
Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2024 SCPE. Volume 25, Issues 1, pp. 147–155, DOI 10.12694/scpe.v25i1.2329

INTELLIGENT PREDICTION OF NETWORK SECURITY SITUATIONS BASED ON DEEP REINFORCEMENT LEARNING ALGORITHM

YAN LU*, YUNXIN KUANG†, AND QIUFEN YANG‡

Abstract. The limitations of traditional network security assessment methods characterized by manual definitions and measurements, data overload, poor performance, and non-negligible drawbacks are addressed in this research. A novel network security system employing a deep learning algorithm is proposed to overcome these challenges. The research unfolds in three key phases. First, a deep self-encoding model is developed to distinguish various network attacks effectively. Subsequently, the creation of missing measurement weights enhances pattern detection, even when dealing with a limited number of training samples. Finally, the model assesses and computes attack issues, assigns impact scores to each attack, and determines the overall network security value. Experimental results demonstrate that the deep auto encoder-based deep neural network (DAEDNN), in conjunction with the proposed unique oversampling weighting (UOSW) algorithm, significantly outperforms traditional methods such as decision trees (DT), support vector machines (SVM), and long short-term memory (LSTM) models. The F1 score of UOSW surpasses these models by approximately 2.77, 10.5, and 5.2, respectively. The deep self-encoding model employed in the proposed system offers superior accuracy and recall rates, leading to more precise and efficient measurement results.

Key words: Network Security, Deep Learning, Deep Self-Encoding Model, Security Assessment, Attack Detection, UOSW Algorithm

1. Introduction. The issue of network security has now evolved into a global concern. Managing the network security landscape and fostering effective network collaboration have become imperative topics that warrant collective discussion among nations worldwide. In the growing "Global Village era", network security assumes an increasingly critical role. The author contends that prioritizing prevention over management is paramount in controlling the network environment or establishing a robust framework of order. Consequently, this study focuses on real-time network security issue prediction. The design and optimization of the proposed model demonstrate its efficacy in timely forecasting network security problems [10].

Network security forecasting encompasses continuously monitoring the ecosystem and anticipating potential network issues, such as viruses and trojans, to safeguard computer security. Through proactive network security prediction, users can identify potential threats within the current network, maintain a historical record of related incidents, and select data patterns with specific characteristics that could impact current network security concerns. Utilizing mathematical models, this approach enables the prediction and tracking of the formation and progression of network security issues. Ultimately, this process furnishes reliable information for effective computer security management, ensuring users' online safety [16].

The term "network security problem" primarily revolves around the functionality of network equipment and the extent to which the actions of network users might jeopardize network security. In simpler terms, it encapsulates the security status of an operational network. Notably, the widespread adoption of the Internet has underscored the critical role of networks in our professional endeavours and daily lives. Consequently, numerous network security challenges have emerged, impacting a broad spectrum of users. Moreover, the patterns of network access and cyberattacks have evolved significantly, marked by diverse interactions and significant developments [11].

To anticipate the state of network security more effectively for safeguarding the network environment. Such anticipation is the foundational step toward comprehending the prevailing network security challenges. We can

*Hunan Open University, Changsha, Hunan, 410004, China
†Hunan Railway Professional Technology College, Zhuzhou Hunan, 412001, China (Corresponding author: yunxinkuang6@163.com)
‡Hunan Open University, Changsha, Hunan, 410004, China

harness this data as a foundational resource for forecasting the security landscape by extracting pertinent information regarding network security issues. This entails predicting the evolution of network security issue configurations over time within the dynamic realm of network attacks and defences, thereby enabling the proactive prevention of real-time network threats and the resolution of security concerns while enhancing prediction accuracy [3].

In practical forecasting, the landscape of attacks and vulnerabilities constantly evolves, posing a formidable challenge for security managers to address them effectively. Concurrently, monitoring the other four categories of network security content typically necessitates the application of time-quantitative analysis algorithms to forecast their security issues. Consequently, these four aspects of network protection stand as viable strategies to enhance network security measures [13].

Security personnel are crucial in furnishing accurate insights into evolving security landscapes. Alongside ensuring the consistent operation and enduring lifecycles of safety measures, short-term assessments of security measures remain equally vital. Consequently, those responsible for security management should proactively engage in the timely analysis of security systems. This proactive approach should extend to future-proofing network security configurations, accounting for application impact, prevailing work environments, and potential changes [7].

The structure of the paper is as follows: A comprehensive literature review, surveying existing knowledge and research in network security prediction, is explored in section 2. Section 3 elaborates on the proposed method, which centres around the deep reinforcement learning algorithm for intelligent prediction of network security situations. Section 4 presents the results obtained from applying the proposed method, and Section 5 summarizes the key insights and contributions of the proposed methodology.

2. Literature Review. Numerous factors exert influence over the four key facets of network security issues. Regarding asset configuration, location, quantity, and priority play a central role. Meanwhile, the bedrock of business configuration hinges on metrics such as the frequency of changes and the application of business processes. The topology model, on the other hand, is chiefly influenced by the number of switching nodes and connections. Security policies' impact typically involves parameters like the frequency of changes and access control rights within security settings. Consequently, these dynamic factors remain in constant flux, and their fluctuations carry profound implications for the future security of the network [12].

When there are alterations in the topology structure, previously feasible or infeasible attack paths can transition into feasible or unfeasible ones, consequently impacting all aspects of network security. Hence, a comprehensive analysis of the four dimensions of network security issues mentioned above should be conducted through a temporal lens to gain deeper insights into future network security challenges. In practical forecasting, utilizing the variability in distribution patterns as a basis for prediction and estimation is essential. Building upon this foundation, contrasting the natural layers of forthcoming network content should serve as the framework for budgeting and prediction. Only through this approach can we enhance the accuracy of calculated location values and establish a dependable groundwork for predicting and analyzing network security issues within site maintenance [5].

The users can discern issues within the network, dissect the root causes of these problems, identify data points indicative of network security concerns, and anticipate the trajectory of network security challenges. This process entails the development of mathematical models, fostering computer network security, and disseminating information to ensure the network environment's safety. In cases where the target under examination is extensive or the model exhibits complexity, it is often necessary to define the target through specific conditions [8].

The term "event" originally found its usage in the context of military actions. In military settings, it is commonplace to forecast the unfolding of complex, multifactorial scenarios. However, within the realm of information network construction, if the aim is to create a secure and dependable network environment, there must be shared awareness of the network's current status and a comprehensive understanding of its overall security posture. By examining the frequency, volume, and nature of network security events, we investigate the threat these events pose to the network. This amalgamation of principles about different aspects of information network security informs the broader picture. It provides a summarized narrative of the current state of network security, facilitating predictions regarding its future security trajectory [1].

The analytical focus of the model encompasses extensive and intricate subjects, and the term "condition"

frequently comes into play to characterize the inherent nature of the subject under scrutiny. Originating from a military context, "accident" refers to anticipating complex developments and situational dynamics similar to military operations. In the context of research on information network security, these terms are applied to address the challenges surrounding the creation of secure and resilient networks. This approach enhances our comprehension of network security across networked environments [9].

The swift advancement of artificial intelligence, internet technology, and automatic recognition has been accompanied by deep learning methods such as decision tree classification algorithms, gradient classification algorithms, neural networks, and convolution, rapidly finding applications in computer network security recognition. As a result, they substantially bolster the computing capacity for network security data and enhance data information efficiency, thereby expanding the domain of computer network security applications in deep learning [6].

Building upon the foundation of deep learning, the aim is to attain efficient, precise, and high-quality computer network security identification and management technology. This is achieved by harnessing the algorithmic advantages of deep learning, including feature vector extraction, recognition, information optimization, and classification. A secure, cost-effective, intelligent system is meticulously designed and developed through systematic analysis encompassing its principles, architecture, functional characteristics, and platform implementation. This system furnishes a scientifically sound reference for the design, execution, and application of the computer network security identification management system, incorporating deep learning algorithms' principles [14].

The prediction of network security issues represents a technology that accomplishes a genuine amalgamation of historical event data from diverse sources. It entails the analysis of interconnected events, comprehensive research, and informed decision-making concerning the development of network security. This domain is the primary focus within knowledge technology research on network security challenges. In tandem with the continual evolution of network attacks and defensive strategies, the landscape of network security constantly evolves, heightening the expectations and complexities associated with network attack scenarios. Consequently, the demand for predictive accuracy and real-time responsiveness in network security defences has surged significantly [15].

Renowned for its emphasis on scientific rigour, knowledge acquisition, and security management with a focus on scalability, this research endeavours to gain a deeper understanding of the intricacies and functionality of deep learning algorithms. The overarching goal is to design and implement cutting-edge technologies within computer network security management. By applying these design principles and deploying innovative techniques, this study aims to elevate the stature of network security analysis technology within the framework of deep learning. Furthermore, it seeks to enhance the utilization of deep learning methodologies alongside management control technology. The research process entails designing, predicting, and analyzing computer network security through neural networks and problem-solving training. In contrast to traditional security management processes, deep learning demonstrates remarkable improvements in prediction accuracy for security measures and overall management performance. Additionally, in terms of data augmentation, machine learning is leveraged to expand the capabilities of existing security tools, preconfigure security enhancements, and develop new security protection functionalities, thereby comprehensively fortifying the scope of computer network security management [4].

In light of the shortcomings of the methods above, the author introduces a novel approach to network security analysis rooted in deep learning. To address the challenge of assigning low values to different attack types within the dataset, a weighted data processing algorithm, referred to as UOSW, is proposed. Additionally, network attacks are disentangled by utilizing an energy-absorbing auto-encoder. This methodology involves the assessment of the impact of each attack type when deriving network attack classifications and evaluating network security. Experimental results underscore the efficacy of the author's model in conducting real-time network security assessments. The evaluation outcomes exhibit superior performance, intelligence, and overall performance metrics compared to alternative models [2].

3. Proposed Network Security Assessment Model. The network security analysis model consists of three parts: incident detection, analysis, and evaluation. The structure of the network security model is shown in Figure 3.1.

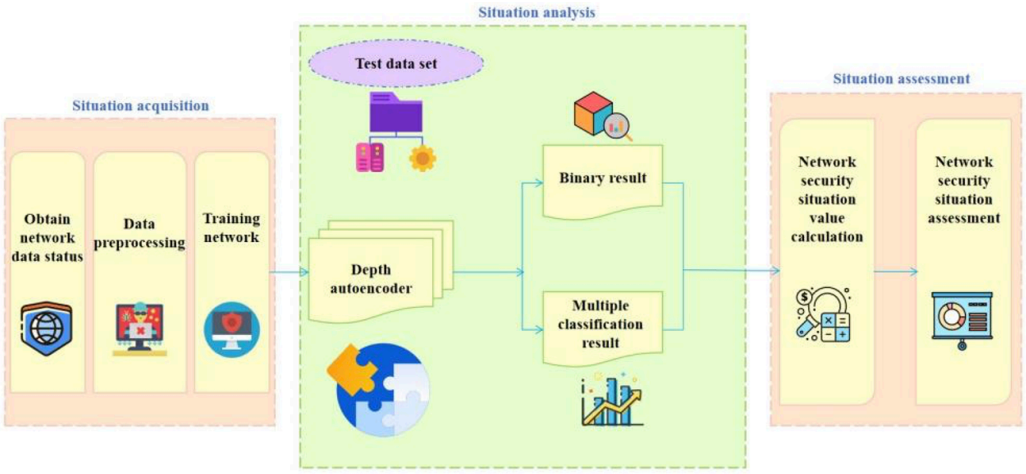

Fig. 3.1: Assessment model of network security

The merger of the three phases, namely "Nature of discovery", "Situation analysis", and "Condition assessment", constitutes what is commonly known as the "Network Security Assessment Process" or simply the "Security Assessment Process". This systematic approach is employed to assess and fortify the security of a network or information system.

(1) Nature of discovery

At this stage, traffic information on the network is received. The above NSL-KDD dataset is chosen as a road network to simulate a network processing large volumes of traffic data. After pre-planned data, the coder's deeply personal ideas in training.

(2) Situation analysis

Test datasets are integrated into training models, and binomial and multivariate distributions of test results are collected and used to calculate quantitative results for network security problems.

(3) Condition assessment

Based on the index attack distribution results, network attack probability and different network attack impact values are calculated. Also, estimate the cost of network security issues and evaluate network security issues. A detailed calculation model is presented below.

3.1. Model structure of depth self-encoder. Deep neural network (DNN) has been widely used in intrusion detection due to its accuracy and efficiency. Because DNN contains multiple hidden layers, its learning ability is significantly improved. The DAEDNN model can be applied not only to binary classification but also to multispecies. When performing a binary operation, the model function is a sigmoid function, and the model values range between 0 and 1.

$$F_{\mathrm{sgm}}(x) = (1 + e^{-x})^{-1} \tag{3.1}$$

When the model performs a multi-class task, the activation function of the model is the softmax function, which also maps the output to the range of 0 and 1:

$$F_{\mathrm{sfm}}(z_i) = e^{z_i} / \sum_{j=1}^{K} e^{z_j}, z = (z_1, z_2, \cdots, z_k) \tag{3.2}$$

K represents that the output can be divided into K classes, and Z_i represents the value obtained by each class.

3.2. Training of depth self-coder model. The training model is structured into three distinct stages:

Table 3.1: KDD - NSL data set information

Data set	Normal	DoS	Probe	R2L	U2R	Total
KDD Train+	68545	42615	10767	986	59	122972
KDD Test+	9812	7365	2530	2853	199	22759

1. The training data is initially input into the DAE network for specialized training, and the weight values are collected post-training.

2. After training the DAE model, the DAE model is integrated with the DNN model to form the DAEDNN model, which is then trained as a combined entity. To achieve the training results of the DAE model within the DAEDNN model, specific weight configurations for the DAE network are set, and the parameters of the DAE layer are designated for either learning or not learning the DNN network. During this phase, network updates are directed solely toward the DNN network.

3. Lastly, when the DAE layers encounter difficulties during training and updates for both the DAE and DNN networks are unsuccessful, adjustments to the training process are implemented. These adaptations serve the dual purpose of achieving desired learning outcomes for the DAE layer and enhancing the information message structure's visual capabilities.

3.3. Under-sampling weighted data resampling algorithm.

3.3.1. Data set description. In network security, the NSL-KDD configuration file is selected to evaluate the basic data entry for searching. The data sets used by the authors are shown in Table 3.1.

3.3.2. Under-sampling weighted data resampling algorithm. The distribution of training data among the five attack types is uneven, as illustrated in Table 3.1. Specifically, the "normal" category boasts the largest dataset, comprising 67,343 instances, while the "DoS" and "U2R" categories are comparatively limited, each consisting of only 52,995 data points. In training deep learning models, inadequate training data can hinder the model's capacity to capture all relevant data features. In contrast, an excessive amount of data may lead to overfitting. Essentially, the model learns essential features from the data, meaning that insufficient information can result in suboptimal model training, diminishing the accuracy of class recognition. In contrast, an abundance of information can lead to the opposite effect.

In data analysis, techniques like oversampling and undersampling are employed to rectify class imbalances within a dataset, commonly referred to as data resampling. Undersampling typically involves the removal of a subset of instances from an overrepresented category, while oversampling entails increasing the volume of data about a few specific data samples to achieve a more balanced dataset. To address the challenge of disparate data distribution and enhance the accuracy of minority class detection, the author introduces a weighted approach combining meticulousness, focus, and severity elements. The algorithm's steps are delineated as follows.

Set the original data set as S^1, the output data set as S^2, the data type to be resampled is $type_i$, and the original data set and sample number are S_i and x_i.

Step 1 Computing weights denoted as "A" for each data type within the dataset. When the data values for each class in the training network exhibit minimal variation, resulting in a narrow range from below the average to around the average, the network's recognition accuracy tends to be exceptionally high. Consequently, the author calculates the disparity between the observed values and the optimal model for each category, utilizing this discrepancy as a weight factor to attain a balance among all categories.

$$w_i = \sum_{i}^{n} x_i/(x_i \times n) \tag{3.3}$$

where n means that the dataset contains n categories.

Step 2 Data under-sampling: For the type with too much data, data under-sampling is performed to make the processed data sample close to the average. Use the "train_test_split" method of the sklearn library in

Table 3.2: Basic information on five types of attacks

Attack type	Description
Denial of service (DoS)	This attack renders a computer or network inoperable and unusable. A DoS attack does this by sending large amounts of traffic or data to a target.
Get permissions (User to Root, U2R)	The attack attempted to obtain the foundation's approval by illegal means.
Remote intrusion (Remote to Local, R2L)	This attack allows an attacker to access a local computer without logging in.
Detection attack (Probe)	This attack gathers network information as needed before other attacks can be launched.
Normal flow (Normal)	Normal network traffic

Table 3.3: Assessment of attack impact value

Index	Impact degree	Impact value
Confidentiality (C)	None (N) / Low (L) / High (H)	0 / 0.21 / 0.55
Integrity (I)	None (N) / Low (L) / High (H)	0 / 0.20 / 0.57
Usability (A)	None (N) / Low (L) / High (H)	0 / 0.21 / 0.58

Python to divide the data set S_i into two data sets $S_{\dot{r}\text{itrain}}$, S_{irremain}. Take S_{itrain} as the training set and add S^2, where the data volume of $S_{\dot{r}\text{itrain}}$ is $s_i = x_i \times w_i$; S_{iremain} is used for the next data oversampling operation, adding it to dataset S_{remain}.

Step 3 Data oversampling: The oversampling algorithm SMOTE is used in sampling groups with small data. The main goal of SMOTE is to create new models that are similar to existing models. The SMOTE algorithm was originally focused on two classification problems, and the following improvements were made to the algorithm due to the author's research on multi-classification problems.

(1) Merge other types of data: Combine the data set S_{remain} under-sampling processing in step 2 with a small number of data sets in the original data set, which is represented as S_{union}.

(2) Change the label. After (1), S_{union} contains data of n categories. Because the algorithm SMOTE is only for binary classification, it is necessary to distinguish the types that need oversampling from other types. Change the label of the dataset S_{union} to the same type, but different from the type.

(3) Determine the size of the data volume. To balance the data set, it is necessary to expand a small number of samples, set the expanded data size to s_i, where $s_i = x_i \times w_i$, w_i is the weight of the data type $type_i$.

(4) Data oversampling. Use the SMOTE method of the timber library in Python, combine it with other data types to generate the required data, and add it to S^2.

Repeat (1)-(4) until the oversampling operation is completed for all types whose data volume exceeds the average value.

3.4. Network attack impact value. NSL-KDD data set includes five types of network data: Normal, DoS, U2R, R2L and Probe. The basic information of the above attacks is shown in Table 3.2.

The attack impact value evaluation table is developed using the Common Vulnerability Scoring System (CVSS). The scores of confidentiality (C), integrity (I) and availability (A) are shown in Table 3.3.

The influence value (I_i) of each attack type is calculated as follows:

$$I_i = C_i + I_i + A_i \tag{3.4}$$

3.5. Quantification of the network security situation. The quantification of network security issues enables an in-depth exploration of various facets of a network. The author's approach to analyzing network

security problems typically comprises four integral components: attack analysis, estimation of attack impact, assessment of network performance security costs, and the quantitative evaluation of network security issues. The workflow for each of these sections is outlined as follows.

(1) Inspection

Randomly select several data groups from the test data set, input them into the DAEANN model, perform binary and multivariate classification, and note the distribution stops found in the second distribution.

(2) Calculate the attack impact value

The C, I, and A values of each type of attack are determined in Table 3.2 and Table 3.3, and the attack value is determined according to formula (3.5).

(3) Estimating network security issues

Network security is important to understand all the attacks on the network and the threat of each attack. Set the value of the network security problem.

$$T = \left\{ p \times \sum_{i}^{n-1} I_i \times t_i \right\} / (N - t_n) \tag{3.5}$$

Here, p is the attack probability in formula (3.1), n and N represent all n types of data and N samples, I_i represents the impact value of each type of attack, and t_i represents the number of attacks each attack, t_n represents the number of occurrences of the typical type. Because normal mode is a normal network data flow, it does not affect the network's confidentiality, integrity, or availability, so its impact score is 0, and it is only necessary to calculate the impact score of $n-1$ idle modes.

(4) Quantitative assessment of network security issues

According to the network security problem values of 0.00∼0.20, 0.21∼0.40, 0.41∼0.60, 0.61∼0.80 and 0.81∼1.00, the severity of network security problems is divided into five levels: security, low risk, medium risk, high risk and super risk.

4. Results and Discussion. The hardware environment for the experiment is Intel (R) Xeon (R) Silver processor, NVIDIAQuadroP2000 graphics card and 32GB memory. The training and test experiments were conducted on the Windows 64-bit operating system. The programming language and machine learning libraries used are Python 3.5 and TensorFlow 2.0. GPU accelerates the model's training and testing.

4.1. Evaluation indicators. The metrics used by the author are as follows.

True Positive (TP): Shows the number of models predicted to be correct when the model stalls.

False Positive (FP): A model is assumed to be normal but is a stopped model.

True Negative (TN): Refers to the number of samples that are expected to be normal and the sample to be normal.

Negative (FN): Indicates a time when the predicted sample is stopped but is a normal sample.

In the following formula, P_T, P_F, N_T and N_F represent true positive, false positive, true negative and false negative, respectively.

Precision (P): Indicates the correct attack sample frequency predicted by the model. The higher the accuracy rate, the lower the false alarm rate. It can be expressed as

$$P = P_T/(P_T + P_F) \tag{4.1}$$

$$R = P_T/(P_T + N_F) \tag{4.2}$$

F1 value: It means that the accuracy and recall of the model are comprehensively considered. It can be expressed as

$$F = 2PR/(P + R) \tag{4.3}$$

4.2. Model Vs classification results. Using the UOSW algorithm, the KDD Test data package was used to test five models: DT, SVM, LSTM, DAEDNN, and DAENDD, and precision, recall, and F1 values were selected as test parameters to compare and analyze different models. The test scores of various models

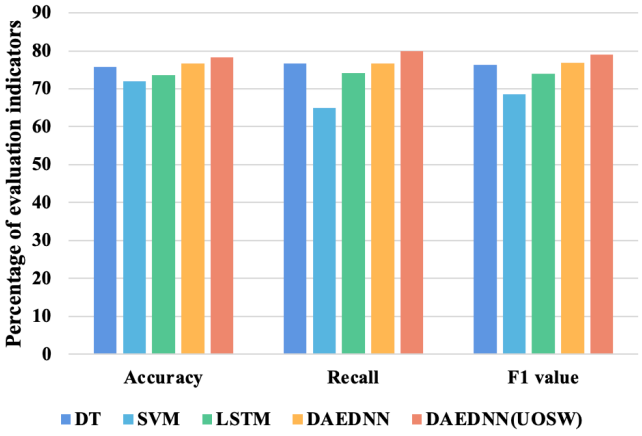

Fig. 4.1: Scores of various indicators of different models

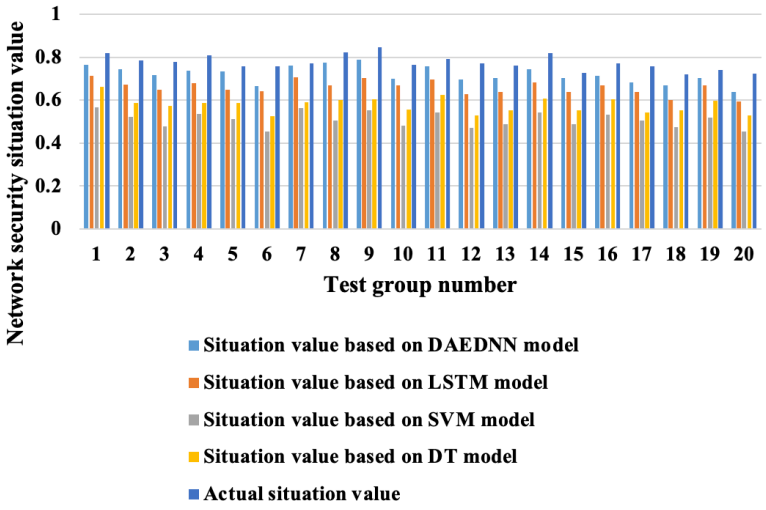

Fig. 4.2: Network security situation value of 20 groups of tests

are shown in Figure 4.1. The setting in the figure represents the percentage of test results; the higher the value, the better the model. It shows that the UOSW model outperforms the other four models regarding precision, recall, and F1 value. Experimental results show that UOSW improves the recall and accuracy of attack types on several training data but does not reduce the detection of attacks with more training examples.

It should be noted that DAEDNN has higher accuracy, recovery speed, and wider capabilities after combining with the original UOSW algorithm. Compared with DT, SVM and LSTM models, the F1 value UOSW increased by approximately 2.77, 10.5 and 5.2, respectively.

To increase the effectiveness of network security problems, the same number of test samples are selected from the test data, and different models estimate the importance of network security problems; the results of 20 groups of network security indicators.

5. Conclusion. The author introduces an innovative approach to network security, leveraging the power of deep learning. This method represents a pioneering endeavour in constructing a DAEDNN model that seamlessly integrates autoencoders and deep neural networks to identify network attacks. Drawing upon the insights

gathered from the analysis, the model calculates outage probabilities and impact values, thereby enabling the computation of network security costs. This multidimensional assessment of security issues offers a specific understanding of its potential impact on network security. The results from experimental trials underscore the efficacy of the author's proposed model, demonstrating its superiority over alternative models in both binary and multi-class classification scenarios. Moreover, the UOSW algorithm further enhances the model's ability to accurately detect attack patterns, particularly in situations with limited training samples. This augmentation empowers the model to effectively discern and evaluate diverse types of network attacks and associated security concerns, ultimately contributing to a more robust and comprehensive network security framework.

Acknowledgements. The study was supported by

1. Scientific research project of Hunan Provincial Department of Education "Research on optimization of DDPG algorithm based on Actor-Critic framework" (No.: 21C1186)

2. Hunan Vocational College Education and Teaching Reform Research Project "Research on Classroom Teaching Evaluation in Higher Vocational Education Based on Deep Learning" (No.: ZJGB2021189)

3. Hunan Natural Science Foundation project "Research on yawn detection algorithm based on AdaBoost "(No.: 2021J60038)

REFERENCES

[1] A. Akhunzada, A. Gani, N. B. Anuar, A. Abdelaziz, M. K. Khan, A. Hayat, and S. U. Khan, *Secure and dependable software defined networks*, Journal of Network and Computer Applications, 61 (2016), pp. 199–221.

[2] A. Årnes, K. Sallhammar, K. Haslum, T. Brekne, M. E. G. Moe, and S. J. Knapskog, *Real-time risk assessment with network sensors and intrusion detection systems*, in Proceedings of the Computational Intelligence and Security: International Conference, vol. 3802, Xi'an, China, 2005, Springer, pp. 388–397.

[3] M. Baykara and R. Das, *A novel honeypot based security approach for real-time intrusion detection and prevention systems*, Journal of Information Security and Applications, 41 (2018), pp. 103–116.

[4] V. Chastikova and V. Sotnikov, *Method of analyzing computer traffic based on recurrent neural networks*, Journal of Physics: Conference Series, 1353 (2019), p. 012133.

[5] M. Husák, J. Komárková, E. Bou-Harb, and P. Čeleda, *Survey of attack projection, prediction, and forecasting in cyber security*, IEEE Communications Surveys & Tutorials, 21 (2018), pp. 640–660.

[6] Y. N. Kunang, S. Nurmaini, D. Stiawan, and B. Y. Suprapto, *Attack classification of an intrusion detection system using deep learning and hyperparameter optimization*, Journal of Information Security and Applications, 58 (2021), p. 102804.

[7] N. M. Kuznetsova, T. V. Karlova, and A. Y. Bekmeshov, *Methods of timely prevention from advanced persistent threats on the enterprise automated systems*, in Proceedings of the International Conference on Quality Management, Transport and Information Security, Information Technologies, Saint Petersburg, Russian Federation, 2022, IEEE, pp. 158–161.

[8] N. Lal, S. M. Tiwari, D. Khare, and M. Saxena, *Prospects for handling 5g network security: Challenges, recommendations and future directions*, Journal of Physics: Conference Series, 1714 (2021), p. 012052.

[9] D. C. Le and N. Zincir-Heywood, *A frontier: Dependable, reliable and secure machine learning for network/system management*, Journal of Network and Systems Management, 28 (2020), pp. 827–849.

[10] H. Lin and J. Wang, *Pinning control of complex networks with time-varying inner and outer coupling*, Mathematical Biosciences and Engineering, 18 (2021), pp. 3435–3447.

[11] Z. Lin, J. Yu, and S. Liu, *The prediction of network security situation based on deep learning method*, International Journal of Information and Computer Security, 15 (2021), pp. 386–399.

[12] S. Rathore, P. K. Sharma, V. Loia, Y.-S. Jeong, and J. H. Park, *Social network security: Issues, challenges, threats, and solutions*, Information Sciences, 421 (2017), pp. 43–69.

[13] B.-C. Seo and W. F. Krajewski, *Statewide real-time quantitative precipitation estimation using weather radar and nwp model analysis: Algorithm description and product evaluation*, Environmental Modelling & Software, 132 (2020), p. 104791.

[14] P. Wei, Y. Li, Z. Zhang, T. Hu, Z. Li, and D. Liu, *An optimization method for intrusion detection classification model based on deep belief network*, IEEE Access, 7 (2019), pp. 87593–87605.

[15] H. Wu, Q. Gao, X. Tao, N. Zhang, D. Chen, and Z. Han, *Differential game approach for attack-defense strategy analysis in internet of things networks*, IEEE Internet of Things Journal, 9 (2021), pp. 10340–10353.

[16] H. Zhang, C. Kang, and Y. Xiao, *Research on network security situation awareness based on the lstm-dt model*, Sensors, 21 (2021), p. 4788.

Edited by: Venkatesan C
Special issue on: Next Generation Pervasive Reconfigurable Computing for High Performance Real Time Applications
Received: Jun 5, 2023
Accepted: Sep 28, 2023