Deep Autoencoder and Deep Q-Network-Based Feature Learning with Multi-Head Residual Attention DNN for Cyberattack Classification

Dharani Kanta Roy, Hemanta Kumar Kalita
Department of Computer Science and Engineering,
Central Institute of Technology Kokrajhar, Assam, India

***Abstract*—As society's digital infrastructure grows, so do the threats. We cannot deny internet access, so we need to safeguard ourselves against cyber threats. Every day, new threats emerge. The most conventional systems classify cyberattacks in a closed set; that is, training and testing use the same set of attack behaviors. In the real world, we cannot train our system on all attack patterns because attack behavior evolves daily. To address this issue, this study proposes open-set cyberattack classification, in which we train our system on a limited set of labeled attack types and, during testing, use a larger unlabeled data sample that includes both known and unknown attack patterns. Our framework integrates a Deep Autoencoder (DAE)-based latent representation, an Explainable Deep Q-Network (Explainable DQN) for feature selection, and a Multi-Head Residual Attention Deep Neural Network (MHRA-DNN) for final classification. The DAE transforms high-dimensional network traffic into a low-dimensional latent feature representation. This reduces redundancy while preserving important feature characteristics. The Explainable DQN selects the features that contribute most to the model's performance. It also improves explainability by highlighting the most influential features. The selected features are then processed by the MHRA-DNN. It captures multiple feature combinations and strengthens learning through attention mechanisms and residual connections. We evaluate the proposed framework on the benchmark cybersecurity datasets UNSW-NB15, APA-DDoS, and CICIDS2017. We set up an open-set environment that involves known and unknown attack classes. The proposed model performs strongly across accuracy, precision, recall, F1-score, specificity, MCC, and NPV. In addition, parameter tuning improves model stability, and SHAP-based analysis provides evidence of feature contributions to model performance. Overall, the proposed model offers an adaptive, explainable, and robust solution to cyberattack classification in an open-set environment.**

***Keywords*—Cyberattack classification; deep autoencoder; feature selection; deep q-network; multi-head attention; deep neural network; SHapley additive exPlanations**

I. Introduction

The rapid growth of network-based services, cloud platforms, Internet of Things devices, and distributed computing environments has significantly increased the complexity of modern cyber threats. Cyberattacks are no longer limited to predefined or previously observed patterns; instead, attackers continuously modify their strategies to evade detection and classification systems. As a result, accurate cyberattack classification has become essential for strengthening network security and enabling timely defensive actions. However, most traditional intrusion detection and classification models assume a closed set, where all attack categories are known during training. This assumption is not fully applicable to real-world cybersecurity environments, where previously unseen attack types may emerge during testing [1].

Conventional machine learning is used for cyberattack classification, but its performance depends on manually selected features and predefined attack labels [2]. Deep learning methods perform better at classifying attacks by automatically learning complex features from network traffic data [3]. High-dimensional traffic features may contain redundant or less informative components, increasing computational complexity. This may also reduce classification reliability. Moreover, many deep learning models offer limited interpretability, making it difficult to understand how features contribute to the classification. In security applications, this lack of transparency may reduce trust.

To address these limitations, this study proposes an open-set cyberattack classification framework. The proposed model is based on the Deep Autoencoder (DAE), the Explainable Deep Q-Network (Explainable DQN), and the Multi-Head Residual Attention Deep Neural Network (MHRA-DNN). The DAE learns compact latent representations from high-dimensional network traffic [4]. The reduced feature representation retains important information while reducing computational complexity. The Explainable DQN functions as a reinforcement learning agent for feature subset selection [5]. It systematically explores various subsets of latent features and evaluates each subset based on feedback from the classifier. Higher rewards are assigned to subsets that improve accuracy and F1-score, reduce false positives, and use fewer features. Through iterative reward-based updates, the DQN identifies the latent feature subset that yields superior classification performance. The selected optimized feature set is finally passed to the MHRA-DNN. The multi-head attention mechanism captures multiple dependencies among features [6], and residual connections [7] improve DNN training stability.

The proposed framework is designed for open-set classification environments. In this model, it is trained on a limited set of labeled attack classes and evaluated on anomalous traffic that may belong to known categories, unknown categories, or zero-day attack types. This setting is more practical for real-world environments than the closed-set setting, where every class is known at training time. We evaluate the framework using benchmark cybersecurity datasets UNSW-NB15, APA-DDoS, and CICIDS2017 [8], [9]. Parameter tuning improves

the model's stability and effectiveness. The SHapley Additive exPlanations (SHAP)-based analysis is used to interpret the contributions of selected features [10].

Overall, this work introduces an adaptive, explainable open-set cyberattack classification framework that combines latent representation learning, reward-based feature selection, attention-based classification, and feature-level interpretability.

A. Problem Statement

Most cyberattack classification models assume closed-set learning, allowing classification only of attack categories seen during training. This approach is inadequate for real-world cybersecurity, where attack types evolve and cannot all be anticipated. As a result, conventional classifiers may misclassify unknown attacks as known types, leading to unreliable results and poor security decisions.

Another challenge is the high dimensionality of network traffic data, which often includes noisy, redundant, or uninformative features. Using all features increases computational complexity and reduces learning efficiency. A compact feature representation is needed to retain critical attack information while minimizing complexity.

Even with a compact latent representation, not all features are equally valuable for cyberattack classification. Some provide strong discrimination, while others add little or introduce noise. Existing feature selection methods often rely on fixed criteria or heuristics and may not directly assess how feature subsets affect classification performance. An adaptive selection mechanism is needed to identify feature combinations that improve classification results.

Many deep learning-based cyberattack classification models lack interpretability and do not explain how selected features influence predictions. This lack of transparency is a significant concern in cybersecurity, where decisions must be understandable and reliable. An adaptive, explainable framework is needed to deliver compact representations, performance-driven feature selection, and reliable classification in open-set environments with both known and unknown attack patterns.

B. Contributions

The following are the novel contributions of this study:

- DAE to transform high-dimensional network traffic to a low-dimensional latent feature representation.
- An explainable DQN-based latent feature subset selection mechanism to select only the most effective latent features. The DQN agent explores latent feature subsets and evaluates their usefulness through a reinforcement-based reward mechanism.
- SHAP-based analysis to strengthen the interpretability of the selected latent features.
- A Multi-Head Residual Attention DNN is designed for final cyberattack classification. The attention mechanism enables the classifier to learn relationships among multiple features. The residual connections preserve useful feature information across deeper layers of the DNN. It reduces the possibility of feature degradation during training.
- Development of an adaptive cyberattack classification in an open-set environment integrating DAE based latent representation learning, Explainable DQN-based performance-oriented feature subset selection, SHAP-based feature interpretability, and Multi-Head Residual Attention-based DNN classification.

C. Organization of the Study

The study is organized as follows: Section II reviews related work and summarizes the research gaps; Section III presents the proposed methodology; Section IV presents the results and discussion; and Section V presents the conclusions.

II. Related Work

Researchers have widely investigated cyberattacks using machine learning and deep learning techniques. Traditional machine learning methods, such as Support Vector Machines (SVM), Decision Trees, Random Forest, K-Nearest Neighbors, and ensemble classifiers, are used for cyberattack classification due to their simplicity and lower computational cost [2]. However, these techniques often depend on manually selected features. They may also fail to capture the non-linear relationships present in modern cyberattacks. As cyberattacks become dynamic and evolving, manually designed feature representations may be insufficient for classification tasks.

Deep learning approaches have significantly improved cyberattack classification. They automatically learn complex feature representations from large-scale network traffic data. Models such as DNNs, Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) have been applied to cyberattack classification [3]. These models can learn high-level feature patterns and improve classification compared to traditional machine learning techniques. However, deep learning models require large amounts of labeled data and may incur high computational complexity when the input feature space is very large. Most importantly, many deep learning models assume a closed set, where all attack types are available during training. These limitations call into question their suitability in a practical cybersecurity environment where unknown attacks may emerge during testing [1].

High-dimensional network traffic may contain redundant, noisy, and less informative features. As a result, the efficiency and reliability of the classification model may decrease. Autoencoder-based representations have been widely used to solve this issue. [4] showed that a DAE learns a low-dimensional code from high-dimensional data that represents the same information. However, although DAE can reduce dimensionality, the complete latent feature space may still contain features with different levels of relevance to the classification task. Therefore, using all latent features generated by the DAE may not always yield the best classifier performance.

Feature selection plays an important role in cyberattack classification. Conventional methods often rely on statistical, filter-based, and heuristic optimization techniques. Although these methods can reduce the number of features, they still

fail to evaluate how different feature subsets will affect the classifier performance. Therefore, in a complex cybersecurity environment, the usefulness of a selected feature subset should be chosen not only based on statistical importance. Rather, it should be selected based on its actual contribution to the classification task.

A reinforcement learning agent can explore different feature-subset combinations and learn from reward feedback, making it a suitable mechanism for feature selection. The DQN framework demonstrated its capability to learn effective decision policies from high-dimensional inputs [5]. Recent studies demonstrate that deep reinforcement learning is suitable for feature selection and classification. ID-RDRL used a feature selection method for in-network intrusion detection. It showed that redundant features can be reduced while improving the classification performance [11]. [12] presented a DQN-based multi-agent feature selection approach for intrusion detection and network attack classification. These studies show how reinforcement learning techniques can support feature selection in cybersecurity. However, many existing approaches do not integrate DAE latent representation learning and reward-based latent feature subset selection.

Attention mechanisms have become an important technique in deep learning. They allow a model to focus on the most relevant parts of the input features. The Transformer architecture introduced multi-head attention mechanism. In this case, the multiple attention heads learn information from different representation subspaces [6]. This idea is highly relevant to cyberattack classification, as different latent features may represent distinct attack characteristics. A single attention mechanism may not capture all relevant feature dependencies.

Residual learning has also been widely used to improve the training of deeper layers in deep neural networks. He et al. [7] introduced residual connections to ease the optimization of deep networks. It also reduces degradation problems during model training. In cyberattack classification, residual connections can preserve important feature information across deeper layers and improve training stability. Therefore, combining multi-head attention and residual connections yields a strong classifier that can learn discriminative relationships from selected latent features.

Explainability is also an important requirement in cybersecurity. In classification tasks, model decisions should be understandable and reliable in critical applications. Many deep learning-based intrusion detection models are considered black-box models that do not clearly explain which features influence the final decision. This lack of transparency can reduce trust in security-sensitive applications. SHAP provides a framework for interpreting model predictions by assigning importance values to features based on their contribution [10]. Therefore, SHAP analysis helps explain how different features influence cyberattack classification decisions.

Recent research has advanced reinforcement learning, open-set recognition, attention-based learning, and unknown intrusion detection in network security. Wu et al. developed an active learning framework utilizing a DQN for zero-day attack detection, demonstrating that DQN supports adaptive decision-making in intrusion detection scenarios [13]. Fang and Xie presented a method for detecting unknown intrusion traffic based on unsupervised learning and open-set recognition, employing InfoGAN and OpenMax to identify unknown traffic while preserving classification performance on known traffic classes [14]. Cai et al. introduced HCRP-OSD, a fine-grained open-set intrusion detection approach that leverages hybrid convolution and adversarial reciprocal points learning to differentiate known attacks and infer unknown attacks [15]. DDP-DAR integrates a denoising diffusion probabilistic model with a dual-attention residual network to enhance feature representation and intrusion detection performance [16]. While these studies contribute to zero-day detection, open-set intrusion detection, attention-based classification, and unknown attack recognition, none simultaneously incorporate DAE-based latent representation learning, Explainable DQN-based latent feature subset selection, SHAP-based interpretation, and MHRA-DNN-based classification within a unified open-set anomalous traffic classification framework.

A. Research Gap Analysis

Machine learning and deep learning methods have shown promising results in cyberattack classification. Traditional machine learning models are simpler and less computationally intensive but rely on manually selected features and may not capture complex nonlinear relationships in network traffic. Deep learning models can automatically learn complex representations but typically require large labeled datasets and often assume a closed set. This limits their effectiveness when attacks go unseen during testing in real network environments.

A further research gap involves feature representation and selection. DAE models can reduce high-dimensional network traffic data to compact latent representations, but the resulting feature sets may not be optimal for classification. Not all latent features contribute equally to distinguishing cyberattacks. Existing feature selection methods often rely on fixed statistical measures, heuristic searches, or manually defined criteria, which may not dynamically assess how different latent feature subsets affect classifier performance. While reinforcement learning-based feature selection has shown promise in intrusion detection, integrating latent representation learning with reward-based selection of latent feature subsets remains underexplored.

A gap also exists in designing final classification models. Many deep learning classifiers use standard DNN, CNN, RNN, or LSTM architectures, which can learn useful patterns but may not fully capture the complex relationships among selected latent features. Different attack classes may rely on distinct feature combinations. An attention-guided classification model is needed to learn diverse, attack-specific patterns. Additionally, residual connections help preserve important feature information and enhance the stability of deeper models.

Finally, many deep learning-based cyberattack classification models function as black-box systems. While they may perform well, they often do not explain which features influence their predictions. In cybersecurity, this lack of interpretability is a significant concern. These gaps underscore the need for a unified open-set cyberattack classification framework that integrates compact latent representation learning, reward-based latent feature subset selection, attention-guided classification, and interpretable feature contribution analysis for anomalous network traffic.

III. Proposed Methodology

A. Problem Formulation

The methodology addresses adaptive cyberattack classification. The semi-supervised anomaly data setting is represented as:

$$D = \{D_a, D_u\} (\text{where } D_a \cap D_u = \emptyset)$$

where, D_a is the limited labeled anomaly subset for training, while D_u is the unlabeled anomaly subset for testing. D_u includes samples from both known and unexplained anomaly classes. The classification objective can be represented as: $f : x^{(i)} \rightarrow \hat{y}^{(i)}$ where $x^{(i)}$ denotes a network traffic sample and $\hat{y}^{(i)}$ is the predicted cyberattack class.

B. Architecture and Working of Adaptive Cyberattack Classification

The sequential classification workflow is designed to classify network traffic into distinct cyberattack categories. Network traffic may be high-dimensional and include redundant or less informative attributes. The first step is to generate a compact and informative representation of this traffic. To achieve this, a DAE is employed to learn latent representations. It compresses high-dimensional traffic features into a lower-dimensional latent space while preserving key attack-related patterns. This process reduces feature complexity and creates a more effective input for subsequent feature selection.

After learning the latent representation, the Explainable DQN selects the most informative subset of latent features. The DQN agent evaluates various feature combinations to identify those that enhance classification performance. This reward-guided selection removes redundant features and improves model interpretability by highlighting the importance of selected features.

The selected latent features are provided to the MHRA-DNN for final classification. The multi-head attention mechanism enables the classifier to focus on key relationships among features, while the residual connection retains useful information during deeper learning. The DNN's dense layers then learn nonlinear relationships from the refined features and predict the cyberattack category. The model's outputs are either a known cyberattack class or an unexplained anomaly. Fig. 1 illustrates the architecture and workflow of the proposed model.

1) Deep autoencoder-based latent representation learning: The first component of the proposed model is DAE-based latent representation learning. This approach reduces the complexity of high-dimensional network traffic features and produces compact representations for cyberattack classification. Network traffic data often contains numerous packet-level, flow-level, and statistical attributes, many of which may be redundant, noisy, or only weakly relevant. The DAE transforms the original feature space into a lower-dimensional latent space while preserving essential traffic patterns [4], [17]. Let the original anomalous traffic feature vector be represented as:

$$x = [x_1, x_2, x_3, \ldots, x_n] \quad (1)$$

where, x is the original input feature vector and x_i is the i^{th} input feature. The encoder of the DAE maps the input vector to a compact latent representation as:

$$h = f(Wx + b) \quad (2)$$

where, h is the learned latent representation, W is the encoder weight matrix, b is the bias vector, and $f(\cdot)$ is the nonlinear activation function. The decoder reconstructs the original input feature vector from the latent representation as:

$$\hat{x} = g(W'h + b') \quad (3)$$

where, $\hat{x}$ is the reconstructed feature vector, W' is the decoder weight matrix, b' is the decoder bias vector, and $g(\cdot)$ is the decoder activation function.

The DAE is trained by minimizing the reconstruction error between the original input x and the reconstructed output $\hat{x}$. The reconstruction loss is represented as:

$$L_{DAE} = |x - \hat{x}|^2 \quad (4)$$

where, L_{DAE} is the reconstruction loss. Minimizing this loss allows the autoencoder to learn a compressed representation that retains key structural information from anomalous traffic data [18].

After training, the encoder output is used as the latent feature representation:

$$Z = [z_1, z_2, z_3, \ldots, z_m] \quad (5)$$

where, Z is the latent feature vector and z_j is the j^{th} latent feature. This compact representation reduces feature dimensionality and improves input for the Explainable DQN-based latent feature subset selection stage. The DAE supports the model by generating meaningful latent features for adaptive cyberattack classification.

2) Explainable DQN-based latent feature subset selection: After the DAE generates a compact latent feature representation, the next objective is to identify the most informative subset of latent features for cyberattack classification. While the autoencoder reduces the dimensionality of the original features, the resulting latent representation may still contain redundant or minimally informative components. To address this, the proposed framework employs an Explainable DQN as a reinforcement-learning-based feature-subset selection agent. We achieve interpretability of the selected subset by analyzing the contributions of reward components. To improve interpretability, we analyze the selected latent features using SHAP-based explanations to estimate each feature's contribution to the model's decision. As a result, the Explainable DQN module improves feature selection effectiveness and process transparency before passing the selected features to the MHRA-DNN classifier.

At this stage, the agent selects candidate subsets of latent features. Reinforcement learning-based feature selection is formulated as a decision-making problem in which the agent

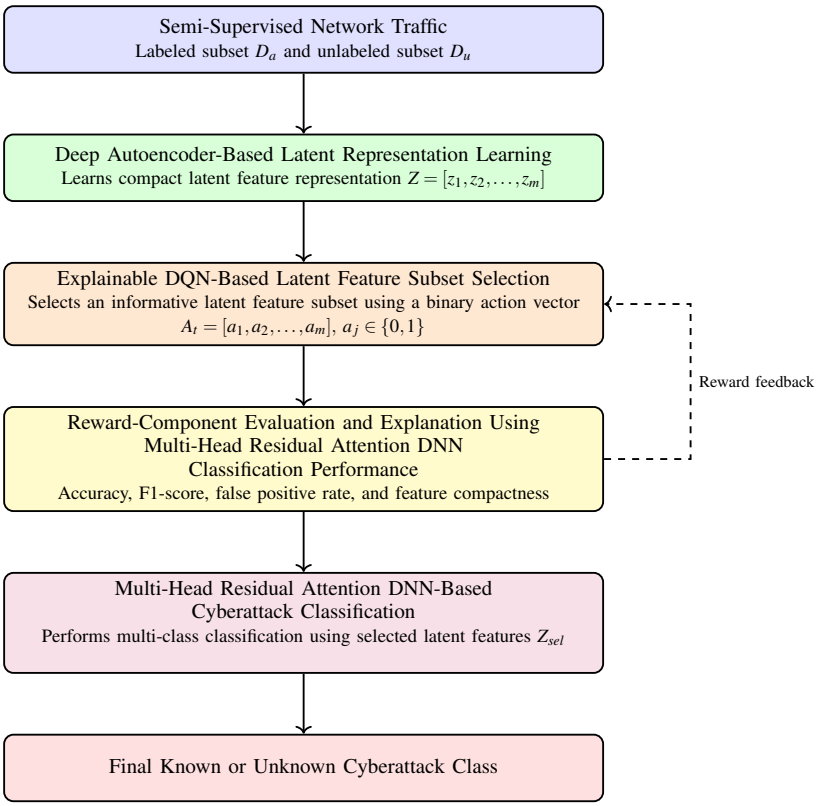


Fig. 1. Architecture and working of adaptive cyberattack classification framework.

chooses feature subsets and receives reward feedback based on downstream classification performance [19].

The latent feature vector produced by the DAE is defined in Eq. (5). The DQN agent identifies a subset of latent features using a binary selection vector:

$$A_t = [a_1, a_2, \ldots, a_m], \qquad a_j \in \{0,1\} \tag{6}$$

Here, $a_j = 1$ signifies that the corresponding latent feature z_j is selected, while $a_j = 0$ indicates that the feature is excluded. Thus, each DQN action represents a candidate subset of latent features.

The selected latent feature subset is represented as:

$$Z_{sel} = \{z_j \mid a_j = 1,\ j = 1, 2, \ldots, m\} \tag{7}$$

In this context, Z_{sel} denotes the subset of latent features selected by the DQN agent. The DQN learns the feature subset selection policy through reward feedback. After a candidate subset is selected, the MHRA-DNN classifier is evaluated using that subset. The DNN's classification performance is then used to compute the DQN's reward. We assign a higher reward when the selected subset improves classification accuracy, F1-score, and false-positive reduction while using fewer features.

The DQN estimates the expected reward of selecting a particular feature subset using the Q-value function:

$$Q(s_t, a_t; \theta) \tag{8}$$

In this formulation, s_t denotes the current feature selection state, a_t indicates the selected feature subset action, and θ represents the trainable parameters of the Q-network [5], [20]. During training, the DQN policy is implicitly derived from the Q-value function using an epsilon-greedy strategy. The agent explores random feature-subset actions with probability ε and exploits the action with the highest estimated Q-value with probability (1-ε).

The optimal feature subset action is selected as:

$$\hat{a}_t = \arg\max_{a_t} Q(s_t, a_t; \theta) \tag{9}$$

This approach ensures that the DQN selects the latent feature subset expected to yield the highest reward according to the learned Q-value estimates. Explainability is incorporated through reward-component decomposition. Rather than treating the selected feature subset as a black-box decision, the framework explains the DQN action by analyzing the contribution of each reward component. These components include the accuracy contribution, the F1-score contribution, the false-positive penalty, and the feature-size penalty.

During training, the DQN updates its Q-network based on reward feedback from the MHRA-DNN classification stage. When the selected feature subset yields strong DNN classification performance with fewer features, the DQN receives

a higher reward. Conversely, if the subset results in poor classification performance, high false positives, or excessive feature usage, the reward is reduced or becomes negative. Through repeated interactions, the DQN incrementally learns which latent feature subsets are most effective for cyberattack classification.

3) Reward-component explanation mechanism: The reward function is central to Explainable DQN-based latent feature subset selection. DQN's learning relies on rewards based on the MHRA-DNN classifier's evaluation of the selected feature subset. The reward function is structured to improve classification performance, minimize unnecessary feature use, and enhance cyberattack detection.

Once the DQN selects a candidate subset of latent features, the MHRA-DNN classifier evaluates it. The DNN's classification performance determines the reward, which is calculated using key metrics: classification accuracy, F1-score, false-positive rate, and feature selection ratio. Accuracy reflects overall correctness, while the F1-score balances precision and recall. The reward function penalizes high false-positive rates to maintain cybersecurity reliability and penalizes excessive feature selection to reduce redundancy. The reward function is defined as:

$$R_t = \lambda_1 Acc_t + \lambda_2 F1_t - \lambda_3 FPR_t - \lambda_4 FR_t \tag{10}$$

Here, Acc_t represents classification accuracy, $F1_t$ denotes the F1-score. FPR_t denotes the false positive rate, and FR_t represents the selected feature ratio at time step t [20].

The selected feature ratio is calculated as follows:

$$FR_t = \frac{|Z_{sel}|}{|Z|} \tag{11}$$

In this context, $|Z_{sel}|$ denotes the number of selected latent features, and $|Z|$ represents the total number of latent features. The coefficients λ_1, λ_2, λ_3, and λ_4 are non-negative weighting factors that balance classification performance, reduction of false positives, and feature compactness.

Each term in the reward function provides interpretability for the feature subset the DQN selects. The accuracy term reflects the selected subset's contribution to overall classification correctness. The F1-score term indicates its impact on balanced detection performance. The false-positive rate term penalizes false alarms, indicating whether the selected subset increases them. The selected feature ratio indicates whether the subset is compact or unnecessarily large.

We introduce a negative reward if the selected feature subset performs below a predefined baseline reward. The adjusted reward formulation is given by:

$$R_t = \lambda_1 Acc_t + \lambda_2 F1_t - \lambda_3 FPR_t - \lambda_4 FR_t - R_{base} \tag{12}$$

Here, R_{base} represents the baseline reward penalty. If the final reward value is negative, the selected feature subset is considered ineffective for classification. This mechanism discourages the DQN agent from repeatedly selecting weak or redundant feature combinations.

Therefore, the reward-component explanation mechanism guides the Explainable DQN agent to select compact, discriminative, and classification-relevant latent feature subsets. Simultaneously, it enhances transparency by decomposing the reward into interpretable components, thereby clarifying the preference for specific subsets of latent features in cyberattack classification.

4) Multi-head residual attention DNN-based cyberattack classification: Fig. 2 presents the architecture of the MHRA-DNN cyberattack classifier. After the Explainable DQN-based latent feature subset selection, the chosen latent features are passed to the Multi-Head Residual Attention block. This block further enhances the latent representation before final classification. While the Explainable DQN identifies the most informative features, different attack classes may rely on various combinations and relationships among them. The multi-head attention mechanism captures multiple feature-relation patterns within the selected subset.

Eq. (7) represents the selected latent feature subset. The selected latent features are transformed into multiple attention heads, with each head learning a distinct representation of the same feature subset. Within this framework, one attention head may emphasize strong dependencies among highly correlated latent features, while another may capture complementary relationships among features that are individually less dominant but collectively beneficial for classification. A separate attention head may focus on class-discriminative feature interactions that help separate attack classes, whereas others may reduce the influence of redundant or less informative feature patterns. Consequently, multiple attention heads enable the classifier to analyze the selected latent features from multiple perspectives, rather than relying on a single attention pattern.

For the i^{th} attention head, the selected latent feature representation is transformed into query, key, and value representations as follows:

$$Q_i = Z_{\text{sel}} W_i^Q \tag{13}$$

$$K_i = Z_{\text{sel}} W_i^K \tag{14}$$

$$V_i = Z_{\text{sel}} W_i^V \tag{15}$$

Here, (Q_i), (K_i), and (V_i) represent the query, key, and value representations of the i^{th} attention head, respectively. (W_i^Q), (W_i^K), and (W_i^V) are learnable weight matrices. The query and key representations compute attention scores, while the value representation provides the feature information weighted by these scores.

The output of the i^{th} attention head is computed as follows:

$$\text{head}_i = \text{softmax}\left(\frac{Q_i K_i^T}{\sqrt{d_k}}\right) V_i \tag{16}$$

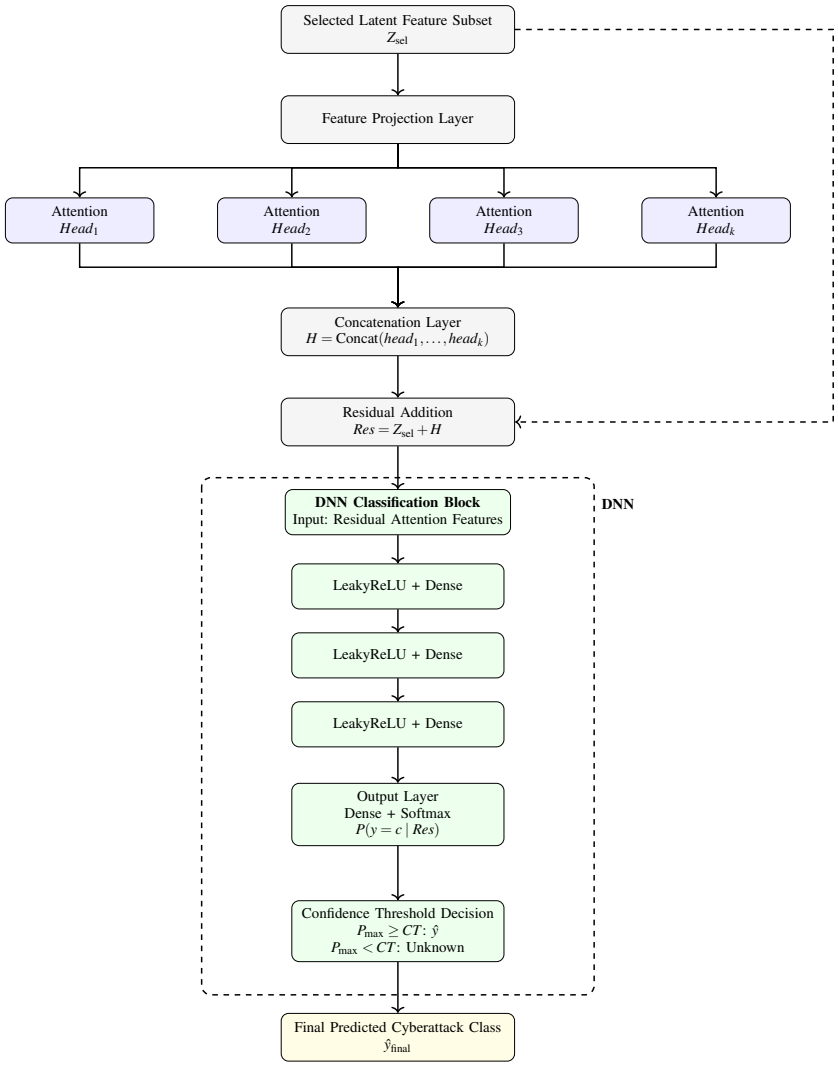


Fig. 2. Architecture of the multi-head residual attention DNN classifier.

In this context, (d_k) denotes the dimension of the key representation. Within the attention block, the softmax function normalizes the attention scores to generate attention weights. Higher attention weights correspond to more significant feature relationships, whereas lower weights indicate less important relationships. As a result, each attention head learns to focus on distinct and useful relationships among the selected latent features.

The combined attention output is:

$$H = Concat(head_1, head_2, \ldots, head_N) \tag{17}$$

Here, H is the combined attention output, $head_i$ is the output of the i^{th} attention head, and N is the number of attention heads. This structure allows the classifier to focus on key relationships among latent features from multiple perspectives.

A residual connection is applied between the selected latent feature representation and the attention-enhanced output to preserve important feature information and support stable, deeper learning. The residual representation is:

$$Res = Z_{sel} + H \tag{18}$$

Here, Res is the residual attention-enhanced feature representation. This connection retains key information from the selected latent features while enabling the attention mechanism to enhance discriminative patterns.

The residual attention-enhanced representation is then processed to the DNN classification block. The dense neural layers learn nonlinear relationships among the selected latent features, thereby generating a discriminative representation for cyberattack classification. The transformation of the l^{th} hidden layer is:

$$h^{(l)} = f(W^{(l)} h^{(l-1)} + b^{(l)}) \tag{19}$$

Here, $h^{(l)}$ is the output of the l^{th} hidden layer, $W^{(l)}$ is the weight matrix, $b^{(l)}$ is the bias vector, and $f(\cdot)$ is the nonlinear

activation function. We use the leaky ReLU activation function.

For multi-class cyberattack classification, the output layer applies the softmax function to generate class probability scores. The probability of class c is:

$$P(y = c|Z_{sel}) = \frac{\exp(o_c)}{\sum_{j=1}^{C} \exp(o_j)} \tag{20}$$

Here, $P(y = c|Z_{sel})$ is the predicted probability of class c, o_c is the output score for class c, and C is the total number of cyberattack classes.

After the MHRA-DNN generates class probability scores, the highest score is compared to a predefined confidence threshold. This threshold separates known cyberattack classes from unexplained or unseen anomaly classes. Since the classifier is trained only on known anomaly classes in D_a, an unseen attack sample from D_u may not match any known class closely. As a result, the maximum predicted probability for these samples is typically lower than for known attack samples. The maximum confidence score is computed as:

$$P_{max} = \max_{c} P(y = c|Z_{sel}) \tag{21}$$

The predicted class is obtained as:

$$\hat{y} = \arg\max_{c} P(y = c|Z_{sel}) \tag{22}$$

The known/unknown decision is then made as:

$$Final_{class} = \begin{cases} \hat{y}, & if \quad P_{max} \geq CT, \\ Unknown, & if \quad P_{max} < CT \end{cases} \tag{23}$$

Here, CT represents the confidence threshold. If P_{max} is greater than or equal to CT, the sample is assigned to the predicted known attack class. Conversely, if P_{max} is less than CT, the sample is categorized as an unexplained or unknown anomaly class.

The threshold CT is determined using validation data from known anomaly classes. During validation, the model's confidence values are analyzed to select a threshold that reduces the risk of misclassifying unknown attacks as known classes. A higher threshold increases caution and improves detection of unknown samples, while a lower threshold assigns more samples to known classes. The threshold should balance classification accuracy with the detection of unknown attacks.

The confidence threshold enables open-set cyberattack classification by preventing the model from assigning every anomaly to a known attack category. It also allows unexplained anomalies to be identified separately.

C. Training and Testing Phase

The proposed model employs a workflow comprising a DAE, an Explainable DQN, and an MHRA-DNN. The labeled network traffic subset D_a is used for training, while the unlabeled traffic subset D_u is used for testing. Algorithm 1 is used during training; the model extracts compact latent representations, selects the most informative latent features, and trains the final classifier. Algorithm 2 is used during testing; the trained encoder, optimized latent feature subset, classifier, and confidence threshold are used to classify new anomalous samples as either known attack classes or unexplained anomaly classes.

The proposed model predicts the cyberattack class for each network traffic sample. The model uses a confidence-threshold mechanism. If the highest class probability meets or exceeds the threshold, the sample is assigned to a known cyberattack class category. Otherwise, it is classified as an unknown anomaly.

Algorithm 1 Training Phase of the Proposed Model

Require: Training dataset D_a.
Ensure: Optimized latent feature subset Z_{opt}, trained Multi-Head Residual Attention DNN classifier, and confidence threshold CT.

1: Train the Deep Autoencoder using samples from D_a.
2: Encode each input sample x using encoder Eq. (2).
3: Reconstruct the input sample using the decoder Eq. (3).
4: Minimize the reconstruction loss using Eq. (4).
5: Extract the latent feature representation from the trained encoder as in Eq. (5).
6: Initialize the Explainable DQN agent for latent feature subset selection.
7: **for** each training episode **do**
8: Generate a binary feature-selection action using Eq. (6).
9: Select the latent feature subset using Eq. (7).
10: Evaluate the Multi-Head Residual Attention DNN classifier using $Z_{sel}^{(t)}$.
11: Predict the class label using the classifier in Eq. (22).
12: Compute the selected feature ratio using Eq. (11).
13: Compute the reward using Eq. (10).
14: Generate the reward-component explanation using accuracy contribution, F1-score contribution, false positive penalty, and feature-size penalty.
15: **end for**
16: Select the best feature-selection action using Eq. (9).
17: Obtain the optimized latent feature subset:

$$Z_{opt} = \{z_j \mid a_j^* = 1, j = 1, 2, \ldots, m\}$$

18: Train the final Multi-Head Residual Attention DNN classifier using Z_{opt}.
19: Select the confidence threshold CT using validation confidence scores obtained from known anomaly classes.
20: Return Z_{opt}, trained Multi-Head Residual Attention DNN classifier, and confidence threshold CT.

IV. Result and Discussion

This section presents the results and discussion of the proposed model. In this work, we analyze our model performance on three benchmark datasets: UNSW-NB15, APA-DDoS, and CICIDS2017.

Algorithm 2 Testing Phase of the Proposed Model

Require: Testing dataset D_u, trained DAE encoder, optimized latent feature subset Z_{opt}, trained Multi-Head Residual Attention DNN classifier, and confidence threshold CT.

Ensure: Final predicted cyberattack class $\hat{y}_{final}$.

1: Pass test sample x_{test} through the trained DAE encoder.

2: Obtain the latent feature representation:

$$Z_{test} = [z_1, z_2, \ldots, z_m]$$

3: Select only the optimized latent features identified during training:

$$Z_{test}^{opt} = \{z_j \in Z_{test} \mid z_j \in Z_{opt}\}$$

4: Feed Z_{test}^{opt} into the trained Multi-Head Residual Attention DNN classifier.

5: Compute the class probability with Eq. (20).

6: Compute the maximum confidence score:

$$P_{max} = \max_c P(y = c|Z_{test}^{opt})$$

7: Obtain the predicted class:

$$\hat{y} = \arg\max_c P(y = c|Z_{test}^{opt})$$

8: Apply the confidence-threshold decision:

$$\hat{y}_{final} = \begin{cases} \hat{y}, & \text{if } P_{max} \geq CT, \\ Unknown, & \text{if } P_{max} < CT. \end{cases}$$

9: Return $\hat{y}_{final}$.

A. Dataset Description and Evaluation Setting

1) UNSW-NB15 dataset: The UNSW-NB15 dataset serves as a benchmark for network traffic classification. The Cyber Range Lab at UNSW Canberra developed it. The IXIA PerfectStorm tool was used to generate network packets. It combines real modern normal activities with synthetic contemporary attack behaviors. It has a training set (175,341 records) and a testing set (82,332 records) [21]. Table I shows category-wise samples of the UNSW-NB15 dataset.

TABLE I. ATTACKS DISTRIBUTION IN UNSW-NB15 DATASET

Categories	Numbers
Normal	93000
Generic	58871
Exploits	44525
Fuzzers	24246
DoS	16353
Reconnaissance	13987
Analysis	2677
Backdoor	2329
Shellcode	1511
Worms	174

2) APA-DDoS dataset: The APA-DDoS dataset is a well-established benchmark dataset for denial-of-service (DDoS) detection methods. Publicly accessible on Kaggle, this dataset features ACK and PUSH-ACK flooding traffic, specifically targeting DDoS attack patterns. It contains 151,200 samples [22]. Table II displays the distribution of attack categories of the APA-DDoS dataset.

3) CICIDS2017 dataset: The Canadian Institute for Cybersecurity developed the CICIDS2017 dataset. It includes both common cyberattacks and benign network traffic. CICFlowMeter was used to generate PCAP and CSV files. The sample structure mimics real-world network environments [23]. Table III displays attack categories distribution in the CICIDS2017 dataset.

TABLE II. ATTACK DISTRIBUTION IN APA-DDOS DATASET

Categories	Numbers
Benign	75600
DDoS-PSH-ACK	37800
DDoS-ACK	37800

TABLE III. ATTACK DISTRIBUTION IN CICIDS2017 DATASET

Categories	Numbers
BENIGN	4378008
DoS Hulk	462146
PortScan	317860
DDoS	256054
DoS GoldenEye	20586
FTP-Patator	15876
SSH-Patator	11794
DoS slowloris	11592
DoS Slowhttptest	10998
Bot	3932
Web Attack Brute Force	1507
Web Attack XSS	652
Infiltration	72
Heartbleed	22
Web Attack Sql Injection	21

To establish a clear open-set evaluation framework, we divided attack categories into known and unknown groups for each dataset. We use the known categories for model training, while separate test samples from these categories assess known-class classification performance. Unknown categories are excluded from training and are introduced only during testing to evaluate the model's capacity to detect previously unseen anomalous traffic. Tables IV and V display the distributions of known-class train-test samples and unknown categories used in the open-set evaluation, respectively.

TABLE IV. DISTRIBUTION OF KNOWN ATTACK CATEGORIES ACROSS DATASETS USED FOR TRAINING AND TESTING

Dataset	Category	Training	Testing
UNSW-NB15	Generic	41,210	17,661
	Exploits	31,167	13,358
	Fuzzers	16,972	7,274
	DoS	11,447	4,906
	Reconnaissance	9,791	4,196
CICIDS2017	DoS Hulk	323,502	138,644
	PortScan	222,502	95,358
	DDoS	179,238	76,816
	DoS GoldenEye	14,410	6,176
	FTP-Patator	11,113	4,763
	SSH-Patator	8,256	3,538
	DoS slowloris	8,114	3,478
APA-DDoS	DDoS-ACK	26,460	11,340
	DDoS-PSH-ACK	26,460	11,340

We exclude the Normal class because the primary objective is adaptive cyberattack classification rather than binary normal-versus-anomaly detection. Previous work [24] separated normal and anomalous traffic using an anomaly detection framework. Consequently, this study assumes that a prior detection stage has already filtered out normal traffic. The proposed classification model receives only anomalous traffic, which it further analyzes to classify known cyberattack categories and

TABLE V. DISTRIBUTION OF UNKNOWN ATTACK CATEGORIES ACROSS DATASETS FOR OPEN-SET EVALUATION

Dataset	Unknown Attack Category	Number of Samples
UNSW-NB15	Analysis	2,677
	Backdoor	2,329
	Shellcode	1,511
	Worms	174
	Total	**6,691**
CICIDS2017	Bot	3,932
	DoS Slowhttptest	10,998
	Heartbleed	22
	Infiltration	72
	Web Attack–Brute Force	1,507
	Web Attack–SQL Injection	21
	Web Attack–XSS	652
	Total	**17,204**
APA-DDoS	Synthetic Unknown	30,000
	Total	**30,000**

detect unexplained anomalous behavior.

This design enables the proposed model to address the more complex task of distinguishing among various attack categories and unknown anomalous samples. Including the Normal class would shift the evaluation focus toward normal-versus-attack discrimination, which does not align with this study's primary objective. Additionally, normal traffic may dominate overall performance in imbalanced datasets and may not accurately represent the model's fine-grained capability for cyberattack classification. Accordingly, the evaluation is limited to anomalous traffic to provide a focused and reliable assessment of the proposed model under open-set cyberattack classification conditions.

We evaluate the proposed model using accuracy, precision, F1-score, sensitivity, specificity, Matthews correlation coefficient, negative predictive value, false positive rate, and false negative rate. These metrics collectively provide a comprehensive assessment of classification accuracy, false-alarm reduction, missed-attack reduction, and overall reliability.

B. Performance and Comparative Analysis of the Proposed Model

This subsection discusses the proposed model's performance and comparative analysis on the UNSW-NB15, APA-DDoS, and CICIDS2017 datasets.

1) Performance analysis of the proposed model: The confusion matrix evaluates the proposed model's class-wise classification performance. While overall metrics such as accuracy, precision, recall, and F1-score offer a general assessment of the model's effectiveness, the confusion matrix provides a more granular analysis of classification behavior for each cyberattack class. In this matrix, diagonal entries indicate correctly classified samples, whereas off-diagonal entries correspond to misclassified samples. Consequently, the confusion matrix facilitates the identification of both accurately detected attack categories and classes where misclassification is prevalent. Fig. 3, 4, and 5 present the confusion matrices of the proposed method for the UNSW-NB15, APA-DDoS, and CICIDS2017 datasets, respectively.

2) Comparative analysis of the proposed model: Table VI demonstrates that the proposed model outperforms ResNet, DNN, Bi-LSTM, and Bi-GRU on the UNSW-NB15 dataset.

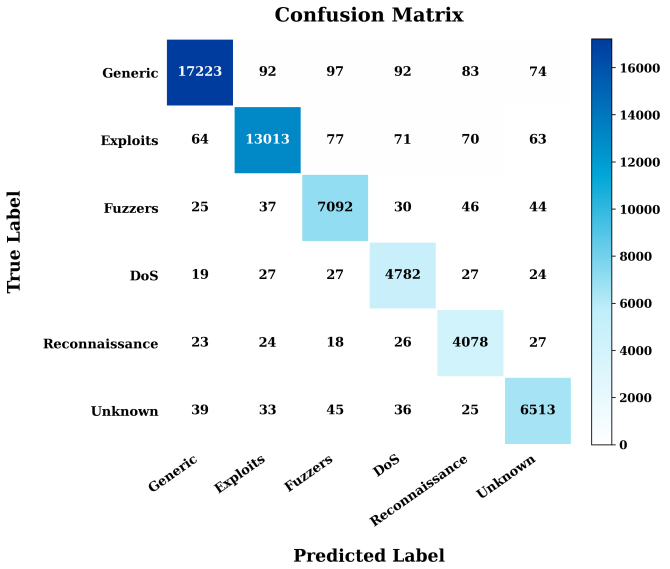


Fig. 3. Confusion matrix of proposed model on UNSW-NB15 dataset.

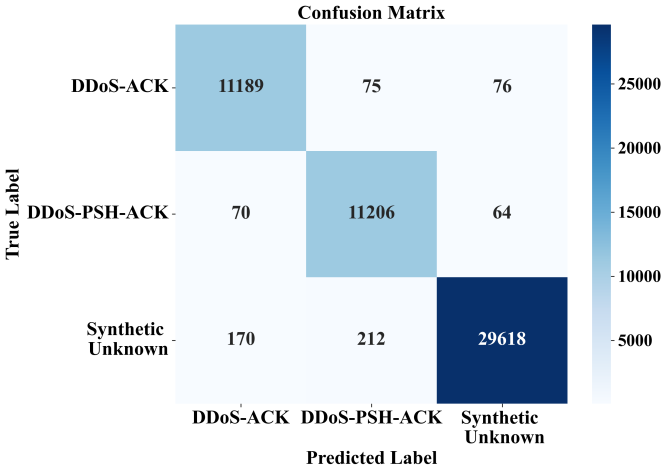


Fig. 4. Confusion matrix of proposed model on APA-DDoS dataset.

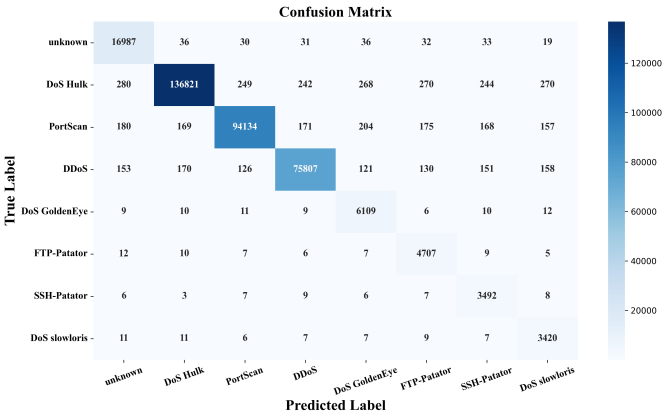


Fig. 5. Confusion matrix of proposed model on CICIDS2017 dataset.

The proposed model achieves the highest accuracy of 0.9744,

while ResNet, DNN, Bi-LSTM, and Bi-GRU achieve 0.9559, 0.9374, 0.9078, and 0.8782, respectively. In terms of precision, the proposed model attains 0.9659, surpassing ResNet (0.9425), DNN (0.9188), Bi-LSTM (0.8823), and Bi-GRU (0.8475). The proposed model also achieves the highest F1-score (0.9699), compared with 0.9490, 0.9279, 0.8935, and 0.8610 for the other models. These results suggest that the proposed model achieves a superior balance between accurate attack classification and minimizing false classifications.

The proposed model shows superior specificity, sensitivity, Matthews correlation coefficient (MCC), and negative predictive value (NPV). Specifically, it achieves a specificity of 0.9949, sensitivity of 0.9741, MCC of 0.9675, and NPV of 0.9945. In contrast, ResNet records 0.9912, 0.9561, 0.9441, and 0.9905, while DNN achieves 0.9875, 0.9382, 0.9208, and 0.9866 for these metrics. Bi-LSTM attains 0.9816, 0.9068, 0.8837, and 0.9804, and Bi-GRU achieves 0.9757, 0.8780, 0.8468, and 0.9741. The proposed model also yields the lowest error rates, with a false positive rate (FPR) of 0.0051 and a false negative rate (FNR) of 0.0259. These values are lower than those of ResNet (0.0088, 0.0439), DNN (0.0125, 0.0618), Bi-LSTM (0.0184, 0.0932), and Bi-GRU (0.0243, 0.1220). Collectively, these findings show that the proposed framework delivers improved classification performance, fewer false alarms, and fewer missed attacks on the UNSW-NB15 dataset.

Table VII compares the performance of the proposed model with ResNet, DNN, Bi-LSTM, and Bi-GRU on the APA-DDoS dataset. The proposed model achieves the highest accuracy (0.9873), outperforming ResNet (0.9627), DNN (0.9437), Bi-LSTM (0.9247), and Bi-GRU (0.9057). It also attains the highest precision (0.9831), compared to 0.9509, 0.9272, 0.9040, and 0.8814 for the other models. The proposed model's F1-score is 0.9852, exceeding those of ResNet (0.9568), DNN (0.9347), Bi-LSTM (0.9140), and Bi-GRU (0.8921). These findings demonstrate that the proposed model achieves superior overall classification performance while maintaining a robust balance between precision and sensitivity on the APA-DDoS dataset.

The proposed model also achieves the highest values for specificity, sensitivity, Matthews correlation coefficient (MCC), and negative predictive value (NPV). Specifically, it records a specificity of 0.9937, sensitivity of 0.9874, MCC of 0.9784, and NPV of 0.9921. In contrast, ResNet achieves 0.9814, 0.9630, 0.9368, and 0.9771, while DNN attains 0.9718, 0.9429, 0.9050, and 0.9660, respectively. Bi-LSTM achieves 0.9624, 0.9256, 0.8739, and 0.9545, and Bi-GRU records 0.9525, 0.9047, 0.8425, and 0.9440. Furthermore, the proposed model yields the lowest false positive rate (FPR) of 0.0063 and false negative rate (FNR) of 0.0126, compared to ResNet (0.0186, 0.0370), DNN (0.0282, 0.0571), Bi-LSTM (0.0376, 0.0744), and Bi-GRU (0.0475, 0.0953). These results indicate that the proposed framework reduces both false alarms and missed attack classifications more effectively than the other models. Collectively, the findings confirm the proposed model's strong performance on the APA-DDoS dataset.

Table VIII presents a performance comparison between the proposed model and ResNet, DNN, Bi-LSTM, and Bi-GRU on the CICIDS2017 dataset. The proposed model achieves the highest accuracy (0.9870), while ResNet, DNN, Bi-LSTM, and Bi-GRU achieve 0.9754, 0.9630, 0.9436, and 0.9350, respectively. The proposed model achieves the highest precision (0.9286), compared with 0.8782, 0.8316, 0.7743, and 0.7522 for the other models. Regarding F1-score, the proposed model attains 0.9559, surpassing ResNet (0.9210), DNN (0.8864), Bi-LSTM (0.8387), and Bi-GRU (0.8190). These findings demonstrate that the proposed model delivers superior overall classification performance and achieves a more robust balance between precision and sensitivity on the CICIDS2017 dataset.

The proposed model also shows superior specificity, sensitivity, Matthews correlation coefficient (MCC), and negative predictive value (NPV). Specifically, it achieves a specificity of 0.9981, sensitivity of 0.9870, MCC of 0.9818, and NPV of 0.9977. In comparison, ResNet achieves 0.9965, 0.9754, 0.9658, and 0.9956, while DNN records 0.9947, 0.9644, 0.9487, and 0.9934, respectively. Bi-LSTM achieves 0.9919, 0.9436, 0.9224, and 0.9900, whereas Bi-GRU yields 0.9907, 0.9346, 0.9107, and 0.9885. Furthermore, the proposed model produces the lowest false positive rate (FPR) of 0.0019 and false negative rate (FNR) of 0.0130, compared to ResNet (0.0035, 0.0246), DNN (0.0053, 0.0356), Bi-LSTM (0.0081, 0.0564), and Bi-GRU (0.0093, 0.0654). These results suggest that the proposed framework more effectively reduces both false alarms and missed attack classifications than the compared models on the CICIDS2017 dataset.

Fig. 6a to 10 show the performance of the proposed model compared to ResNet, DNN, Bi-LSTM, and Bi-GRU on the UNSW-NB15, APA-DDoS, and CICIDS2017 datasets.

C. Component-Wise Ablation Design

A component-wise ablation design is presented to clarify the functional role of each major module within the proposed framework. The model integrates DAE-based latent representation learning, Explainable DQN-based latent feature subset selection, MHRA-DNN-based classification, SHAP-based interpretation, and IFSA-based parameter tuning, with each component contributing to a distinct stage of the classification process. Table IX summarizes the model variants and their respective purposes. The baseline DNN performs direct classification using preprocessed features. The DAE + DNN variant demonstrates the impact of latent representation learning. The DAE + DQN + DNN variant elucidates the contribution of reward-based latent feature subset selection. The DAE + DQN + MHRA-DNN variant illustrates the effect of attention and residual learning. The complete proposed model encompasses feature learning, feature selection, classification, interpretation, and classifier-level parameter tuning.

D. Analysis of Testing Loss and Accuracy

This subsection examines the proposed model's testing loss and accuracy. Because the model is intended to classify previously unseen anomalous traffic, the testing loss and accuracy curves serve to evaluate its generalization capability and classification stability.

Fig. 11, 12, and 13 compare testing loss and testing accuracy for the UNSW-NB15, APA-DDoS, and CICIDS2017 datasets, respectively. In these figures, the testing loss curve represents the model's classification error on previously unseen anomalous samples, whereas the testing accuracy curve reflects

TABLE VI. COMPARISON ANALYSIS OF PROPOSED MODELS WITH EXISTING MODELS ON UNSW-NB15 DATASET

UNSW-NB15					
Metrics	**Proposed**	**ResNet**	**DNN**	**Bi-LSTM**	**Bi-GRU**
Accuracy	0.9744	0.9559	0.9374	0.9078	0.8782
Precision	0.9659	0.9425	0.9188	0.8823	0.8475
F1-Score	0.9699	0.9490	0.9279	0.8935	0.8610
Specificity	0.9949	0.9912	0.9875	0.9816	0.9757
Sensitivity	0.9741	0.9561	0.9382	0.9068	0.8780
MCC	0.9675	0.9441	0.9208	0.8837	0.8468
NPV	0.9945	0.9905	0.9866	0.9804	0.9741
FPR	0.0051	0.0088	0.0125	0.0184	0.0243
FNR	0.0259	0.0439	0.0618	0.0932	0.1220

TABLE VII. COMPARISON ANALYSIS OF PROPOSED MODEL WITH EXISTING MODELS ON APA-DDOS DATASET

APA-DDoS					
Metrics	**Proposed**	**ResNet**	**DNN**	**Bi-LSTM**	**Bi-GRU**
Accuracy	0.9873	0.9627	0.9437	0.9247	0.9057
Precision	0.9831	0.9509	0.9272	0.9040	0.8814
F1-Score	0.9852	0.9568	0.9347	0.9140	0.8921
Specificity	0.9937	0.9814	0.9718	0.9624	0.9525
Sensitivity	0.9874	0.9630	0.9429	0.9256	0.9047
MCC	0.9784	0.9368	0.9050	0.8739	0.8425
NPV	0.9921	0.9771	0.9660	0.9545	0.9440
FPR	0.0063	0.0186	0.0282	0.0376	0.0475
FNR	0.0126	0.0370	0.0571	0.0744	0.0953

TABLE VIII. COMPARISON ANALYSIS OF PROPOSED MODEL WITH EXISTING MODELS ON CICIDS2017 DATASET

CICIDS2017					
Metrics	**Proposed**	**ResNet**	**DNN**	**Bi-LSTM**	**Bi-GRU**
Accuracy	0.9870	0.9754	0.9630	0.9436	0.9350
Precision	0.9286	0.8782	0.8316	0.7743	0.7522
F1-Score	0.9559	0.9210	0.8864	0.8387	0.8190
Specificity	0.9981	0.9965	0.9947	0.9919	0.9907
Sensitivity	0.9870	0.9754	0.9644	0.9436	0.9346
MCC	0.9818	0.9658	0.9487	0.9224	0.9107
NPV	0.9977	0.9956	0.9934	0.9900	0.9885
FPR	0.0019	0.0035	0.0053	0.0081	0.0093
FNR	0.0130	0.0246	0.0356	0.0564	0.0654

TABLE IX. COMPONENT-WISE ABLATION DESIGN OF THE PROPOSED FRAMEWORK

Model Variant	**Purpose**
DNN	Baseline classifier using preprocessed features.
DAE + DNN	Demonstrates the role of DAE-based latent representation learning in generating compact feature representations.
DAE + DQN + DNN	Explains the contribution of reward-based latent feature subset selection through the Explainable DQN module.
DAE + DQN + MHRA-DNN	Examines the impact of multi-head attention and residual learning on cyberattack classification.
Proposed Model	Describes the complete framework, which integrates DAE-based latent feature learning, Explainable DQN-based feature selection, MHRA-DNN-based classification, SHAP-based interpretation, and IFSA-based classifier-level parameter tuning.

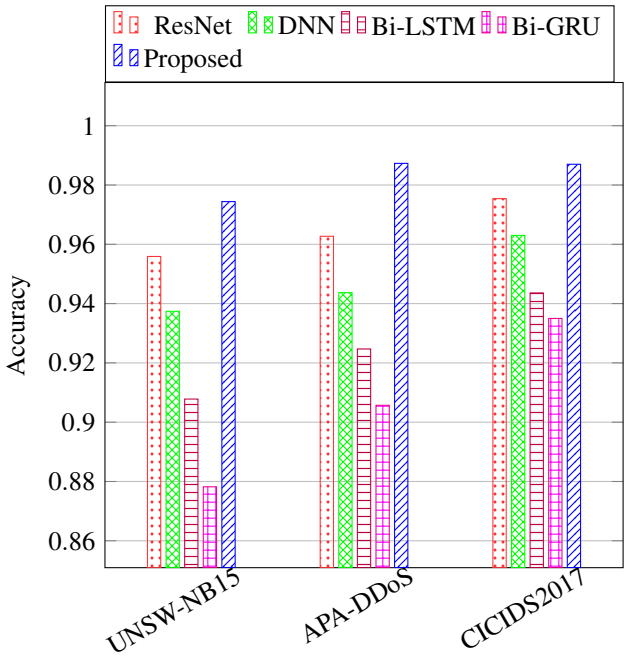
the model's capacity to classify known attack classes and detect unknown anomalous behavior.

The proposed model achieves lower testing loss and higher testing accuracy than existing methods. These results indicate that the model learns more discriminative attack representations and generalizes more effectively to previously unseen anomalous traffic. Overall, the analysis of testing loss and accuracy supports the proposed model's effectiveness for adaptive cyberattack classification.

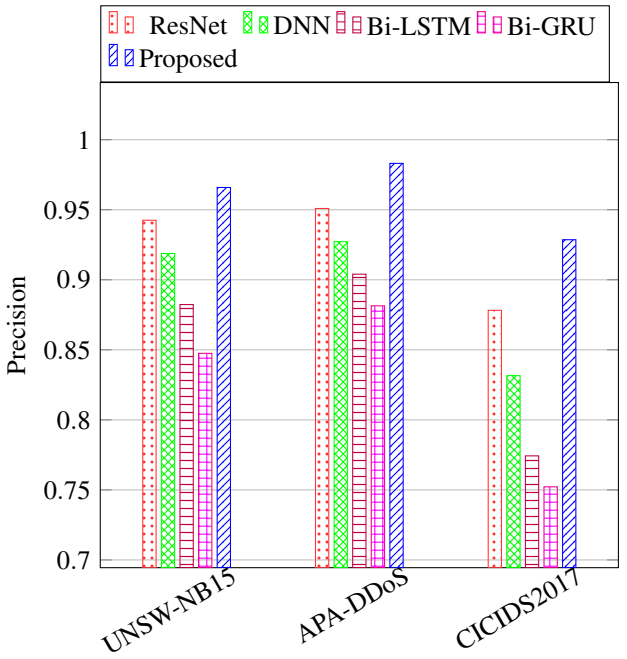
E. ROC Curve Analysis

This subsection presents the receiver operating characteristic (ROC) curves for the proposed adaptive cyberattack classification model, alongside those of existing comparison methods. The ROC curve evaluates each model's capacity to distinguish between anomalous traffic classes by plotting the true positive rate against the false positive rate. A curve closer to the top-left region indicates superior classification performance.

Fig. 14, 15, and 16 present ROC curve comparisons for the UNSW-NB15, APA-DDoS, and CICIDS2017 datasets, respectively. The proposed model demonstrates superior ROC performance compared to existing methods. This result suggests enhanced discriminative capability in classifying both known attack classes and previously unseen anomalous behaviors. The ROC analysis substantiates that the proposed adaptive cyberattack classification model delivers reliable performance on novel anomalous traffic.

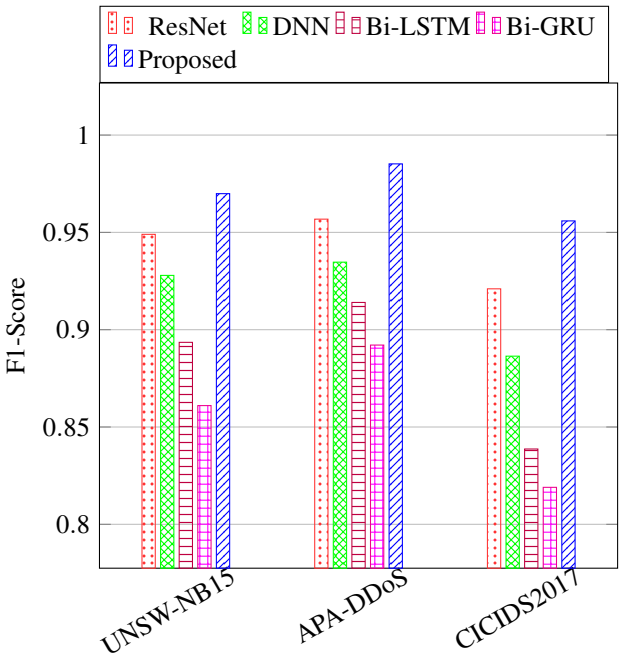


(a) Graphical accuracy comparison of the proposed model with existing models.

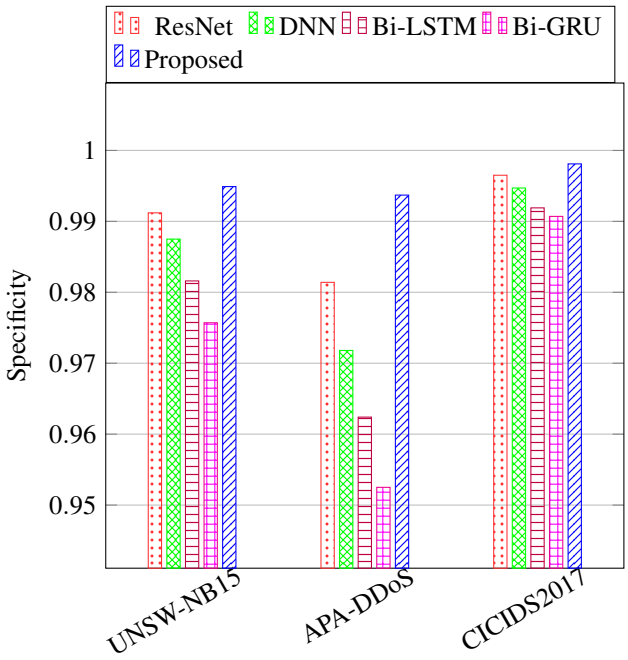


(b) Graphical precision comparison of the proposed model with existing models.

Fig. 6. Accuracy and precision comparison of the proposed model with existing models.



(a) Graphical F1-score comparison of the proposed model with existing models.

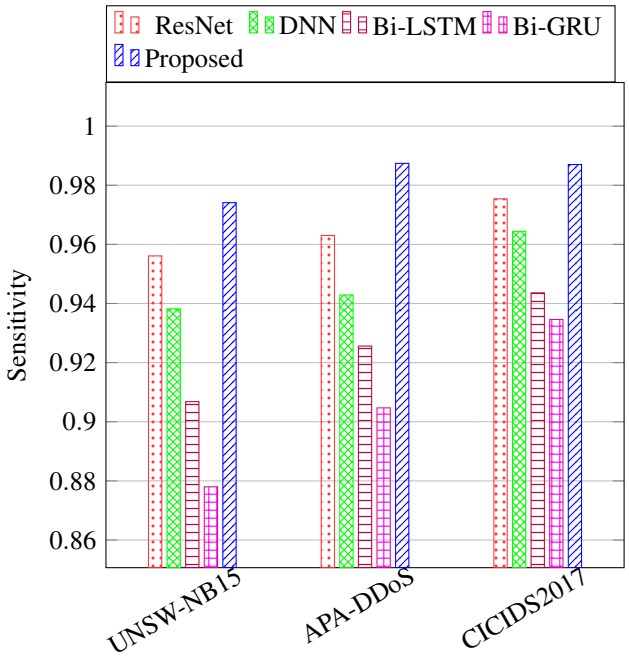


(b) Graphical specificity comparison of the proposed model with existing models.

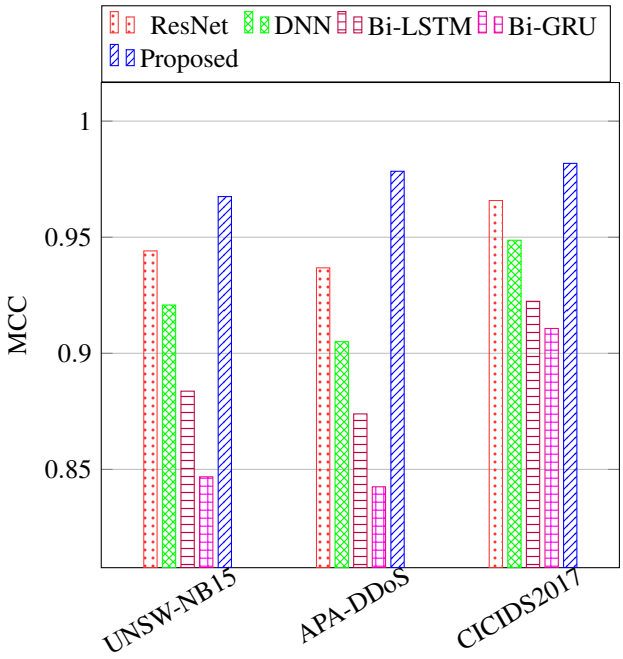
Fig. 7. F1-score and specificity metric comparison of the proposed model with existing models.

F. SHAP-Based Explainability Analysis

Fig. 17 presents the SHAP-based explainability analysis of the proposed model applied to the UNSW-NB15 dataset. In the SHAP summary plot, the vertical axis denotes the selected latent features, and the horizontal axis denotes the SHAP values. Positive SHAP values signify that a feature increases its influence on the model's decision, while negative SHAP values signify a reduction in influence. Each point's color represents the value of the corresponding latent feature for a given sample, with red indicating higher values and blue

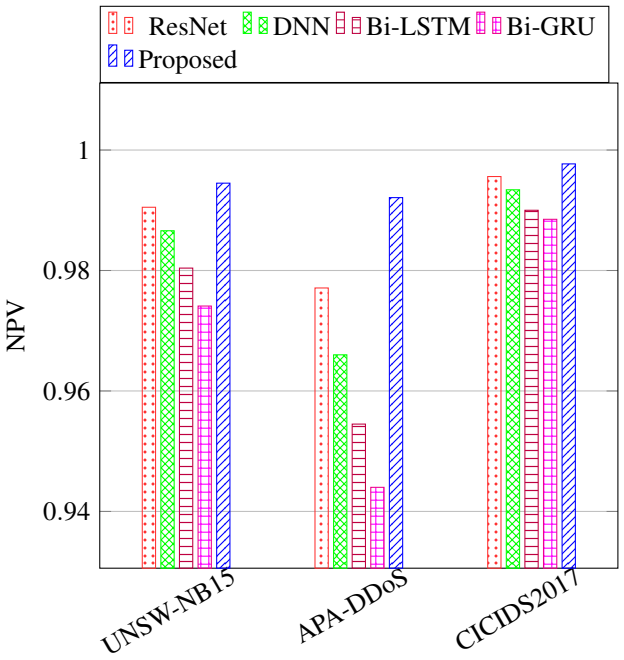


(a) Graphical sensitivity comparison of the proposed model with existing models.

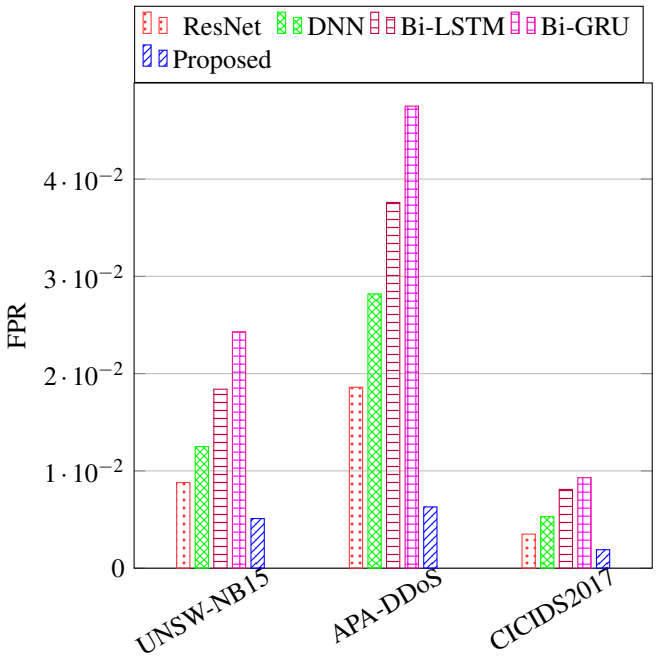


(b) Graphical MCC comparison of the proposed model with existing models.

Fig. 8. Sensitivity and MCC comparison of the proposed model with existing models.



(a) Graphical NPV comparison of the proposed model with existing models.



(b) Graphical FPR comparison of the proposed model with existing models.

Fig. 9. NPV and FPR comparison of the proposed model with existing models.

indicating lower values.

The figure demonstrates that Features 56, 16, 8, 9, and 2 are the most influential latent features, as evidenced by their wider SHAP value distributions and greater deviations from zero. Features 23, 1, 49, 61, and 45 also contribute to the classification decision, although their influence is comparatively lower. The SHAP analysis indicates that the proposed model assigns greater importance to discriminative latent features rather than treating all selected features equally. This approach enhances the model's transparency and supports the interpretability of

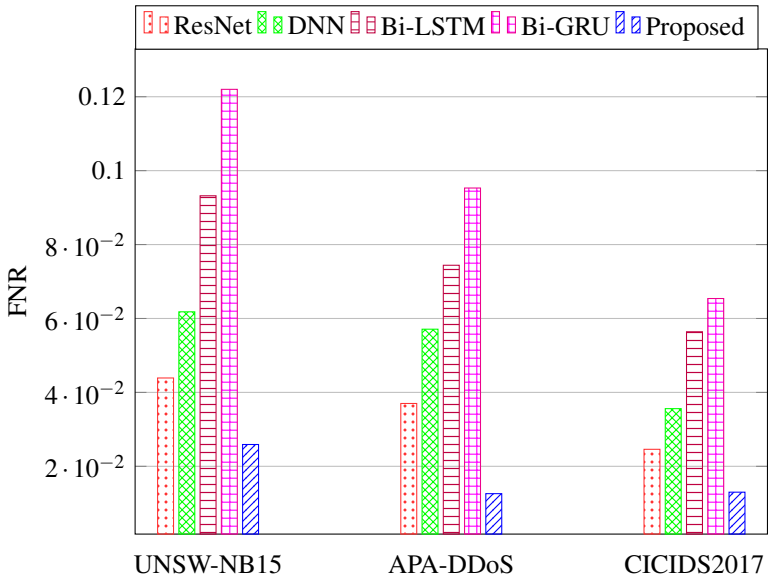
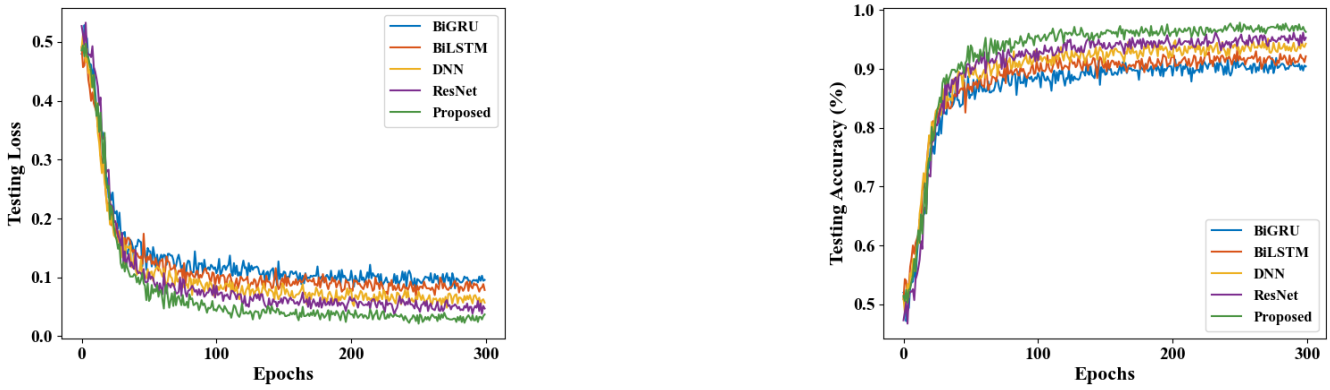


Fig. 10. Graphical FNR comparison of the proposed model with existing models.



(a) Testing loss.

(b) Testing accuracy.

Fig. 11. Comparison of testing loss and accuracy between the proposed model and existing models on the UNSW-NB15 dataset.

the Explainable DQN-based feature selection stage.

Fig. 18 presents the SHAP-based explainability analysis of the proposed model applied to the APA-DDoS dataset. The results indicate that Features 4, 47, 40, 54, and 46 are the most influential latent features, as evidenced by their broader SHAP value distributions and greater deviations from zero. In contrast, Features 41, 52, 67, 12, and 6 also contribute to the classification decision, but their influence is comparatively lower.

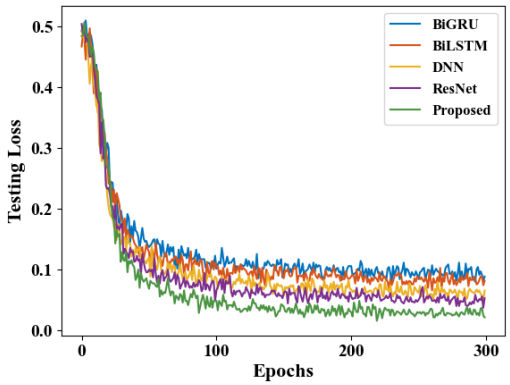
The SHAP value distribution demonstrates that the proposed model assigns varying levels of importance to latent features, rather than treating all features equally. This finding confirms that the Explainable DQN-based feature selection stage effectively identifies discriminative latent features, thereby enhancing the interpretability of the adaptive cyberattack classification model on the APA-DDoS dataset.

Fig. 19 presents the SHAP-based explainability analysis of the proposed model on the CICIDS2017 dataset. The analysis indicates that Features 37, 55, 30, 51, and 20 exert the greatest influence on the classification decision, as evidenced by their broader SHAP value distributions and greater deviations from zero. In contrast, Features 18, 13, 32, 31, and 65 also contribute to the model output, but their impact is comparatively lower.

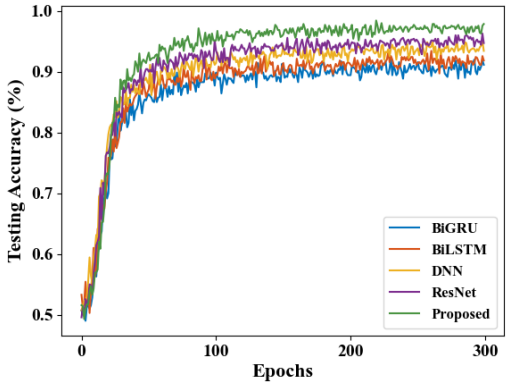
The SHAP value distribution confirms that the proposed model assigns varying levels of importance to distinct latent features. This analysis demonstrates that the Explainable DQN-based feature selection stage effectively identifies key discriminative latent features, thereby enhancing the interpretability of the adaptive cyberattack classification model on the CICIDS2017 dataset.

G. Effect of Parameter Tuning

We tuned parameters to improve classification performance and training stability of the MHRA-DNN. This study applied

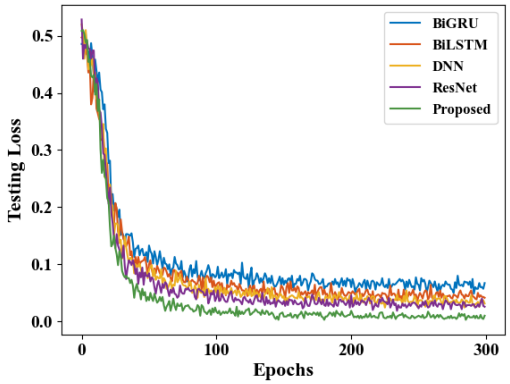


(a) Testing loss.

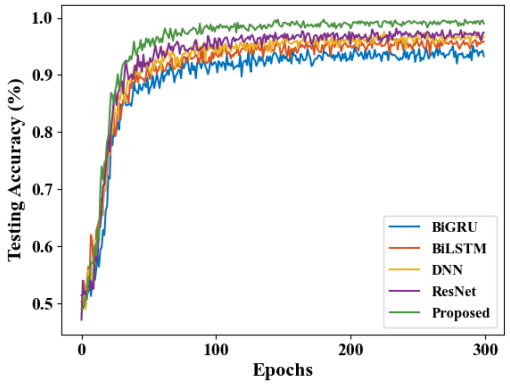


(b) Testing accuracy.

Fig. 12. Comparison of testing loss and accuracy between the proposed model and existing models on the APA-DDoS dataset.



(a) Testing loss.



(b) Testing accuracy.

Fig. 13. Comparison of testing loss and accuracy between the proposed model and existing models on the CICIDS2017 dataset.

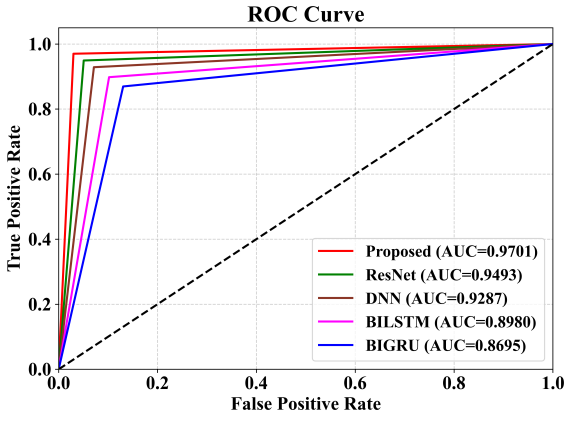


Fig. 14. ROC Curve comparison between the proposed and existing models on the UNSW-NB15 Dataset.

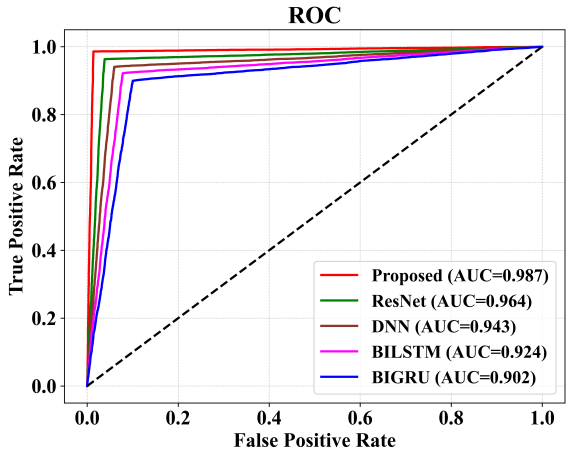


Fig. 15. ROC Curve comparison between the proposed model and existing models on the APA-DDoS Dataset.

Improved Flamingo Search Algorithm (IFSA)-based tuning to the final classification model. The process targeted optimal

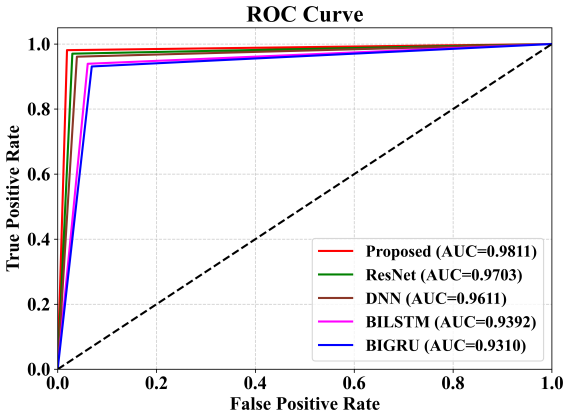


Fig. 16. ROC Curve comparison between the proposed model and existing models on the CICIDS2017 Dataset.

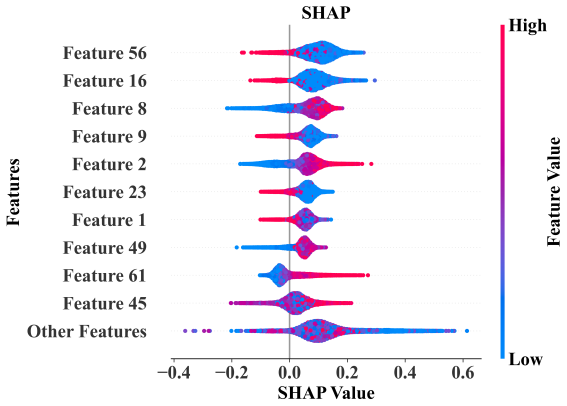


Fig. 17. SHAP analysis of selected latent features on the UNSW-NB15 dataset.

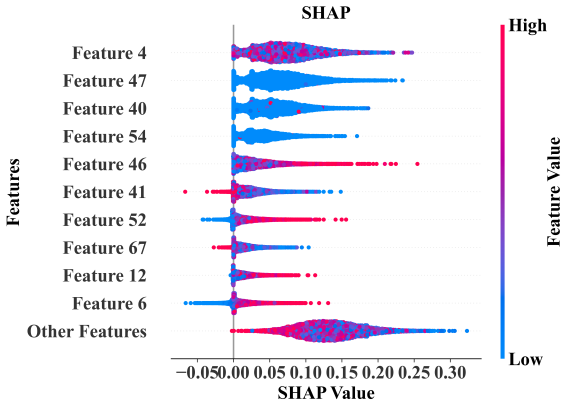


Fig. 18. SHAP analysis of selected latent features on the APA-DDoS dataset.

values for key parameters such as learning rate, batch size, dropout rate, number of attention heads, number of hidden neurons, and other attention-related settings. The goal was to optimize the classifier configuration after the Explainable DQN module selected the latent feature subset.

The results show that tuning classifier-level parameters improves the proposed model's learning behavior. With optimized parameters, the MHRA-DNN converges better, achieves lower testing loss, and maintains more stable testing accuracy. These gains result from selecting appropriate parameter values, which help the classifier learn more effective relationships among the chosen latent features. As a result, IFSA-based parameter tuning strengthens the final classification stage and enhances the overall reliability of the adaptive cyberattack classification model.

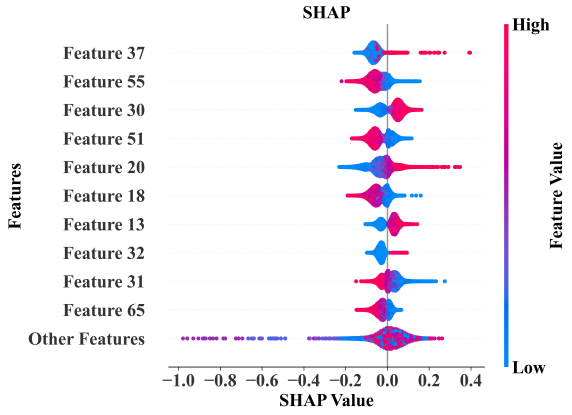


Fig. 19. SHAP analysis of selected latent features on the CICIDS2017 dataset.

V. Conclusion

This study presents an adaptive and explainable cyberattack classification model for anomalous network traffic. This approach focuses exclusively on anomalous samples, assuming a preceding anomaly detection stage filters out normal traffic. The model integrates a DAE, an explainable DQN, SHAP analysis, and an MHRA-DNN. The DAE extracts compact latent features from high-dimensional traffic data. The Explainable DQN identifies the most informative subset of latent features using a reward-based selection mechanism. MHRA-DNN functions as the final classifier. SHAP analysis provides further insight into how the selected latent features contribute to the final classification decision.

We evaluated the model on the UNSW-NB15, APA-DDoS, and CICIDS2017 datasets. Results demonstrate superior performance compared to existing methods across multiple metrics, including accuracy, precision, F1-score, sensitivity, specificity, Matthews correlation coefficient (MCC), negative predictive value (NPV), false positive rate (FPR), and false negative rate (FNR). Additional analyses, such as the confusion matrix, receiver operating characteristic (ROC) curve, test loss, test accuracy, and SHAP analysis, further validate the model's effectiveness. The proposed framework provides reliable, interpretable cyberattack classification for open-set anomalous traffic. Future research may extend this model for real-time deployment, incremental learning, and integration with autonomous response systems.

References

[1] W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult, "Toward open set recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 7, pp. 1757–1772, 2012.

[2] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications surveys & tutorials*, vol. 18, no. 2, pp. 1153–1176, 2015.

[3] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE access*, vol. 7, pp. 41 525–41 550, 2019.

[4] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[5] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[8] N. Moustafa and J. Slay, "Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set)," in *2015 military communications and information systems conference (MilCIS)*. Ieee, 2015, pp. 1–6.

[9] I. Sharafaldin, A. H. Lashkari, A. A. Ghorbani *et al.*, "Toward generating a new intrusion detection dataset and intrusion traffic characterization." *ICISSp*, vol. 1, no. 2018, pp. 108–116, 2018.

[10] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[11] K. Ren, Y. Zeng, Z. Cao, and Y. Zhang, "Id-rdrl: a deep reinforcement learning-based feature selection intrusion detection model," *Scientific reports*, vol. 12, no. 1, p. 15370, 2022.

[12] K. Ren, Y. Zeng, Y. Zhong, B. Sheng, and Y. Zhang, "Mafsids: a reinforcement learning-based intrusion detection model for multi-agent feature selection networks," *Journal of Big Data*, vol. 10, no. 1, p. 137, 2023.

[13] Y. Wu, Y. Hu, J. Wang, M. Feng, A. Dong, and Y. Yang, "An active learning framework using deep q-network for zero-day attack detection," *Computers & Security*, vol. 139, p. 103713, 2024.

[14] J. Fang and C. Xie, "Unknown intrusion traffic detection method based on unsupervised learning and open-set recognition," *Scientific Reports*, vol. 15, no. 1, p. 17001, 2025.

[15] N. Cai, Y. Jiang, H. Zhang, Z. Yang, L. Zhong, and Q. Chen, "Hcrp-osd: Fine-grained open-set intrusion detection based on hybrid convolution and adversarial reciprocal points learning," *Concurrency and Computation: Practice and Experience*, vol. 37, no. 4-5, p. e70010, 2025.

[16] S. Cai, Y. Zhao, J. Lyu, S. Wang, Y. Hu, M. Cheng, and G. Zhang, "Ddp-dar: Network intrusion detection based on denoising diffusion probabilistic model and dual-attention residual network," *Neural Networks*, vol. 184, p. 107064, 2025.

[17] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[18] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[19] W. Fan, P. Zhong, R. Zhai, S. Liu, and X. Zhang, "Reinforcement learning for feature selection," *ACM Transactions on Knowledge Discovery from Data*, vol. 15, no. 6, pp. 1–28, 2021.

[20] Z. Juozapaitis, A. Koul, A. Fern, M. Erwig, and F. Doshi-Velez, "Explainable reinforcement learning via reward decomposition," in *IJCAI/ECAI Workshop on explainable artificial intelligence*, 2019.

[21] "UNSW-NB15," Aug. 2024, [Online; accessed 21. Aug. 2024]. [Online]. Available: https://research.unsw.edu.au/projects/unsw-nb15-dataset

[22] "APA-DDoS Dataset," Aug. 2024, [Online; accessed 21. Aug. 2024]. [Online]. Available: https://www.kaggle.com/datasets/yashwanthkumbam/apaddos-dataset

[23] "Network Intrusion dataset(CIC-IDS- 2017)," Aug. 2024, [Online; accessed 21. Aug. 2024]. [Online]. Available: https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset

[24] D. K. Roy and H. K. Kalita, "Anomaly detection in an open set environment using reinforcement learning," *IET Information Security*, vol. 2025, no. 1, p. 7990749, 2025.