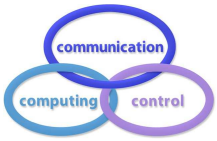
INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 20, Issue: 6, Month: December, Year: 2025
Article Number: 6832, https://doi.org/10.15837/ijccc.2025.6.6832



CCC Publications

Intrusion Detection in Marine Networks Based on Feature Dimension Reduction and Graph Convolution

Hongyan Xing, Zhiwei Ni, Wei Xu

Hongyan Xing*
1. School of Electrical and Energy Engineering, Nantong Institute of Technology, Nantong Jiangsu, 226001, China
2. School of Electronics and Information Engineering, Nanjing University of Information Science and Technology, Nanjing Jiangsu, 210044, China
*Corresponding author: xinghy@nuist.edu.cn

Zhiwei Ni
School of Electronics and Information Engineering, Nanjing University of Information Science and Technology, Nanjing Jiangsu, 210044, China
202212490213@nuist.edu.cn

Wei Xu
School of Electronics and Information Engineering, Nanjing University of Information Science and Technology, Nanjing Jiangsu, 210044, China
xw@nuist.edu.cn

Abstract

In response to the challenges in the realm of intrusion detection for marine meteorological sensor networks, such as difficulties in model training, inadequate detection performance, and low operational efficiency, we propose an advanced intrusion detection model. This model leverages feature dimensionality reduction, utilizing the Genetic Algorithm based on Random Forest (GARF) technique, to discern the most effective feature subset, thereby streamlining the original network intrusion detection dataset. An Approximate Nearest Neighbor (ANN) algorithm is employed to convert network traffic data into a graph structure, and the Approximate Nearest Neighbor-based Graph Convolutional Neural Network (AGCN) is constructed for traffic classification prediction on this graph-structured data. Our simulation experiments conducted on the NSL-KDD dataset have yielded positive results, demonstrating the model's enhanced intrusion detection capabilities and efficiency, and affirming its potential to provide robust security measures for marine meteorological sensor networks.

Keywords: Network intrusion detection, Marine meteorological sensor network, Graph convolutional network, Feature dimensionality deduction.

1 Introduction

The Internet of Things (IoT), benefiting from the rapid progression in smart device technology and innovative communication methods, has been integrated into various sectors, including marine meteorological sensor networks (MMSN) [1, 2, 3]. The MMSN is designed to collect comprehensive regional marine meteorological and hydrological data, encompassing seawater temperature, current dynamics, tidal phases, and their intensities. Such information is of paramount importance to maritime navigation, oilfield operations, and aquaculture activities [4].

However, while IoT technology brings convenience to MMSN, it also increases network security risks [5]. The high-frequency communication between marine meteorological sensors, utilizing public protocols, renders the MMSN inherently vulnerable to attacks from unauthorized networks. This exposure significantly jeopardizes the cybersecurity of the MMSN. Furthermore, the potential leakage of marine meteorological and hydrological information poses a substantial threat to national strategic security [6, 7, 8]. The heterogeneous characteristics of MMSN devices make it difficult to design and deploy an effective intrusion prevention mechanism for MMSN [9]. Facing increasingly diverse attack types and attack methods, establishing a reliable and effective intrusion prevention mechanism is urgent for the further development of the marine, and traditional Passive defense mechanisms, such as firewalls, traffic encryption, user authentication, etc., are difficult to cope with [10]. Intrusion Detection System (IDS) is widely used as an active defense mechanism against network intrusion attacks by intruders to enhance system security and is widely used with IoT security, smart grids, and maritime transportation systems [11, 12]. Therefore, the reliability and security of MMSN will be improved by establishing an MMSN intrusion detection system to identify network anomalous traffic with aggressiveness in MMSN.

IDS are mainly divided into two categories: Host-based Intrusion Detection Systems (HIDS) and Network-based Intrusion Detection Systems (NIDS) [13]. HIDS keeps an eye on system logs and response records, spotting intrusions by checking how files are handled, and it's typically found on local gear like PCs and database devices [14]. Feature-based NIDS and anomaly-based NIDS are two typical types of NIDS [15]. Feature-based NIDS and anomaly-based NIDS represent the two principal categories of network-based (NIDS). A feature-based NIDS operates by establishing a set of rules to ascertain whether a given behavior constitutes an intrusion. These rules are essentially feature-based representations of recognized attack patterns. While this approach excels in detecting known attacks, it is less effective in identifying previously unseen attacks, commonly referred to as zero-day attacks [16]. Anomaly-based NIDS constructs a profile of typical behavior by extensively analyzing statistics derived from network traffic data(NTD). This methodology exhibits enhanced generalization capabilities, which has garnered significant attention in recent years.

In the scenarios under consideration for the MMSN, several notable characteristics are present, including the harsh operational environment and the heterogeneity of the devices utilized. However, a significant majority of these DL methodologies operate on the assumption of centralized training data and unlimited computational resources, neglecting the time cost associated with the intrusion detection process. Given that real NTD typically possesses high-dimensional features, the application of complex detection neural networks can entail significant computational demands, extensive time consumption, and may even impact real-time detection capabilities. Consequently, the practical implementation of these methods within the MMSN environment is rendered infeasible.

To solve the dilemma faced by MMSN and to transmit and store marine meteorological and hydrological information more securely, this paper proposes a GARF-based intrusion detection method for AGCN networks, and the main contributions of this paper are as follows:

1. Propose a reliable feature dimensionality reduction algorithm combining GA and RF to reduce the feature dimensions of NTD and introduce it to NID.

2. An ANN algorithm is proposed to convert NID data into graph-structured data, and a GCN model is constructed for efficient graph classification of graph-structured data.

3. The simulation experiment was performed on the GARF-AGCN using the NSL-KDD dataset, and it was benchmarked against several other state-of-the-art methods. The experimental outcomes indicate that the GARF-AGCN intrusion detection model demonstrates superior performance and efficiency in MMSN intrusion detection.

The remainder of this paper is organized as follows: The related work are presented in Section 2.The design of our proposed model is discussed in Section 3. The experimental design and results are presented in Section 4. The conclusions are discussed in Section 5.

2 Related work

IDS can be conceptualized as a classification issue pertaining to network behavior. Leveraging Machine Learning (ML) and Deep Learning (DL) methodologies, several conventional classification algorithms have been integrated into the domain of network intrusion detection (NID) [17, 18]. Traditional Machine Learning techniques, including logistic regression (LR), support vector machine (SVM), and random forest (RF), are extensively employed in the realm of NIDS for the identification of normal and abnormal network behaviors. Additionally, hybrid models are typically utilized to enhance the efficacy of IDS [19]. Vijayanand et al. have developed a novel intrusion detection system for wireless networks, which incorporates genetic algorithm-based feature selection and multiple support vector machine classifiers. Their experimental findings indicate that this system exhibits a high degree of accuracy in detecting network intrusions [20]. Disha et al. utilized the Gini Impurity-based Weighted Random Forest model as an embedded feature selection technique, effectively reducing the feature dimensionality and enhancing the efficiency of intrusion detection [21]. Gu et al. have developed an SVM and Naive Bayes-based IDS framework. This framework employs Naive Bayes technology to modify the original features, which are then utilized to train the SVM classifier, thereby creating an effective intrusion detection model. This strategy demonstrates superior performance in terms of detection rate and false alarm rate, and exhibits greater robustness compared to other technologies [22].

DL techniques have advanced considerably since the 2010s, with Convolutional Neural Networks (CNN), Deep Neural Networks (DNN), Recurrent Neural Networks (RNN), and other associated algorithms being extensively employed in NID. DL approaches address the limitations of conventional ML methods in effectively handling high-dimensional and large-scale data [23, 24]. CNN, DNN, and RNN excel in extracting abstract features from large volumes of NTD, thereby overcoming the limitations of shallow ML algorithms. These techniques provide a novel approach to achieving more precise intrusion detection [25]. Ji Z et al. have proposed a novel method for intrusion detection, combining CNN with Conditional Generative Adversarial Network (CGAN). This approach achieves data balance, utilizing CGAN for data augmentation, and employs CNN as a classification model to enhance the accuracy of intrusion detection [26]. Mushtaq et al. have proposed an intrusion detection methodology that employs Autoencoders (AE) to extract optimal features and subsequently utilizes Bidirectional Long Short-Term Memory (BILSTM) networks for classifying normal and abnormal samples, thereby achieving a relatively high level of intrusion detection accuracy [27]. Fu Y et al. have proposed an intrusion detection approach that initially extracts sequence features from data traffic through CNN. This is followed by redistributing the weight of each channel using the attention mechanism, and subsequently employing BiLSTM networks to learn from the sequence feature network, thereby further enhancing the performance of NID [28]. One of the key strengths of the aforementioned DL-based intrusion detection methods is their superior performance compared to those based on traditional ML. However, these methods are accompanied by challenges such as increased model complexity, difficulties in model training, higher computational resource requirements, and elevated time costs for detection. Existing techniques continue to struggle in achieving a harmonious balance between intrusion detection performance and efficiency.

3 System description

3.1 MMSN intrusion detection system

The structure of the Marine Meteorological Sensor Network (MMSN) is shown in Figure 1. MMSN mainly consists of marine servers (coastal servers, marine base station servers, etc.), marine mobile terminal nodes (meteorological buoys, meteorological drones, etc.), marine switches (ship meteorological stations, meteorological detection ships, etc.), and satellites [29].

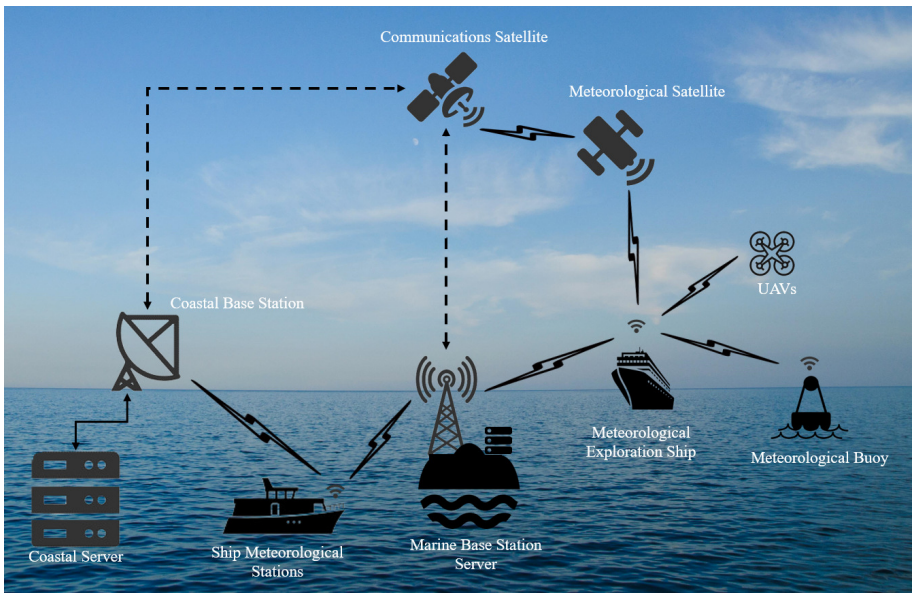


Figure 1: Marine meteorological sensor network

Each marine meteorological mobile terminal node can transmit and forward network traffic, and the mobile terminal node has limited computing resources, making it impossible to implement large-scale network protection mechanisms. Marine base station servers have sufficient computing power to deploy network security mechanisms. The marine switch close to the marine meteorological mobile terminal node can capture the network traffic of the marine meteorological mobile terminal node of the marine base station server.

Marine meteorological mobile terminal nodes are characterized by their diverse types, complex hierarchical structures, substantial data transmission capacities, and challenging deployment environments [30]. Within the Marine Meteorological and Marine Network (MMSN), the marine switch serves as the primary forwarding device, capable of forwarding or discarding traffic based on the flow table. The transmission process of all network traffic can be conceptualized as:

$$H_{src} \to (S_1, R_1) \to \cdots \to (S_i, R_i) \to H_{des} \tag{1}$$

where, H denotes an IP host, src and des denote source and destination addresses, respectively, and S_i and R_i denote the i-th network node and its traffic processing rules.

In addition, marine base station servers and marine switches have strong computing capabilities and computing resources and can be deployed for advanced defense. It is assumed that criminals invade MMSN network nodes from the external network by initiating malicious network traffic, further harming and controlling vulnerable MMSN network nodes, and placing MMSN network nodes in an untrustworthy state. An infected node within the MMSN due to malicious network traffic loses its intended functionality and transforms into a malicious MMSN node. Subsequently, it can initiate a substantial amount of malicious traffic, targeting other normal MMSN nodes, which could potentially lead to the paralysis of the entire network, resulting in severe consequences. To prevent the MMSN from being compromised and to ensure its normal operation, it is imperative to conduct timely NID within the MMSN.

In light of the specific characteristics of the MMSN and the requisite functionalities for NID, the framework design of the MMSN Intrusion Detection System is depicted in Figure 2.

This framework comprises a data collection module, a NID module, and a feedback processing module. The data collection module primarily functions to collect the network traffic data flow within the MMSN, extract the relevant data characteristics from this traffic, and subsequently transfer this information to the NID module. The NID module is mainly responsible for pre-processing the network traffic data, using efficient classifiers to classify and detect the incoming network traffic data features, identifying abnormal network traffic data and network traffic attacks, and passing the NID results to the response feedback module. The response feedback module is designed to terminate services

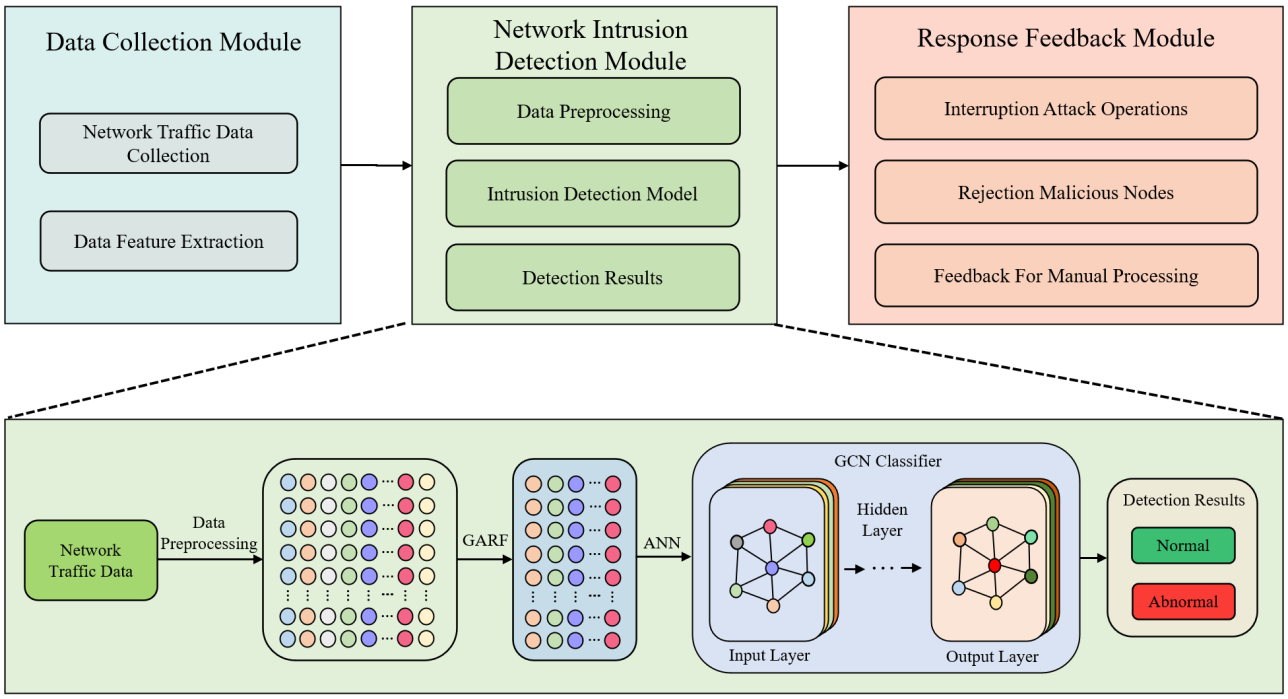


Figure 2: Marine meteorological sensor network

at the affected node in the event of an attack, to filter out network traffic originating from malicious nodes, and to provide detailed reports to human personnel for subsequent action regarding the network security incident.

The NID model based on GARF-AGCN proposed in this paper mainly consists of three parts, such as the GARF feature dimensionality reduction module, and ANN graph structure construction module, and the GCN classification module. First of all, the data preprocessing such as numerical and standardization is performed on the network traffic data. Then we do feature dimensionality reduction by the GARF algorithm to find the optimal feature subset and accelerate the training speed of the model; then we use the ANN algorithm to learn the acquaintance characteristics of neighboring nodes, get the information of approximate nearest-neighbor nodes, and construct the optimal graph structure data, and then we make efficient classification prediction on the graph structure data by the GCN classifier, and finally, we get the intrusion detection results of the network traffic.

3.2 Data preprocessing module

In the context of marine meteorological sensor NID, the preprocessing of raw data is imperative to enhance the precision of feature identification. This paper presents a data preprocessing module specifically designed for the MMSN, which is demonstrated to effectively distill the principal features from network traffic datasets. This module significantly reduces the dimensionality of features, optimizes computational resource utilization, and boosts the efficacy of NID processes. The module is structured into two integral components: Numerical processing. Symbolic features within the NID dataset must be transformed into digital format for model training purposes. To achieve this, One hot encoding will be employed to convert the symbols features in the dataset into digital features.

Standardized processing. In NID datasets, the variability in the range of values across different features can be substantial. Such a wide disparity in feature values may hinder the model's ability to converge effectively. Consequently, it is imperative to employ a standardization process that standardizes the computation of eigenvalues for each dataset sample, thereby converting the data into a normal distribution with a mean of 0 and a standard deviation of 1. Standardization is a critical process that facilitates the model's more accurate learning of data features, thereby enhancing the efficacy of model training. This is achieved by computing the mean and standard deviation for each feature. The standardization formula is expressed in the following equation:

$$X_{standard} = \frac{X - \bar{X}}{\sigma(X)} \tag{2}$$

Where $\bar{X}$ is the mean of X and $\sigma(X)$ is the standard deviation of X. The standardization of each feature is conducted utilizing Equation (2) . This standardized feature matrix not only accelerates the model's processing speed but also guarantees the model's convergence in an appropriate manner.

The standardized feature matrix optimizes the efficiency of matrix operations. When computing the covariance matrix $C = X^T X$, the unstandardized feature matrix may lead to numerical instability due to large differences in feature value ranges. In contrast, the standardized feature matrix X_{standard} ensures that the computation of the covariance matrix $C_{\text{standard}} = X_{\text{standard}}^T X_{\text{standard}}$ is more stable, thereby improving the efficiency of matrix operations.

3.3 GARF module

Feature dimensionality reduction entails the selection of an optimal subset of features from the original feature set. Its key objective is to eliminate extraneous and superfluous features, thereby retaining only the pertinent features [31]. First, a GARF feature dimensionality reduction method is proposed to effectively reduces the dimensionality of the feature space, expedites the learning algorithm's execution, and enhances the prediction accuracy of the learning algorithm. Genetic Algorithms (GA) [32] offer the advantage of effectively addressing non-linear distributions and performing feature dimensionality reduction through the evaluation of the relevance and importance of each feature, interpretability, and robustness in the presence of noise and outliers. Consequently, GA are utilized as the search strategy, with the Random Forest (RF) algorithm serving as the classifier in the feature dimensionality reduction stage. This collaboration establishes the GARF feature dimensionality reduction algorithm to identify the most effective feature subset and reduce the dimensionality of the original dataset's features.

Fitness Function: The computation of fitness scores is essential in evaluating the capabilities of GA. The fitness function is derived from the classification accuracy of the RF and the quantity of selected features. The chromosome exhibiting the highest accuracy and the least number of features attains the maximum fitness value. The adaptation score for each chromosome within the population is calculated utilizing the formula provided in Equation (3).

$$fitness(X) = \alpha \times Accuracy(X) + (1 + \alpha) \times \left(\sum_{i=1}^{n} x_i\right)^{-1} \tag{3}$$

where X is a binary vector representing the current subset of features, and α is a predefined weight adjusted according to the user's requirements to indicate the importance of RF accuracy relative to the length of the subset. Accuracy(X) serves as an evaluation metric for the RF's performance based on the feature subset X, crucial for assessing the classifier's efficacy, and the summation $\sum_{i=1}^{n} x_i$ provides the total count of features within the given subset.

RF is an integrated algorithm consisting of a Decision Tree (DT), RF is composed of many DTs, and there is no association between different DTs. When the classification task is performed, new input samples come in, and each DT in the forest judges and classifies them separately, each DT will get a classification result of its own, and whichever of the DT's classification results has the most classifications, the RF will take that result as the final result.

The algorithm proposed in this paper for GARF-based feature dimensionality reduction is shown below.

The initial step involves the random generation of an initial population, comprising N chromosomes, denoted as P. To create a new population, P_{new} based on the preceding generation P, genetic algorithm operations are implemented. Initially, two chromosomes, labeled X_{P_1} and X_{P_2}, are randomly selected from P for crossover, employing the roulette selection technique. $X_0 = Crossover(X_{P_1}, X_{P_2})$

Subsequently, a mutation operation is conducted on the offspring in accordance with the predefined mutation rate. The resulting mutated offspring, identified as X_{O_1} and X_{O_2}, are subsequently incorporated into the newly formed population P_{new}.

Algorithm 1 GARF-based feature dimensionality reduction

Input: Original data set
Output: Subset of features
$P = Generate\_Initial\_Population(N)$
$i = 1$
while $Chromosomes(P)$ and $i \leq G$ **do**
    $P_{new} = empty$
    repeat(N/2 times)
        Selection (P, X_{p1}, X_{p2})
        $X_0 = Crossover(X_{p1}, X_{p2})$
        Mutation (X_{o1}, X_{o2})
        Add to P_{new} $(P_{new}, X_{o1}, X_{o2})$
    until completed
    Compute Fitness with RF(P_{new})
    Save Best Chromosome(P_{new})
    $P = P_{new}$
    $i = i + 1$
end while

Thirdly, the fitness of each chromosome within the newly formed population P_{new} is computed, considering the accuracy and the quantity of features of the RF algorithm. The chromosome exhibiting the highest fitness throughout all iterations is preserved in memory. This sequence of operations is iterated until all chromosomes within P exhibit similarity or the predetermined number of iterations G is attained.

3.4 ANN module

Graph Convolutional Networks (GCN), as an extension of CNNs from regular lattices to irregular graphs, have been widely studied for graph data representation and learning [33]. Whereas GCN computation is based on graph-structured data, graph-structured data consists of two parts, the feature matrix, and the adjacency matrix, so it is necessary to transform the dataset into graph-structured data. Literature [34] transforms datasets into graph-structured data through K-Nearest Neighbor (KNN), but constructing graph-structured data using the KNN method suffers from long computationally time-consuming computations and difficulty in dealing with large datasets of network traffic. Approximate Nearest Neighbor [35] (ANN) has the advantages of faster computation and fewer computational resources compared to KNN. Therefore, this module adopts the ANN method to convert marine meteorological sensor network traffic data into graph-structured data, which reduces the construction time and saves computational resources. The ANN method considers each piece of data in the network traffic dataset as a node and calculates the Euclidean distance between the nodes, and the Euclidean distance calculation process is:

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{4}$$

where x and y are the feature vectors of the two nodes and n is the number of features. The ANN-based algorithm for graph structure data construction is shown in Algorithm 2:

Algorithm 2 ANN-based algorithm for graph-structured data construction

Require: dataset after feature dimensionality reduction
Ensure: graph-structured data
Read the dataset
Select the node identification column
Extraction characteristics
Select the value of K
Initializing ANN models using Euclidean distance metrics
Adding data points to the model
Constructing ANN Indexes Using K Trees
Create an empty list to store side connections
for each node **do**
    Retrieve K's approximate nearest neighbors
    Excluding self-connections
    Add the found edge to the list
end for
Returns the edge index
Convert to an adjacency matrix

By cyclically traversing the nodes in the dataset, obtaining the indexes of similar neighboring nodes of node i, obtaining the node feature matrix, adding the information of the edges to the edge list, obtaining the edge connectivity data, and finally converting it to the adjacency matrix, i.e., transforming the network traffic data into graph-structured data for the classification prediction of the graph-structured data by the GCN classifiers.

3.5 GCN module

GCN exhibits high computational efficiency with graph-structured data, capable of processing it directly. It utilizes the feature matrix X and the adjacency matrix A as its initial input to extract features from graph-structured data. The operational principle of GCN is depicted in Figure 3, illustrating how nodes aggregate information from adjacent nodes and subsequently update their features.

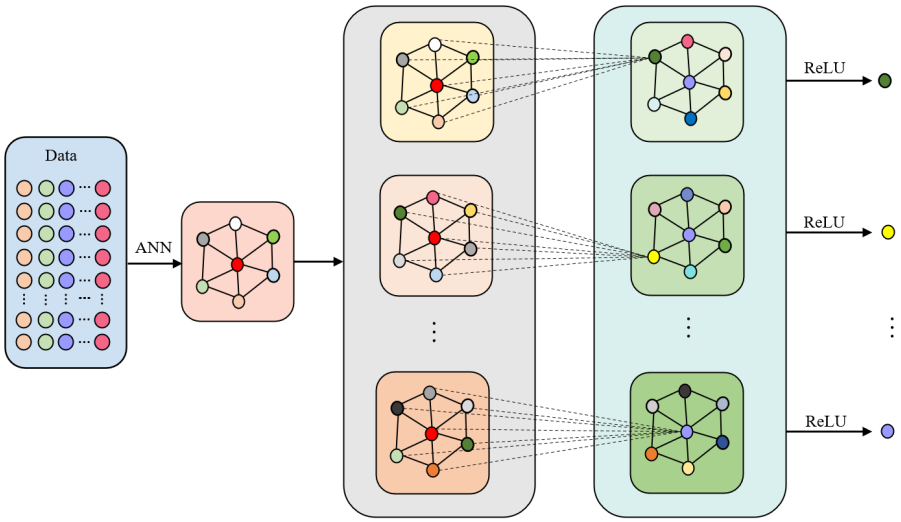


Figure 3: Working principle of GCN

Given a graph with N nodes, each possessing d-dimensional features, the graph's structure and the features of its nodes are represented by $A \in R^{N\times N}$ and $X \in R^{N\times d}$, respectively. The GCN unit operates with X and A as inputs, executing a graph convolution operation utilizing a localized first-order approximation. The GCN architecture comprises several neural network layers, which propagate

information across layers in a defined manner:

$$Z = \sigma(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}XW) \tag{5}$$

Where Z represents the node representation matrix obtained after the graph convolution operation and σ is a nonlinear activation function. The term $\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}$ represents an approximate graph convolution filter. $\hat{A}$ is the self-loop adjacency matrix, and $\hat{A} = A + I$, $\hat{D}$ refers to the degree matrix, while W is the weight matrix associated with inputs to the hidden layer. GCN facilitates the learning of hidden representations of nodes by encoding the graph structure and node features, thereby enabling the classification of network traffic and achieving the objective of NID.

4 Experimental results and analysis

4.1 NSL-KDD dataset

The NSL-KDD dataset [36], recognized as a benchmark in the field of NID, has been extensively utilized in recent years for research in this domain. Therefore, it has been selected as the primary dataset for our experimental analysis. The NSL-KDD dataset comprises 148,517 instances of network traffic data, with each entry featuring 43 distinct attributes. Of these attributes, 41 pertain to the traffic input itself, while the remaining two represent labels (indicating whether the traffic is normal or malicious) and scores (reflecting the severity of the traffic input).The traffic types are classified into five categories: Normal, DoS (Denial of Service), Probe, U2R (User to Root), and R2L (Remote to Local). Within these categories, DoS encompasses 11 attack types, Probe includes 6, U2R comprises 7, and R2L consists of 15 attack types.The dataset is subdivided into two subsets, KDDTrain+ and KDDTest+, and the categorical distribution of the samples within these subsets is detailed in Table 1. Considering that the objective of MMSN-IDS is to identify attack traffic within MMSN traffic, the intrusion detection task can be conceptualized as a binary classification problem. In this scenario, "0" is utilized to denote normal traffic, while "1" signifies abnormal traffic.

Table 1: Distribution of sample categories in the NSL-KDD dataset

Category	Data	Train	Test	Total
Attack	DoS	45,927	7,458	53,385
	Probe	11,656	2,421	14,077
	U2R	995	2,754	3,749
	R2L	52	200	252
Normal		67,343	9,711	77,054
Total		125,973	22,544	148,517

4.2 Evaluation criteria

In order to evaluate the efficacy of the proposed model, we have employed Accuracy, Precision, Recall, and F1-score as our primary evaluation metrics in the experimental phase [37]. Accuracy, being a standard metric in classification model evaluation, quantifies the proportion of samples that are accurately classified in relation to the overall sample set. Precision quantifies the proportion of actual positive labels within the subset of samples that the model predicts positively, thereby assessing the model's proficiency in accurately forecasting positive instances. Conversely, Recall represents the proportion of all positive samples that are accurately predicted by the model, serving as an indicator of its effectiveness in encompassing all positive examples. The F1-score, derived from the harmonic mean of Precision and Recall, is a critical metric for evaluating the performance of classification models. It

provides a nuanced measure of the model's balance between precision and recall, thereby offering a comprehensive evaluation of its classification efficacy.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$F1 - \text{score} = \frac{2 \times Recall \times Precision}{Recall + Precision} \tag{9}$$

Where: TP denotes the total number of positive cases accurately predicted as positive, FP signifies the total number of negative cases inaccurately predicted as positive, FN indicates the total number of positive cases incorrectly predicted as negative, and TN represents the total number of negative cases correctly predicted as negative.

4.3 Analysis of experimental results

4.3.1 Feature dimensionality reduction process based on GARF

The GARF-AGCN model is simulated and experimented with using the Pytorch deep learning framework. Firstly, data preprocessing of the NSL-KDD dataset is required to convert character-based features 'protocol_type', 'service', and 'flag' in the dataset to numerical features using one-hot coding. After one hot coding, the feature dimension of the dataset is increased from 41 dimensions to 122 dimensions and then the dataset is normalized using the normalization method. The GARF algorithm is used for feature dimensionality reduction of the data preprocessed dataset, the ANN algorithm is used to make the data transformed into graph-structured data, and finally, the classification prediction is performed using the GCN classification module to get the NID results.

The experimental parameters are set as follows: the initial population size of GARF is set to 50, the number of iterations is set to 50, the crossover rate is set to 0.8, the mutation rate is set to 0.1, the random seed size of RF is set to 42, the K value of ANN is set to 7, the number of training rounds of GCN is set to 300, the learning rate is set to 0.05, the size of the hidden layer is set to 14, and the discarding rate is set to 0.375. In addition, in the MMSN The most important goal in intrusion detection is to distinguish between normal and malicious traffic in MMSN, therefore, in this paper, binary classification labels are used as model labels for training and testing.

Firstly, the training set is subjected to feature dimensionality reduction using the GARF algorithm, and the results are stabilized after 50 iterations to get the subset of features with different feature dimensions, and the experimental results of the classifier's accuracy under the subset of features with different feature dimensions in the validation set are shown in Figure 4.

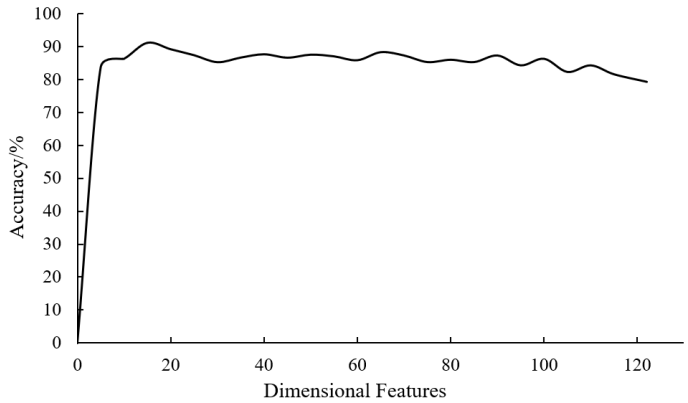


Figure 4: Effect of different feature dimensions on model performance

As can be seen in Figure 4, when the feature dimension is 122 full features, the accuracy of the classifier is about 80%. As the feature dimension decreases, the accuracy of the classifier gradually increases, and when the feature dimension is reduced to 16 features, the classifier reaches a maximum accuracy of about 90%, and continuing to reduce the feature dimension, the accuracy of the classifier will drop dramatically. Therefore, the feature subset F16 with a feature dimension of 16 is selected as the best feature subset of the original dataset to achieve the purpose of feature dimensionality reduction, reduce the model training time, and improve the model performance. The 16 features in the best feature subset F16 are shown in Table 2.

Table 2: Feature names of the best feature subset F16

ID	Feature Name	ID	Feature Name	ID	Feature Name	ID	Feature Name
1	protocol_type_udp	5	service_nnsp	9	flag_RSTR	13	hot
2	service_domain_u	6	service_pm_dump	10	flag_S0	14	root_shell
3	service_http,	7	service_supdup	11	flag_SF	15	num_shells
4	service_kshell	8	flag_REJ	12	wrong_fragment	16	num_access_files

4.3.2 Construction process of graph-structured data based on ANN

Secondly, the ANN algorithm is used to make the data transformed into graph structured data, the key parameter of ANN algorithm is the K value, different K values control the number of nearest neighbor nodes for each node, When K=7, the ANN algorithm finds for each node its 6 nearest neighbor nodes as its neighbor nodes. Different numbers of neighbor nodes will affect the model classification performance and training duration. The effect of the K value of the ANN algorithm on the classification model is shown in Table 3.

Table 3: The effect of different K values on the classification performance of the model

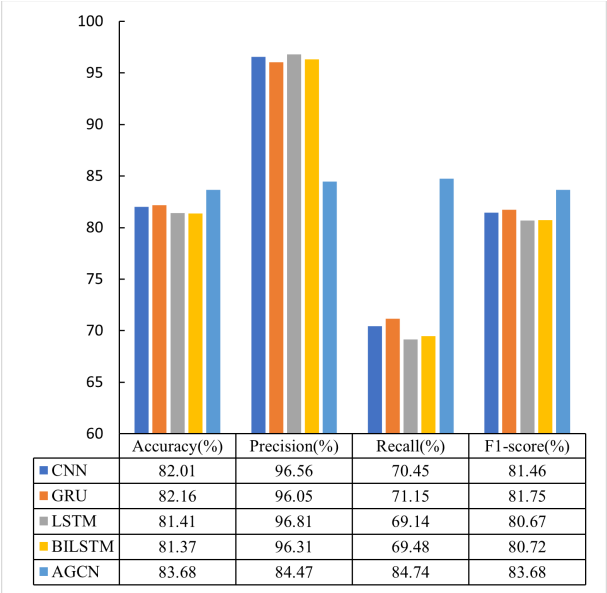
K value	Accuracy(%)	Precision(%)	Recall(%)	F1score(%)	Time(s)
K=2	91.86	91.63	92.06	91.78	9.83
K=3	91.83	91.77	92.41	91.79	10.82
K=4	91.67	91.59	92.22	91.63	11.72
K=5	91.78	91.69	92.31	91.74	12.72
K=6	91.87	91.81	92.43	91.84	13.59
K=7	92.02	91.88	92.49	91.97	14.75
K=8	91.75	91.65	92.27	91.71	15.61
K=9	91.73	91.63	92.25	91.69	16.58
K=10	91.75	91.65	92.27	91.71	17.58

From Table 3, with the increase of the K value, the number of each node with other neighboring nodes will increase, the classification performance of the classifier will gradually enhance, and the training time will also increase; when K=7, the classifier's reaches the highest accuracy rate of 92.02%. Therefore, K=7 is chosen as the optimal K value for the ANN algorithm to get the best approximation of the graph structure data of neighbor nodes.

4.3.3 Performance evaluation results of NID based on GARF-AGCN

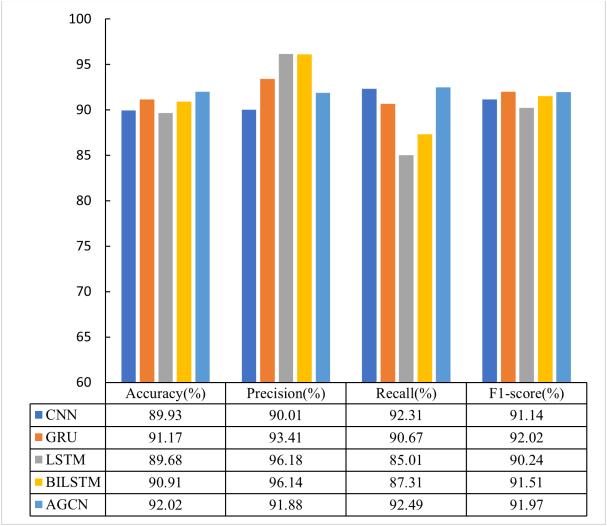
To verify the effectiveness of the GARF algorithm, the original dataset F122 and the feature subset F16 are utilized for the training and testing of classification models. Initially, commonly used DL intrusion detection classification models, such as CNN, GRU, LSTM, and BILSTM, are employed as comparative models against the AGCN model. The results are presented in Figure 5 and Figure 6.

From Figure 5, it can be seen that all five classification models have poor classification results on the original dataset F122, with lower accuracy, recall, and F1, and higher precision, which indicates that the classification models failed to learn effective features on the original dataset F122, and that the



	Accuracy(%)	Precision(%)	Recall(%)	F1-score(%)
CNN	82.01	96.56	70.45	81.46
GRU	82.16	96.05	71.15	81.75
LSTM	81.41	96.81	69.14	80.67
BILSTM	81.37	96.31	69.48	80.72
AGCN	83.68	84.47	84.74	83.68

Figure 5: Classification results of different DL models on the original dataset F122



	Accuracy(%)	Precision(%)	Recall(%)	F1-score(%)
CNN	89.93	90.01	92.31	91.14
GRU	91.17	93.41	90.67	92.02
LSTM	89.68	96.18	85.01	90.24
BILSTM	90.91	96.14	87.31	91.51
AGCN	92.02	91.88	92.49	91.97

Figure 6: Classification results of different DL models on the feature subset F16

prediction results of the classification models are biased towards the normal traffic, and the sensitivity for the detection of the attack traffic is relatively low.

As can be seen in Figure 6, the classification performance of the five classification models on the feature subset F16 is greatly improved, the average test Accuracy, Recall, and F1-score are all improved, and the average test Precision of the false high is alleviated, which indicates that the GARF feature dimensionality reduction can improve the sensitivity of the classification model to the attack traffic, which in turn improves the classification model's NID performance.

Comparing Figure 5 and Figure 6, it can be found that after GARF feature dimensionality reduction, the test accuracies of the five DL classification models, CNN, GRU, LSTM, BILSTM, and AGCN, are improved by 7.92%, 9.01%, 8.27%, 9.54%, and 8.32%, respectively. The AGCN classification model has the highest average test Accuracy, Recall, and F1-score of 92.02%, 91.63%, 91.88%, and 91.97%, respectively, which are higher than the other 4 DL classification models, indicating that AGCN model has a better ability to recognize the attack traffic and has a better classification performance.

4.3.4 Ablation experiments based on GARF-AGCN

To verify the effectiveness of the proposed GARF-AGCN model, ablation experiments are conducted on the NSL-KDD dataset to test the intrusion detection performance and intrusion detection efficiency with different conditions, and the results are shown in Table 4.

Table 4: The effect of GARF-AGCN ablation experiments different K values on the classification performance of the model

Different Conditions	Accuracy(%)	Precision(%)	Recall(%)	F1 score(%)	Time(s)
F122 K=2	83.27	84.95	84.71	83.27	12.21
F122 K=7	83.68	84.47	84.74	83.68	17.88
F16 K=2	91.86	91.63	92.06	91.78	9.83
F16 K=7	92.02	91.88	92.49	91.97	14.75

As shown in Table 4, a comparative analysis of two experimental configurations (F122 K=2 and F122 K=7) with preserved 122-dimensional features reveals that optimizing ANN parameters by adjusting the neighbor count K from 2 to 7 enhances model performance. Specifically, the accuracy and F1 scores improve from 83.27% and 83.27% to 83.68% and 83.17% respectively, indicating that K=7 facilitates the construction of optimal graph-structured datasets for feature extraction.

Furthermore, comparison between configurations F122 K=7 and F16 K=7 (maintaining identical ANN parameters) demonstrates that implementing GARF-based feature dimensionality reduction to obtain the optimal 16-dimensional feature subset yields substantial improvements. The accuracy and F1 scores escalate from 83.68% and 83.17% to 92.02% and 91.97% respectively, representing relative improvements of 8.34% and 8.80%. Concurrently, the training duration decreases from 17.88 s to 14.75 s. These results confirm that GARF effectively extracts critical features from network traffic data while optimizing computational efficiency.

This systematic evaluation demonstrates dual optimization benefits:

1. ANN parameter tuning ($K = 7$) enhances graph structure construction, and
2. GARF-driven dimensionality reduction (F16) achieves superior feature representation with reduced computational overhead.

4.3.5 Comparative results based on different graph data construction methods

To verify the effectiveness of the ANN algorithm, ANN and KNN are compared on the original dataset F122 and feature subset F16, and the node and edge information obtained from the two algorithms are constructed into graph structure data, and finally the constructed graph structure data are used for the training and testing of GCN model. The test results of GCN based on KNN and ANN algorithms and the construction time of graph structure data are shown in Table 5.

Table 5: The effect of Comparison experiments between ANN and KNN GARF-AGCN ablation experiments different K values on the classification performance of the model

Different Conditions	Accuracy(%)	Precision(%)	Recall(%)	F1score(%)	Time(s)
F122-KNN-GCN	83.71	84.55	84.79	83.71	284.49
F122-ANN-GCN	83.68	84.47	84.74	83.68	6.05
F16-KNN-GCN	91.83	91.76	92.39	91.79	3.61
F16-ANN-GCN	92.02	91.88	92.49	91.97	2.72

From Table 5, it is evident that the GCN models based on KNN and ANN algorithms exhibit superior classification performance on both the original dataset F122 and the feature subset F16. Notably, the classification accuracy and F1 score of GCN on the feature subset F16 exceed 91%, indicating that the graph structures constructed by both algorithms are of high quality and suitable for subsequent GCN training. Additionally, the graph structure construction times for KNN and ANN on the original dataset F122 are 284.49 seconds and 6.05 seconds, respectively, highlighting the inefficiency of KNN in

constructing graph structures when dealing with high-dimensional features, while ANN demonstrates relatively higher efficiency. Overall, KNN and ANN exhibit similar capabilities in constructing graph structure data, but ANN proves to be more efficient when handling high-dimensional feature datasets.

4.3.6 Comparative results of NID based on ML and DL methods

In order to validate the NID performance of the GARF-AGCN model, it is compared with the common ML-based and DL intrusion detection models on the original dataset F122 and the feature subset F16, and the results of the experimental comparison of the intrusion detection of different models are shown in Table 6.

Table 6: Effect of GARF-AGCN Ablation Experiments with Different K Values on Classification Performance

Models	F122				F16			
	Acc(%)	Pre(%)	Rec(%)	F1(%)	Acc(%)	Pre(%)	Rec(%)	F1(%)
LR	76.29	79.06	78.07	76.24	84.67	85.47	85.74	84.67
RF	79.83	82.53	81.58	79.79	80.82	83.72	82.64	80.78
XGB	82.28	84.66	83.95	82.26	83.86	85.27	85.19	83.86
SVM	76.77	79.39	78.49	76.72	85.59	86.87	86.88	85.58
CNN	82.01	96.56	70.45	81.46	89.93	90.01	92.31	91.14
GRU	82.16	96.05	71.15	81.75	91.17	93.41	90.67	92.02
LSTM	81.41	96.81	69.14	80.67	89.68	96.18	85.01	90.24
BILSTM	81.37	96.31	69.48	80.72	90.91	96.14	87.31	91.51
GARF-AGCN	83.68	84.47	84.74	83.68	92.02	91.88	92.49	91.97

Table 6 shows that among ML models (LR, RF, XGB, SVM), XGB performs best with 82.28% accuracy and 82.26% F1 score. For DL models (CNN, GRU, LSTM, BILSTM, AGCN), AGCN leads with 83.68% in both metrics. This indicates that on high-dimensional network traffic data, both ML and DL models struggle to learn key features, resulting in subpar classification.

On the F16 feature subset, ML and DL intrusion detection models show improved performance. ML models (LR, RF, XGB, SVM) see accuracy and F1 score increases of 8.38%, 0.98%, 1.58%, 8.82% and 8.43%, 0.99%, 1.60%, 8.86% respectively. For DL models (CNN, GRU, LSTM, BILSTM, AGCN), the improvements are 7.92%, 9.01%, 8.27%, 9.53%, 8.33% in accuracy and 9.68%, 10.27%, 9.56%, 10.79%, 8.28% in F1 score. This indicates that after GARF dimensionality reduction, both types of models better capture effective features, enhancing classification. The top ML model is SVM with 85.59% accuracy and 85.58% F1 score, while the best DL model, AGCN, achieves 92.02% accuracy and 91.97% F1 score, showing DL models' superior overall performance. Among all, the AGCN model excels in NID.

4.3.7 Comparative results of training durations for NID based on ML and DL

The length of time to train the intrusion detection model to a stable state is also an important indicator of intrusion detection in marine networks, which not only directly affects the intrusion response time, but also reflects the computational overhead of NID, which is very important for MMSN with limited computational power. To verify the NID efficiency of the GARF-AGCN model, the test set of NSL-KDD is used to test the computational efficiency of ML models such as LR, SVM, XGB, RF, and DL models such as CNN, GRU, LSTM, BILSTM, AGCN, etc., where the training rounds are set to 300 rounds. The experimental comparison results are shown in Figure 7 and Figure 8.

Figure 7 and Figure 8. show that DL-based intrusion detection models have longer training times than ML-based ones, but both see reduced times after GARF dimensionality reduction. On the F122 dataset, LR (ML) trains fastest at 2.08 s, while SVM (ML) is slowest at 398.98 s; for DL, BILSTM is slowest at 422.46 s, and AGCN is quickest at 40.68 s.

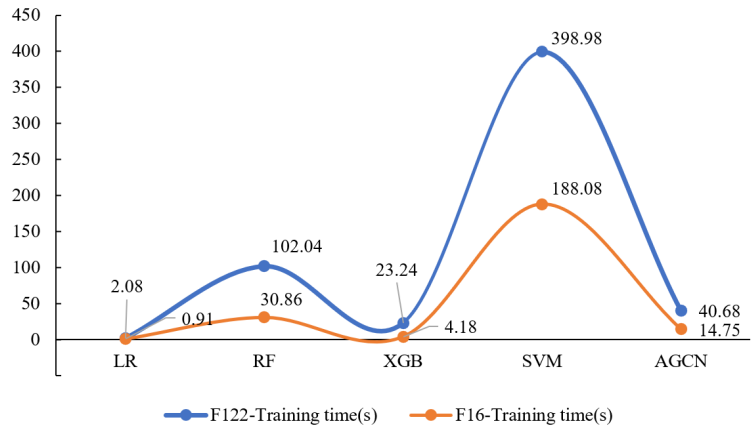


Figure 7: Training duration of different ML classification models on F122 and F16 datasets

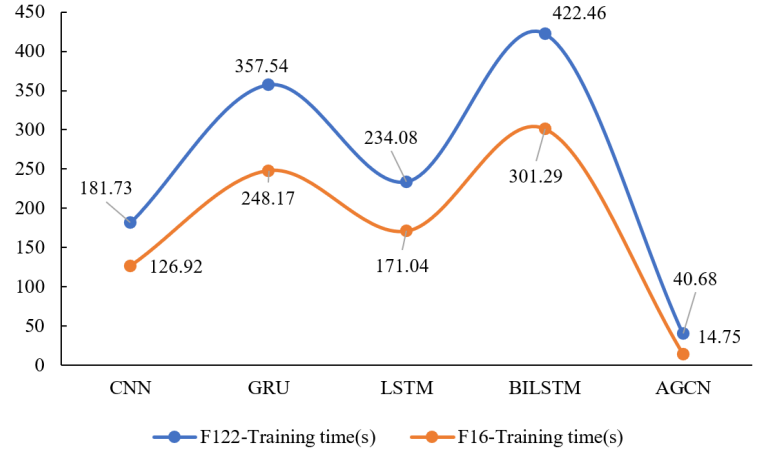


Figure 8: Training duration of different DL classification models on F122 and F16 datasets

On the F16 feature subset, LR (ML) and SVM (ML) have their training duration shortened to 0.91 s and 188.08 s, respectively, whereas BILSTM (DL) and AGCN (DL) take 301.29 s and 14.75 s. Despite LR's fast training, its accuracy and F1 score are below 85%, making it ineffective for MMSN. BILSTM and GRU (DL) perform better, with BILSTM scoring 90.91% and 91.51% and GRU 91.17% and 92.02% in accuracy and F1, respectively. However, their long training times and difficulty in convergence make them unsuitable for such networks.

The GARF-AGCN model has a training time of 14.75 s, outperforming both DL and ML models, including RF and SVM. With an accuracy of 92.02% and F1 score of 91.97%, it demonstrates superior detection performance and efficiency. Thus, the GARF-AGCN model is well-suited for detecting and protecting against malicious attacks in MMSN.

4.3.8 Comparative results of different NID models on the NSL-KDD Dataset

he intrusion detection performance of some existing studies and the GARF-AGCN model proposed in this paper are compared and the results are shown in Table 7.

Table 7: Classification performance of different methods on NSL-KDD dataset

Model	Accuracy(%)	F1score(%)
CNN-CGAN[26]	79.70	79.60
AE-BILSTM[27]	89.00	91.00
CNN-BILSTM[28]	90.73	89.65
GARF-AGCN	92.02	91.97

Table 7 shows that the GARF-AGCN model outperforms other models in intrusion detection. Literature [26] used CGAN for data equalization and CNN for classification, achieving only 79.70% accuracy and 79.60% F1 score. Literature [27] improved performance to 89.00% accuracy and 91.00% F1 score by using AE for feature selection and BILSTM for classification. Literature [28] combined CNN, Attention, and BILSTM, reaching 90.73% accuracy and 89.65% F1 score, which is the best among the three but still below the GARF-AGCN's 92.02% accuracy and 91.97% F1 score, indicating its superior performance.

5 Conclusion

In this paper, a GARF-AGCN-based intrusion detection model for MMSN is designed. The feature dimensionality reduction module GARF is utilized to select the best feature subset F16, the ANN module is utilized to transform the marine network traffic data into graph-structured data, and the GCN module is utilized to classify and predict the graph-structured data. Simulation comparison experiments were conducted on the NSL-KDD dataset, and the intrusion detection accuracy and F1 score of the GARF-AGCN model reached 92.02% and 91.97%, respectively, with a training duration of 14.75 s for 300 rounds.

The experimental results show that the NID performance and intrusion detection efficiency of the GARF-AGCN model are higher than that of the existing ML intrusion detection models such as LR, SVM, and DL intrusion detection models such as CNN, LSTM, etc., which verifies the validity of the intrusion detection model of marine meteorological sensor network based on the GARF-AGCN=, and illustrates the importance and application value of the GARF-AGCN intrusion detection model in the marine meteorological sensor network field has important theoretical significance and application value. However, marine meteorological devices are deployed in marine environments and face harsh climatic conditions such as salt water corrosion, strong ultraviolet rays, and strong winds and waves, so application tests in real marine meteorological sensor network environments are still needed to verify the actual performance of the models.

Although the GARF-AGCN model has demonstrated high detection accuracy and efficiency in experiments, several limitations remain. Firstly, the model's feature extraction capability may be constrained by the complexity and diversity of the data, particularly when dealing with high-dimensional, nonlinear, or dynamically changing marine meteorological data. The comprehensiveness and accuracy of feature extraction need further improvement. Secondly, the model's performance in real-world marine environments has not yet been validated. Harsh climatic conditions such as salt spray corrosion, strong ultraviolet radiation, and severe wind and waves, which marine meteorological equipment often faces, may impact the model's stability and reliability.

To address these limitations, future research could focus on the following improvements:

1. **Incorporation of Attention Mechanisms:** By integrating self-attention or channel attention mechanisms, the model can better capture key features in the data, thereby enhancing the precision and robustness of feature extraction.

2. **Multimodal Data Fusion:** Combining multi-source data (e.g., temperature, humidity, wind speed) from marine meteorological sensor networks to design a multimodal fusion model can provide a more comprehensive description of network states and improve detection performance.

3. **Real-World Environment Testing:** Conducting application tests in actual marine meteorological sensor network environments to verify the model's stability and reliability under adverse climatic conditions.

The intrusion detection method based on GARF-AGCN proposed in this paper alleviates the computational limitations of MMSN devices by significantly reducing feature dimensionality through the GARF feature reduction algorithm, thereby decreasing computational load. Additionally, by converting network traffic data into graph-structured data using the ANN algorithm and leveraging GCN for feature extraction, the model enhances training efficiency and improves the real-time performance of

MMSN-IDS. This provides a novel solution for securing resource-constrained marine IoT systems. Furthermore, the feature reduction and graph-structured data processing methods employed in this model can serve as a reference for other resource-constrained IoT systems, promoting further advancements in IoT security technologies.

Funding

This work is supported by the National Natural Science Foundation of China (No. 62171228)

Author contributions

The authors contributed equally to this work.

Conflict of interest

The authors declare no conflict of interest.

References

[1] Ashraf, I.; Khan, M.A.; Khan, S.U.; et al. (2022). A survey on cyber security threats in IoT-enabled maritime industry, *IEEE Transactions on Intelligent Transportation Systems*, 24(2), 2677–2690, 2022.

[2] Su, X.; Meng, L.; Huang, J. (2020). Intelligent maritime networking with edge services and computing capability, *IEEE Transactions on Vehicular Technology*, 69(11), 13606–13620, 2020.

[3] Hou, T.; Xing, H.; Liang, X.; et al. (2024). Traffic flow prediction method for the intelligent marine meteorological sensor network, *Ocean Engineering*, 309, 118296, 2024.

[4] Dorle, P.K.; Ranade, S.D. (2021). Hydrology and Hydrological Information Systems (HIS) in India, *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, 9(2), 2021.

[5] Su, X.; Zhang, G.; Zhang, M.; Ye, B.; Xing, H. (2022). Intrusion Detection for Marine Meteorological Sensor Network. In *2022 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)* (pp. 190-196). IEEE, 2022.

[6] Gyamfi, E.; Khan, M.A.; Khan, S.U.; et al. (2022). An adaptive network security system for IoT-enabled maritime transportation, *IEEE Transactions on Intelligent Transportation Systems*, 24(2), 2538–2547, 2022.

[7] Jahanbakht, M.; Xiang, W.; Hanzo, L.; et al. (2021). Internet of underwater things and big marine data analytics—a comprehensive survey, *IEEE Communications Surveys & Tutorials*, 23(2), 904–956, 2021.

[8] Tiwari, D.; Bhati, B.S.; Nagpal, B.; et al. (2021). An enhanced intelligent model: To protect marine IoT sensor environment using ensemble machine learning approach, *Marine Engineering*, 242, 110180, 2021.

[9] Hou, T.; Xing, H.; Liang, X.; et al. (2023). A marine hydrographic station networks intrusion detection method based on LCVAE and CNN-BiLSTM, *Journal of Marine Science and Engineering*, 11(1), 221, 2023.

[10] Liu, W.; Zhang, G.; Su, X.; et al. (2022). Intrusion detection for maritime transportation systems with batch federated aggregation, *IEEE Transactions on Intelligent Transportation Systems*, 24(2), 2503–2514, 2022.

[11] Sun, C.C.; Cardenas, D.J.S.; Hahn, A.; et al. (2020). Intrusion detection for cybersecurity of smart meters, *IEEE Transactions on Smart Grid*, 12(1), 612–622, 2020.

[12] Kumar, P.; Kumar, S.; Kumar, A.; et al. (2021). DLTIF: Deep learning-driven cyber threat intelligence modeling and identification framework in IoT-enabled maritime transportation systems, *IEEE Transactions on Intelligent Transportation Systems*, 24(2), 2472–2481, 2021.

[13] Singh, A.P.; Singh, M.D. (2014). Analysis of host-based and network-based intrusion detection system, *International Journal of Computer Network and Information Security*, 6(8), 41–47, 2014.

[14] Martins, I.; Resende, J.S.; Sousa, P.R.; et al. (2022). Host-based IDS: A review and open issues of an anomaly detection system in IoT, *Future Generation Computer Systems*, 133, 95–113, 2022.

[15] Kim, T.; Pak, W. (2023). Integrated Feature-Based Network Intrusion Detection System Using Incremental Feature Generation, *Electronics*, 12(7), 1657, 2023.

[16] Ahmad, R.; Alsmadi, I.; Alhamdani, W.; et al. (2023). Zero-day attack detection: a systematic literature review, *Artificial Intelligence Review*, 56(10), 10733–10811, 2023.

[17] Oliveira, N.; Praça, I.; Maia, E.; et al. (2021). Intelligent cyber attack detection and classification for network-based intrusion detection systems, *Applied Sciences*, 11(4), 1674, 2021.

[18] Ahmad, Z.; Shahid Khan, A.; Wai Shiang, C.; et al. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches, *Transactions on Emerging Telecommunications Technologies*, 32(1), e4150, 2021.

[19] Saranya, T.; Sridevi, S.; Deisy, C.; et al. (2020). Performance analysis of machine learning algorithms in intrusion detection system: A review, *Procedia Computer Science*, 171, 1251–1260, 2020.

[20] Vijayanand, R.; Devaraj, D.; Kannapiran, B. (2018). Intrusion detection system for wireless mesh network using multiple support vector machine classifiers with genetic-algorithm-based feature selection, *Computers & Security*, 77, 304–314, 2018.

[21] Disha, R.A.; Waheed, S. (2022). Performance analysis of machine learning models for intrusion detection system using Gini impurity-based weighted random forest (GIWRF) feature selection technique, *Cybersecurity*, 5(1), 1, 2022.

[22] Gu, J.; Lu, S. (2021). An effective intrusion detection approach using SVM with naïve Bayes feature embedding, *Computers & Security*, 103, 102158, 2021.

[23] Ashiku, L.; Dagli, C. (2021). Network intrusion detection system using deep learning, *Procedia Computer Science*, 185, 239–247, 2021.

[24] Halbouni, A.; Gunawan, T.S.; Habaebi, M.H.; et al. (2022). CNN-LSTM: hybrid deep neural network for network intrusion detection system, *IEEE Access*, 10, 99837–99849, 2022.

[25] Chauhan, S.; Mahmoud, L.; Gangopadhyay, S.; et al. (2022). A comparative study of LAD, CNN and DNN for detecting intrusions. In *International Conference on Intelligent Data Engineering and Automated Learning* (pp. 443-455). Cham: Springer International Publishing, 2022.

[26] Ji, Z.; Gao, X. (2022). CNN-CGAN: A New Approach for Intrusion Detection Based on Generative Adversarial Networks. In *International Conference on Emerging Networking Architecture and Technologies* (pp. 324-335). Singapore: Springer Nature Singapore, 2022.

[27] Mushtaq, E.; Zameer, A.; Umer, M.; et al. (2022). A two-stage intrusion detection system with auto-encoder and LSTMs, *Applied Soft Computing*, 121, 108768, 2022.

[28] Fu, Y.; Du, Y.; Cao, Z.; et al. (2022). A deep learning model for network intrusion detection with imbalanced data, *Electronics*, 11(6), 898, 2022.

[29] Su, X.; Zhang, G.; Xing, H.; et al. (2023). Research on Intrusion Detection in Marine Meteorological Sensor Network Based on Balanced Generative Adversarial Networks [in Chinese], *Journal of Communications*, 44(4), 124–136, 2023.

[30] Su, X.; Tian, T.; Gong, Z.; et al. (2023). Research on Intrusion Detection Methods in Marine Meteorological Sensor Networks Based on Abnormal Behavior [in Chinese], *Journal of Communications*, 44(7), 86–99, 2023.

[31] Abdulhammed, R.; Musafer, H.; Alessa, A.; et al. (2019). Features dimensionality reduction approaches for machine learning based network intrusion detection, *Electronics*, 8(3), 322, 2019.

[32] Almomani, O. (2020). A feature selection model for network intrusion detection system based on PSO, GWO, FFA and GA algorithms, *Symmetry*, 12(6), 1046, 2020.

[33] Deng, X.; Zhu, J.; Pei, X.; et al. (2022). Flow topology-based graph convolutional network for intrusion detection in label-limited IoT networks, *IEEE Transactions on Network and Service Management*, 20(1), 684–696, 2022.

[34] Sun, B.; Yang, W.; Yan, M.; Wu, D.; Zhu, Y.; Bai, Z. (2020). An encrypted traffic classification method combining graph convolutional network and autoencoder. In *2020 IEEE 39th International Performance Computing and Communications Conference (IPCCC)* (pp. 1-8). IEEE, 2020.

[35] Wang, M.; Xu, X.; Yue, Q.; et al. (2021). A comprehensive survey and experimental comparison of graph-based approximate nearest neighbor search, *arXiv preprint arXiv:2101.12631*, 2021.

[36] Ravipati, R.D.; Abualkibash, M. (2019). Intrusion detection system classification using different machine learning algorithms on KDD-99 and NSL-KDD datasets-a review paper, *International Journal of Computer Science & Information Technology (IJCSIT)*, 2019.

[37] Salih, A.A.; Abdulazeez, A.M. (2021). Evaluation of classification algorithms for intrusion detection system: A review, *Journal of Soft Computing and Data Mining*, 2(1), 31–40, 2021.

This journal is a member of, and subscribes to the principles of,
the Committee on Publication Ethics (COPE).
https://publicationethics.org/members/international-journal-computers-communications-and-control

Cite this paper as:

Hongyan Xing, Zhiwei Ni, Wei Xu (2025). Intrusion Detection in Marine Networks Based on Feature Dimension Reduction and Graph Convolution, *International Journal of Computers Communications & Control*, 20(6), 6832, 2025.
https://doi.org/10.15837/ijccc.2025.6.6832