (the supporting theory of co-domain sampling). According to Fig. 1, improving the sparsity of data can improve the accuracy of data collection methods and reduce the errors introduced when restoring the original monitoring data using the feature data of monitoring quantities; Data recovery requires knowing the compressed measurement sampling value y and sparse basis matrix Ψ at the other end of the communication channel. The amount of data transmitted between the two ends, y and Ψ, determines the speed of the data collection method.

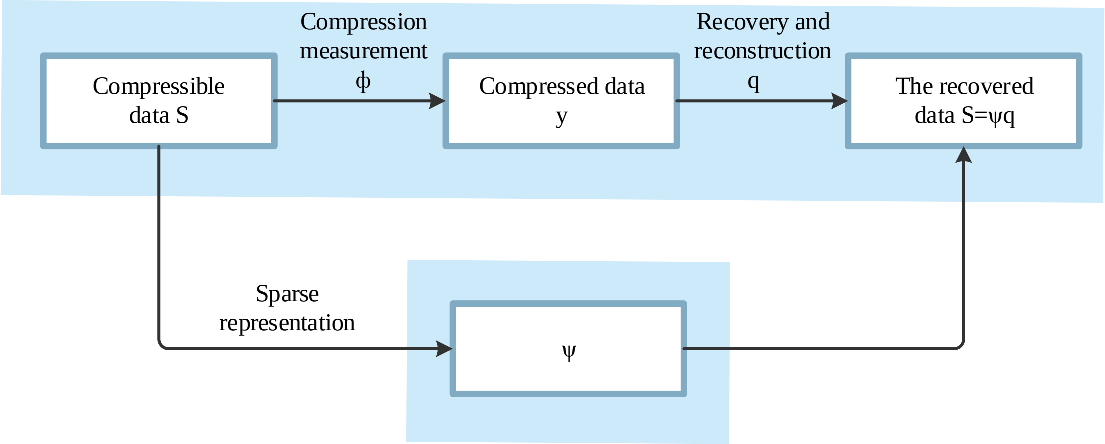

Fig. 1. Compressed sensing flow of monitoring data.

1) Theoretical basis of compression perception algorithm: In compressed sensing [13], the K-means Singular Value Decomposition (K-SVD) dictionary learning algorithm is an adaptive data sparse method. Due to the fact that data collected in a university's cyberspace can always be sparsely represented in an unknown space, and the excellent adaptive characteristics of the K-SVD dictionary enable it to adapt to the collected data in the university's cyberspace through dictionary learning algorithms, so as to effectively find a sparse space suitable for university's cyberspace data and achieve sparse representation of university's cyberspace collected data.

The mathematical model of the K-SVD dictionary learning algorithm is as follows:

$$\begin{cases} D, \theta = argmin\{\|s - D\theta\|_r^2\} \\ s.t. \forall i, \|\theta_i\|_0 \leq T_0 \end{cases} \tag{1}$$

where, s represents monitoring data containing different data types within a sampling period; θ is sparse coefficient vector; T_0 is the upper limit of the number of non-zero components in sparse coefficients; the dictionary is $D = [d_1, d_2, \cdots, d_k] \in R^{n \times k}$, where each column d_k represents an atom; r is the norm.

The data s parasitisation steps based on the K-SVD dictionary learning algorithm are as follows.

Initialise the dictionary. Randomly select K data samples from the general data samples as the atoms of dictionary D. This paper takes the monitoring data s as the initial atom of D, and its standardisation process is as follows:

$$d_i^* = \left| \frac{d_i}{\sqrt{\sum_{i=1}^{N} d_i 2}} \right| \tag{2}$$

where, N is any length of compressible discrete real value data.

a) Sparse encoding. Under the condition that the dictionary D is fixed, it can solve the optimisation model of the above equation using the Orthogonal Matching Pursuit(OMP) to obtain the sparse coefficient θ′ of the collected data s.

b) Dictionary update. Update the dictionary column by column. When updating the k -th column d_k of the dictionary, make E_k the error generated by removing d_k from the data, that is:

$$E_k = s - \sum_{j \neq k} d_j \theta_T^j \tag{3}$$

In the equation, θ_T is the jth line of θ. Then perform singular value decomposition on the error to obtain the updated d_k and θ.

c) Repetitive sparse encoding and dictionary updates. If the termination condition is met, the output data is based on the adaptive sparse dictionary D of the K-SVD dictionary learning algorithm.

In the application of data collection in the university's cyberspace, based on the K-SVD dictionary learning algorithm, every atom of the optimal dictionary D of signal s is identical, and the coefficient vector θ decomposed on this dictionary is 1-sparse, ensuring the high sparsity of data s. This shows the feasibility and progressiveness of the K-SVD dictionary learning algorithm applied to a sparse representation of monitoring data in university network space.

2) Compression measurement: For the compression measurement of the university's cyberspace data, data x is compressed to obtain a measurement vector y with a length of $M(M \times N)$, which is expressed as:

$$y = \Phi x \tag{4}$$

Compression measurement no longer uses the method of measuring the university's network spatial data x itself first and then compressing it. Instead, a measurement matrix $\Phi \in R^{M \times N}$ is used to directly compress and measure data x, converting data sampling into sampling of M projections on the measurement matrix Φ.

3) Compression reconstruction: For the restoration and reconstruction of the university's cyberspace data, it is the process of restoring the original data x from vector y. In this process, the measurement vector y is represented as:

$$y = \Theta\theta \tag{5}$$

In the equation, Θ is the perception matrix and satisfies the isometric constraint criterion. The goal of the university's cyberspace data recovery and reconstruction is to find the optimal sparse coefficient vector θ, which can be expressed as an optimisation problem under the l_1 norm, namely:

$$\begin{cases} \theta = \arg\min \|\theta\|_1 \\ s.t. \quad y = \Theta\theta \end{cases} \tag{6}$$

Finally, the optimal recovery value x of the original data can be obtained. And then, the space data collection of the university's network using compressed sensing is realised. Based on the above compression perception, compression measurement, and compression reconstruction processes, data collection and output of university cyberspace security threat detection results are shown in Table I:

TABLE I. SECURITY THREAT DETECTION RESULTS OF UNIVERSITY CYBERSPACE

Connect IP	Time stamp	Detection result
192.168.1.2	2022-10-05 09:30:20	Normal
192.168.1.4	2022-10-05 09:35:10	DDoS attacks
192.168.1.5	2022-10-05 09:40:15	DNS hijacking
192.168.1.7	2022-10-05 09:45:30	Normal
192.168.1.2	2022-10-05 09:50:40	Normal

B. Machine Learning-based Threat Detection for University's Cyberspace Security

1) Theoretical basis of ConvLSTM-CNN model: A CNN is a deep feedforward neural network with a large number of convolution calculation processes. Different from the conventional neural network model, the neurons in each layer of CNN are arranged three-dimensionally. A two-dimensional data's length and width are the neurons' length and width, while the depth represents the third dimension used to activate the data volume. It is precisely because of this characteristic of the CNN model that it can better obtain the spatial features of university network data [14].

At present, CNN models have been widely applied in the field of cyberspace security in universities, such as intrusion detection systems and situational awareness. The structure diagram of a typical CNN model is shown in Fig. 2. Convolutional neural networks are typically composed of input, hidden, and output layers, with hidden layers typically including convolutional, pooling, and fully connected layers.

The data is first input in the input layer, and then the university's network data perceived in Section II (A) is preprocessed. By normalising the data, it is then passed into the hidden layer for calculation.

The first layer to enter is the convolutional layer, which is the core of the CNN model structure. It has multiple convolutional kernels inside and can extract features from the input data. After the convolutional layer, there is usually a pooling layer, which appears alternately in the convolutional neural network. Each convolutional layer corresponds to a pooling layer. The fully connected layer in convolutional neural networks has a similar network structure to conventional neural networks, where each neuron structure constructs connections with all neurons in the previous layer, playing the role of a "classifier" in the entire network, integrating and mapping local sample information into the corresponding space.

Compared with other neural network models, CNN has better automatic feature extraction ability [15] and is widely used in computer vision, natural language processing, cyberspace security and other directions. It has made much progress and can be used for attack classification, image recognition, target segmentation, and other fields. In this paper, we utilised the advantages of CNN models in the perceptual extraction of spatial dimension features and optimised their structure to make them more suitable for modelling traffic time series in the field of cyberspace security situational awareness in universities.

Although the CNN model has obvious advantages in data space feature extraction, it is not good at feature extraction for sequence data such as network traffic. Therefore, it needs to be optimised based on fully utilising its advantages.

LSTM model is a special Recurrent Neural Network (RNN) model which has the ability to store the long-term state of data. Sometimes, due to the long data sequence length, the training process of ordinary RNN models is prone to the problem of vanishing or exploding gradients. The LSTM model performs better in training longer sequence data than this.

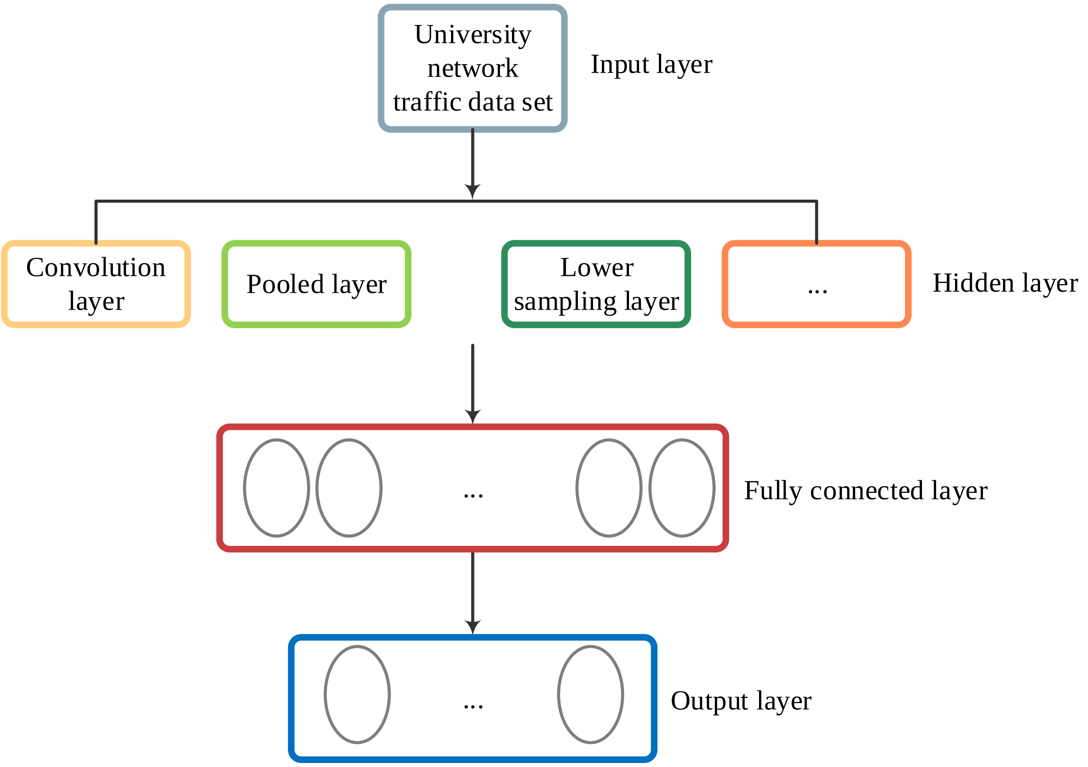

Fig. 2. Schematic diagram of convolutional neural network model.

In order to control the cell state, three gates are designed in the LSTM model to control or delete the added unit state information, namely the forgetting gate, memory gate, and output gate:

a) Forgetting gate: It used to forget or discard part of attribute information, accept a short-term memory output from the previous unit module, and decide which part to retain and forget.

b) Memory gate: It determines the information stored in the cellular state. For the attribute relationships discarded in the forgetting gate, it can find and fill in the corresponding new attribute information in this unit module to supplement the attribute information discarded by the forgetting gate.

c) Output gate: It determines the output value based on the cell state.

The working principle of the LSTM model is shown in Eq. (7):

$$\begin{cases} i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci} \circ c_{t-1} + b_i) \\ f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf} \circ c_{t-1} + b_f) \\ c_t = f_t \circ c_{t-1} + i_t \circ tanh(\, W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\ o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co} \circ c_t + b_o) \\ h_t = o_t \circ tanh(\, c_t) \end{cases} \tag{7}$$

where, i_t、f_t、c_t、o_t、h_t represents input gate, forgetting gate, cell state, output gate, and transmission state; $\circ$ is the multiplication of the corresponding elements of the matrix, also known as Hadamard product, and σ is the sigmoid function; W is the weight matrix, and b is the bias top.

The LSTM model, especially for high-dimensional time series data such as network data, can effectively analyse the logical relationships between input features and the complex temporal correlations between information. Meanwhile, compared to the CNN model, the LSTM model can better simulate human thinking patterns and cognitive processes and exhibits better processing ability for complex tasks closely related to time series. Therefore, this paper will use the temporal features extracted from LSTM network data combined with the spatial features extracted from CNN network data as the result of feature extraction for university network data.

2) Multi-feature dimensionality reduction processing of university's network data based on non-negative matrix decomposition algorithm

Due to the multi-dimensional problem of obtaining the features of the university's network data in Section II (B) (1), in order to improve the accuracy of threat detection in the university's network space security, a non-negative matrix decomposition algorithm is used to perform multi-feature dimensionality reduction on university network data [16]. The non-negative matrix decomposition algorithm has the advantages of simple decomposition implementation and small space occupation. The specific process of multi-feature dimensionality reduction for university network data based on the non-negative matrix decomposition algorithm is as follows:

Fuzzy classification of university network data features is carried out, and the initial value of the non-negative matrix decomposition base matrix to be composed of the classification centroid is set [17]. In contrast, the non-negative matrix decomposition rank is the number of classifications. The feature set of university network data is set to $\lambda = [\lambda_1, \lambda_1, \cdots, \lambda_n]$, where n represents the number of samples in the feature set of university network data. After fuzzy l classification, the set is $F = [F_1, F_2, \ldots, F_l]$, where l represents the number of feature classifications of university network data. While the membership degree of the university's network data features to the fuzzy vector is expressed by U_{ij}, the membership degree matrix representation is expressed as $U = [U_{ij}]_{l\times n}$. It should be noted that U_{ij} meets $\sum_{i=1}^{l} U_{ij} = 1$, and the membership degree update function is specified as:

$$U_{ij} = \begin{cases} 0 & \|\lambda_j - z_i\|_2 > 0 \\ 1 & \|\lambda_j - z_i\|_2 = 0 \\ \left[\sum_{a=1}^{l} (\|\lambda_j - z_i\|_2 / \|\lambda_j - z_a\|_2)^{2/(\omega-1)}\right]^{-1} & \|\lambda_j - z_i\|_2 < 0 \end{cases} \tag{8}$$

where, λ_j represents the jth university network data feature; z_i represents the i-th clustering centre; z_a represents the initial clustering centre; ω represents the fuzzy weighted index.

Thus, the cluster centre set is determined as follows:

$$Z_i = \frac{\sum_{j=1}^{n} (U_{ij})^{\omega} \lambda_j}{\sum_{j=1}^{n} (U_{ij})^{\omega}} \tag{9}$$

The objective function is:

$$D = \sum_{i=1}^{l} \sum_{j=1}^{n} (U_{ij})^{\omega} \|\lambda_j - z_i\|_2^2 \tag{10}$$

If the above equation calculates the result $D < \tau$ (threshold), then the iteration stops, and the initial value of the base matrix P in non-negative matrix decomposition is set to be composed of the cluster centre set z_i.

According to the non-negative matrix decomposition algorithm, the feature matrix $\lambda_{m\times n}$ of university network data is decomposed into two non-negative matrices $Q_{m\times n}$ and $P_{m\times n}$, and the product between them infinitely approximates the original non-negative matrix, namely $P_{m\times n} \approx Q_{m\times n}$. Where, $P_{m\times n}$ represents the base matrix; $Q_{m\times n}$ represents the coefficient matrix, using minimising the remaining Frobenius as the objective function, and the expression is:

$$\min_{P,Q} F(P,Q) = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} (\lambda_{ij} - (PQ)_{ij})^2 \tag{11}$$

When, $\lambda = PQ$, the value is 0.

Before the convergence of the objective function, fuzzy classification is used to obtain the centroid initialisation basis matrix, and the matrices P and Q are alternately updated. The matrix iteration rules are expressed as follows:

$$\begin{cases} P_{ia} \leftarrow P_{ia} \sum_j \frac{\lambda_{ij}}{(PQ)_{ij}} Q_{ai} \\ P_{ia} \leftarrow \frac{P_{ia}}{\sum_i P_{ia}} \\ Q_{aj} \leftarrow Q_{aj} \sum_i \frac{\lambda_{ij}}{(PQ)_{ij}} \end{cases} \tag{12}$$

After repeated iterations, the base matrix and coefficient matrix are finally obtained as P_t and Q_t. The coefficient matrix Q_t is used to replace the original sample matrix, thus achieving multi-feature dimensionality reduction of university network data.

3) ConvLSTM-CNN-based threat detection model for university's cyberspace attacks: As mentioned earlier, traditional CNN models cannot fully consider the sequence characteristics of data, and using LSTM models alone cannot fully consider the spatial characteristics of data and the correlation relationships between various features. Therefore, this paper intends to introduce the idea of the ConvLSTM model to model the problem of situational awareness of cyberspace security threats in universities from the dimensions of time and space features. Considering the limited computational performance of the ConvLSTM model and the higher computational cost and longer training time of each layer compared to convolutional layers, this paper proposes a ConvLSTM-optimized CNN model. It applies to the field of cyberspace security threat situational awareness in universities.

ConvLSTM was first proposed in 2015 and has shown good performance in predicting spatiotemporal data with both temporal and spatial features. This model extends the fully connected LSTM to have a convolutional structure in both input-to-state and state-to-state transitions, as shown in Fig. 3, which is the ConvLSTM network architecture diagram.

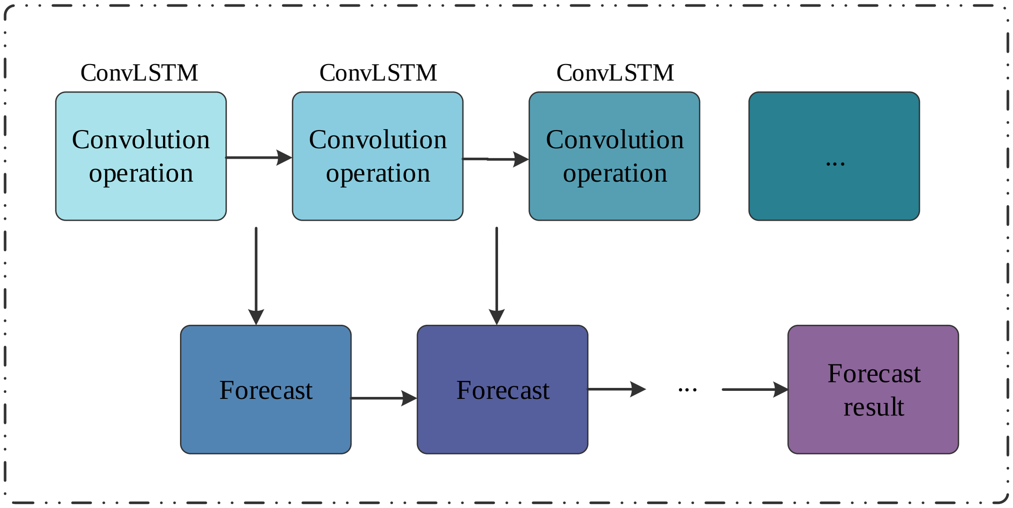

Fig. 3. Internal architecture of ConvLSTM model.

The working principle of ConvLSTM is shown in Eq. (13):

$$\begin{cases} i_t = \sigma(W_{xi} * \chi_t + W_{hi} * H_{t-1} + W_{ci} \circ c_{t-1} + b_i) \\ f_t = \sigma(W_{xf} * \chi_t + W_{hf} * H_{t-1} + W_{cf} \circ c_{t-1} + b_f) \\ c_t = f_t \circ c_{t-1} + i_t \circ tanh(W_{xc} * \chi_t + W_{hc} * H_{t-1} + b_c) \\ o_t = \sigma(W_{xo} * \chi_t + W_{ho} * H_{t-1} + W_{co} \circ c_t + b_o) \\ h_t = o_t \circ tanh(c_t) \end{cases} \tag{13}$$

In the equation, i_t, f_t, c_t, o_t, H_t represents input gate, forgetting gate, cell state, output gate, and transmission state, respectively; $\circ$ is the multiplication of the corresponding elements of the matrix, also known as Hadamard product, and σ is the sigmoid function; W is the weight matrix, and b is the bias top. Unlike LSTM, where χ, c, H, i, f, o represents three dimensions, f and o represent the information of data rows and columns.

For the classification and prediction requirements of spatiotemporal sequence data, such as university's cyberspace data, the ConvLSTM model has the memory storage and computing ability for long-time sequences [18]. Therefore, this paper uses a combination of the ConvLSTM and CNN models to construct the ConvLSTM-CNN model, with the specific structure shown in Fig. 4.

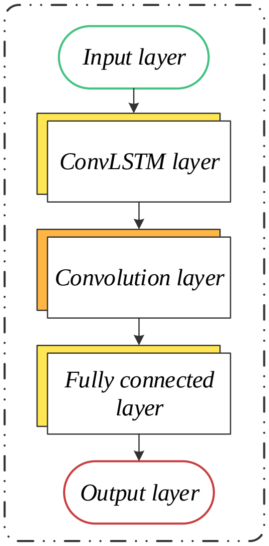

Fig. 4. Structure diagram of ConvLSTM-CNN model.

a) Input layer: First, encode the labels of input data in the input layer, convert them from textual data to continuous numerical variables, fit the data, and then perform standardisation and normalisation processing. Next, convert the data into a specific dimensional matrix that can be accepted by the next ConvLSTM layer.

b) ConvLSTM layer: The first layer to enter after the input layer is the ConvLSTM layer [19], which identifies the time dimension and spatial dimension of data, captures the spatiotemporal characteristics of data, and has a long memory function for solving the problem of serial data modelling with a long time. At the same time, unlike the feedforward fully connected form between the input and each gate in the LSTM model, the model's weight is convolutionally calculated and connected using the convolutional form, which allows for the acquisition of spatial features of the data while considering temporal features.

c) Convolutional layer: After capturing the spatiotemporal features of the data, it enters the convolutional layer for calculation. By utilising the typical sparse connection characteristics of the convolutional layer, it can effectively extract data features while shortening the convergence time of the model [20]. In forward fully connected neural networks, every element in the matrix needs to participate in computation, which undoubtedly increases the computational time and complexity of the algorithm. In this layer, the parameters of each convolutional kernel can be shared with each other, allowing only one operation to be performed on the same parameters in the university network model, which not only improves the runtime but also preserves its advantages in spatial feature extraction [21]. In the design of this model, the number of convolutional layers is set to 3, with a gradually decreasing number of convolutional kernels.

d) Fully connected layer: Located at the last layer of the entire network structure, it is used to integrate the feature information output from the previous layer.

e) Output layer: make classification and probability mapping for the model output and map it to the corresponding labels.

4) Optimisation parameter selection: In a neural network, the key work to enable it to solve the nonlinear problem of a sparse matrix, such as a university's cyberspace security situation awareness, is how to select and use appropriate activation functions, introduce nonlinear factors, retain and map the features of some activated neurons, and remove redundant and irrelevant features in the data, so as to enable neural networks to have hierarchical nonlinear mapping learning capabilities that linear models do not possess [22]. After completing the construction of the neural network model, it is necessary to search for the optimal solution for the model and select a suitable optimiser. Usually, the idea of gradient descent is used to calculate the loss and gradient of the model in order to obtain the parameters of each layer of the university's cyberspace security threat detection model and generate the optimal solution.

a) Activation function: In the structure and calculation process of ConvLSTM, similar to LSTM, there is a tanh operation in the related calculations of its input and output gates, which is used to generate candidate memories. As shown in Eq. (14), the calculation equation for the tanh function is shown. It can be seen that the tanh function requires a fourth power operation during the calculation process, and more power operations may be performed during its backpropagation differentiation, which undoubtedly increases the computational complexity of the ConvLSTM layer. Meanwhile, the tanh function may also cause gradient saturation during model training, which is a major reason for its low computational efficiency.

$$tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{14}$$

where, e is a function.

In order to reduce the computational load of the security threat detection model of the university's cyberspace and solve the gradient saturation problem in ConvLSTM, this paper introduces the linear rectification function (ReLU) as the activation function in ConvLSTM-CNN and CNN layers and Eq. (15) is the calculation equation of ReLU function.

$$\mathrm{ReLU}(x) = max(0, x) \tag{15}$$

In the full connection layer, the activation function [23] is not used by default; that is, each layer only performs simple matrix multiplication. In this layer, the design and use of appropriate activation functions can play the role of adding nonlinear factors to neurons and the entire network structure, making neurons approach nonlinear networks. Therefore, this paper uses Softmax as the activation function in the full connection layer to obtain the probability value of each sample corresponding to the input so as to achieve the purpose of data classification. The calculation equation for the sigmoid function is shown in Eq. (16).

$$\mathrm{soft}\,max(x) = \frac{e^{x}}{\sum_{x=1}^{T} e^{x}} \tag{16}$$

where, T is the number of types of samples.

b) Optimiser: After the establishment of the threat detection model for cyberspace security in basic colleges and universities, multiple iterations is needed to continuously optimise the model, find appropriate parameters, reduce the value of the loss function as much as possible, and improve the accuracy of the threat detection model for cyberspace security in colleges and universities. In order to adjust the parameters of the university's cyberspace security threat detection model during the training process and find the optimal solution, we introduced an optimiser to calculate and update the network parameters that could affect the training and output of the university's cyberspace security threat detection model, so that the results of the university's cyberspace security threat detection model can approximate or achieve the best [24], [25]. In this paper, the Adaptive Time Estimation Method (Adam) is introduced to calculate the adaptive learning rate of each parameter in the university's cyberspace security threat detection model. Adam algorithm has higher convergence speed and more efficient computational efficiency than other adaptive learning rate algorithms. At the same time, it also has a lower occupancy rate of system computing resources and is more suitable for situations such as sparse gradients or high gradient noise. The Adam algorithm has made improvements such as bias correction and gradient algorithm optimisation on the basis of AdaGrad and RMSProp. Still, it retains its advantage of dynamically and adaptively optimising the learning rate. The calculation process of the Adam algorithm is shown in Eq. (17), which includes storing the exponential decay average value v_t of the historical square gradient and maintaining the exponential decay average value m_t of the historical gradient.

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$$
$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \tag{18}$$

where, β is the learning rate, and g_t is the fitness function.

The calculation process of the Adam algorithm after introducing the deviation correction step is shown in Eq. (18) to reduce the impact of deviation on the early training of the university's cyberspace security threat detection model.

$$m_t' = \frac{m_t}{1-\beta_1^t}$$
$$v_t' = \frac{v_t}{1-\beta_2^t} \tag{18}$$

After correcting the deviation, as shown in Equation (19), it can adaptively calculate the updated step size η_t from the perspectives of gradient mean m_t' and gradient square v_t'.

$$\eta_t = \eta_{t-1} - \alpha * m_t / (\sqrt{v_t} + \varepsilon) \tag{19}$$

where, α is the Lagrange multiplier and ε is the relaxation variable.

C. Defense Methods for Cyberspace Security in Universities

According to the ConvLSTM-CNN threat detection model in Section II (B), security threats in the university's cyberspace are detected, and the university's cyberspace security defense is carried out through the following methods.

1) Security defense against DNS attack threats: For example, the Domain Name System (DNS) denial of service attack threat is prevented in the network blocking attack detected by the ConvLSTM-CNN threat detection model. DNS denial of service attack threat security defense requires a detailed record of the number of DNS requests and hit rate in the time window. Based on the network attack threat detected by ConvLSTM-CNN, threat detection is performed on DNS attack data [26]. The attack data is divided into a blacklist. Normal data is divided into a whitelist and efficient network space inside the socket is used to transfer data information from the two lists so that it is assigned to filtering work, to filter attacks and redirect them, and implement constraints on the number of DNS requests that each source IP passes through per unit time to achieve security defense. The specific approach is as follows:

a) Prohibit DNS requests from the blacklist;

b) For whitelist lists, IP can be allowed to pass through a large number of DNS requests;

c) For non-whitelist lists, it constrains the number of DNS requests that IP passes through, where the allowed non-frequent DNS requests per unit time (1-2 requests) can be limited by using the tc tool. To provide commonly used domain name responses on behalf of servers in cases where there are a large number of DNS requests and to randomly discard some requests that are not included in the DNS frequency list, filters can be set in front of the attacked server. The security defense model steps for network blocking attack threats are shown in Fig. 5.

A filter function is mounted on the NetFilter framework chain of the Linux kernel to discard DNS requests with the source IP on the blacklist. In university's cyberspace data packets, using this framework can achieve the application of custom behaviour. Blocking IP addresses on the blacklist improves filtering efficiency and compensates for the missing functionality of the original DNS protocol [27].

The created queue is set on the output port and constraints university network traffic reasonably based on routing selection. Filters, queues, and classifications together form the tc tool. The following is the operation process:

a) Create CBQ queues and apply them to network physical devices;

b) Create the classification in the CBQ queue;

c) Create filters for each category, and the filters need to be created based on the routing;

d) Routing tables can be created by matching filters.

Usually, only one queue needs to be created, with each queue containing a root classification that includes subcategories. The smaller the classification number is, the more effective it becomes. If a certain classification rule is met, the data packet can be sent using that classification, and subsequent classifications will lose effectiveness. This completes the security defense against DNS attacks that threaten the cyberspace of universities.

2) Security defense against DDoS attack threats: The essence of security defense is how to defend against DDoS attack threats detected through the ConvLSTM-CNN threat detection model [28]. In SDN networks, OpenFlow flow tables can be used for implementation. As shown in Fig. 6, in the SDN network, when an attacker controls the botnet to launch a DDoS attack against the server, the attacking university's network space traffic first reaches the edge switch S1 and S1 detects whether there is a flow table matching the flow label in the university's network traffic space. If there is one, it will be forwarded directly according to the flow table rules. If not, it will upload the flow information to the controller and wait for the controller to make a decision. After receiving the flow information, the controller executes the

ConvLSTM-CNN threat detection model to detect the traffic in the university's cyberspace and determines whether the flow is DDoS attack traffic. If it is not, it calculates forwarding and sends a flow table containing routing information to the switch. The switch forwards according to the flow table rules; If so, the controller issues a table containing discarded instruction flows to the switch, and then the switch will discard this DDoS attack. In order to avoid the overflow of switch flow table space caused by excessive traffic in the university network space, a soft timeout time for the university's cyberspace traffic is set. Since DDoS attacks on university network space traffic are usually instantaneous, the soft timeout time is set to 1 second, which means that the flow table will delete itself if it is not matched for more than a second. This completes the security defense against DDoS attacks that threaten the cyberspace of universities.

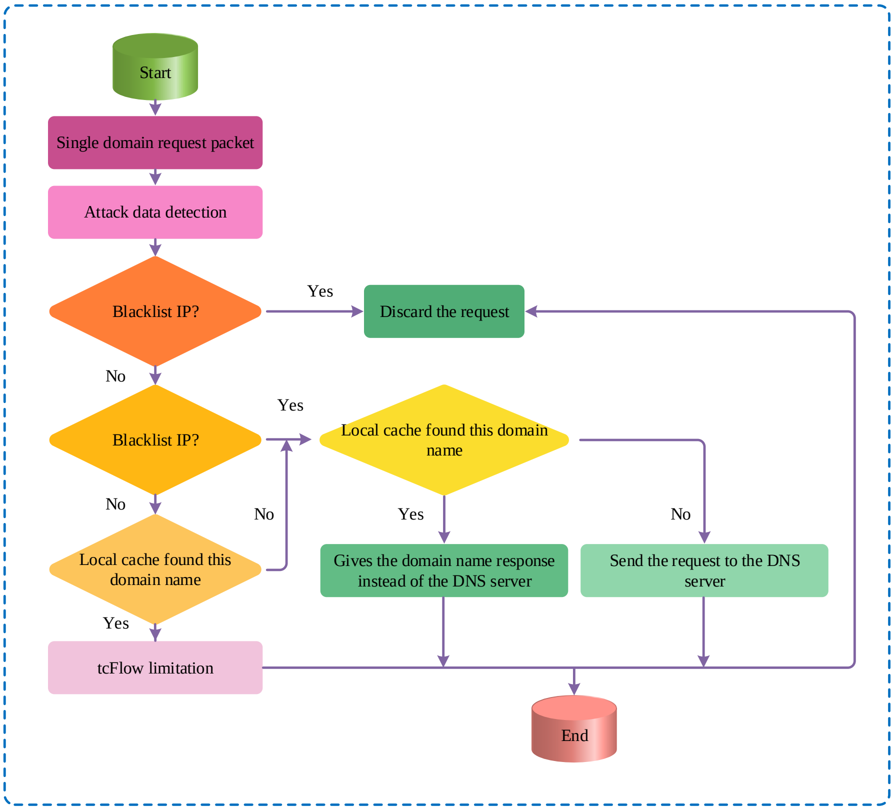
Start
Single domain request packet
Attack data detection
Blacklist IP?
Yes
Discard the request
No
Blacklist IP?
Yes
Local cache found this domain name
No
No
Yes
No
Local cache found this domain name
Gives the domain name response instead of the DNS server
Send the request to the DNS server
Yes
tcFlow limitation
End

Fig. 5. Steps for defending against network blocking attacks.

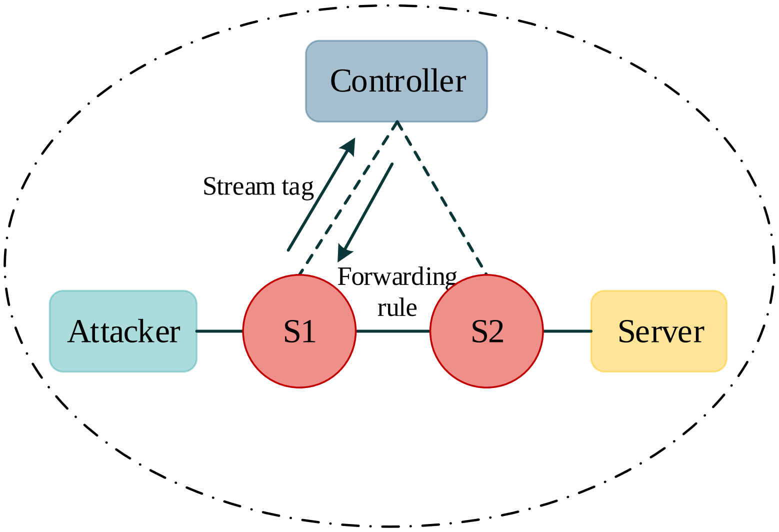

Fig. 6. Network diagram.

III. Experimental Analysis

Taking a certain university's cyberspace as the experimental object, based on big data analysis technology operating parameters, the control centre selects the Mininet+POX platform, simulates the virtual university's cyberspace environment through Mininet, follows the custom remote network interaction process, and runs the POX controller within the platform as the core selection. Based on the connectivity of remote university's cyberspace attacks, six switches are used to connect with each other. Among them, one switch is used to connect to the remote subnet, and another attack host is prepared. The other four are the teaching building, library, experimental building, and cafeteria of a certain university, and the topology of the university's network space attack is completed based on the remaining operating bandwidth of the switch. Using the built-in traffic generation tool Trafgen to attack university network space, the attack host's attack traffic is generated. Based on the attack traffic source IP address and source port number parameters, the attack content in the host file is continuously configured, and iPerf3 is used to generate and configured attack packet-related parameters. The environmental design scheme is described in Fig. 7.

After building the attack topology structure, it simulates the running environment of the university's big data analysis technology using the background traffic in the attack package to call Scapy on the normal host and complete the writing of the main function code of big data analysis technology. Based on the instructions sent and the actual response interactions generated, the traffic in the network space of attacking universities within the topology structure is resolved into multiple protocols that support attack instructions, which can control the attacking host, form identity mapping, generate mapping requests with attacking instructions, extract query ports generated by port access, and use them as fixed identity numerical markers to debug the remote attack process between the attacking host and other hosts.

To ensure the effectiveness of the experiment, the parameters of the proposed method are set as follows:

CNN model parameters: Convolutional kernel size is 3 × 3. The step size is 1, and the activation function is ReLU.

LSTM model parameters: The hidden layer size is 256, and the activation function is tanh.

Non negative matrix decomposition algorithm parameter: Select to retain the first 50 features as the dimensionality reduction result.

CNN model: Convolutional layer 1 has 32 convolutional kernels, with a kernel size of 3x3 and a step size of 1. Convolutional layer 2 has 64 convolutional kernels, with a kernel size of 3×3 and a step size of 1. The activation function is ReLU.

LSTM model: The LSTM layer has 64 hidden layer units.

Fully connected layer: The output layer has two neurons, representing the categories of normal and attack.

Loss function: Using cross entropy loss function.

The selected comparison methods use the parameters during their testing period, and will not be listed here.

The DDoS attack and DNS attack are used to verify the security defense capability of the model in this paper against the network space security of colleges and universities. At the same time, the network active security defense model based on the K-means algorithm in reference [4], the network security defense model based on the improved particle swarm optimisation algorithm in reference [5], the network security defense model oriented to digital transformation based on endogenous security framework in reference [6], the network security defense decision-making model based on timing game in reference [7], and the network data resource security defense model based on cloud computing in reference [8] are tested as a comparison method for the model in this paper. The test results are shown in Table II.

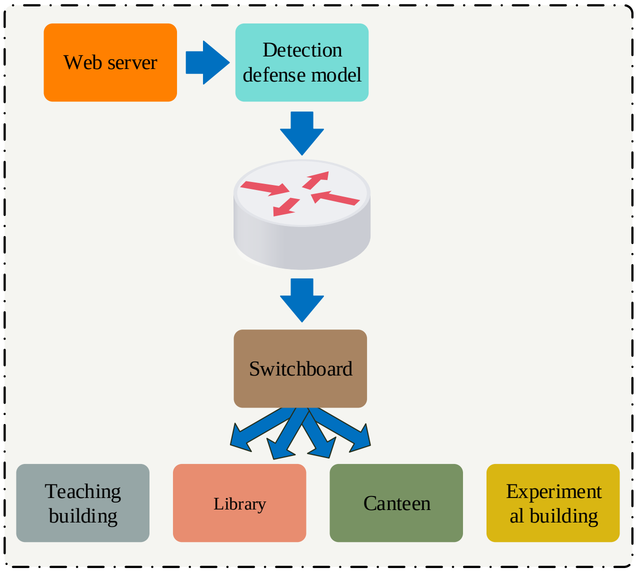

Fig. 7. Design of security defense model environment for cyberspace attacks in universities.

TABLE II. SECURITY DEFENSE RESULTS OF DIFFERENT ENVIRONMENTS IN UNIVERSITY NETWORK SPACE

Host		Teaching building	Library	Canteen	Experimental building
Model in this paper	DDoS attack	Success	Success	Success	Success
	DNS attack	Success	Success	Success	Success
Reference [4] Model	DDoS attack	Failure	Success	Success	Failure
	DNS attack	Failure	Failure	Failure	Failure
Reference [5] Model	DDoS attack	Failure	Failure	Success	Failure
	DNS attack	Success	Failure	Failure	Failure
Reference [6] Model	DDoS attack	Failure	Failure	Failure	Success
	DNS attack	Success	Success	Success	Success
Reference [7] Model	DDoS attack	Failure	Success	Success	Failure
	DNS attack	Success	Success	Success	Failure
Reference [8] Model	DDoS attack	Success	Success	Success	Failure
	DNS attack	Success	Failure	Failure	Failure

From Table II, it can be seen that in the simulated cyberspace attack environment outside a certain university's teaching building, library, cafeteria, and experimental building, the models in reference [4] and the reference [5] only successfully defended two times when they attacked DDoS and DNS cyberspace eight times, while the other six times failed. The defense effect of the models in reference [6], reference [7], and reference [8] are better than the first two models, the defense against threat attacks is above 50%, but it has not achieved the desired security defense effect. And through the model in this paper, the defense against DDoS and DNS network space attacks is all successful, indicating that the model in this paper can achieve security defense in network space under different network attacks in different environments and has strong security defense capabilities.

The experiment sets the attack host traffic value between 200-1200MB to ensure the accuracy of the test results. In this attack traffic mode, the fixed controller has 18 running windows. Based on the running window values and attack traffic parameters, it completes the statistical work of the actual intercepted traffic data of the defense software. Then it calculates the attack interception rate of the defense software.

Equation $F = (AR + TN) \div (TA + TN + FA + AR) \times 100\%$ is used to represent numerical relationships. Among them, F represents the interception rate of defense software attacks; AR represents the number of attack data correctly intercepted by the defense software, TN indicates successful tagging of attack data traffic; TA represents the number of intercepted windows supported by the controller, FA represents the set attack traffic value; traffic interception rate statistics for defense software attacks can be conducted based on different traffic attacks. The results are shown in Table III.

Based on the numerical relationship between different attack interception values, statistical analysis is conducted on the attack interception rates of the models in reference [4], reference [5], reference [6], reference [7], reference [8], and the defense software in this paper. The average is read for the attack interception rates under different supply scales. From Table III, it can be seen that the average attack interception rates of the defense software in reference [5] model and reference [6] model are 77.7% and 77.2%, respectively, indicating weak defense capabilities against university cyberspace security; The average attack interception rates of the defense software for reference [4] model, reference [7] model, and reference [8] model are 85.2%, 85.4%, and 87.9%, respectively, indicating an improvement in defense capabilities; The average attack interception rate of the defense software designed in this model is 97.6%, which is the strongest defense capability for university cyberspace security compared to the other five models.

Based on two types of network attack methods, DDoS attack and DNS attack, the congestion status of the university's cyberspace is tested for six attack defense models. The network space attack defense models of reference [4], reference [5], reference [6], reference [7], and reference [8] are used as comparative models. Fig. 8 shows the network congestion rate results of the six models for the university's cyberspace attack defense.

TABLE III. INTERCEPTION RATES OF UNIVERSITY'S CYBERSPACE ATTACKS USING DIFFERENT MODELS (%)

Host attack traffic	Reference [4] Model	Reference [5] Model	Reference [6] Model	Reference [7] Model	Reference [8] Model	Model in this paper
200	82.1	79.2	75.6	82.6	89.2	97.6
400	86.2	77.5	76.8	84.3	86.9	98.9
600	85.6	79.4	79.5	85.9	88.4	96.4
800	87.1	75.3	77.9	87.5	86.1	97.7
1000	85.2	76.1	78.2	86.7	89.4	98.3
1200	84.9	78.7	75.3	85.3	87.5	96.9

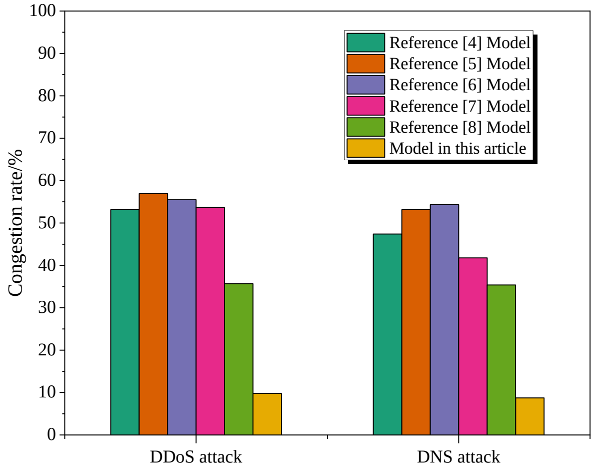

Fig. 8. Comparison of defense congestion rates of different defense methods.

Analysing Fig. 8, it can be seen that after being defended by six different network space security defense models, the congestion rate of DDoS attacks and DNS attacks is less than 60%. However, for the two different attacks, DDoS attacks and DNS attacks, the congestion rate of the model in this paper is the lowest, below 10%. The university's cyberspace security defense model proposed in this paper can effectively reduce the congestion rate of the university's cyberspace and protect the security of the university's cyberspace; the model in this paper can defend against different types of attack methods and has excellent defense capabilities against network space blocking attacks in universities.

In order to verify the effectiveness of the university's cyberspace security threat detection model in this paper and compare the impact of different numbers of feature quantities on detecting the university's cyberspace security threats, the number of feature quantities is set to 5, 10, 15, 25, 25, 30, 35, 40, and 45. The security threat detection accuracy of the university's cyberspace is obtained on different numbers of feature quantities, and the detection results are shown in Fig. 9.

From Fig. 9, it can be seen that through the feature extraction of the university's cyberspace using this model, as the number of features continues to increase, the model's accuracy in detecting security threats in the university's cyberspace has also improved. When the number of features is 35, the model has the highest accuracy in detecting security threats in the university's cyberspace, at around 0.98. However, when the number of features exceeds 35, the detection accuracy of the model in this paper for threats to the security of the university's cyberspace has begun to decline. Therefore, the model in this paper is used to reduce the dimensionality of the features in the university's cyberspace, setting the feature amount to 35, laying the foundation for the next step of the university's cyberspace security detection and improving detection accuracy.

In order to further verify the detection performance of the model in this paper for university's cyberspace security threats, DDoS attacks, DNS attacks, and DoS attacks are set as attack sources. The detection accuracy of the models in reference [4], reference [5], reference [6], reference [7], reference [8], and the model in this paper for university's cyberspace security threats are used as evaluation indicators and compared. The detection results are shown in Table IV.

From Table IV, it can be seen that the average detection accuracy of the model in reference [8] for DDoS attacks, DNS attacks, and DoS attacks is 0.76, which is relatively low in detection accuracy; The average detection accuracy of the models in reference [4], reference [5], reference [6], and reference [7] for three different network attacks is 0.82, 0.83, 0.80, and 0.80, all of which are above 0.8. Compared with the model in reference [8], the detection accuracy has been improved; the model in this paper has the highest detection accuracy for these three different types of network attacks, with an average detection accuracy of 0.98. The detection accuracy for each network attack is 0.97 or above, indicating that the model in this paper is the most effective in detecting security threats in the university's cyberspace and ensuring the security of the university's cyberspace.

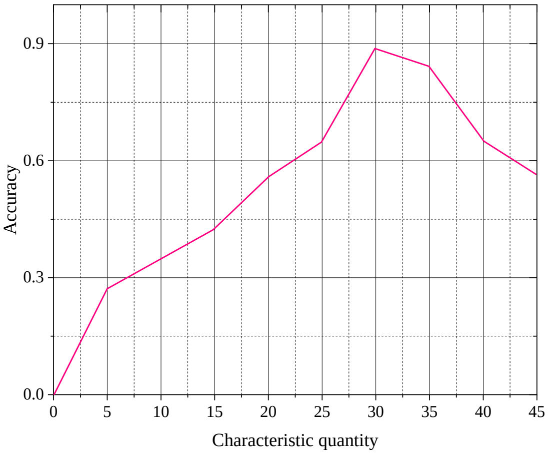

Fig. 9. University network security detection accuracy of different characteristic quantities.

TABLE IV. ACCURACY OF SECURITY DETECTION IN UNIVERSITY'S CYBERSPACE BY DIFFERENT MODELS

Model	DDoS	DNS	DoS
Reference [4] Model	0.85	0.79	0.83
Reference [5] Model	0.84	0.80	0.85
Reference [6] Model	0.79	0.81	0.81
Reference [7] Model	0.83	0.76	0.82
Reference [8] Model	0.73	0.77	0.79
Model in this paper	0.97	0.98	0.98

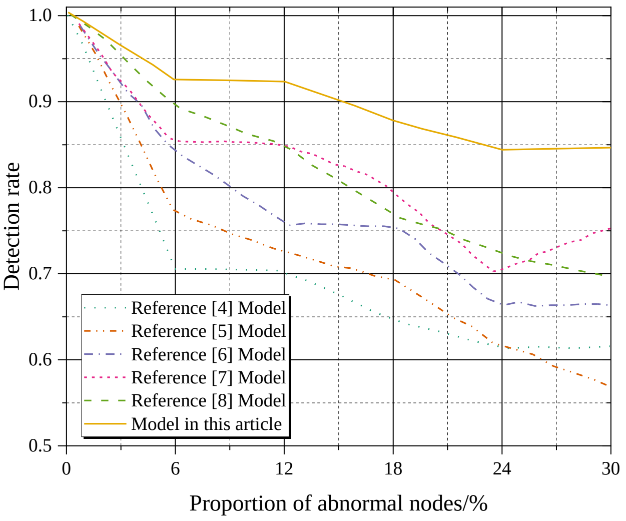

Fig. 10. Comparison of data detection rates collected by different models.

The core goal of the data collection model based on the compressed sensing principle is to ensure the high accuracy of the data. Distinguishing whether there is a threat to the security data nodes in the university's cyberspace is an important indicator to measure the model in this paper. Based on the model oriented towards the university's cyberspace security threat data in this paper, the detection rate of the algorithm is defined as the proportion of the detected university's cyberspace security threat data nodes to the total data nodes. In order to verify the effectiveness of the model proposed in this paper in detecting threats to the university's cyberspace security, 200 sensor nodes are set up in the experiment. The detection rates of the model proposed in this paper are compared with those of models in reference [4], reference [5], reference [6], reference [7], and reference [8] under different abnormal node ratios. The results are shown in Fig 10.

As shown in Fig. 10, as the proportion of nodes posing security threats to the university's cyberspace increases, the detection rates of the models in references [4], [5], [6], [7], [8] and this paper all show a downward trend. However, the detection rates of this paper are all above 0.9, while the detection rates of other models are all below 0.9. In comparison, the model in this paper has the highest detection rate. This model can effectively distinguish the types of efficient cyberspace security threat data nodes, provide decision-making support for data reliability collection, and improve the detection rate of cyberspace security threats in universities. Based on the above experimental results, it can be seen that the machine learning based university cyberspace security defense method proposed in this article has achieved significant results in practice. Compared with previous studies, the method proposed in this paper not only enhances existing research results, but also provides innovative solutions and improvement strategies.

Firstly, the method proposed in this article has achieved certain results in the defense of cyberspace security in universities. The model in this article combines data collection based on compressed sensing, security threat detection based on machine learning algorithms, and optimization parameter selection, providing comprehensive security defense support for university cyberspace.

Secondly, the method results presented in this article demonstrate novelty that was not discovered in previous studies. By introducing a compression aware data collection method, this article extracts effective information from large-scale network data, avoiding the related problems of storing and transmitting all the original data in traditional methods. Meanwhile, by using different machine learning models and feature selection methods, this article can accurately detect and prevent security threats in university cyberspace from multiple perspectives.

Finally, the research findings of this article provide new ideas and insights for the field of cyberspace security in universities. By effectively combining machine learning algorithms with knowledge in the field of network security, this method not only improves security defense capabilities, but also improves the accuracy and efficiency of security threat detection. These innovative achievements not only have significant academic significance, but also provide specific guidance and reference for the practice of cyberspace security in universities, and provide new solutions for network security protection in practical applications.

IV. Conclusion

The security threats and defense issues of university cyberspace are not only related to campus security but also closely related to the daily lives of teachers and students. There is a large amount of data, such as network traffic, log information, and system signals, in the cyberspace of universities. This study proposes a new method for university cyberspace security defense through the comprehensive application of compressed sensing based data collection method and deep learning model. After experimental verification, the model has achieved significant results in attack interception rate and provided effective protection for the security of university cyberspace. These results demonstrate the innovation and feasibility of the model, providing new ideas and insights for research and practice in the field of cyberspace security in universities. Due to the large amount of attack data, more types of data will be added to the model in the future, such as network traffic data, log data, user behavior data, etc., to improve the breadth and accuracy of security detection.

V. Data Availability

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation."

VI. Conflicts of Interest

The authors declared that they have no conflicts of interest regarding this work."

Acknowledgment

This work supported by Shanxi Provincial Universities Philosophy and Social Sciences Research Project 2022W170, Research on the Logical and Practical Path of the Construction of University Cyberspace Security.

REFERENCES

[1] C. Yin, “Application of Virtual Private Network Technology in University Network Information Security,” in *Journal of Physics: Conference Series*, IOP Publishing, 2021, p. 042071.

[2] S. Ding, Z. Zhang, and J. Xie, “Network security defense model based on firewall and IPS,” *Journal of Intelligent & Fuzzy Systems*, vol. 39, no. 6, pp. 8961–8969, 2020.

[3] Y. Niu, W. Du, and Z. Tang, “Computer Network Security Defense Model,” in *Journal of Physics: Conference Series*, IOP Publishing, 2022, p. 012041.

[4] H. Wang and W. Li, “DDosTC: A transformer-based network attack detection hybrid mechanism in SDN,” *Sensors*, vol. 21, no. 15, p. 5047, 2021.

[5] N. Oliveira, I. Praça, E. Maia, and O. Sousa, “Intelligent cyber-attack detection and classification for network-based intrusion detection systems,” *Applied Sciences*, vol. 11, no. 4, p. 1674, 2021.

[6] Q. Abu Al-Haija and M. Al-Dala’ien, “ELBA-IoT: an ensemble learning model for botnet attack detection in IoT networks,” *Journal of Sensor and Actuator Networks*, vol. 11, no. 1, p. 18, 2022.

[7] J. Zhang, L. Pan, Q.-L. Han, C. Chen, S. Wen, and Y. Xiang, “Deep learning-based attack detection for cyber-physical system cybersecurity: A survey,” *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 3, pp. 377–391, 2021.

[8] Ö. Tonkal, H. Polat, E. Başaran, Z. Cömert, and R. Kocaoğlu, “Machine learning approach equipped with neighbourhood component analysis for ddos attack detection in software-defined networking,” *Electronics (Basel)*, vol. 10, no. 11, p. 1227, 2021.

[9] K. P. Gurumanapalli, and N. Muthuluru, “Feistel Network Assisted Dynamic Keying based SPN Lightweight Encryption for IoT Security,” *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 5, p. 377-392, 2021.

[10] R. Majeed, N. A. Abdullah, and M. F. Mushtaq, “IoT-based Cyber-security of Drones using the Nave Bayes Algorithm,” *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 7, p. 422-427, 2021.

[11] I. M. Ali, “LP Based Integration of Computer Network and Security in University College,” in *Journal of Physics: Conference Series*, IOP Publishing, 2021, p. 012028.

[12] G. Sharma, “Secure Remote Access IPSEC Virtual Private Network to University Network System,” *Journal of Computer Science Research*, vol. 3, no. 1, pp. 16–27, 2021.

[13] E. Balestrieri, P. Daponte, L. De Vito, F. Picariello, S. Rapuano, and I. Tudosa, “A Wi-Fi internet-of-things prototype for ECG monitoring by exploiting a novel compressed sensing method,” Acta Imeko, vol. 9, no. 2, pp. 38–45, 2020.

[14] X. Xu et al., “Crack detection and comparison study based on faster R-CNN and mask R-CNN,” Sensors, vol. 22, no. 3, p. 1215, 2022.

[15] J. Lu, L. Tan, and H. Jiang, “Review on convolutional neural network (CNN) applied to plant leaf disease classification,” Agriculture, vol. 11, no. 8, p. 707, 2021.

[16] J. Zhang, X. Zhang, and L. Jiao, “Sparse nonnegative matrix factorization for hyperspectral unmixing based on endmember independence and spatial weighted abundance,” Remote Sens (Basel), vol. 13, no. 12, p. 2348, 2021.

[17] P. Lu and W. Chen, “Vertex centrality of complex networks based on joint nonnegative matrix factorization and graph embedding,” Chinese Physics B, 2022.

[18] A. Gallardo-Antolín and J. M. Montero, “On combining acoustic and modulation spectrograms in an attention LSTM-based system for speech intelligibility level classification,” Neurocomputing, vol. 456, pp. 49–60, 2021.

[19] A. Agga, A. Abbou, M. Labbadi, and Y. El Houm, “Short-term self-consumption PV plant power production forecasts based on hybrid CNN-LSTM, ConvLSTM models,” Renew Energy, vol. 177, pp. 101–112, 2021.

[20] Y. Ma, L. Wu, and Z. Li, “A novel face presentation attack detection scheme based on multi-regional convolutional neural networks,” Pattern Recognit Lett, vol. 131, pp. 261–267, 2020.

[21] J. Kim, J. Kim, H. Kim, M. Shim, and E. Choi, “CNN-based network intrusion detection against denial-of-service attacks,” Electronics (Basel), vol. 9, no. 6, p. 916, 2020.

[22] B.-Z. Han and W.-X. Huang, “Active control for drag reduction of turbulent channel flow based on convolutional neural networks,” Physics of Fluids, vol. 32, no. 9, p. 095108, 2020.

[23] X. Liu and X. Di, “TanhExp: A smooth activation function with high convergence speed for lightweight neural networks,” IET Computer Vision, vol. 15, no. 2, pp. 136–150, 2021.

[24] Y. Liu, J. Wang, H. He, G. Huang, and W. Shi, “Identifying important nodes affecting network security in complex networks,” Int J Distrib Sens Netw, vol. 17, no. 2, p. 1550147721999285, 2021.

[25] I. Khan, W. Farrelly, and K. Curran, “A Demonstration of Practical DNS Attacks and their Mitigation Using DNSSEC,” International Journal of Wireless Networks and Broadband Technologies (IJWNBT), vol. 9, no. 1, pp. 56–78, 2020.

[26] C. Zhou, Y. Yu, S. Yang, and H. Xu, “Intelligent immunity-based security defense system for multi-access edge computing network,” China Communications, vol. 18, no. 1, pp. 100–107, 2021.

[27] R. Kadhim and M. Gaata, “A hybrid of CNN and LSTM methods for securing web application against cross-site scripting attack,” Indones. J. Electr. Eng. Comput. Sci, vol. 21, pp. 1022–1029, 2020.

[28] S. Janarthanam, N. Prakash, and M. Shanthakumar, “Adaptive learning method for DDoS attacks on software defined network function virtualization,” EAI Endorsed Transactions on Cloud Systems, vol. 6, no. 18, pp. e6–e6, 2020.