Transactions on Computational and Scientific Methods | Vo. 4, No. 3, 2024
ISSN: 2998-8780
https://pspress.org/index.php/tcsm
Pinnacle Science Press

Semantic Graph-Based Modeling for Protocol Anomaly Detection in Heterogeneous Computing Systems

Ming Gong
University of Pennsylvania, Philadelphia, USA
mgong108@gmail.com

Abstract: This paper addresses the challenges of protocol behavior anomaly detection in high-concurrency heterogeneous systems, including structural complexity, semantic heterogeneity, and the absence of labeled data. It proposes an unsupervised anomaly detection method based on a request semantic graph autoencoder. The method transforms raw protocol requests into structured semantic graphs by modeling entity associations and semantic dependencies, then applies a graph autoencoder to extract node embeddings and reconstruct structural information to capture latent behavioral deviations. A semantic perturbation view is introduced alongside a consistency regularization term to enhance embedding stability and semantic alignment. To evaluate the method's effectiveness, experiments are conducted under various protocol fusion and structural disturbance settings, analyzing factors such as encoder depth, embedding dimension, anomaly ratio, and graph depth. The results demonstrate that the proposed approach achieves stable and superior performance across multiple unsupervised anomaly detection metrics, effectively characterizing structural patterns and distributional anomalies in protocol behaviors with strong adaptability and discriminative capability in complex systems.

Keywords: Semantic graph modeling; graph autoencoder; protocol behavior modeling; unsupervised anomaly detection

1. Introduction

In modern distributed systems and cloud computing platforms, service components collaborate through various communication protocols, forming complex request paths and interaction behaviors. As system scale continues to expand and business logic grows in complexity, protocol-level behaviors exhibit high heterogeneity, dynamic evolution, and multi-stage dependencies. At the same time, abnormal behaviors are becoming increasingly concealed. They may manifest as subtle changes in request content or deviations in behavioral patterns within protocol sequences. These challenges render traditional rule-based or single-metric detection approaches ineffective, making it difficult to accurately identify potential risks in backend service systems[1].

Protocol behavior serves as a critical representation of system operational states. It not only carries interaction semantics between services but also implicitly reflects behavioral paths related to task scheduling, state transitions, and logic execution. Deep modeling and semantic understanding of protocol-level behaviors help reveal the root causes and propagation mechanisms of anomalies from a global perspective. However, due to the high-dimensional sparsity, structural complexity, and semantic misalignment of protocol data, it remains a major research challenge to build a modeling framework that can effectively express protocol semantics and accommodate heterogeneous behavioral features. Especially under unsupervised or weakly

supervised settings, capturing anomalous deviations during behavioral evolution and enhancing sensitivity and generalization in anomaly detection remain critical technical bottlenecks.

In recent years, semantic graphs have emerged as a unified representation that integrates structural information and contextual semantics, and have been increasingly adopted in behavior modeling and system analysis tasks. By constructing node entities and semantic relational edges from protocol requests, semantic graphs with topological structures can be generated, enabling structured modeling of complex behavioral sequences[2]. Based on this, employing a graph autoencoder framework for semantic graph representation learning allows for preserving local dependencies while capturing global behavioral patterns. This approach alleviates the curse of dimensionality inherent in raw request data and provides more semantically discriminative embeddings for anomaly detection, offering technical support for identifying potential threats effectively.

Moreover, backend anomalies often involve localized behavioral perturbations, abnormal cross-node interactions, or nonlinear shifts in periodic patterns. These complexities demand that detection models possess strong structural modeling capabilities and semantic consistency. Constructing protocol semantic graphs provides a unified paradigm for representing abnormal behaviors. This enables models to mine potential anomaly clues in protocol behaviors from multiple levels and perspectives. Through the propagation mechanism inherent in graph structures, the model can capture the transmission paths of anomaly signals across nodes, uncovering deep dependencies and behavioral variations that are difficult to detect using traditional methods[3]. This facilitates more intelligent risk perception for maintaining the stability of backend systems.

In summary, anomaly detection targeting protocol behaviors holds significant theoretical value and practical importance for ensuring the stability, security, and maintainability of modern backend systems. As distributed systems become increasingly complex and service interactions grow in scale and heterogeneity, detecting subtle anomalies at the protocol level has emerged as a critical challenge. Protocol behaviors inherently encode rich semantic and structural information, reflecting the underlying operation logic, state transitions, and service dependencies of a system. Effectively identifying deviations within such behaviors can provide early warning signals for potential faults or intrusions, enabling proactive system protection and operational resilience[4].

With the continuous accumulation of multi-source observational data-including system logs, invocation traces, network packets, and protocol request flows-there exists a growing opportunity to construct a unified, semantically meaningful representation that captures both structural patterns and contextual semantics. In this context, developing a modeling framework based on request semantic graph autoencoders becomes highly promising. This approach allows for the transformation of raw protocol data into structured semantic graphs, from which latent abnormal patterns and behavioral irregularities can be uncovered without the need for labeled data.

2. Related work

In the field of system anomaly detection, researchers have extensively explored methods based on multi-source data such as logs, metrics, and invocation traces. Traditional approaches often rely on static rules, clustering, or statistical feature analysis. These methods were initially effective in meeting basic detection needs. However, with the continuous evolution of distributed system architectures, the interaction logic among backend components has become increasingly complex. Anomalous behaviors have also evolved to exhibit dynamic characteristics, contextual dependencies, and cross-phase correlations. These changes present major challenges to traditional techniques, which struggle to capture subtle anomaly signals hidden in complex behavioral patterns. In particular, when dealing with large-scale protocol data, traditional methods are prone to high rates of missed detections and false alarms[5].

To address these challenges, researchers have begun to adopt deep learning approaches, especially those involving sequence modeling and graph-based learning, to achieve more semantically rich modeling of system behavior sequences. For example, sequence modeling methods based on recurrent neural networks and attention mechanisms aim to capture temporal dependencies and contextual relationships within protocol behaviors. These approaches have demonstrated certain advantages. However, sequence models are inherently limited in their ability to model structural dependencies. They often fail to fully capture the complex interaction paths and semantic structures among entities in protocol requests. This limitation is particularly evident in multi-stage request processes or heterogeneous protocol integration scenarios, where the representational power of sequence models becomes insufficient[6].

Building on this, graph neural networks have gradually been introduced into protocol behavior modeling and anomaly detection tasks. By abstracting requests, responses, and operations as nodes and constructing semantic edges among them, it becomes possible to form a semantic graph that represents the entire behavioral process. Graph neural networks have a natural advantage in capturing high-order dependencies and contextual information between nodes. They are effective in extracting key features from behavioral structures and in modeling anomaly propagation paths. However, most existing methods still operate in supervised or semi-supervised settings, requiring a large amount of labeled data for training. This makes them unsuitable for real-world scenarios where labels are scarce or entirely unavailable[7].

To tackle the problem of structural anomaly detection under unsupervised conditions, graph autoencoders have shown increasing modeling potential as a framework for low-dimensional embedding learning. These models use an encoder to extract latent representations of protocol semantic graphs, and a decoder to reconstruct the graph structure or adjacency relationships. Without relying on labeled data, graph autoencoders can learn the distributional characteristics of nodes and edges, enabling the identification of anomalous structures that deviate from global behavioral patterns[8]. When integrated with multi-view generation, perturbation-based augmentation, and contrastive learning strategies, these methods further enhance discriminative ability and robustness in protocol behavior modeling. As a result, they offer a more general and flexible solution for backend anomaly detection.

3. Method

The network architecture is built upon semantic graphs constructed from protocol requests. It first employs a graph autoencoder to embed the original graph structure, enabling high-order aggregation of node features and semantic compression. A lightweight perturbation mechanism is then used to generate perturbed embedding views, and a consistency constraint is introduced to enhance the model's robustness and discriminative ability for abnormal behavior. The overall framework unifies structure reconstruction and embedding consistency under unsupervised conditions, providing structure-aware and semantically aligned representations for high-quality anomaly detection in backend protocol behaviors. The model architecture is shown in Figure 1.

This study proposes a protocol behavior anomaly detection method based on a request semantic graph autoencoder. The raw backend protocol data is first parsed into structured behavioral representations, where nodes represent request entities and edges denote semantic relationships between protocol elements. By embedding semantic information from fields such as parameters, paths, and methods, and performing entity abstraction, a unified graph-based representation is constructed. This representation not only preserves the structural characteristics of behavior but also captures contextual dependencies at the semantic level, providing a robust foundation for subsequent graph representation learning.

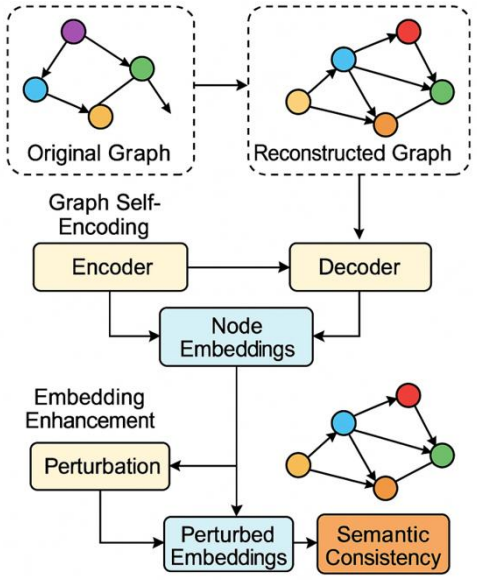

Figure 1. Semantic-Aware Graph Encoder for Protocol Behavior Detection

In the graph structure modeling stage, a graph autoencoder framework is designed to perform unsupervised embedding learning on the protocol semantic graph. The encoder part uses a graph convolution structure to extract the local context information of each node and realizes the aggregation and fusion of high-order adjacent features through a multi-layer propagation mechanism. Let $G = (V, E)$ trepresents the protocol semantic graph, where V is a collection of nodes, E is an edge set. Then the node $v_i \in V$ representation is updated as follows:

$$h_i^{(l+1)} = \sigma(\sum_{j \in N(i)} \frac{1}{c_{ij}} W^{(l)} h_j^{(l)})$$

Where $N(i)$ represents the neighbor set of node i, c_{ij} is the normalization coefficient, $W^{(l)}$ is the weight matrix of the l th layer, and σ is the activation function.

To enhance the model's robustness to local perturbations and its ability to model structural consistency, a decoder is introduced to reconstruct the connection relationship between nodes. By minimizing the difference between the original image adjacency matrix A and the reconstructed matrix $\hat{A}$, the encoder is guided to learn a discriminative graph embedding. The reconstruction mechanism is defined as follows:

$$\hat{A}_{ij} = \sigma(z_i^T z_j)$$

$$L_{recon} = \sum_{(i,j) \in V \times V} (A_{ij} \log \hat{A}_{ij} + (1 - A_{ij}) \log(1 - \hat{A}_{ij}))$$

In addition, to improve the sensitivity of the embedding space to abnormal behavior, a perturbation consistency regularization term is introduced. Specifically, a slight perturbation is introduced into the protocol semantic graph to generate a pseudo view, and the original embedding and the perturbation embedding are required to maintain semantic consistency. Let the original embedding be Z and the perturbation embedding be $\tilde{Z}$; then the consistency loss is defined as:

$$L_{cons} = \sum_{i=1}^{|V|} \| z_i - \widetilde{z}_i \|_2^2$$

The final optimization goal combines reconstruction loss and consistency loss to achieve unified modeling of structural information compression and behavioral feature extraction through joint training. The overall loss function is defined as follows:

$$L_{total} = L_{recon} + \lambda L_{cons}$$

Where λ is a balancing hyperparameter that controls the weight relationship between structural reconstruction and semantic consistency. This method can extract discriminative patterns in protocol behaviors under unsupervised conditions and provide stable and robust representation support for backend anomaly detection.

4. Experimental Results

4.1 Dataset

This study adopts the MAWILab network traffic dataset as the foundational data source for protocol behavior modeling and anomaly detection. The dataset is collected from real backbone network routers on the Internet. It captures request behaviors across various communication protocols, including TCP, UDP, and ICMP, and reflects typical backend service interaction characteristics. Anomalies are labeled through cross-validation by multiple detectors, covering various types of abnormal patterns such as port scans, data injection, and denial of service. This provides a reliable basis for modeling anomalies in complex scenarios.

In the data preprocessing stage, key behavioral fields are extracted, including the packet five-tuple (source IP, destination IP, source port, destination port, and protocol type), session timing, traffic direction, and packet size. Each communication request is abstracted into nodes and edges in a graph, constructing a request semantic graph. This preserves the temporal order, interaction behavior, and semantic dependencies of protocol activities, enabling structured graph modeling. In this graph, nodes represent communication entities, and edges capture semantically related protocol interactions, forming a complete representation of protocol behavior.

To support the unsupervised representation learning required by the model, the processed semantic graph samples are used without explicit labels during training. Anomaly labels are introduced only during the evaluation phase to assess detection performance. The dataset contains diverse protocol behavior patterns and structural heterogeneity, effectively supporting the modeling requirements of the proposed backend anomaly detection framework based on request semantic graph autoencoders.

4.2 Experimental Results

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

Table1: Comparative experimental results

Model	F1 Score	AUC	Precision	KS Score
Ours(SAGE-Detect)	0.872	0.911	0.854	0.683
GDN[9]	0.791	0.842	0.765	0.538
Anomaly Transformer[10]	0.804	0.861	0.779	0.557
Timeautoad [11]	0.769	0.828	0.754	0.521

NeuTraL[12]	0.782	0.837	0.746	0.534

Based on the experimental results, the proposed SAGE-Detect outperforms the baseline models across multiple key metrics. It demonstrates stronger capabilities in protocol behavior modeling and anomaly identification. The F1 score reaches 0.872, showing an improvement of over 8% compared to the traditional graph-based model GDN. This indicates a higher overall discriminative ability in capturing semantic graph structures and abnormal behavior features.

In terms of precision, SAGE-Detect achieves a score of 0.854, surpassing existing Transformer-based anomaly detection models such as Anomaly Transformer and Timeautoad. This reflects its advantage in semantic consistency modeling and filtering of pseudo-anomalies. These improvements are attributed to the designed graph autoencoder framework and the perturbation consistency mechanism, which effectively suppresses non-structural noise in protocol behavior and improves the accuracy in identifying localized anomalies.

From the perspective of structural consistency, SAGE-Detect achieves a KS score of 0.683, which is significantly higher than other models. This result indicates that the semantic graph modeling enhances the discriminative power of feature representations and improves the model's sensitivity to structural anomalies. Compared to methods that rely solely on sequential modeling, this approach exhibits stronger expressiveness in capturing complex structural changes across stages and entities in protocol paths.

In summary, the proposed model integrates request semantic graph construction with unsupervised graph autoencoder training to produce more expressive structural embeddings. This effectively enhances anomaly detection sensitivity and stability. Under highly heterogeneous and dynamically evolving backend systems, the method shows strong adaptability and practical value, validating the effectiveness of structure-aware modeling and consistency constraints in anomaly detection tasks.

This paper also experiments on the synergistic impact of encoder depth and detection performance in multi-protocol heterogeneous scenarios. The experimental results are shown in Figure 2.

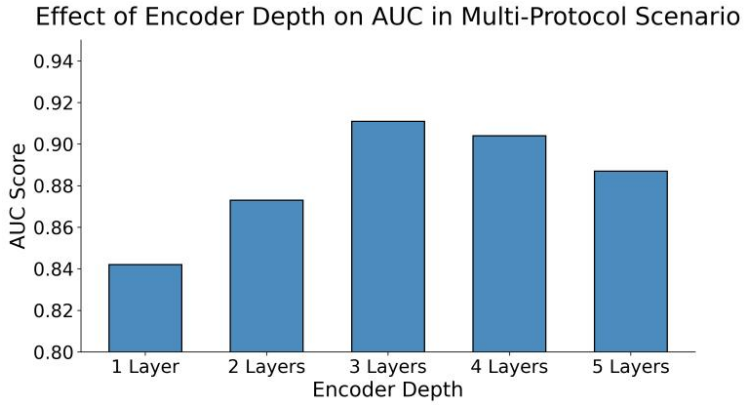

Figure 2. The synergistic impact of encoder depth and detection performance in multi-protocol heterogeneous scenarios

The experimental results show that as the depth of the encoder increases, the anomaly detection performance of the model in multi-protocol heterogeneous scenarios first improves and then stabilizes. When the encoder has only one layer, the AUC score is relatively low. This indicates that the model has limited ability to capture high-order structural dependencies and complex interaction relationships in the protocol graph. The embedding representation is insufficient, making it difficult to accurately model the structural characteristics of anomalous behaviors.

When the encoder depth reaches three layers, the model achieves the highest AUC score. This suggests that the graph autoencoder at this depth effectively integrates semantic associations across different protocol types and generates more discriminative node embeddings. The enhanced representation captures cross-protocol path features among requests, which strengthens the model's ability to recognize anomaly distributions in complex structures.

However, when the depth increases to four or five layers, model performance slightly declines. This may be caused by overfitting or over-smoothing due to excessive network depth. As a result, node representations become homogenized and lose local structural distinctions. This degradation reduces the separability of anomalous nodes in the embedding space and weakens the model's overall discriminative power.

This paper also conducts comparative experiments on the cross-sensitivity of node embedding dimension, anomaly injection ratio, and detection accuracy. The experimental results are shown in Figure 3.

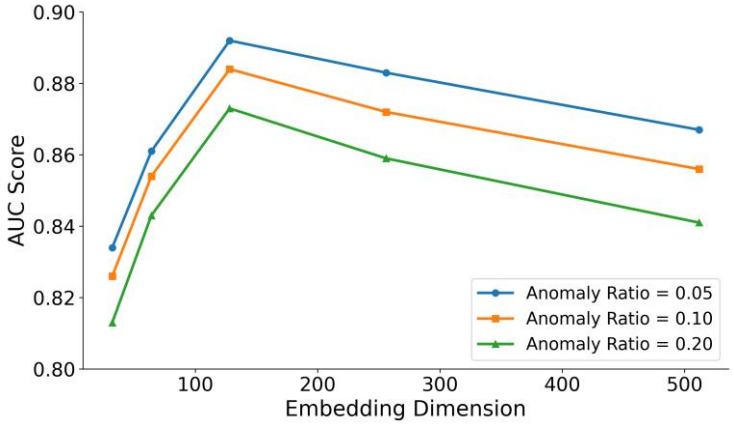

Figure 3. Cross-sensitivity experiment on node embedding dimension, anomaly injection ratio, and detection accuracy

The experimental results show that the node embedding dimension has a significant impact on anomaly detection performance. This trend remains highly consistent under different anomaly injection ratios. When the embedding dimension increases from 32 to 128, the AUC score rises significantly. This suggests that lower dimensions cannot fully capture the complex structural dependencies and semantic relationships in the protocol semantic graph, limiting the model's ability to identify anomalous behaviors. With higher dimensions, the model can capture more high-order graph features, enhancing its capacity to model weak anomaly signals in cross-protocol interaction patterns.

Under a 5% anomaly injection setting, the model achieves its highest AUC value of 0.892 with a 128-dimensional embedding. This indicates that this embedding size offers optimal semantic compression and behavior separation in low-interference environments. When the embedding dimension further increases to 256 and 512, performance slightly declines. This may be due to the enlarged representation space introducing redundancy, which results in sparser embeddings and weakens the graph autoencoder's ability to distinguish anomalous samples during node comparison.

In high anomaly ratio settings, such as 20%, the overall detection performance slightly decreases. This reflects the model's reduced ability to discriminate boundary samples when the proportion of anomalies increases. Nonetheless, the model maintains strong robustness at 128 and 256 dimensions, indicating that the semantic graph autoencoder relies on a moderately sized embedding space to preserve semantic consistency and structural stability when modeling imbalanced data.

This experiment further validates the cross-sensitivity between node embedding dimension and anomaly ratio. It highlights the importance of setting an appropriate embedding size in highly heterogeneous protocol scenarios. A suitable embedding space not only strengthens the model's representation of protocol behavior

features but also maintains structural separability of anomalous nodes under varying levels of anomaly injection. This is essential for ensuring the stability and accuracy of backend anomaly detection.

This paper also analyzes the coupling relationship between the embedding dimension and the decoding and reconstruction capability under the condition of multi-source protocol fusion. The experimental results are shown in Figure 4.

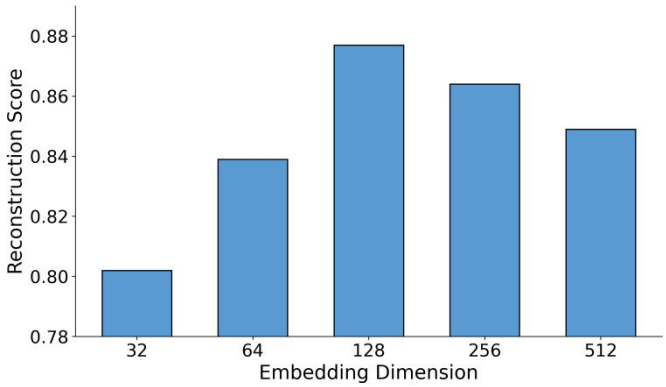

Figure 4. Coupling between Embedding Dimension and Reconstruction Ability under Multi-Protocol Fusion

The experimental results indicate that, in the context of multi-source protocol integration, the node embedding dimension has a clear impact on the decoder's structural reconstruction capability. As the embedding dimension increases from 32 to 128, reconstruction accuracy continues to improve. This suggests that a moderate expansion of the embedding space helps the graph autoencoder capture high-order interactions and structural features across multiple protocols. As a result, the model can more accurately restore the topological structure and semantic dependencies of the original graph. This process enhances the overall modeling capacity of protocol behavior and provides a foundation for high-quality anomaly detection.

When the embedding dimension is set to 128, the model achieves its best reconstruction performance. This indicates that, in a multi-protocol integration setting, this configuration preserves essential structural features while reducing redundancy in the embedding space. It produces stable and discriminative representations. However, further increasing the embedding dimension to 256 and 512 leads to a slight drop in performance. This suggests that redundant features interfere with the decoder's ability to recover key structural patterns. This high-dimensional sparsity is more evident in heterogeneous protocol graphs, highlighting the importance of embedding dimension control in preserving structural integrity.

In addition, high-dimensional embeddings may introduce semantic drift. This is especially critical in multi-source protocol integration, where the semantic boundaries between different protocol types become less distinct. An overly large embedding space may lead to confusion between features of different protocols, weakening the decoder's perception of local structure. This effect is particularly noticeable at the 512-dimensional level, suggesting that graph autoencoders should avoid unnecessary representational redundancy when modeling protocol-level behaviors.

Finally, this study tested the comprehensive performance of graph structure depth and abnormal ratio changes in high-concurrency heterogeneous systems, as shown in Figure 5.

The experimental results show that graph structure depth has a significant impact on anomaly detection performance. In high-concurrency heterogeneous protocol systems, the number of graph propagation steps directly affects the model's ability to capture cross-node behavioral dependencies. As the graph depth increases from one to three layers, the F1 score consistently improves across all anomaly injection ratios. This indicates that shallow graph models are insufficient to cover long-range dependencies in protocol behavior paths. Increasing the propagation range helps the model access richer contextual information and enhances its ability to detect global anomaly features.

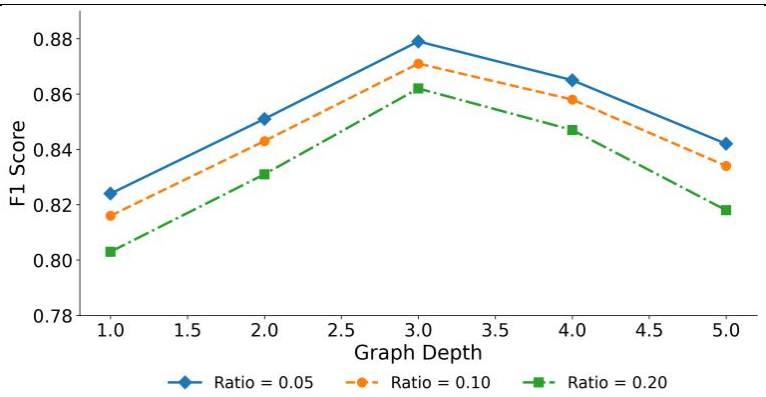

Figure 5. Comprehensive Evaluation of Graph Depth and Anomaly Ratio in Heterogeneous High-Concurrency Systems

At a depth of three, the F1 score reaches its peak under all three anomaly ratios. This suggests that the graph autoencoder achieves a better balance between semantic expressiveness and structural generalization. In complex scenarios involving concurrent multi-protocol interactions, a three-hop neighborhood can effectively cover key nodes and routing boundaries in protocol behavior paths. It preserves upstream and downstream semantics in the structure and enables high-quality modeling of anomaly propagation paths.

When the depth increases to four and five layers, detection performance gradually decreases, with a slight drop in F1 score. This may result from over-smoothing caused by excessive graph propagation. Node representations become more homogeneous, reducing the model's ability to express local structural differences. In protocol behavior graphs, deeper propagation may also introduce structural noise. In protocol subgraphs with weak connections and sparse boundaries, too much neighbor information may interfere with the discriminative quality of embeddings.

Changes in anomaly ratio also affect overall performance. As the ratio increases from 5% to 20%, the model remains stable but shows a slight performance drop. This suggests that the graph autoencoder retains strong robustness even under high anomaly interference. The embeddings integrate structural information from multi-protocol behaviors through multi-hop propagation. As a result, the model maintains structural discriminative power to some extent. These findings confirm the coupled effect between graph depth and anomaly ratio.

5. Conclusion

This study proposes an unsupervised detection method based on request semantic graph autoencoders for protocol behavior modeling and backend anomaly detection in multi-protocol heterogeneous systems. The approach transforms raw protocol requests into structured semantic graphs. It applies a graph autoencoder framework to learn node embeddings and reconstruct structural information. A consistency-based perturbation mechanism is introduced to improve the robustness and discriminative power of the embeddings. The overall architecture integrates structural awareness, semantic compression, and anomaly representation. It effectively captures hidden abnormal patterns in backend systems without labeled data, demonstrating strong adaptability and generalizability.

The experimental design evaluates the model's stability and effectiveness across multiple dimensions. These include protocol structure modeling, embedding space construction, and dynamic changes in graph topology. Sensitivity analyses are conducted on encoder depth, embedding dimension, anomaly injection ratio, protocol fusion intensity, and system concurrency scale. The results reveal the coupling relationships among graph depth, input semantic complexity, and model robustness. They demonstrate that modeling based on request semantic graphs can accurately recover multi-source behavior paths in complex systems and improve the granularity of anomaly identification.

This study introduces a new paradigm that combines structure-driven modeling with semantic alignment for protocol-level anomaly detection. It overcomes the limitations of traditional methods based on log or metric analysis. The proposed approach offers theoretical and technical support for system security operations, distributed tracing, and network intrusion detection. In domains such as industrial internet, financial backend services, and cloud platform scheduling, where system stability is critical, the method shows high application potential. It can be integrated as a core component of behavior modeling and anomaly detection within existing monitoring systems.

6. Future Research

Future research may further extend the proposed model to support cross-protocol transfer learning, multimodal data integration, and collaborative modeling in federated environments, to meet the demands of more complex and dynamic system operations. In multi-protocol systems, significant differences exist in structural hierarchies, semantic expressions, and interaction mechanisms. Designing transfer strategies with structural abstraction and semantic alignment capabilities can improve the model's adaptability across different protocol types. As the need for integrating logs, metrics, traces, and network traffic continues to grow, building a unified graph representation framework to jointly model semantic correlations across modalities will enhance the model's perception of behavioral evolution. In practical deployments, large-scale systems and heterogeneous computing environments raise new challenges. Improving online update efficiency and cross-node deployment performance will be essential for enabling cloud-edge collaboration. Designing autoencoder architectures with incremental learning capability and low communication overhead can support continuous modeling and fast response in dynamic systems. In addition, incorporating causal inference mechanisms to capture stable dependencies in protocol behaviors, integrating graph attention mechanisms to strengthen the representation of anomaly-sensitive structures, and applying adaptive perturbation strategies to regulate embedding disturbances can further improve the model's overall performance in structural modeling, anomaly localization, and generalization robustness. These directions will provide structurally aware and semantically expressive techniques for intelligent system operations and maintenance.

References

[1] Chen X., Li X., Li Z., et al. Hierarchical Federated Graph Learning for Cross-Organization Anomaly Detection.AAAI 2022.

[2] Pazho A D, Noghre G A, Purkayastha A A, et al. A survey of graph-based deep learning for anomaly detection in distributed systems[J]. IEEE Transactions on Knowledge and Data Engineering, 2023, 36(1): 1-20.

[3] Liu Y, Li Z, Pan S, et al. Anomaly detection on attributed networks via contrastive self-supervised learning[J]. IEEE transactions on neural networks and learning systems, 2021, 33(6): 2378-2392.

[4] Zhou K, Huang X, Song Q, et al. Auto-gnn: Neural architecture search of graph neural networks[J]. Frontiers in big Data, 2022, 5: 1029307.

[5] Sun Z, Teixeira A M H, Toor S. GNN-IDS: Graph Neural Network based Intrusion Detection System[C]//Proceedings of the 19th International Conference on Availability, Reliability and Security. 2024: 1-12.

[6] Jiang W. Graph-based deep learning for communication networks: A survey[J]. Computer Communications, 2022, 185: 40-54.

[7] Xu J, Wu H, Wang J, et al. Anomaly transformer: Time series anomaly detection with association discrepancy[J]. arXiv preprint arXiv:2110.02642, 2021.

[8] Feng Y, Chen J, Liu Z, et al. Full graph autoencoder for one-class group anomaly detection of IIoT system[J]. IEEE Internet of Things Journal, 2022, 9(21): 21886-21898.

[9] Deng A, Hooi B. Graph neural network-based anomaly detection in multivariate time series[C]//Proceedings of the AAAI conference on artificial intelligence. 2021, 35(5): 4027-4035.

[10]Chen L, You Z, Zhang N, et al. UTRAD: Anomaly detection and localization with U-transformer[J]. Neural Networks, 2022, 147: 53-62.

[11]Jiao Y, Yang K, Song D, et al. Timeautoad: Autonomous anomaly detection with self-supervised contrastive loss for multivariate time series[J]. IEEE Transactions on Network Science and Engineering, 2022, 9(3): 1604-1619.

[12]Qiu C, Pfrommer T, Kloft M, et al. Neural transformation learning for deep anomaly detection beyond images[C]//International conference on machine learning. PMLR, 2021: 8703-8714.