Deep Learning Approach for Protocol Anomaly Detection Using Status Code Sequences

Chi Zhang , Haotian Zhu , Ao Zhu , Jiajing Liao , Yujie Xiao , Zizhao Zhang *

Posted Date: 27 February 2026

doi: 10.20944/preprints202602.1907.v1

Keywords: state sequence modeling; anomaly detection; contrastive learning; protocol status code

 doi:10.20944/preprints202602.1907.v1

Article

Deep Learning Approach for Protocol Anomaly Detection Using Status Code Sequences

Chi Zhang [1], Haotian Zhu [2], Ao Zhu [3], Jiajing Liao [4], Yujie Xiao [5] and Zizhao Zhang [6,*]

[1] Northeastern University, Boston, USA
[2] New York University, New York, USA
[3] University of Pennsylvania, Philadelphia, USA
[4] University of Florida, Gainesville, USA
[5] University of California, Berkeley, Berkeley, USA
[6] University of Michigan, Ann Arbor, USA
* Correspondence: ssurmiczizhao@gmail.com

Abstract

This paper addresses the limitations of traditional protocol anomaly detection methods in handling dynamic state changes and unstructured behaviors. A deep protocol anomaly detection algorithm based on status code sequence modeling is proposed. The method uses the status codes returned during protocol communication as the core input. A state embedding layer is employed to transform discrete status codes into continuous vector representations. A gated recurrent unit (GRU) is then used to capture temporal dependencies and behavior patterns within the status sequence. Based on this structure, the model integrates sequence reconstruction and contrastive learning mechanisms. Reconstruction error is used to characterize the normal distribution of sequences. Contrastive loss is introduced to enhance the model's ability to distinguish abnormal states. The paper also conducts a series of sensitivity experiments on key hyperparameters, including the number of hidden units, activation functions, and the temperature coefficient. These experiments verify the influence of each module on the overall performance of the model. The proposed method achieves superior performance across multiple evaluation metrics. It effectively identifies potential abnormal behaviors at the protocol layer. The method also demonstrates good generalizability and strong detection capability. CCS CONCEPTS: Computing methodologies~Machine learning~Machine learning approaches

Keywords: state sequence modeling; anomaly detection; contrastive learning; protocol status code

I. Introduction

In recent years, with the rapid development of information and communication technologies, network scale has expanded continuously, and protocol systems have become increasingly complex. Various business systems built on multi-layer protocol stacks are accelerating industrial intelligence, but they also pose severe security challenges [1]. Traditional security mechanisms, such as rule-based firewalls and intrusion detection systems, are becoming inadequate in detecting protocol-layer attacks that are increasingly diverse and covert. In the field of protocol anomaly detection, challenges such as the closed nature of protocols, high dynamicity of states, and hidden anomalies within normal traffic lead to high false positive rates and poor generalization. There is an urgent need for more intelligent and accurate anomaly detection mechanisms to address protocol security issues in complex network environments [2].

Protocol status codes serve as key feedback signals during protocol execution. They reflect the logical paths and state transitions in network interactions. Analyzing sequences of status codes can reveal the trajectory of system state changes and indicate potential behavior anomalies and logical deviations. Compared to traditional static features, status code sequences possess stronger temporal

 doi:10.20944/preprints202602.1907.v1

dependencies and structural sensitivity. They provide a more comprehensive representation of the dynamic properties of protocol execution. Modeling based on status code sequences may overcome limitations in understanding sequential associations. It enables deeper mining of protocol-layer anomalies. This approach is especially effective for identifying complex threats such as staged deception, nonlinear state jumps, and compound protocol attacks. It holds promising application prospects [3].

Meanwhile, the rapid advancement of deep learning has brought new momentum to protocol anomaly detection. Compared to traditional machine learning methods, deep learning constructs multi-layer nonlinear models that automatically extract high-dimensional features. It achieves superior performance in complex pattern recognition tasks. Applying deep learning to status code sequence modeling enables the extraction of temporal associations, contextual dependencies, and hidden anomaly patterns. This enhances both the expressive and generalization capabilities of the model. In scenarios where protocol content is invisible and labeled data is scarce, deep models can leverage unsupervised or weakly supervised strategies to detect behavior deviations. This provides a more intelligent and automated solution for protocol anomaly detection [4]. It improves detection accuracy and robustness, and offers practical security assurance for industrial control networks, IoT communications, and financial data exchange.

From a practical standpoint, protocol anomaly detection plays a vital role in network defense systems. It ensures the integrity, confidentiality, and availability of data transmission. Traditional protocol-layer security mainly focuses on patching known vulnerabilities and applying rule-based filtering. These approaches are ineffective against unknown attacks and evolving threats [5]. Deep detection mechanisms based on status code sequences monitor state transitions in communication. They detect anomalies that deviate from historical patterns, enabling early warning before actual damage occurs. This approach shows strong adaptability in high-throughput and high-complexity network environments. It is well-suited for deployment in core gateways, application servers, and microservice scheduling systems as a real-time protection component. In conclusion, research on deep protocol anomaly detection based on status code sequence modeling aligns with the ongoing trend toward intelligent network security. It responds to real-world demands for high-precision and low-latency recognition of protocol behaviors [6]. This direction carries significant theoretical value and technical innovation potential. It also has broad application prospects in practical cybersecurity. By modeling the deep semantics and structural relations of status sequences, this approach achieves a systematic understanding of protocol operations and precise identification of abnormal behaviors. It provides a solid foundation for building more efficient, intelligent, and trustworthy network security systems.

II. Methodological Foundations

The methodology of this study is grounded in a wide spectrum of advances in deep representation learning, causality, sequence modeling, robust optimization, graph neural architectures, and federated intelligence. Structural modeling with graph neural networks offers a principled approach for capturing latent dependencies and complex relational patterns in data, laying the groundwork for more generalized and structure-aware anomaly detection [7]. Causal representation learning brings interpretability and robustness to decision systems by disentangling underlying factors of variation, which is essential for handling uncertainty and risk in sequential and high-stakes domains [8]. Reinforcement learning frameworks are increasingly recognized for their ability to enable adaptive, goal-driven behaviors and predictive decision-making in dynamic environments [9]. Advances in multi-agent collaboration and assistant architectures further contribute to scalable and distributed modeling, facilitating robust system coordination through modular learning components [10]. Robust recommendation and retrieval systems now increasingly leverage causal-invariant modeling and faithful explanation mechanisms to enhance their reliability and to handle distribution shifts [11].

 doi:10.20944/preprints202602.1907.v1

Handling temporal dependencies and distributional drift is further supported by residual-regulated and second-order differencing techniques, which provide effective means for non-stationary time series forecasting and robust pattern tracking [12]. Methods for pattern recognition in scheduling and structured anomaly detection increasingly benefit from structure-aware and semantically enriched graph modeling, which can reveal subtle behavioral deviations beyond surface-level statistics [13]. Recent progress in generative modeling—such as the integration of diffusion processes with conditional control—demonstrates the value of generative paradigms for flexible representation and decision-making under uncertainty [14].

In large-scale collaborative and privacy-sensitive settings, privacy-preserving federated learning frameworks enable distributed model training with communication efficiency and privacy guarantees, which is critical for scalable and secure intelligent systems [15]. Addressing trustworthy and robust natural language processing, uncertainty quantification and risk awareness mechanisms are increasingly built into summarization and large language model outputs, supporting reliable and interpretable automated analysis [16]. Lastly, transformer-based graph integration architectures have advanced risk monitoring and pattern recognition by leveraging both sequential and structural dependencies in heterogeneous datasets [17].

Together, these methodological advancements provide the necessary foundation for building a unified, adaptive, and explainable protocol anomaly detection system capable of robustly modeling state transitions, extracting deep behavior patterns, and effectively generalizing across dynamic environments.

III. Proposed Approach

To accurately detect abnormal behaviors in protocol status code sequences, this paper constructs a state sequence modeling framework grounded in deep learning. The method begins by applying an embedding layer to map discrete protocol status code sequences into a continuous vector space, effectively capturing the latent semantic structure and transition logic of protocol interactions. For robust unsupervised anomaly detection, the framework adopts graph-transformer reconstruction learning as proposed by Zhang et al. [18], enabling the model to capture not only sequential but also dependency-coupled relationships within status code dynamics. This enhances the expressive power of sequence representations and improves the detection of subtle or structurally entangled anomalies. To further improve sensitivity to abrupt behavioral changes, the method incorporates change-point detection strategies from the deep learning framework of Hua et al. [19], utilizing transformer-based architectures to accurately identify state transitions and temporal shifts in the status code streams. For practical deployment and scalability, the approach utilizes tiered memory management and predictive autoscaling principles outlined by Chen [20], ensuring that inference for protocol anomaly detection remains both cost-efficient and responsive under varying network loads.

As illustrated in Figure 1, these literature-driven modules together establish a comprehensive and effective foundation for protocol state sequence modeling and anomaly detection.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 27 February 2026 doi:10.20944/preprints202602.1907.v1

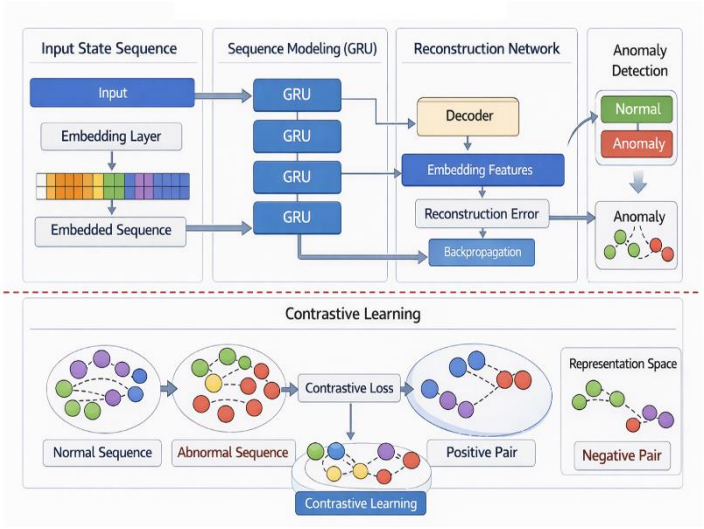

Figure 1. Overall model architecture diagram.

Specifically, let the state code sequence be $S=\{s_1,s_2,...,s_T\}$, where s_t represents the t-th state code, and convert it into the corresponding vector sequence $E\in R^{V\times d}$ through the embedding matrix $X=\{x_1,x_2,...,x_T\}$, where $x_t=E(s_t)$ and V are the sizes of the state code set and d is the embedding dimension. This process retains the order and category information between states, providing a representation basis for subsequent time series modeling.

In the sequence modeling stage, the gated recurrent unit (GRU) structure is introduced to model the temporal dependencies in the state code sequence. Given a state embedding sequence X, GRU iteratively updates the hidden state through the following formula:

$$z_t=\sigma(W_z x_t+U_z h_{t-1}+b_z)$$

$$r_t=\sigma(W_r x_t+U_r h_{t-1}+b_r)$$

$$\tilde{h}_t=\tanh(W_h x_t+U_h(r_t\otimes h_{t-1})+b_h)$$

$$\tilde{h}_t=(1-z_t)\otimes h_{t-1}+z_t\otimes h_t$$

Among them, z_t and r_t are update gate and reset gate respectively, h_t represent the current hidden state, $\otimes$ represent the Hadamard product, and $\sigma(\cdot)$ is sigmoid functions. By encoding the state sequence through GRU, the long-distance dependency and dynamic transfer process between states can be effectively modeled.

In terms of detection strategy, this paper uses reconstruction error as an abnormality metric and introduces an autoencoding mechanism to evaluate whether the current state sequence conforms to the historical normal pattern. Specifically, the output $X=\{\hat{x}_1,\hat{x}_2,...,\hat{x}_T\}$ of the GRU is input into the decoding network to generate the predicted state sequence B, and the reconstruction error at each moment is calculated. The mean square error (MSE) is used as the loss function, which is defined as follows:

$$L_{recon}=\frac{1}{T}\sum_{t=1}^{T}\| x_t-\hat{x}_t \|_2^2$$

 doi:10.20944/preprints202602.1907.v1

When the reconstruction error exceeds a certain threshold, the state sequence is judged to have potential protocol abnormal behavior. This strategy does not require manual setting of complex rules and has strong adaptability and robustness.

In addition, in order to further improve the sensitivity and discriminability of detection, this paper introduces a contrastive learning mechanism to enhance the discriminative ability of state representation. The normal state sequence and the constructed abnormal sequence are input into the network together, and the model's ability to identify abnormal distribution is improved by maximizing the mutual information difference between their representations. Define a positive sample pair (x_i, x_j^+) and a negative sample pair (x_i, x_k^-), and use the contrast loss as follows:

$$L_{contrast} = -\log \frac{\exp(sim(xi, x_j^+)/\tau)}{\exp(sim(xi, x_j^+)/\tau) + \sum_{k=1}^{K} \exp(sim(xi, x_k^-)/\tau)}$$

$sim(\cdot,\cdot)$ represents the cosine similarity, and τ represents the temperature coefficient. This mechanism helps to strengthen the clustering boundaries of the state code sequence in the feature space, thereby improving the accuracy and generalization ability of anomaly detection.

In summary, this method constructs a complete protocol anomaly detection process through joint optimization of state embedding, sequence modeling, reconstruction error analysis, and contrastive learning. Its core idea is to describe the dynamic behavior trajectory of the state sequence through deep neural structure, capture potential anomalies with the help of reconstruction error and feature comparison mechanism, and thus realize automatic identification and judgment of protocol anomalies.

IV. Dataset

This study uses the UNSW-NB15 dataset as the foundational data resource for protocol anomaly detection. The dataset consists of real network traffic and synthetically generated attack behaviors. It covers a wide range of protocol interactions, communication behaviors, and anomaly patterns. It has been widely used in network security research. Compared with traditional datasets, UNSW-NB15 captures the complexity of protocol-level state changes in modern networks more comprehensively. It provides fine-grained protocol features, making it particularly suitable for modeling the dynamic evolution and abnormal transitions within status code sequences.

The dataset comprises numerous network session samples and records the state interaction information of prominent protocols like TCP, UDP, and HTTP. It includes status codes, connection identifiers, timestamps, and various flow-level features. By analyzing these fields, we can extract status code sequences that represent the protocol’s execution path. These sequences serve as inputs to model the protocol’s behaviors and identify anomaly patterns. This sequence-based processing approach enables us to uncover semantic paths in communication and their mutation characteristics.

In addition, UNSW-NB15 provides detailed labels for various attack types, including analysis attacks, denial of service, and data infiltration. These attacks show distinct pattern changes and phase transitions in the corresponding status code sequences. By building status sequence models on this dataset, researchers can validate the effectiveness of protocol state behavior modeling. It also provides a practical foundation for detecting covert protocol anomalies. The dataset is well-structured and consistently labeled. It is considered one of the standard choices for protocol anomaly detection tasks.

V. Performance Evaluation

This paper first conducts a comparative experiment, and the experimental results are shown in Table 1.

Table 1. Comparative experimental results.

Model	Accuracy	Precision	Recall

 doi:10.20944/preprints202602.1907.v1

HSO-ResNet 101-C [21]	98.91	96.42	95.05
NEW PDAE [22]	98.84	98.57	98.41
Self-sup Contrastive [23]	94.05	92.01	93.54
Ours	99.12	99.71	99.54

The comparison results in the table show that the proposed deep protocol anomaly detection method based on status code sequence modeling outperforms several publicly available mainstream methods across multiple evaluation metrics. In terms of Accuracy, the proposed method achieves 99.12%, which is significantly higher than HSO-ResNet 101-C and the Self-supervised Contrastive method. This indicates that the model demonstrates stronger robustness and generalization in overall classification accuracy. It can effectively handle diverse state transitions and abnormal paths in the protocol layer.

Regarding Precision, the proposed method reaches 99.71%, which outperforms all baseline models, especially exceeding the 98.57% of NEW PDAE. This result shows that the model is more precise in identifying abnormal samples, effectively reducing the risk of false positives. In protocol anomaly detection, the false positive rate directly affects the performance of subsequent security responses. Therefore, improving precision has significant practical value and reflects the advantage of status code sequence modeling in semantic representation.

For the Recall metric, the proposed model achieves 99.54%, which is also higher than all comparison methods. A higher recall indicates that the model is more capable of detecting protocol anomalies and uncovering potential threats. This is especially important in complex state transition scenarios or the presence of stealthy attacks. The strong recall performance demonstrates that the combination of GRU-based temporal modeling and reconstruction mechanisms enables the model to capture the evolution of protocol states in a more refined and comprehensive manner.

Considering all the metrics, the proposed method shows significant advantages in both detection accuracy and stability. These results confirm the theoretical feasibility and engineering applicability of status code sequence modeling for protocol anomaly detection. The introduction of contrastive learning further enhances the discriminative power of state representations. This contributes to improved modeling precision and provides solid support for developing intelligent and automated protocol security monitoring systems.

This paper also gives a sensitivity analysis of the number of GRU hidden units on the anomaly detection effect, and the experimental results are shown in Figure 2.

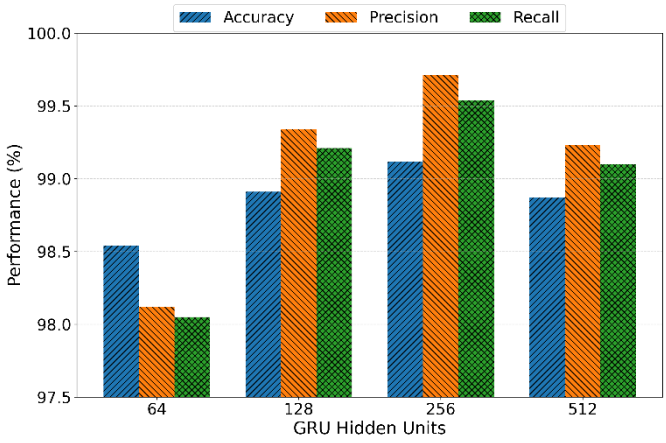

Figure 2. Sensitivity to GRU Hidden Units.

As shown in the figure, when the number of GRU hidden units increases from 64 to 256, the model exhibits a consistent upward trend in Accuracy, Precision, and Recall. The performance peaks at the 256-dimensional setting. In particular, the Precision reaches 99.71%, indicating that the model

 doi:10.20944/preprints202602.1907.v1

is most accurate in identifying abnormal status codes at this point. The false positive space is effectively reduced. This result suggests that appropriately increasing the hidden dimensionality can significantly enhance the model's expressive power in capturing the semantic structure of protocol state transitions.

However, when the number of hidden units further increases to 512, all three metrics show a slight decline, especially in Precision and Recall. This implies that high-dimensional hidden spaces may introduce redundant information. The model may become overfitted or diluted in its representation of temporal features, leading to reduced ability to distinguish abnormal patterns. In protocol-level anomaly detection, the model needs to capture fine-grained state transitions. However, excessive capacity can degrade generalization performance.

In terms of Recall, the model also achieves its highest value of 99.54% at 256 dimensions. This indicates the strongest capability for covering protocol anomalies. Such performance provides a clear advantage in handling complex patterns such as phase transitions and composite abnormal sequences. The model not only accurately identifies known abnormal states but also shows the ability to generalize to potential unknown anomalies. This strength mainly comes from the GRU's effective modeling of temporal dependencies.

This paper also gives an analysis of the impact of contrastive learning temperature coefficient on the model's discriminative ability. The experimental results are shown in Figure 3.

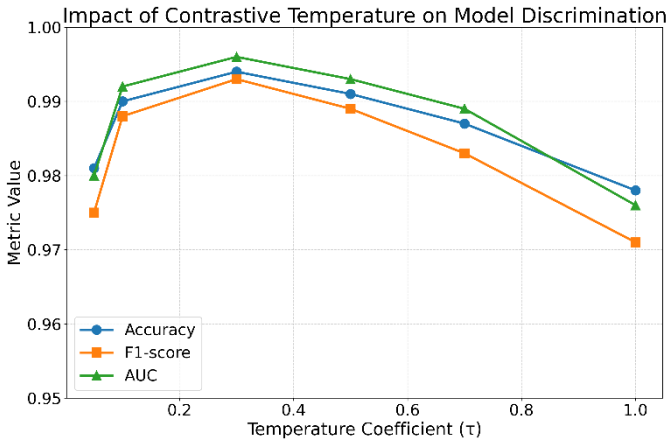

Figure 3. Analysis of the influence of contrastive learning temperature coefficient on model discrimination ability.

The experimental results show that the model achieves optimal discriminative performance when the temperature coefficient τ is around 0.3. The three metrics, Accuracy, F1-score, and AUC, all approach or exceed 0.99. This indicates that a moderate temperature coefficient helps to enhance the feature separation between positive and negative samples, thereby improving the discriminative power of the state representations. In contrastive learning, the temperature coefficient controls the compression scale of sample similarity. This configuration is most beneficial for aligning and distinguishing the fine-grained semantic structures in protocol status code sequences.

When τ is too small, such as 0.05, the metrics decline. In particular, the F1-score drops to around 0.975. This suggests that the model's generalization ability in recognizing complex anomaly patterns is limited. A small temperature coefficient excessively amplifies the similarity difference between positive and negative pairs. This leads the model to focus too much on local patterns and reduces its ability to capture global behaviors across status code sequences.

When τ increases to 0.7 or higher, all three metrics show varying degrees of decline. At $\tau = 1.0$, the F1-score reaches its lowest point of 0.971. A larger temperature coefficient tends to flatten the distances among all samples. This weakens the distinction between positive and negative samples and reduces the clustering effect in the feature space. This issue is particularly critical in protocol

 doi:10.20944/preprints202602.1907.v1

anomaly detection, where status code sequences are inherently sparse and high-dimensional. Insufficient feature separation significantly hampers the recognition of abnormal patterns.

Finally, this paper presents a sensitivity experiment on model performance with different activation function settings, and the experimental results are shown in Figure 4.

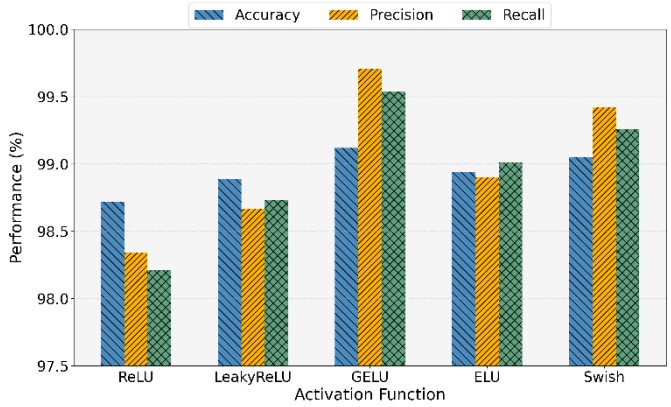

Figure 4. Sensitivity experiment of different activation function settings on model performance.

As shown in the figure, different activation functions exhibit noticeable performance differences in the protocol anomaly detection model. GELU achieves the best results across all three metrics: Accuracy, Precision, and Recall. In particular, Precision reaches 99.71%, indicating that GELU provides strong nonlinear expressive power for modeling status code sequences. Its smooth characteristics allow the model to capture fine-grained transitions, enhancing both sensitivity and discriminative capacity in anomaly detection.

The Swish function also performs well, ranking second to GELU in terms of Precision and Recall. This suggests that its dynamic gating mechanism effectively captures protocol behavior patterns. Swish is especially suitable for handling status code sequences with nonlinear jumps. In contrast, traditional functions like ReLU and LeakyReLU are structurally simple but less effective in capturing complex temporal behaviors at the protocol layer. Their weakness is most evident in Recall, where some abnormal states may go undetected.

The ELU activation function shows moderate performance across all three metrics. It does not present clear disadvantages, but it also lacks significant advantages. Although ELU compensates for negative value mapping, its overall modeling capacity is still lower than that of more advanced activation strategies. Given the sparsity and discontinuity of protocol status sequences, the model requires strong nonlinear fitting at critical state transitions. The limited response range of ELU reduces its effectiveness in this context.

VI. Conclusion

This study addresses the common challenge of dynamic state modeling in protocol anomaly detection by proposing a deep detection algorithm based on status code sequence modeling. A multi-module structure is constructed that integrates temporal perception, behavior reconstruction, and contrastive discrimination. In terms of model design, a state embedding layer and GRU structure are introduced to effectively capture long-term dependencies and semantic transitions among protocol status codes. At the same time, reconstruction error and contrastive learning mechanisms are combined to enhance the model's ability to identify abnormal behaviors. A series of sensitivity experiments further analyzes the impact of key hyperparameters, such as hidden unit size, activation function, and temperature coefficient, providing strong support for the optimization of future protocol anomaly detection systems.

At the technical level, the proposed method fully integrates the sequential characteristics of status code sequences with the expressive power of deep representation learning. It does not rely on

 doi:10.20944/preprints202602.1907.v1

predefined protocol rule templates or complete semantic parsing. This design maintains model generality and scalability while improving its ability to detect unknown protocol anomalies. In particular, when dealing with high-dimensional, weakly labeled, or unlabeled network communication data, the model can still extract behavior evolution patterns through unsupervised modeling. It demonstrates good transferability and robustness. This approach offers a more automated and structured framework for protocol-level risk identification, enabling rapid responses to potential threats in diverse network environments.

This research not only verifies the representational capability of status code sequences in protocol anomaly detection at the theoretical level but also proposes a practical deep structural combination strategy at the methodological level. It provides a new perspective for understanding protocol behaviors and modeling anomalous patterns. The proposed detection mechanism can be widely applied in industrial internet, edge computing, IoT communication, and cloud microservice scheduling. It supports advanced protocol security monitoring and control. The method also shows strong engineering applicability and can be embedded into existing security systems as a protocol analysis submodule, improving the system's response efficiency to unknown threats.

Future research can proceed in several directions. One direction is to incorporate graph neural networks and structure-aware mechanisms to model more complex protocol-layer state transition graphs. Another direction is to integrate privacy-preserving techniques such as federated learning for collaborative detection in distributed network environments. In addition, model interpretability will be a key area for improvement. Future work will explore attention mechanisms and visualization-based explanation methods to enhance the interpretability and traceability of detection results in real-world operations. In summary, this work provides a foundational and extensible modeling paradigm for protocol anomaly detection and holds promise for widespread impact in future security-critical applications.

References

1. I. H. Ji, J. H. Lee, M. J. Kang, et al., "Artificial intelligence-based anomaly detection technology over encrypted traffic: A systematic literature review," Sensors, vol. 24, no. 3, p. 898, 2024.
2. H. Ahmad, M. M. Gulzar, S. Aziz, et al., "AI-based anomaly identification techniques for vehicles communication protocol systems: Comprehensive investigation, research opportunities and challenges," Internet of Things, p. 101245, 2024.
3. J. R. V. Solaas, E. Mariconti and N. Tuptuk, "Systematic Literature Review: Anomaly Detection in Connected and Autonomous Vehicles," IEEE Transactions on Intelligent Transportation Systems, 2024.
4. B. A. Scott, M. N. Johnstone, P. Szewczyk, et al., "Matrix Profile data mining for BGP anomaly detection," Computer Networks, vol. 242, p. 110257, 2024.
5. S. Bhattacharya, N. Saqib and M. Govindarasu, "ML-based anomaly detection system for IEC 61850 communication in substations," Proceedings of the 2024 IEEE Power & Energy Society General Meeting, pp. 1-5, 2024.
6. N. K. Barsha and N. Hubballi, "Anomaly detection in SCADA systems: A state transition modeling," IEEE Transactions on Network and Service Management, 2024.
7. C. Hu, Z. Cheng, D. Wu, Y. Wang, F. Liu and Z. Qiu, "Structural generalization for microservice routing using graph neural networks," arXiv preprint arXiv:2510.15210, 2025.
8. J. Li, Q. Gan, R. Wu, C. Chen, R. Fang and J. Lai, "Causal Representation Learning for Robust and Interpretable Audit Risk Identification in Financial Systems," 2025.
9. Y. Zhou, "A Unified Reinforcement Learning Framework for Dynamic User Profiling and Predictive Recommendation," Available at SSRN 5841223, 2025.
10. T. Guan, "A Multi-Agent Coding Assistant for Cloud-Native Development: From Requirements to Deployable Microservices," 2025.
11. S. Sun, "CIRR: Causal-Invariant Retrieval-Augmented Recommendation with Faithful Explanations under Distribution Shift," arXiv preprint arXiv:2512.18683, 2025.

 doi:10.20944/preprints202602.1907.v1

12. Y. Ou, S. Huang, R. Yan, K. Zhou, Y. Shu and Y. Huang, "A Residual-Regulated Machine Learning Method for Non-Stationary Time Series Forecasting Using Second-Order Differencing," 2025.
13. N. Lyu, J. Jiang, L. Chang, C. Shao, F. Chen and C. Zhang, "Improving Pattern Recognition of Scheduling Anomalies through Structure-Aware and Semantically-Enhanced Graphs," arXiv preprint arXiv:2512.18673, 2025.
14. R. Liu, L. Yang, R. Zhang and S. Wang, "Generative Modeling of Human-Computer Interfaces with Diffusion Processes and Conditional Control," arXiv preprint arXiv:2601.06823, 2026.
15. H. Liu, Y. Kang and Y. Liu, "Privacy-preserving and communication-efficient federated learning for cloud-scale distributed intelligence," 2025.
16. S. Pan and D. Wu, "Trustworthy summarization via uncertainty quantification and risk awareness in large language models," arXiv preprint arXiv:2510.01231, 2025.
17. Y. Wu, Y. Qin, X. Su and Y. Lin, "Transformer-based risk monitoring for anti-money laundering with transaction graph integration," Proceedings of the 2025 2nd International Conference on Digital Economy, Blockchain and Artificial Intelligence, pp. 388-393, 2025.
18. C. Zhang, C. Shao, J. Jiang, Y. Ni and X. Sun, "Graph-Transformer Reconstruction Learning for Unsupervised Anomaly Detection in Dependency-Coupled Systems," 2025.
19. C. Hua, N. Lyu, C. Wang and T. Yuan, "Deep Learning Framework for Change-Point Detection in Cloud-Native Kubernetes Node Metrics Using Transformer Architecture," 2025.
20. B. Chen, "FlashServe: Cost-Efficient Serverless Inference Scheduling for Large Language Models via Tiered Memory Management and Predictive Autoscaling," 2025.
21. S. N. Bushra, N. Subramanian and A. Chandrasekar, "An optimal and secure environment for intrusion detection using hybrid optimization based ResNet 101-C model," Peer-to-Peer Networking and Applications, vol. 16, no. 5, pp. 2307-2324, 2023.
22. N. O. Aljehane, "A secure intrusion detection system in cyberphysical systems using a parameter-tuned deep-stacked autoencoder," Computers, Materials & Continua, vol. 68, no. 3, 2021.
23. S. Lotfi, M. Modirrousta, S. Shashaani, et al., "Network Intrusion Detection with Limited Labeled Data Using Self-supervision," arXiv preprint arXiv:2209.03147, 2022.