

Multi-period Balancing Attribute Subgraph and Heterogeneous Graph Diffusion Enhancement for Network Anomaly Detection

Xiang Long

Huazhong University of Science and Technology

Haili Sun

Wuhan Textile University

Dingkun Xu

Huazhong University of Science and Technology

Lansheng Han

hanlansheng@hust.edu.cn

Huazhong University of Science and Technology

Article

Keywords:

Posted Date: August 25th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10120575/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Multi-period Balancing Attribute Subgraph and Heterogeneous Graph Diffusion Enhancement for Network Anomaly Detection

Xiang Long^{1,2,+}, Haili Sun^{3,+}, Dingkun Xu¹, and Lansheng Han^{1,*}

¹Huazhong University of Science and Technology, School of Cyber Science and Engineering, Wuhan, 430074, China

²Hubei Vocational College of Bio-Technology, Wuhan, 430070, China

³Wuhan Textile University, School of Cyber Science and Engineering, Wuhan, 430200, China

*hanlansheng@hust.edu.cn

+these authors contributed equally to this work

ABSTRACT

Network traffic anomaly detection is a cornerstone of cybersecurity situational awareness. However, building effective detection models remains challenging due to severe data imbalance and classification uncertainty. Existing approaches often fail to adequately address these issues, resulting in suboptimal detection performance. To overcome these limitations, we propose a novel anomaly detection method that integrates multi-period balanced attribute subgraphs with heterogeneous graph diffusion enhancement. The proposed approach first performs temporal segmentation on the network traffic and constructs an attribute subgraph for each segment. Within each subgraph, the SMOTE algorithm is applied to balance the ratio of normal to anomalous instances. Subsequently, representation learning is performed using a Diffusion-HAN network, which enriches node representations by diffusing information across heterogeneous neighbors and employing a hierarchical attention mechanism that captures both node-level and semantic-level dependencies. Finally, anomalies are classified using a softmax classifier. Experimental results show that our method significantly outperforms state-of-the-art approaches in handling data imbalance and classification uncertainty, achieving a 7% improvement in Unknown Attack Detection (UAD) rate and a 5% gain in F1-score. Moreover, it demonstrates strong robustness and scalability on large-scale network traffic datasets. This work offers a principled framework for network anomaly detection and provides solid theoretical and technical foundations for developing accurate, reliable detection systems.

1 Introduction

With the Internet’s rapid expansion and pervasive integration into society, network security issues have emerged as a critical challenge. The ever-evolving array of network attack techniques poses severe challenges to security protection. As a crucial technical means for cybersecurity situational awareness, network traffic anomaly detection plays a key role in identifying and defending against network attacks¹. However, traditional anomaly detection methods have numerous limitations and struggle to cope with the increasingly complex network environment. Among these, data imbalance and classification uncertainty are two core challenges that severely constrain the detection performance and practical effectiveness of existing methods.

The data imbalance problem is primarily manifested in the uneven distribution of sample quantities across different classes within the training data. This leads models to over-focus on majority classes while neglecting minority classes. Such bias directly affects the accuracy and efficacy of models in detecting anomalous behavior. For instance, in intrusion detection systems, attack samples are often far fewer than normal samples, making it difficult for models to learn the characteristics of attack behavior, thereby reducing the detection rate.

The classification uncertainty problem stems from the diversity and complexity of network security threats. There exists ambiguity in the boundaries between the spatial distributions or underlying manifolds of different classes. Existing technical means find it difficult to accurately characterize the differentiating features between classes, causing models to face difficulties in distinguishing and determining the class of network traffic or events. This increases the risk of false positives and false negatives, consequently degrading the overall performance of the models. For example, certain attack behaviors may share similarities with normal behavior, making it hard for models to accurately distinguish between them, leading to false alarms^{2,3}.

To address these challenges, researchers have explored various approaches; however, existing methods often fall short in jointly and effectively tackling the dual issues of data imbalance and classification uncertainty. For instance, to mitigate sample imbalance, techniques like SMOTE can oversample minority classes, yet they risk overfitting and distorting the intrinsic distribution of network traffic data⁴. Regarding classification ambiguity, deep learning-based frameworks excel at automated

feature extraction⁵ but heavily depend on large labeled datasets and generalize poorly to novel attacks. Unsupervised methods, such as those relying on reconstruction error⁶, circumvent the need for labels but struggle to precisely differentiate subtle anomalies from normal variations, leading to high false-positive rates. Meanwhile, graph-based learning methods capture complex relational structures⁷ but often operate on static graph snapshots, lacking the dynamism to adapt to the evolving, multi-period nature of network traffic. Crucially, most existing solutions address data imbalance and representation learning in isolation, failing to provide a unified framework that dynamically balances data across temporal segments while simultaneously enhancing discriminative feature learning in a heterogeneous graph context. This gap motivates our work.

Section 2 will elaborate on related work and research findings, analyzing the strengths and weaknesses of existing network anomaly detection methods, and introducing the innovations and advantages of the proposed method. Section 3 will introduce the fundamental principles and related definitions of multi-period balanced attributed subgraphs and heterogeneous graph diffusion-enhanced representation learning. It will also detail the proposed anomaly detection method based on multi-period balanced attributed subgraphs and heterogeneous graph diffusion enhancement, covering steps such as data preprocessing, multi-period balanced attributed subgraph construction, Diffusion-HAN network training, and softmax anomaly detection classification. Section 4 will present the corresponding experimental results and analysis, validating the effectiveness and superiority of the proposed method on multiple public datasets. Finally, the paper will summarize the work and outline future research directions.

2 Related Work

The rapid development of network technology has led to increasingly diversified network application scenarios. Different scenarios impose varying requirements and focuses on network security, which continually motivates researchers to refine and optimize anomaly detection methods⁵. For instance, in financial network scenarios where data confidentiality and transaction security are paramount, detection must focus particularly on anomalous behaviors involving financial risks⁸. Moreover, the massive volume of network data generated in today's big data environment provides a richer information source for anomaly detection, while also introducing significant challenges in data processing and analysis, such as storage and cleaning.

Driven by these needs, researchers have proposed various methods based on machine learning and deep learning in recent years to improve detection efficiency and accuracy⁹⁻¹¹. Traditional methods, which primarily utilize statistical features of network traffic (e.g., flow size, duration, protocol type), are susceptible to noise and attacker camouflage. This leads to false positives and false negatives, and more fundamentally, limits their ability to capture the complexity and dynamic nature of attack traffic. Machine learning-based methods, such as Support Vector Machines (SVM), can train models to distinguish normal from anomalous traffic¹². Their effectiveness, however, is constrained by a reliance on large amounts of labeled data, a propensity for overfitting, and difficulty in handling scenarios where attack samples are scarce. Deep learning-based methods, including Convolutional Neural Networks (CNN), offer automatic feature representation learning from network traffic data¹³. A recognized drawback is their frequent failure to capture the temporal and spatial relationships between different fields within attack traffic, coupled with generally poor model interpretability.

To overcome these limitations, several improvements have been explored. The SMOTE algorithm, for example, balances datasets by resampling minority or majority classes to improve model detection rates¹⁴. A critical trade-off is the risk of overfitting and the alteration of the dataset's original sample distribution, which can ultimately affect model accuracy. Separately, heterogeneous graph learning has been proposed to effectively learn relationships between different node types¹⁵. The adoption of this approach is often hampered by complex graph structure construction, high computational complexity, and difficulties in processing large-scale datasets¹⁶.

Further addressing these issues, the research community has provided valuable surveys and specialized analyses. Yang et al. presented a survey on deep learning-based techniques, analyzing various models and discussing future directions¹⁷. Eswaran et al. explored fine-grained behavioral attribute analysis, emphasizing relationships between internal attributes of events¹⁸. Wei et al. conducted an in-depth analysis comparing deep learning models in this domain¹⁹. At the model level, Ji et al. proposed a Heterogeneous Graph Transformer that effectively handles heterogeneous graph data²⁰, while Bamber et al. demonstrated a CNN-LSTM hybrid model achieving 95% accuracy on the NSL-KDD dataset by extracting spatial and temporal features²¹. It is important to note that the latter study focused primarily on a single dataset, leaving its generalization capability across different datasets requiring further validation. Similarly, other hybrid models like HIDIM perform well on multi-classification tasks, but their detection capability for zero-day attacks and adversarial samples is not fully validated, with generalization ability limited by training data coverage²².

In summary, existing work has advanced the field but reveals a persistent gap: methods often address data imbalance and complex pattern learning in isolation, lacking a unified framework that dynamically adapts to the temporal structure of network traffic^{23,24}. To bridge this gap and improve detection efficiency and accuracy, this paper investigates an anomaly detection method based on multi-period balanced attributed subgraphs and heterogeneous graph diffusion enhancement. Our method constructs multi-period balanced attributed subgraphs and employs the SMOTE algorithm *within each subgraph*

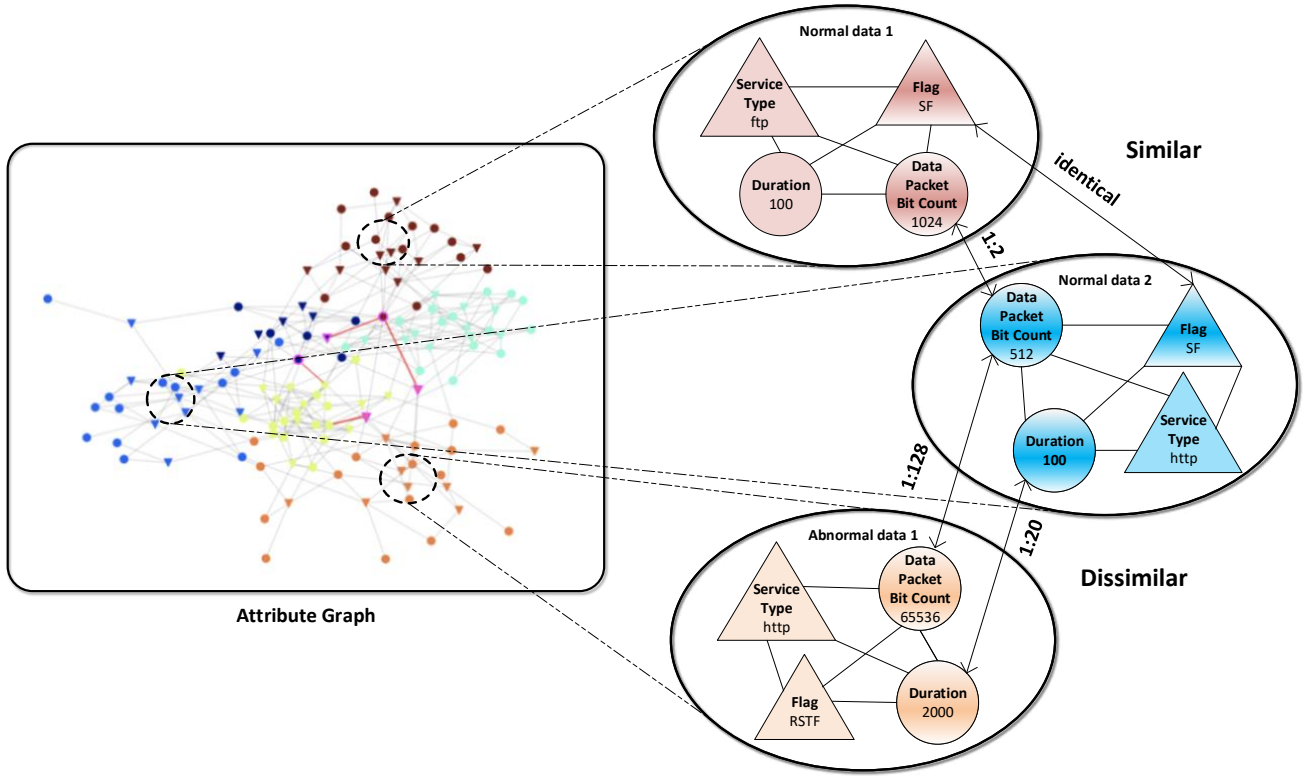


Figure 1. Basic principle – Similarity among nodes with different traffic attributes

to balance normal and anomalous data ratios. This localized balancing mitigates the data imbalance problem while aiming to avoid the overfitting risks associated with global resampling. Simultaneously, it employs a Diffusion-HAN network for representation learning. By incorporating a diffusion process and attention mechanisms, this model enhances its ability to model the spatio-temporal relationships among different fields in attack traffic, thereby improving the model's representational capacity and classification performance.

3 Anomaly Detection Method Based on Multi-Period Balanced Attributed Subgraphs and Heterogeneous Graph Diffusion Enhancement

3.1 Concept Introduction

We first introduce some relevant concepts that appear in this article.

3.1.1 Attributed Graph and Attributed Subgraph

An attributed graph is a fundamental type of heterogeneous graph, formally defined as a tuple $G = (V, E, R)$, where: $V = \{v_1, v_2, \dots, v_n\}$ is the node set (each node $v \in V$ represents a distinct event attribute value); $E \subseteq V \times V$ is the undirected edge set (each edge $e \in E$ represents a relationship between nodes $u, v \in V$); and R is the set of relation types, with each edge annotated as a triple $e = (u, v, r)$ for $r \in R$. This structure jointly encodes attribute semantics and relational patterns.

The concept of an attributed subgraph is derived from the attributed graph. It refers to a sequence of attributed graphs constructed by first sorting the training dataset chronologically, segmenting it into subsets using equal-frequency discretization (i.e., ensuring each subset contains approximately the same number of samples), and then building an attributed graph for each subset.

3.1.2 Representation Learning

Representation learning aims to map nodes into low-dimensional vector embeddings that preserve structural and attributive information crucial for downstream tasks such as anomaly detection. Given a heterogeneous graph $G = (V, E, R)$, the goal is to learn a mapping function $f: V \rightarrow R^d$ that projects each node $v \in V$ into a dense vector $\mathbf{h}_v \in R^d$, where the embedding dimensionality d is much smaller than the total number of nodes $|V|$.

3.2 Experimental Process

3.2.1 Method Overview

Building upon the attributed graph, we introduce the concept of a multi-period attributed subgraph. To construct these subgraphs, the raw data undergoes systematic preprocessing. First, duplicate and incomplete entries are removed. The data is then sorted chronologically and partitioned into distinct temporal subsets using the equal-width discretization method, with each subset corresponding to a specific period. Within each period, continuous values are further standardized and discretized (again via equal-width discretization) to facilitate attributed graph construction.

A critical challenge in this setting is the severe class imbalance, where anomalous instances are vastly outnumbered by normal ones, leading to model bias toward the majority class. To mitigate this, we apply the Synthetic Minority Over-sampling Technique (SMOTE) locally within each attributed subgraph. SMOTE operates through the following steps:

1. Identify k -Nearest Neighbors: For each minority-class (anomalous) sample, we identify its k nearest neighbors within the same class, typically using Euclidean distance.
2. Generate Synthetic Samples: For each minority sample, one neighbor is randomly selected. A new synthetic sample is created via linear interpolation between the original sample $\mathbf{x}_i$ and the selected neighbor $\mathbf{x}_{nn}$,

$$\mathbf{x}_{\text{new}} = \mathbf{x}_i + \lambda \cdot (\mathbf{x}_{nn} - \mathbf{x}_i), \quad (1)$$

where $\lambda \in [0, 1]$ is a random weight. This process is repeated until a predefined number of synthetic samples is generated.

3. Integrate Synthetic Samples: All synthetic samples are added to the original subset, balancing the class distribution within that period.

For continuous features, synthetic values are generated via the linear interpolation described above. For discrete features (e.g., status flags), values are sampled from a multinomial distribution based on the observed frequencies in the original minority-class data. Through this process, each temporally partitioned subset is transformed into a balanced attributed subgraph, collectively forming a multi-period, balanced representation of the network traffic.

3.2.2 Constructing Multi-Period Balanced Attributed Subgraphs

We construct an attributed subgraph for each balanced temporal subset. In this construction, each data record is decomposed into a set of attribute nodes, where each node corresponds to a specific attribute-value pair (e.g., Service Type=http, Duration=2000(ms)). As illustrated in Figure 1, features such as service type, duration, packet bits, and flags are thus mapped to nodes. Within a single data record, all its constituent attribute nodes are fully connected, capturing intra-record dependencies. Crucially, attribute nodes sharing identical values across different records are merged into a single, shared node. This merging process establishes implicit connections between different data records that possess common attribute values, thereby weaving the isolated records into a coherent graph structure that encapsulates both intra-record and inter-record relationships.

Once the multi-period balanced attributed subgraphs are constructed, we proceed with multi-head attention representation learning using the Diffusion-enhanced Heterogeneous Graph Attention Network with Single-Hop Meta-paths (Diffusion-HAN).

3.2.3 Representation Learning Based on Diffusion-HAN

Initial Node Embedding via Pre-trained Vectors. The input to our representation learning model is the set of attribute nodes from the constructed subgraphs. Assigning meaningful initial vector representations to these nodes is crucial. Manually setting weights between all node pairs is impractical and requires extensive domain knowledge. Instead, we utilize widely-adopted pre-trained word vectors (e.g., FastText) to generate the initial embeddings. These vectors, trained on large-scale corpora, provide a dense and semantically informed starting point for each attribute-value pair (e.g., service=http), serving as an effective prior in lieu of expert-crafted features.

Representation Learning with Diffusion-HAN We propose the Diffusion-HAN network (Heterogeneous graph representation network based on spatiotemporal attribute subgraph diffusion enhancement) for representation learning, with its architecture depicted in Figure 2. The model operates on the attributed subgraph, which is inherently a heterogeneous graph comprising nodes of different types and multiple types of relationships between them. The primary objective of Diffusion-HAN is to discover relationships between different attribute types. Consequently, we utilize meta-paths of length 1, meaning the set of meta-paths directly corresponds to all the distinct relationship types present in the attributed graph.

The learning process employs a hierarchical attention structure based on these meta-paths. Node-level attention is applied first: for each meta-path φ , a node aggregates information from its neighbors connected via that meta-path. This step utilizes a self-attention mechanism, similar to that in Graph Attention Networks (GAT). Specifically, for a given node pair (i, j) connected

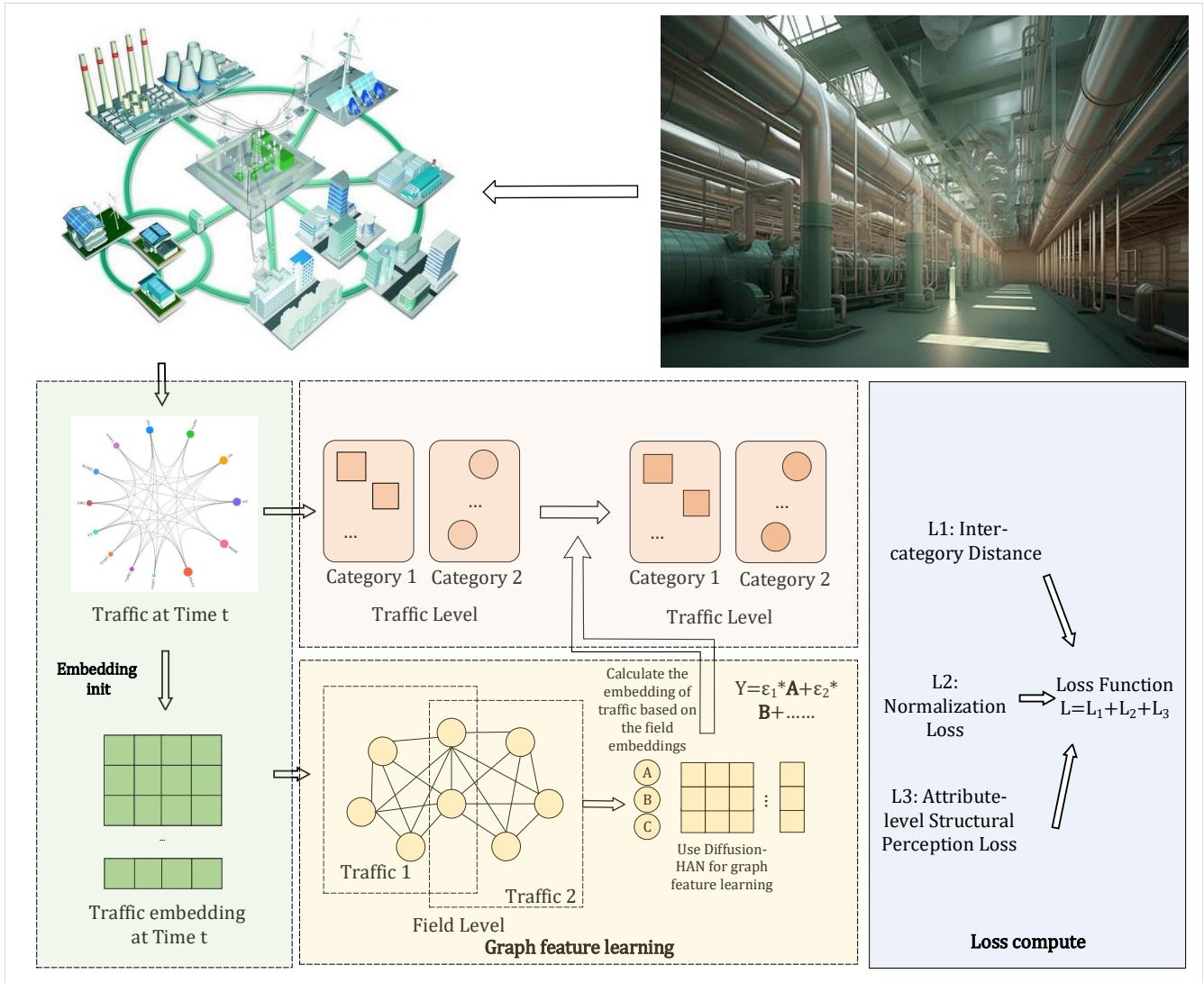


Figure 2. Overall structure

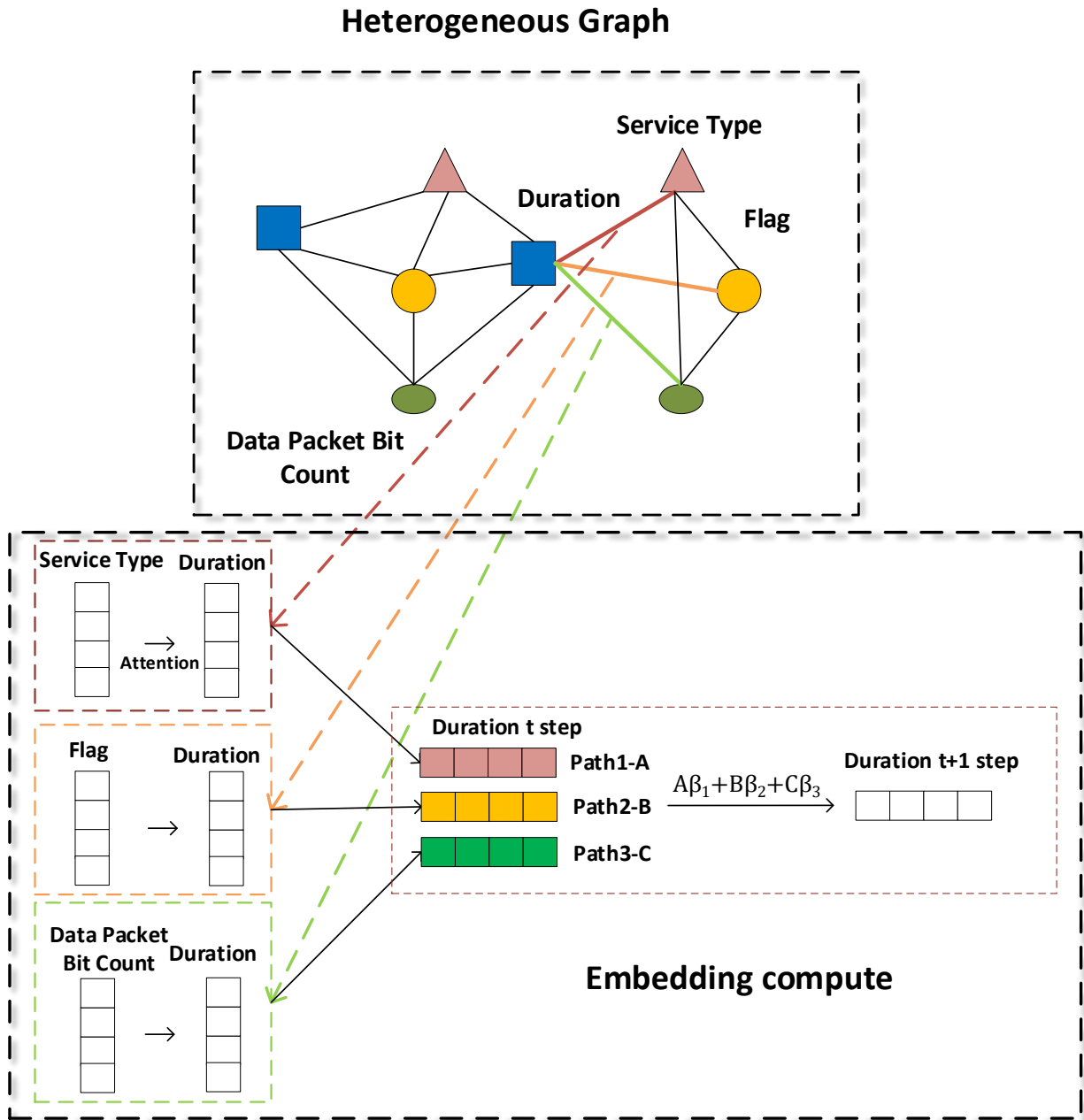


Figure 3. Node's neighbor

under meta-path φ , we first compute their transformed representations. Then, the attention weight α_{ij}^φ , which indicates the importance of neighbor j to node i , is calculated as:

$$\alpha_{ij}^\varphi = s_j(\sigma(\alpha_\varphi^T [\mathbf{h}_i \| \mathbf{h}_j])) \quad (2)$$

The node-level attention weight α_{ij}^φ represents the attention weight of attribute node i to attribute node j in meta-path φ ; σ represents the activation function; The feature vector of an attribute node (e.g., node i or j) is denoted by its initial embedding $\mathbf{h}_i$ or $\mathbf{h}_j$, respectively; α_φ^T represents the transpose of the node-level attention vector of meta-path φ to be learned. Then, based on the attention weights, the node features of the attribute node after attention processing are obtained, namely:

$$\mathbf{h}'_i = \sigma(\sum_{j \in N_\varphi(i)} \alpha_{ij}^\varphi \mathbf{h}_j) \quad (3)$$

In formula 3, $\mathbf{h}'_i$ represents the node features of attribute node i after attention processing; $N_\varphi(i)$ denotes all nodes of attribute node i regarding meta-path φ , and in this paper, $N_\varphi(i)$ includes attribute node i itself.

To further enhance the representation, Diffusion-HAN incorporates a diffusion process—a physical analogy describing how quantities naturally smooth out over a structure. After the attention-based aggregation, a first-order diffusion step is applied to propagate and balance features across the graph, thereby capturing broader neighborhood information:

$$\mathbf{Z}_i = \mathbf{h}'_i + \beta \text{div}(\alpha_{ij}^\varphi \nabla \mathbf{h}'_i) \quad (4)$$

$$\text{div}(\alpha_{ij}^\varphi \nabla \mathbf{h}'_i) = \sum_{j \in N_\varphi(i)} \alpha_{ij}^\varphi \frac{\mathbf{h}'_i - \mathbf{h}'_j}{|\mathbf{h}'_i - \mathbf{h}'_j|} \quad (5)$$

where $\mathbf{Z}_i$ denotes the path node embedding of attribute node i under meta-path φ and β is a preset hyperparameter controlling the strength of diffusion.

A balanced data subgraph is defined based on P meta-paths, where $P \geq 1$. Combining the path node embeddings of all attribute nodes in the p -th meta-path of a certain balanced data subgraph can obtain the meta-path embedding (group) $\mathbf{Z}_{\varphi_p}$ corresponding to meta-path φ_p .

Node-level attention assigns weights to a node's neighbors within a specific meta-path, capturing their relative importance in that semantic context. To integrate information across multiple meta-paths, we employ semantic-level attention, which evaluates and weights the importance of different meta-paths themselves. This allows the model to focus on the most relevant semantic perspectives for the downstream task. The aggregation is formulated as follows:

$$\omega_{\varphi_p} = \frac{1}{|U|} \sum_{i \in U} q^T \cdot \tanh(W \cdot \mathbf{z}_i^{\varphi_p} + f) \quad (6)$$

The importance score ω_{φ_p} for the p -th meta-path φ_p is computed using shared parameters: a weight matrix W , a bias vector f , and a semantic-level attention vector q^T . $\mathbf{z}_i^{\varphi_p}$ denotes the meta-path-specific embedding of attribute node i under meta-path φ_p (i.e., the diffusion-enhanced embedding $\mathbf{Z}_i$ for that node under that specific meta-path). The node set within this meta-path is denoted by U . After obtaining the importance scores ω_{φ_p} for all meta-paths, they are normalized via softmax to produce the final meta-path weights:

$$\beta_{\varphi_p} = \frac{\exp(\omega_{\varphi_p})}{\sum_{p=1}^P \exp(\omega_{\varphi_p})} \quad (7)$$

Each attribute node in the balanced attributed subgraph participates in multiple meta-paths. Its final embedding $\mathbf{Z}$ is therefore computed by aggregating its meta-path-specific embeddings, weighted by the corresponding β_{φ_p} :

$$\mathbf{Z} = \sum_{p=1}^P \beta_{\varphi_p} \mathbf{Z}_{\varphi_p} \quad (8)$$

The final node embeddings $\mathbf{Z}_i$ are aggregated to obtain an event-level representation $\mathbf{b}$ for the entire balanced attributed subgraph. This is achieved through a learnable weighted average:

$$\mathbf{b} = \frac{1}{n} \sum_{j=1}^n \varepsilon_j \mathbf{Z}_j \quad (9)$$

where n is the number of attribute nodes, $\mathbf{Z}_j$ is the embedding of node j , and ε_j is a learnable importance weight for node j .

To enhance the discriminative power of these event embeddings for anomaly detection, we introduce an inter-class distance loss L_1 . This loss encourages embeddings of normal and abnormal events to be maximally separated in the representation space. Specifically, we first compute the class centroid $\mathbf{c}_k$ for each class k (normal or abnormal) as the mean of its event embeddings:

$$\mathbf{c}_k = \frac{1}{N_k} \sum_{b \in B_k} \mathbf{b} \quad (10)$$

Let N_k and B_k denote the number and set of events of class $k \in \{normal, abnormal\}$, respectively. Given that there are only two event types, the inter-class distance loss L is formulated as:

$$L_1 = -\|\mathbf{c}_{normal} - \mathbf{c}_{abnormal}\| \quad (11)$$

In the process of continuous training and optimization of the loss function L_1 , the value of the parameter ε_j tends to become very large, so it is also necessary to use the normalized loss function L_2 , where

During training, the parameter ε_j tends to grow excessively under the influence of L_1 , potentially leading to numerical instability. To mitigate this, we introduce a normalization loss L_2 :

$$L_2 = \sum_{i=1}^n |\varepsilon_i| \quad (12)$$

Additionally, Diffusion-HAN employs an attribute-level structure-aware loss L_3 :

$$L_3 = \sum_{v=1}^V Y_v \ln(C \cdot \mathbf{Z}_v) \quad (13)$$

Here, Y_v represents the label of attribute node v , and C denotes the classifier parameters. The total loss function L is obtained by summing these three components:

$$L = L_1 + L_2 + L_3 \quad (14)$$

After determining the total loss function L , it can be used to train and optimize the network parameters of the Diffusion-HAN layer, and finally obtain the optimal parameters.

During training, the Diffusion-HAN model processes all balanced attribute subgraphs sequentially, sharing the same set of network parameters across all subgraphs. After iterating through the entire training set of subgraphs, the trained model effectively captures discriminative node features and is subsequently used for classifying test events.

For each event, we compute the similarity between every pair of connected attribute nodes based on their embeddings. Specifically, given the embeddings $\mathbf{Z}_j$ and $\mathbf{Z}_k$ of two adjacent nodes j and k , their similarity is defined as $\text{sim}(\mathbf{Z}_j, \mathbf{Z}_k)$. These pairwise similarities are then aggregated (e.g., by averaging) to form an event-level similarity feature.

$$\text{sim}(\mathbf{Z}_j, \mathbf{Z}_k) = \frac{\mathbf{Z}_j \cdot \mathbf{Z}_k}{\|\mathbf{Z}_j\| \cdot \|\mathbf{Z}_k\|} \quad (15)$$

By traversing all attribute nodes in event $\mathbf{b}$, we obtain its similarity group. Based on this similarity group, we compute the average similarity and variance for event $\mathbf{b}$, and append them to the similarity group to better capture the similarity relationships among attributes within the event. The average similarity and variance are given by, respectively:

$$\text{sim_avg}(b) = \frac{1}{n(n-1)} \sum_{j=1}^{n-1} \sum_{k=j+1}^n \text{sim}(\mathbf{Z}_j, \mathbf{Z}_k) \quad (16)$$

$$\text{sim_var}(b) = \frac{2}{n(n-1)} \sum_{j=1}^{n-1} \sum_{k=j+1}^n v(\mathbf{Z}_j, \mathbf{Z}_k) \quad (17)$$

Where $v(\mathbf{Z}_j, \mathbf{Z}_k) = (\text{sim}(\mathbf{Z}_j, \mathbf{Z}_k) - \text{sim_avg}(b))^2$

The similarity group of the event is fed into a softmax classifier, which is trained iteratively in a supervised manner. The classifier parameters are optimized using the softmax cross-entropy loss. Finally, the trained classifier is applied to detect abnormal data in network traffic for network anomaly detection.

Algorithm 1: The Procedure of Anomaly Detection

Input: The set of behavioral events B , the time point t for splitting training and test sets
Output: A binary classification result indicating whether each event in the test set is an intrusion or not

- 1 Divide B into training set B_{train} and test set B_{test} according to the time point t ;
- 2 Construct multi-temporal balanced attribute subgraphs from B_{train} using SMOTE for balancing normal and anomaly data proportions;
- 3 **for** $G_i \in B_{\text{train}}$ (each temporal balanced attribute subgraph) **do**
- 4 Perform representation learning on G_i using Diffusion-HAN network with hierarchical attention mechanism;
- 5 Initialize constructed features $f_i \leftarrow \emptyset$ for each event $b_i \in G_i$;
- 6 Transform each event b_i into a feature vector $\mathbf{b}_i$ based on the learned representations;
- 7 **for** $\mathbf{Z}_j \in \mathbf{b}_i$ (each node embedding in the event) **do**
- 8 **for** $\mathbf{Z}_k \in \mathbf{b}_i$ (each other node embedding) **do**
- 9 Compute $\text{sim}(\mathbf{Z}_j, \mathbf{Z}_k)$ using Eq. 15;
- 10 $f_i \leftarrow f_i.\text{append}(\text{sim}(\mathbf{Z}_j, \mathbf{Z}_k))$;
- 11 **end**
- 12 **end**
- 13 Compute $\text{sim_avg}(\mathbf{b}_i)$ and $\text{sim_var}(\mathbf{b}_i)$ using Eq. 16 and Eq. 17;
- 14 $f_i \leftarrow f_i.\text{append}(\text{sim_avg}(\mathbf{b}_i))$;
- 15 $f_i \leftarrow f_i.\text{append}(\text{sim_var}(\mathbf{b}_i))$;
- 16 **end**
- 17 Train a softmax classifier $\text{CLF}(f_i; \phi_c)$ based on all constructed features f_i from all subgraphs G_i ;
- 18 Extract validation samples from training set to determine the optimal parameters ϕ_c^* ;
- 19 Initialize judgment results $R \leftarrow \emptyset$;
- 20 **for** $b_i \in B_{\text{test}}$ **do**
- 21 Construct corresponding attribute subgraph G'_i for b_i ;
- 22 Perform representation learning on G'_i using Diffusion-HAN with previously learned parameters;
- 23 Transform b_i into feature vector $\mathbf{b}'_i$ based on the learned representations;
- 24 Get the classification result $r_i = \text{CLF}(\mathbf{b}'_i; \phi_c^*)$;
- 25 Append the binary classification result to R : $R \leftarrow R.\text{append}(r_i)$;
- 26 **end**
- 27 **return** judgment results R ;

4 Model Evaluation

We evaluate the proposed model in terms of classification performance and computational efficiency.

4.1 Experimental Settings

To comprehensively assess classification performance, we conduct experiments on four widely-used network traffic datasets: **Kyoto**, **UNSW-NB15**, **NSL-KDD**, and **Darknet**. Their key characteristics are summarized below.

Kyoto Dataset. Collected by the Information Security Center at Kyoto University from 2006 to 2009, the Kyoto dataset provides real-world network traffic traces that include both normal activities and diverse attack patterns.

Darknet Dataset. This dataset comprises traffic captured from unallocated or reserved IP address space (often referred to as the “darknet”). Since legitimate services rarely use these addresses, the traffic largely reflects malicious activities such as scanning, backdoor communication, and exploitation attempts.

UNSW-NB15 Dataset. Created by the Cyber Security and Cyber Crime Research Center of the University of New South Wales, Australia, UNSW-NB15 is a comprehensive intrusion detection benchmark containing a mix of normal behavior and several attack categories, including DoS, reconnaissance, malware, and exploits.

NSL-KDD Dataset. Derived from the well-known KDD Cup 1999 dataset, NSL-KDD was curated by researchers at Dalhousie University to address redundancy and imbalance issues in the original data. It covers normal traffic as well as four major attack types: Denial of Service (DoS), Remote-to-Local (R2L), User-to-Root (U2R), and Probing.

The composition of the four datasets is shown in the following table:

	Total Samples	Positive Samples	Negative Samples
nsL_kdd	148,517	77,054	71,463
kyoto	126,216	119,670	6,546
darknet	110,442	48,175	62,267
unsw	1,325,038	74,964	1,250,074

Table 1. Statistical Information of Different Datasets

In our model, we construct attributed graphs to represent these raw data, and then apply the node representations obtained from the event-attribute composite model to the detection model. We select a specific time point in the time series, dividing the data before the selected time point as the training set and the data after as the test set.

Experiments were conducted on a Linux system equipped with an Intel® Xeon® Platinum 8488C CPU @ 3.8GHz, 48GB of RAM, and an NVIDIA GeForce RTX 3090 GPU.

To adapt the raw network data to our model, we first transform them into attributed graphs. The node representations learned by the event-attribute composite model are then fed into the detection module. For temporal evaluation, we select a cutoff timestamp in each dataset’s time series; data before this point constitute the training set, and data after it form the test set.

All experiments were performed on a Linux server with the following hardware configuration:

- **CPU:** Intel® Xeon® Platinum 8488C @ 3.8GHz
- **Memory:** 48 GB RAM
- **GPU:** NVIDIA GeForce RTX 3090 (24 GB VRAM)

4.2 Experimental Results

Our experimental evaluation metrics mainly include the PRC curve and the area under the PRC curve, i.e., the PRC-AUC value, as well as the time complexity of the experiment.

4.2.1 PRC Curve and PRC-AUC (Analysis of Anomaly Detection Effectiveness)

Given the severe class imbalance inherent in intrusion detection datasets (where normal traffic vastly outnumbers attack traffic), conventional metrics such as accuracy are inadequate. A trivial classifier that predicts all instances as normal would achieve high accuracy while failing to detect any attacks. Therefore, we employ the **Precision-Recall (PR) curve** and its **Area Under the Curve (AUC)** as primary evaluation metrics.

The PR curve depicts the trade-off between precision (the proportion of true attacks among predicted attacks) and recall (the proportion of attacks correctly identified) across different classification thresholds. The AUC-PR summarizes the overall performance, with values closer to 1.0 indicating better detection capability for the minority (attack) class.

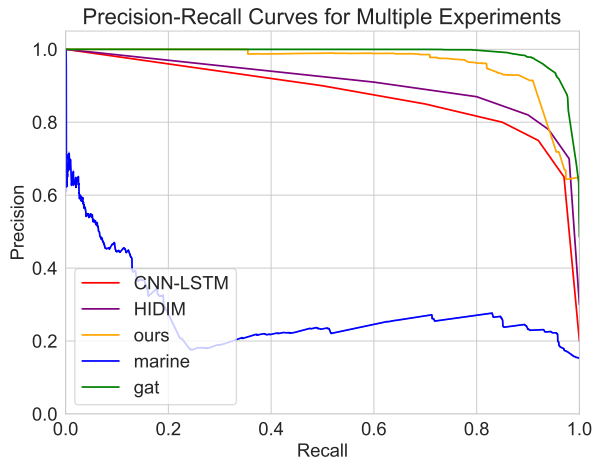
On both the NSL-KDD and UNSW-NB15 datasets, our method and GAT achieve closely matched, high performance, with PRC-AUC values exceeding 0.98. This substantially outperforms other baselines: DLL-IDS (0.89 on NSL-KDD), CNN-LSTM (0.95 on NSL-KDD), and HIDIM (0.96 on NSL-KDD and 0.94 on UNSW-NB15).

Notably, the **marine** method exhibits clear limitations here. First, it failed to scale to the large-scale UNSW-NB15 dataset due to prohibitive training time and excessive GPU memory consumption, which prevented its execution. Second, on NSL-KDD—where network connections exhibit strong temporal dependencies—marine struggles to capture these patterns, resulting in significantly degraded performance.

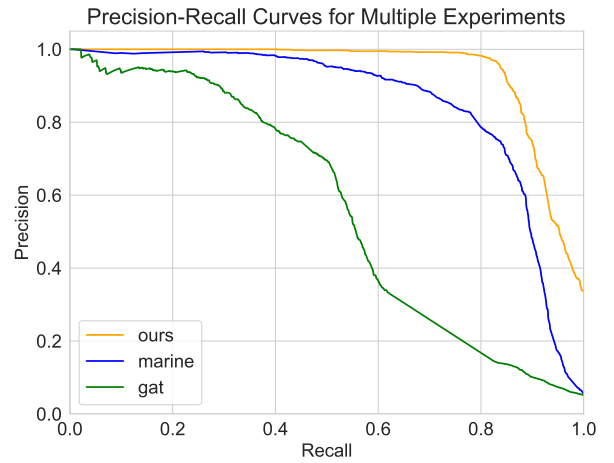
On the Darknet dataset, our method achieves a PRC-AUC approximately 0.06 higher than the strongest baseline, with its PRC curve consistently dominating the others. The advantage becomes even more pronounced on the Kyoto dataset: our

	darknet	kdd	kyoto	unsw
ours_method	0.93	0.99	0.94	0.98
gat	0.85	0.99	0.55	0.98
marine	0.87	0.26	0.85	-
Logistic Regression	0.72	0.96	0.46	0.92
dll-ids	-	0.89	-	-
CNN-LSTM	-	0.95	-	-
HIDIM	-	0.96	-	0.94

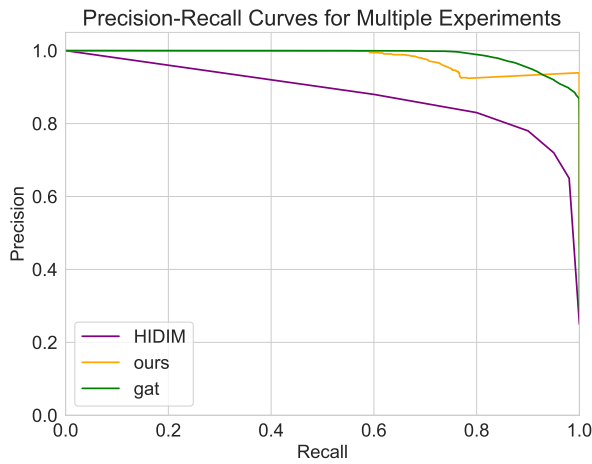
Table 2. PRC-AUC Values of Different Methods on Various Datasets



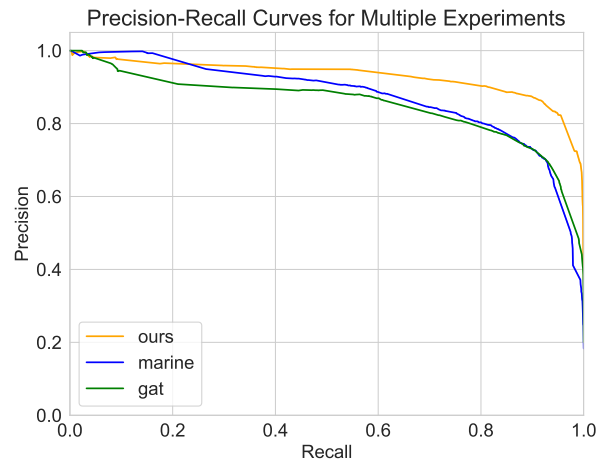
(a) kdd_prc_curves



(b) kyoto_prc_curves



(c) unsw_prc_curves



(d) darknet_prc_curves

Figure 4. PRC Curves of the Model on Four Datasets

method’s PRC curve remains stably above both baselines, yielding a PRC-AUC that surpasses the second-best (marine) by 0.09 and outperforms GAT by 0.37.

Across these two datasets, both GAT and marine show notable performance instability, whereas our method delivers consistent and robust detection accuracy.

4.2.2 Time Complexity

In practical intrusion detection systems, time complexity is not only a critical metric but also directly impacts real-time responsiveness and overall system performance. Low time complexity enables the detection algorithm to maintain high speed and accuracy when processing large-scale data—critical for timely identification and mitigation of security threats. Moreover, reduced computational overhead lowers resource consumption, thereby decreasing operational costs and enhancing system scalability and stability. Consequently, time complexity must be carefully considered in the design and selection of intrusion detection methods.

We evaluated the training and inference times of marine, GAT, and our method on five datasets. The total time—from model training on the training set to prediction on the test set—is reported in the figure below.

As shown, the marine method is significantly slower than the other two approaches and fails to complete execution on larger datasets under the same hardware configuration.

Our method, which operates on subgraphs, incurs slightly higher computational overhead than GAT’s direct training on a single attributed graph, resulting in modestly longer runtime. Nevertheless, its overall time consumption remains comparable to that of GAT, and both methods demonstrate favorable efficiency.

These results indicate that our approach achieves a balanced trade-off between computational efficiency and prediction accuracy, offering advantages over existing intrusion detection methods.

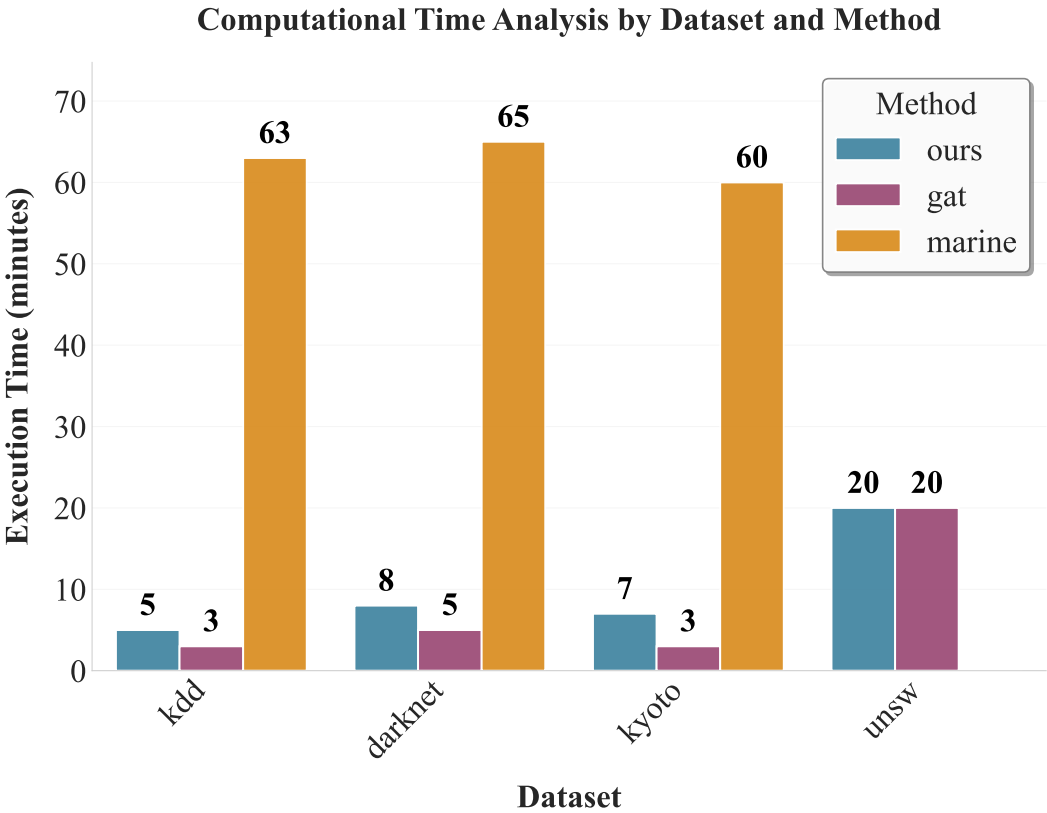


Figure 5. Time Consumption of the Model on Different Datasets

4.2.3 Hyperparameter Analysis

We conducted a hyperparameter analysis of our method to determine the optimal subgraph size. In our method, the parameter that has the greatest impact on the experiment is the proportion of data in a subgraph to the total data. In this section, we conducted experiments with 20%, 30%, and 35% respectively, and the results are shown in the following table:

	darknet	kdd	kyoto	unsw
0.2	0.83	0.97	0.86	0.97
0.3	0.63	0.96	0.87	-
0.35	0.94	0.95	0.93	-

Table 3. PRC-AUC Values Obtained Using Different Subgraph Proportions

The results show that a sampling proportion of 35% achieves the best performance. However, on very large datasets, using a higher proportion to construct attribute subgraphs can lead to excessive GPU memory consumption, causing the program to fail due to out-of-memory (OOM) errors. In UNSW, proportions of 30% and 35% cause insufficient GPU memory. Therefore, this proportion can be adjusted according to the actual situation to adapt to different environments.

4.2.4 Experimental Design Analysis

We employ a representation learning approach to encode the input data into rich feature representations, which are subsequently used for classification. Compared to standard neural networks, our method captures more discriminative patterns, leading to improved detection performance.

To further validate its effectiveness, we compare against Logistic Regression (LR), which operates directly on the original two-dimensional feature matrix (samples $\times$ attributes). As shown in Table II, LR exhibits unstable performance and lags significantly behind our model. This demonstrates that the representation learning framework adopted in our approach offers clear advantages over shallow linear models.

4.2.5 Ablation Study

We conduct an ablation study to investigate the contribution of each component in our method.

First, we merge all subgraphs into a single attributed graph and evaluate the resulting model. This variant allows us to assess the effectiveness of our subgraph-based design.

Second, we replace the Diffusion-HAN module with a standard GAT for representation learning, thereby isolating the impact of our proposed architecture.

The PRC-AUC results of these variants are reported in Table iv.

	darknet	kdd	kyoto	unsw
our method	0.93	0.99	0.94	0.98
No-HAN (GAT)	0.85	0.99	0.55	0.98
No-Sub (Single Graph)	0.87	0.26	0.85	-

Table 4. PRC-AUC Values of Ablation Studies

The curves labeled "nohan" and "nosub" in the figure are the PRC curves of the ablation experiments on the Diffusion-HAN network and the attribute subgraph, respectively. Our method has the highest PRC-AUC value, and the PRC curve is above the two ablation experiments, indicating that these two improvements are effective for improving model performance.

5 Application Scenario and Effectiveness: Anomaly Detection in Smart Grid Industrial Control Systems

5.1 Scenario and Challenges

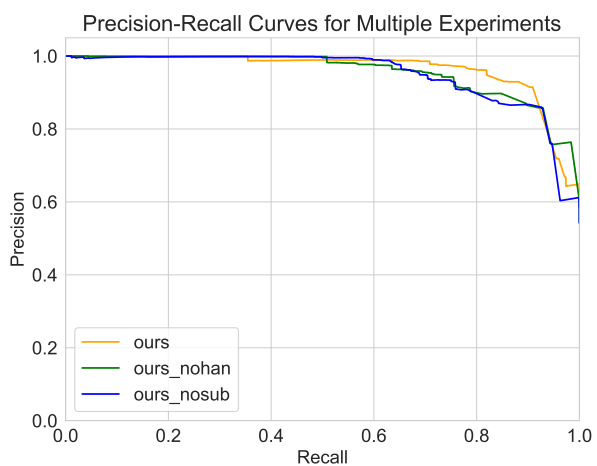
In the smart grid substation monitoring system (which operates under the IEC 61850 protocol), anomaly detection faces two major challenges:

Extreme data imbalance: Over 99.9% of the traffic consists of normal operations, while attack samples—such as unauthorized setting-value modifications—are extremely rare. Strong attack stealth: Malicious instructions closely resemble legitimate operational traffic in their characteristics, making them difficult for traditional methods to distinguish.

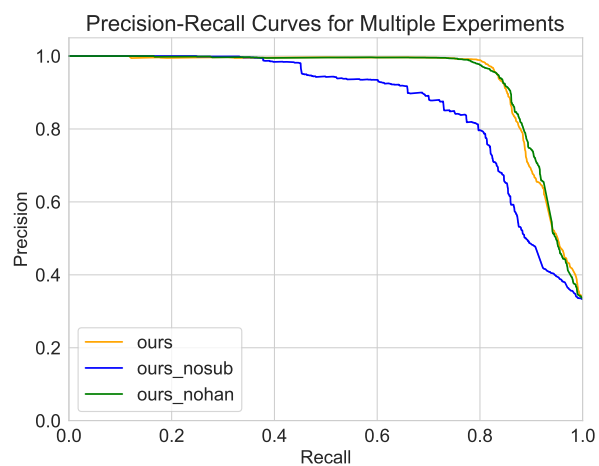
5.2 Application of the Method

To address these challenges, our method is implemented in this scenario as follows:

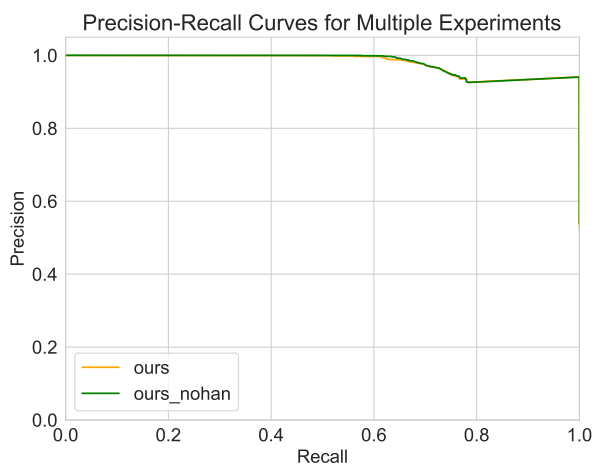
Construction of multi-temporal attribute subgraphs: Network traffic is divided into multiple operational periods (e.g., stable intervals and peak operation windows) based on power grid operating rules, and an attribute subgraph is constructed for each



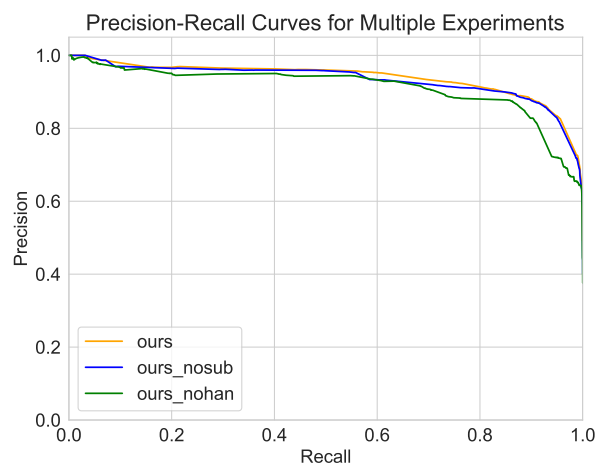
(a) kdd_prc_curves



(b) kyoto_prc_curves



(c) unsw_prc_curves



(d) darknet_prc_curves

Figure 6. PRC Curves of Our Model on Four Datasets

period. SMOTE-based data balancing: Rare anomalous samples within each subgraph are oversampled using SMOTE to effectively mitigate severe class imbalance. Diffusion-HAN representation learning: The Diffusion-HAN module leverages heterogeneous graph diffusion and a hierarchical attention mechanism to accurately capture complex interaction patterns among devices and communication protocols.

5.3 Core Effectiveness

Experiments on real-world power grid traffic datasets demonstrate that our method achieves significantly better performance than mainstream solutions:

	Precision	Recall	F1-Score	UAD
Diffusion-HAN (Ours)	0.968	0.951	0.959	0.923
GAT	0.921	0.897	0.909	0.854
Commercial Rule-based IDS	0.990	0.705	0.825	0.451

Table 5. Performance Comparison of Different Methods on the Smart Grid Dataset

The proposed method achieves high precision (low false positives) and high recall (low false negatives), yielding the best overall performance in terms of F1-score. Notably, its detection rate for unknown attack variants (UAD) substantially exceeds that of all baseline methods, demonstrating strong generalization capability and practical value for securing critical infrastructure.

6 Conclusion and Outlook

This paper addresses network traffic anomaly detection—a core component of cybersecurity situational awareness—and identifies data imbalance and classification uncertainty as the main obstacles to building effective detection models. To tackle these challenges, we propose an innovative approach that combines multi-temporal balanced attribute subgraph construction with heterogeneous graph diffusion enhancement.

Experimental results show that the method not only significantly improves UAD detection rate and F1-score (by 7% and 5%, respectively), but also maintains robust performance when processing large-scale, highly imbalanced network traffic, confirming its robustness and scalability. In particular, the integration of the Diffusion-HAN network effectively aggregates neighborhood information, and its hierarchical attention mechanism further enhances model performance.

However, the rapid advancement of network technologies and the continuous evolution of attack techniques call for ongoing innovation. Future research can be carried out in the following directions:

Feature fusion and selection: Explore effective ways to fuse diverse network features and attributes, and develop advanced feature selection algorithms to improve model accuracy and generalization. Real-time performance and scalability optimization: Investigate how to enhance the model’s real-time response speed and scalability in large-scale network environments while preserving detection accuracy, to address increasingly complex threats. Cross-domain and cross-platform detection: Study anomaly detection strategies that generalize across different network domains (e.g., IoT, cloud computing) and operating systems (e.g., Windows, Linux), enabling more comprehensive cybersecurity protection. In summary, the method proposed in this paper provides a novel and effective solution to key challenges in network anomaly detection and establishes a solid foundation for building accurate, scalable detection systems. Moving forward, we will continue to deepen our research and explore emerging techniques to address the growing demands of network security.

Funding

This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFB3103402 and in part by the National Natural Science Foundation of China under Grant Nos. 62072200, 62172176, and 62127808.

Data Availability

The public datasets (NSL-KDD, UNSW-NB15, Kyoto, Darknet) analysed during this study are all publicly available open benchmark datasets, which can be downloaded from their official published repositories: NSL-KDD: <https://www.unb.ca/cic/datasets/nsl.html>. UNSW-NB15: <https://research.unsw.edu.au/projects/unsw-nb15-dataset>. Kyoto Dataset: https://www.takakura.com/Kyoto_data/. Darknet Dataset: <https://www.unb.ca/cic/datasets/darknet2020.html>. The smart grid industrial control traffic dataset generated and analysed for practical scenario verification is not publicly available due to the confidentiality requirements of power critical infrastructure operators, which contains sensitive substation operation communication data. This restricted dataset is available from the corresponding author on reasonable request with official permission from the data provider.

References

1. Wang, C. & Zhu, H. Wrongdoing monitor: A graph-based behavioral anomaly detection in cyber security. *IEEE Transactions on Inf. Forensics Secur.* **17**, 2703 (2022).
2. Jing, C., Wang, C. & Yan, C. Thinking like a fraudster: Detecting fraudulent transactions via statistical sequential features. In *Proceedings of Financial Cryptography and Data Security*, vol. 11598, 588–604 (Frigate Bay, St. Kitts and Nevis, 2019).
3. Jiang, Z. *et al.* Target privacy preserving for social networks. In *Proceedings of IEEE ICDE*, 1862–1865 (Dallas, TX, USA, 2020).
4. Chawla, N. V., Bowyer, K. W., Hall, L. O. & Kegelmeyer, W. P. SMOTE: Synthetic minority over-sampling technique. *J. artificial intelligence research* **16**, 321–357 (2002).
5. Lian, J. *et al.* xdeepfm: Combining explicit and implicit feature interactions for recommender systems. In *Proceedings of ACM SIGKDD*, 1754–1763 (London, U.K., 2018).
6. Liu, F. *et al.* Log2vec: A heterogeneous graph embedding based approach for detecting cyber threats within enterprise. In *Proceedings of CCS*, 1777–1794 (London, U.K., 2019).
7. Li, Z., Zeng, J., Chen, Y. & Liang, Z. Attackg: Constructing technique knowledge graph from cyber threat intelligence reports. In *CoRR*, vol. abs/2111.07093 (2021).
8. Chen, X., Song, X., Peng, G., Feng, S. & Nie, L. Adversarial-enhanced hybrid graph network for user identity linkage. In *Proceedings of SIGIR*, 1084–1093 (Virtual Event, CA, USA, 2021).
9. Wang, Z., Zhang, W., Liu, N. & Wang, J. Transparent classification with multilayer logical perceptrons and random binarization. In *Proceedings of AAAI*, 6331–6339 (New York, NY, USA, 2020).
10. He, R. & McAuley, J. J. Fusing similarity models with markov chains for sparse sequential recommendation. In *Proceedings of IEEE ICDM*, 191–200 (Barcelona, Spain, 2016).
11. Zhou, G. e. a. Deep interest network for click-through rate prediction. In *Proceedings of ACM SIGKDD*, 1059–1068 (London, U.K., 2018).
12. Okoro, E. A., Ubom, A. E. & Ukommi, S. U. Evaluation of machine learning model for network anomaly detection: Support vector machine. *J. Eng. Res. Reports* **26**, 326–342 (2024).
13. Tang, J. & Wang, K. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of WSDM*, 565–573 (Marina Del Rey, CA, USA, 2018).
14. Gholampour, S. Impact of nature of medical data on machine and deep learning for imbalanced datasets: Clinical validity of smote is questionable. *Mach. Learn. Knowl. Extr.* **6**, 827– (2024).
15. Peng, H. e. a. Fine-grained event categorization with heterogeneous graph convolutional networks. In *Proceedings of IJCAI*, 3238–3245 (Macao, China, 2019).
16. Peng, G., Weiyong, Y. & Haotian, Z. Detecting unknown threat based on continuous-time dynamic heterogeneous graph network. *Wirel. Commun. Mob. Comput.* **2022** (2022).
17. Wang, J., Wen, R., Wu, C., Huang, Y. & Xion, J. Fdgars: Fraudster detection via graph convolutional networks in online app review system. In *Proceedings of WWW*, 310–316 (San Francisco, CA, USA, 2019).
18. Eswaran, S. e. a. Game action modeling for fine grained analyses of player behavior in multi-player card games (rummy as case study). In *Proceedings of ACM SIGKDD*, 2657–2665 (Virtual Event, CA, USA, 2020).
19. Wei, G. & Wang, Z. Adoption and realization of deep learning in network traffic anomaly detection device design. *Soft Comput.* **25**, 1–12 (2020).
20. Ji, Z., Kong, D. & Yang, Y. Assl-hgat: Active semi-supervised learning empowered heterogeneous graph attention network. *Knowledge-Based Syst.* **290**, 111567 (2024).
21. Bamber, S. S., Katkuri, A. V. R., Sharma, S. & Angurala, M. A hybrid cnn-lstm approach for intelligent cyber intrusion detection system. *arXiv* (2024).
22. Zhou, W. *et al.* Hidim: A novel framework of network intrusion detection for hierarchical dependency and class imbalance. *IEEE Transactions on Inf. Forensics Secur.* (2024).
23. Narmadha, S. & Balaji, N. Improved network anomaly detection system using optimized autoencoder - lstm. *Expert. Syst. with Appl.* **273**, 126854 (2025).

24. Nkashama, D. K. *et al.* Anadoe: Autoencoder-based network anomaly detection with outlier exposure. *IEEE Internet Things J.* **13**, 17272–17285 (2026).