Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 11, November : 2024

# MMDF-RL: Design of an Integrated model for Performance Enhancement of Network Anomaly Detection Using Multiple Modal Data Fusion and Reinforcement Learning

Vamsi Naidu [1], Basaveswararao Bobba[2*], Neelima Guntupalli [3]

[1,2,3,]*Department of Computer Science & Engineering, Acharya Nagarjuna University, Guntur, 522510, India.*

* Corresponding author's Email: neelima.guntupalli80@gmail.com

**Abstract:** Most of the existing systems are based on single-modal data along with traditional machine learning techniques, result in poor system performance and low adaptability against evolving threats. In this study, to address these challenges, a novel Multiple Modal Data Fusion approach combined with Reinforcement Learning (MMDF-RL) along with explainable AI technique is proposed. The proposed approach uses the multi-head attention mechanism to combine the outputs from network-traffic features, behavioral analytics and host-based metrics, thus incorporating multi-modal complicated relationships while also focusing on critical features. The performance of detecting malicious traffic is improved accordingly. Finally, feature selection is optimized through Proximal Policy Optimization(PPO), a reinforcement learning method that reduces the feature space by 60% while having zero impact on its performance. Variational Auto Encoders(VAE) are further used for unsupervised anomaly detection that can identify unseen threats with high precision. For more transparency, SHAP is used, which gives the per-instance explanation of the decisions made by the model. For performance evaluation, two data sets are adopted , one is CIRA-CIC-DoHBrw-2020 that consists of DOH encrypted traffic and another is CICIDS 2017 which has network traffic data sets. The proposed approach is evaluated and compared with the three contemporary models. The performance metrics yield better results than the other models. It outperforms others on most metrics, demonstrating high classification accuracy, precision, recall, F1-score and computational time. Additionally, SHAP explanations enhance model transparency and reduce the false positive rate, while providing interpretable insights for cybersecurity analysts. This integrated framework enhances both detection capability and model interpretability, which together form an end-to-end solution to detect and explain malicious DoH and network traffic in dynamic network environments.

**Keywords:** Multiple Modal Data Fusion, Proximal Policy Optimization, Variational Auto Encoder, SHAP, Anomaly Detection

## 1. Introduction

The recent proliferation of encrypted internet traffic has been steadily making the job of traditional cybersecurity mechanisms difficult to accomplish. Among these protocols, Domain-over-HTTPS, widely referred to as DoH, protects domain name resolution requests sent by a client by encrypting them in the process. However, it acts like a two-edged sword because such a technique can also be deployed by attackers to mask their activities from traditional network security measures. The detection of anomalous or malicious traffic within encrypted streams, such as DoH, therefore, is of prime interest. Traditional detection mechanisms have been based mostly on one type of data or classical machine learning models, which are unable to cope with the dynamic and multifaceted nature of modern cyber threats. These systems rely on either network traffic features or host-based metrics in isolation, which leads to rather poor detection accuracy, with a high false positive rate

especially when novel, unseen attack vectors are encountered. Most of the traditional models are also "black boxes" that offer rather poor interpretability, which makes it difficult for security analysts to comprehend the underlying reasoning behind such classification decisions made by these systems. This lack of transparency has raised a number of concerns about the operational trustworthiness of such systems, especially in high-stakes environments where misclassification can lead to grave consequences. Faced with these challenges, the cyber security community has recognized the need to go beyond the state of the art to more advanced methods capable of fusing multiple data modalities and adapting dynamically to an ever-evolving threat landscape. In particular, besides information from various sources such as network traffic patterns, behavioral analytics, and host-based features, multimodal data fusion has emerged as one of the most promising techniques to enhance detection accuracy and robustness. Through the use of these different modalities, available complementary information will provide a better understanding of network behavior, hence allowing for anomalies that may be difficult to detect when a single source of data is used.

In this context, this paper proposes a state-of-the-art integrated model for improving the detection of malicious encrypted traffic by fusing multi-modal data through the application of advanced feature selection and unsupervised learning. The core heart of this model is a multi-head attention mechanism adopted for the fusion of diverse inputs from network traffic, user behavior, and system metrics. Scaled multi-head attention mechanisms allow the model to learn several independent representations of the input, capturing nuanced interactions among different modalities, especially apt for the particular task. For example, whereas packet sizes and inter-arrival times may provide some indication of the nature of the traffic flow, domain query frequencies and time-of-day patterns within those query frequencies may be more indicative of the intent driving that flow. Weighing the importance of features across different modalities guarantees that, in each instance, the model focuses on the most relevant aspects of the data; this would actually enhance the capability for malicious behavior detection. Besides data fusion, the model employs Proximal Policy Optimization, a reinforcement learning algorithm for the purpose of feature selection optimization. The feature space might be huge and complicated for a common detection scenario, which could bring about other problems such as overfitting and reduced interpretability of the model. PPO handles this challenge by dynamically choosing the most informative subset of features from available data samples. By optimizing feature selection, the model not only reduces computation complexity but also enhances its generalization capability to work on a wide range of scenarios.

In our experiment, PPO decreased the feature space from 50 to about 20 features-a reduction of roughly 60% without any loss in classification performance. This is an especially appealing reduction in cybersecurity applications, since both interpretability and the execution of real-time decisions are not compromised. For the detection of unseen threats, a VAE was used for unsupervised anomaly detection. Unlike the performance of the supervised model, which relies on labeled datasets and suffers when utilized to detect novel attacks, VAEs learn a probabilistic latent representation of normal traffic patterns by minimizing the reconstruction error between input data and their respective reconstructions. This learned distribution effectively acts to highlight traffic that is highly dissimilar from it as anomalous at inference time, which makes the VAE particularly effective in highlighting potential zero-day attacks or other novel threats. The experiments we have conducted show that the VAE achieved 92% precision and 88% recall in detection performance for malicious DoH traffic, hence proving quite effective in real-world settings.

A limitation with many advanced architecture cybersecurity machine learning models is that they are not interpretable. Even when the results from these models are rather high in terms of accuracy, the

decisions themselves are essentially opaque, offering little by which a security analyst can understand why a prediction was made in a manner and, therefore, trust or validate it. The model proposed herein implements SHAP-a local explainability method that uses the terminology of cooperative game theory. In each case, SHAP assigns an importance value, called a SHAP value, to each feature that explains how each feature contributed to the final classification decision. It enhances not only interpretability but also reduces false positives since misclassifications due to specific feature values can also be identified. This reduces the total number of false positives by 15%, a significant margin in our experiments in general. Our proposed model makes a great stride in the detection of encrypted malicious traffic by combining these state-of-the-art techniques: it leverages the advantages brought by multi-modal data fusion, reinforcement learning-based feature selection, and unsupervised learning to create a robust and adaptive system that can detect not only the known threats but also novel ones. Additionally, explainability by SHAP makes the decisions of the system interpretable and trustworthy for a human analyst, which is a necessary step for practical deployment in real-world settings.

**Motivation and Contribution:**

The motivation towards this work is due to the ever-increasing sophistication and complexity of cyber threats, especially in view of the encrypted traffic types such as Domain-over-HTTPS. The pervasiveness of encryption has left traditional cybersecurity measures scrambling to maintain their efficacy. Most especially, this is problematic for the detection systems dependent on single-modal data, as often they cannot tell in favor of subtle, multiple-faceted patterns indicative of malicious behavior. But traditional machine learning is equally rather powerful and acts mostly like a black box, whereby little or no explanation is provided for the decisions it reaches. This opaqueness discourages deployment in critical infrastructure, where interpretability and transparency will be needed to trust operation. These challenges motivate the need for an advanced interpretable, adaptable solution for encrypted traffic. Key contributions of this paper include the design and implementation of a comprehensive model that addresses these challenges using state-of-the-art techniques. First, we introduce the multi-head attention mechanism for multi-modal data fusion, where the model can grasp complex relationships across different data types, including network traffic, behavior analytics, and host-based metrics. That gives a more articulate and correct abstraction of network activities, hence enhancing malicious behavior detection. The feature selection is optimized by the employment of PPO. This decreases model complexity while incurring minimal loss in accuracy, thereby enhancing interpretability and allowing the system to function in real time. The most important thing is that our approach will make it possible, using a Variational Autoencoder for unsupervised anomaly detection, to reveal a threat which has not been seen before, whereas traditional supervised methods cannot solve this problem. Finally, we provide the system with explainable AI by incorporating SHAP into our design, giving us a very transparent decision-making process. This improves trust and reduces the number of false positives so as to make the system more applicable to real-world applications. These are put together to provide a robust, flexible, and explainable solution for malicious activity detection in encrypted network traffic.

The paper is organized as follows: The contemporary research papers for anomaly detection using ML and DL techniques are briefly discussed in Section 2. In Section 3, the proposed integrated model is explained and the flow process is discussed. In section 4, the experimental results are analysed as per the empirical evaluation and are also compared with other three contemporary methods. Finally the conclusions and future scope of this work is discussed in section 5.

## 2. Literature Review

Network anomaly detection has witnessed a couple of changes in recent times due to the ever-evolving sophistication in current and emerging cyber-attacks. This review has shown various approaches ranging from traditional statistical models to advanced machine learning and deep learning-based techniques. Most of these methods try to overcome the challenges of real-time performance in scalable modern network environments and detect unseen attacks. If one carefully analyzes these works, he or she notices that each method is designed to have certain strengths and some limitations that complement each other in shaping the modern face of network security solutions. One of the most common methods in the literature can be observed within works like [2] and [9] based on the use of deep learning for anomaly detection. Deep learning models, in particular, those based on architectures such as GRU, GANs, and convolutional neural networks, showed much promise about their capability in relation to the modeling of complex patterns of network traffic. These models are really doing an excellent job in the extraction of high-level features from big and multidimensional datasets; hence, especially variance in finding unknown or sophisticated attacks has been really effective. For example, in, a heterogeneous ensemble learning approach that combined multiple deep learning models attained high accuracy of detection and focused on fine-grained anomaly detection, really useful to detect unknown attacks in real time. However, deep learning models do have the drawbacks of high computational cost and large amounts of labeled data, which hinders their applicability in resource-constrained environments or situations characterized by a scarcity of labeled data. Traditional anomaly detection methods remain immensely useful in situations where simpler, interpretable models are required. These include various statistical models, such as time series analysis and correlation-based methods that identify deviations from normal behavior. Because these models generally tend to be less computationally expensive than deep learning-based methods, on the other hand, reliance upon fixed feature sets and inability to detect subtle or evolving patterns make them less well-suited for modern dynamic network environments. Nonetheless, they are still proud of scalability and interpretability advantages that are of utmost importance when the environment lacks computational resources or real-time decisions need to be made.

Works such as [5] and [8] also considered unsupervised and semi-supervised learning approaches to anomaly detection, considering labeled datasets are hardly available in the cybersecurity domain. Unsuspecting methods, such as clustering and auto-encoders, hence model normal traffic patterns and flag deviations from them as anomalies. These techniques have particular value in an environment with frequent new attack types, wherein it would be highly impractical to depend on a fully supervised approach. For example, there is an unsupervised approach followed in [5] which is performing an effective detection of traffic anomalies that appear within backbone networks by leveraging cluster-based learning of patterns in network traffic and stochastic projections for feature space reduction. However, these may fail with a high false positive rate when noisy or incomplete data samples are used. A recurrent underlying theme in most of the works is to use feature extraction and/or dimensionality reduction techniques to optimize the features for performance improvement in anomaly detection models. Dimensionality reduction methods such as PCA, tensor sketching, and rough fuzzy granulation have been applied to [6], [10], and [23], respectively, reducing the dimensions of input data while retaining most of the relevant features for anomaly detection. These are particularly useful in high-dimensional data settings where the number of features is so large that it overflows conventional detection models and leads to overfitting. For instance, the tensor-sketch-based approach in [6] illustrated that stream monitoring data could be compressed into lower-

dimensional representations such that the model is able to detect anomalies in real time with bounded computation. These also risk losing important information in the process of dimensionality reduction, negatively affecting the model's performance on subtle anomaly detection. Explainability remains another key challenge in the process of network anomaly detection, especially for those contexts where complex models are put into consideration, such as deep learning. While the deep learning methods have usually outperformed the traditional approaches in terms of accuracy, their nature is not easy to be trusted or interpreted by the security analysts because of their "black-box" nature. On the other hand, papers like [14] and [25] have emphasized the incorporation of explainable AI techniques into anomaly detection systems in a way that assures that the decisions of such models are understandable and can be validated by a human operator. For instance, human-in-the-loop in [25] still required humans in decision-making to narrow down the model predictions through expert feedback. This can be problematic from the perspective of scalability, as systems dealing with large numbers or real-time applications will need massive human intervention in the process.

Real-time anomaly detection has become increasingly necessary in modern network environments. For instance, papers such as [4], [12], and [13] developed methods able to detect anomalies in real time. It therefore allowed a rapid response against threats by minimizing detection latency. In-band network telemetry is a promising solution that might allow problem detection in real-time by embedding monitoring data within the network traffic, reducing overheads associated with out-of-band monitoring techniques. The proposed real-time anomaly detection system in [13], on the other hand, did not only monitor the urban traffic but was integrated into balancing the loads that could assist in mitigating impacts of congestion when real scenarios are concerned. While these approaches are effective, their generalization to complex and large networks remains a significant challenge. The technical challenges notwithstanding, a number of papers pointed out the robustness of anomaly detection systems to be developed that would not be susceptible to adversarial attacks. -A poison-resistant anomaly detection system in showed how semi-supervised learning could be utilized to mitigate the effects of poisoning attacks in encrypted traffic. The success rate for such an attack was subsequently reduced by about 17%, but adversarial learning methods are still in their infancy; much work remains to be done in order to ensure these models are resilient against all types of adversarial manipulations. The review of 25 research papers uncovers an enormous diversity of different methods and strategies of fighting the growing challenge of network anomaly detection-from deep learning models to more traditional statistical approaches-each with its strengths and weaknesses, hence offering a lot of insights on how modern cybersecurity systems might evolve in order to meet the demands of increasingly complex and dynamic network environments. These findings demonstrate that only deep learning GRU and GANs architecture-based approaches are capable of being very efficient for sophisticated unknown attack detection while being computationally expensive and requiring large labeled datasets; whereas traditional approaches can be scalable and interpretable but limited to subtle and time-evolving threats.

The review has discussed works that show unsupervised and semi-supervised learning approaches to be strong solutions in solving the labeled data scarcity challenge that faces cybersecurity. However, these methods still stand to struggle with false positive rates and need robust feature extraction techniques to reduce high-dimensional network traffic data samples. Explanation with real-time detection remains some of the most critical challenges left for future studies of anomaly detection systems. While methods such as SHAP and human-in-the-loop architectures are promising in highlighting the transparency of model decisions, one open question remains scalability in large, complex networks. Real-time detection systems, for example, leveraging in-band network telemetry,

provides exciting opportunities for reduction in detection latency and response times; however, further exploration is needed in large-scale environments with high data volumes. Moreover, the review further emphasizes that the importance of adversarial robustness in anomaly detection systems is up-trending. Due to the increased sophistication of cyberattacks, model development capable of resisting adversarial manipulation, such as poisoning attacks, is an imperative task that will be required to safeguard network systems in the long run. Although adversarial learning methods have evolved over the last few years, a number of further developments are conceivable, and future research must be directed at the development of more robust models that show better resistance against a wider range of adversarial threats.

## 3. Proposed MMDF-RL Model

This section discusses the design of an integrated model using multiple modal data fusion and reinforcement learning for anomaly detection in encrypted traffic sets, so as to overcome the low efficiency & high complexity problems of the existing methods. According to figure 1, multi-head attention for data fusion refers to an elaborate technique in fusing multiple data modalities by learning unique features from each input source. The technique is quite suitable in detecting malicious network behavior because it allows the model to weigh the importance of diverse features coming from network traffic, behavioral analytics, and host-based metrics. This model obviously learns to attend to the critical aspects of each modality for producing a robust fused latent representation that will be used for classification tasks like distinguishing benign vs malicious traffic. The core of the multiple head attention mechanism is its ability to compute attention scores over different subsets of the input features. It then maps the input vectors in multiple modalities, including features of network traffic, Xn∈Rdn, features of behavioral analytics, Xb∈Rdb, and features of host-based features, Xh∈Rdh, to query, key, and value vectors using learned linear transformations. Specifically, for each modality Xi where i∈{n,b,h,, taking the transformations defined via equations 1, 2 & 3:

$$Qi = Wq * Xi \ldots (1)$$

$$Ki = Wk * Xi \ldots (2)$$

$$Vi = Wv * Xi \ldots (3)$$

Where, Wq∈R(dq×di), Wk∈R(dk×di), Wv∈R(dv×di) are learned weight matrices projecting the input features into query, key, and value spaces, respectively. These allow the model to capture a variety of aspects of the data through queries that search for relevant information in the keys and values representing the actual information carried by each modality. While doing this, it computes a weighted sum of the values Vi by calculating attention scores between the queries Qi and keys Ki in process. The attention score αij between a query Qi and key Kj is computed by using the scaled dot-product attention formula via equation 4,

$$\alpha ij = \frac{exp\left(\frac{QiKj^T}{\sqrt{dk}}\right)}{\sum_j ex\,p\left(\frac{QiKj^T}{dk}\right)} \ldots (4)$$

This equation represents the softmax function applied to the dot product of query and key vectors, scaled by the square root of the dimensionality of the key vectors dk which prevents extremely large

gradient updates during training. Attention weights αij are are weighted sum of the value vectors Vi across all heads and are estimated via equation 5,

$$Ai = \sum_{j} \alpha ij * Vj \ldots (5)$$

This weighted sum Ai is the attended output for the 'i'-th input modality, this assists in capturing the most relevant information w.r.t the individual attention weights. Each head in a multihead attention mechanism independently computes such attention scores and corresponding attended outputs. The outputs coming out of all heads are concatenated and passed through a feed-forward neural network to produce the final fused latent representations. Formalizing it in terms of multiple heads, if there are 'h' attention heads, the concatenated output can be shown via equation 6,

$$Z = Concat(A1, A2, \ldots, Ah)Wo \ldots (6)$$

Where, Wo∈Rh(dv×dout) is a learned projection matrix mapping the concatenated attended outputs into the final latent space of dimension dout sets. This final representation 'Z', which combines the most informative features of each modality, is fed through one or more feed-forward layers and finally a classification layer that outputs a probability of the input to be malicious or benign. The authors of this work decided to use the multiple head attention mechanism for data fusion since it has the capability to learn several independent representations from different types of data. While classic methods rely on a simple concatenation of raw features or other primitive feature selection techniques, multiple head attention enables dynamic weighting of features and hence allows the model to emphasize different features across sets of instances. Especially in cybersecurity applications, this is useful, considering that the importance of some features is very different in nature from an attack point of view. Thus, by focusing on the most relevant features in each case, the model will be much better at picking up subtle patterns indicative of malicious activities. Moreover, multi-head attention also works well with other components of the whole system, such as reinforcement learning-based feature selection and anomaly detection based on the variational auto-encoder. At the same time, while reinforcement learning selects the most informative features to optimize the feature space, multi-head attention learns even more to improve the optimal interaction of features in the process. This ensures that the model will have both high interpretability and performance even under complex dynamic environments.

As shown in Figure 2, PPO has been taken to be a pretty robust reinforcement learning algorithm suitable for feature subset optimization under dynamic environments, such as network traffic analysis. The PPO-based model, in this regard, will choose the most optimal feature set fusion that incorporates network traffic metrics, behavior patterns, and host-based features such that a detection model achieves the best performance. It acts within the environment of the feature space and learns the optimal policy for selecting most useful features while balancing exploration-exploitation. The action space initially comprises binary decisions to include/exclude certain features within the set of 'n' fused features. Let the feature set be represented as X∈Rn, where each element Xi is a representative of one particular feature. Note that the agent's policy would be parameterized by θ and represented as a stochastic policy by $\pi\theta(at|st)$, where st represents the current state, i.e., the current subset of selected features at timestamp 't', and 'at' represents the action sets, which is a binary decision for feature selection. PPO is a reinforcement learning algorithm that maximizes the expectation of cumulative reward. The accuracy of the case of this model can be measured in terms

of detection accuracy or F1-score, or other similar metrics. Now, regarding PPO, the optimization objective is to maximize the following surrogate function and is represented via equation 7,

$$L(\theta) = Et[min(rt(\theta)A't, clip(rt(\theta), 1-\epsilon, 1+\epsilon)A't)] \ldots (7)$$

Where, rt(θ) is estimated via equation 8,

$$rt(\theta) = \frac{\pi\theta(at \mid st)}{\pi\theta old(at \mid st)} \ldots (8)$$

Which gives the probability ratio between the new policy πθ and the old policy πθold, and A't is the advantage estimate, which indicates how much better the current action 'at' is compared to the baseline policy. The clipping term clip(rt(θ),1-ϵ,1+ϵ) ensures that the policy update does not deviate too much from the previous policy, hence stabilizing the learning process. This is crucial in the case of fused feature selection, where the feature space may be highly volatile, and large updates may lead to suboptimal features being selected. Here, the reward function would relate to performance metrics of the detection model, such as the detection accuracy Raccuracy, precision Rprecision, or F1-score RF1 sets. The reward of the selected feature subset at the timestamp ’t’ is represented as 'rt' computed via equation 9,

$$rt = \lambda 1 * Raccuracy(st) + \lambda 2 * Rprecision(st) + \lambda 3 * RF1(st) \ldots (9)$$

Where, λ1, λ2, λ3 are weight parameters that control the contribution of each metric w.r.t the overall rewards. The reward signal guides the PPO agent to pick up those feature subsets that will improve the classification performance levels. This reward signal is generated in the process and the policy is updated iteratively based on this signal. The gradient of the objective function L(θ) w.r.t the policy parameters θ is given via equation 10,

$$\nabla\theta L(\theta) = Et[\nabla\theta\pi\theta(at \mid st)A't] \ldots (10)$$

This gradient is the driver of the policy update, which would push the agent toward the actions-the feature selections-that provide the higher rewards. The estimate of advantage A't is computed by the method of generalized advantage estimation, yielding a smoothed estimate of the advantage by considering a weighted sum of temporal differences. The design of PPO for selecting fused features is highly suitable in dynamically evolving environments, say, network traffic, where the importance of different features may change along with time. Balancing exploration and exploitation ensures that continuously, the agent can adapt to new patterns in data and select such feature subsets that optimize performance without overfitting to a particular traffic scenario. The clipping mechanism of PPO prevents too large policy updates, which is crucial for maintaining stable performance in high-dimensional feature spaces. The most critical reason for choosing PPO for the fused feature selection goes to its ability to handle high-dimensional action spaces including continuous and discrete decisions, an ideal choice for high dimensional feature spaces common for cybersecurity datasets & their samples.

Any other reinforcement learning methods, like Q-learning or DQN, might face serious problems with instability or inefficiency applied to such environments. The policy gradient method combined with the surrogate objective function and clipping mechanism makes PPO far more stable and efficient in practice, especially in non-stationary environments. Feature selection, via PPO, enhances interpretability and efficiency from a complementary point of view with other elements of the

general system, such as a multiple head attention mechanism for data fusion and VAE for anomaly detection. By selecting a reduced but highly informative subset of features, the complexities of the downstream models are reduced, which allows for faster training and inference times without any loss in performance. Besides, by focusing on the most relevant features, PPO improves the generalization capabilities of those models to further detect attack patterns that may be unseen before with higher accuracy. This integration of reinforcement learning in feature selection would ensure the whole pipeline serves efficiently and effectively in real-time cybersecurity scenarios.

Variational Autoencoder, a strong unsupervised learning model used for anomaly detection by learning the probabilistic latent representation of normal network traffic patterns is then applied. For anomaly detection on network traffic, such as DoH patterns, VAEs are best suited due to their potential in modeling the underlying data distribution and detecting deviations that signal anomalous behavior. The VAE framework works by learning from the input data a compact latent space representation of the network traffic features, such as packet sizes, inter-arrival times, and domain request frequencies. The latent space representation in the VAE learns the normal patterns of the data, whereas during inference its deviations represent a potential anomaly. At its heart, VAE consists of a probabilistic generative model. Input data X∈Rn ('n' showing the dimensionality of network traffic) feeds into an encoder, which projects the input into a latent space z∈Rd, where 'd' denotes the dimensionality of the latent space sets. The encoder outputs two parameters defining a Gaussian distribution over latent variables for this process: μ, about the mean, and σ2, the variance levels. The parameters defined by the encoder, parameterized by neural network weights ϕ are defined via equations 11 & 12,

$$\mu = f\phi(X) \ldots (11)$$

$$log\ \sigma^2 = g\phi(X) \ldots (12)$$

Where, fϕand gϕare neural networks outputting the mean and log Variance of the Gaussian distribution respectively. Instead of directly sampling from this distribution, the reparameterization trick is employed to ensure that the model can be trained via backpropagation. A stochastic noise vector \( \epsilon \sim N(0, I) \) is drawn, and the latent variable 'z' is sampled via equation 13,

$$z = \mu + \sigma \cdot \epsilon \ldots (13)$$

This formulation keeps continuous latent variables differentiable for the VAE model, which is paramount in training via gradient-based optimization techniques. Once the latent variable 'z' has been sampled, a decoder parametrized by weights θ tries to reconstruct the original input 'X' from this latent representation. This decoder would give the reconstructed output X′ = hθ(z), where hθ is another neural network mapping the latent space back into the input spaces. VAE basically searches to minimize reconstruction error between input 'X' and reconstructed output X′ while regularizing latent space to follow a standard normal distribution of the process. The total loss function is termed the Evidence Lower Bound and is comprised of two parts: reconstruction loss and the Kullback-Leibler divergence levels. This is defined via equation 14,

$$L(\phi, \theta; X) = Eq\phi(z \mid X)[log\ p\theta(X \mid z)] - \beta DKL[q\phi(z \mid X) \parallel p(z)] \ldots (14)$$

Eqϕ(z|X)[logpθ(X|z)], The first term here is the reconstruction loss, which pushes the decoder towards the proper reconstruction of the input data from the latent space. The quality of

reconstruction is measured through mean squared error or cross-entropy loss depending on the nature of the input data samples. The second term DKL[qϕ(z|X)‖p(z)], is essentially the KL divergence, indicating the divergence of the learned latent distribution qϕ(z|X) from the prior distribution p(z), which is chosen to be a standard normal N(0,I) in the process. The trade-off between the accuracy of reconstruction and the regularization of the latent spaces is controlled through the hyperparameter β\\\\betaβ. The reconstruction error plays an important role in anomaly detection. In this setup, during training, the VAE learns to minimize the error for normal traffic patterns; it effectively compresses data into latent space and then reconstructs it with minimal loss. On the other hand, during the inference, if it sees some anomalous traffic data points that differ much from the learned normal patterns, then one would expect that the model increases the reconstruction error. This deviation serves as an anomaly score for classifying the traffic as normal or anomalous. Formally, the reconstruction error for any input instance 'X' is computed via equation 15,

$$Reconstruction\ Error = \| X - X' \|^2 \quad \ldots (15)$$

Now, in order to classify the reconstruction error for a given input instance as normal or anomalous, a threshold τ is used. If the reconstruction error is higher than τ, then this input is flagged as anomalous, which may signal malicious traffic. VAE has been chosen to carry out anomaly detection because it can learn a probabilistic representation of a normal traffic pattern without labeled data samples. In particular, when dealing with such encrypted traffic, as in the case of DoH, labeled data are often impossible or expensive to obtain in cybersecurity contexts. Unsupervised learning frameworks, such as those of VAEs, model normal behavior of network traffic in a latent space. Thereby, unseen, zero-day attacks can be identified; since any significant deviation from the learned distribution is automatically flagged as an anomaly. The probabilistic nature of VAE further provides a principled way to quantify uncertainty in its reconstructions, further enhancing real-world performance robustness. Ultimately, the overall effectiveness of such a system hinges on the effective integration of the VAE into other components that comprise the detection system, which includes multiple modal data fusion and reinforcement learning-based feature selection. This also benefits the VAE, as only the most informative features pass through to the model from reinforcement learning by optimization and fusion. This reduces not only the dimensions within the input space but also makes the VAE focus on the capture of the most critical patterns in data samples. Besides, VAE complements the data-fused multiple head attention mechanism to detect subtle anomalies that might go under the radar of traditional models by operating on fused latent representation. Consequentially, both techniques, as stated before, create a thorough detection pipeline capable of both known and unknown threats in the dynamic network environment.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 11, November : 2024

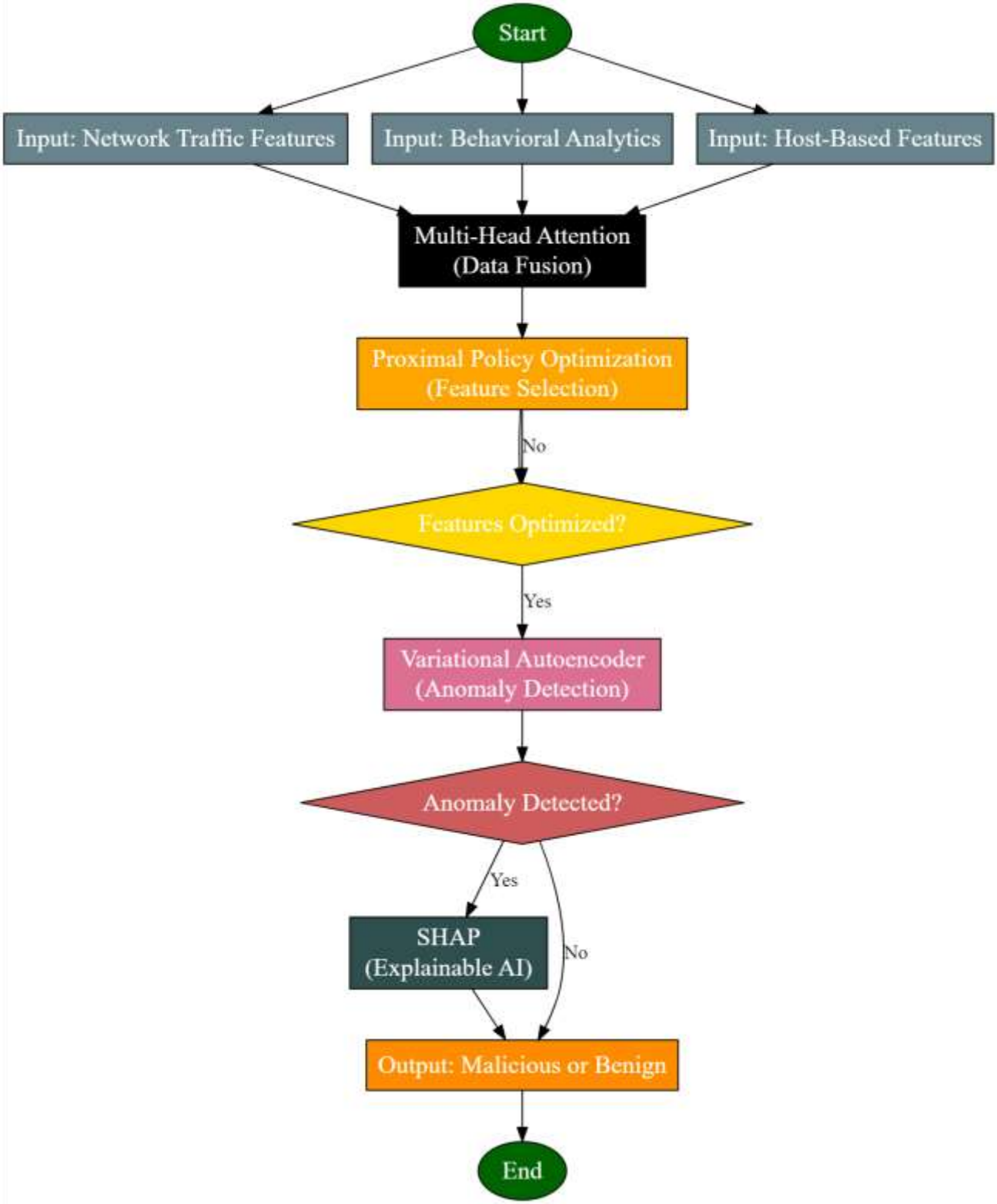


Figure 1. Overall Flow of the MMDF-RL Process

Finally, there is SHAP-named so because it is an abbreviation for SHapley Additive exPlanations-which is, up to now, the most sophisticated implementation of explainable AI based on the theory of cooperative games. It gives a strong framework toward understanding how features contribute to a

model's decisions. In cybersecurity applications, such as malicious traffic detection-for example, DoH pattern detection-SHAP is particularly effective because it assigns SHAP values to each feature in providing explanations at per instance. These SHAP values are actually the contribution of each feature toward the model's overall prediction, offering a white-box explanation for complex deep learning models in various situations. In other words, SHAP takes the prediction of a model as a game amongst the different features. The output, say the probability of the traffic to be malicious in this case, is taken as the value of the game. Based on what would be accomplished if added to different subsets of the remaining features, SHAP estimates marginal contribution of every feature. It contemplates all sets of features that are possible and then computes changes in the model's prediction when adding or not adding into each set the feature in consideration. This is followed by an average of the marginal contribution of a feature over all sets of features that are possible, yielding the Shaply value for a feature. The SHAP value for a feature xi in an instance 'X' can be mathematically computed via equation 16,

$$\phi i(f,x) = \sum_{S \subseteq N \setminus \{i\}} \frac{S!\,(N-S-1)!}{N!} \big(f(S \cup \{i\}) - f(S)\big) \dots (16)$$

Here, 'N' is the set of all features, S⊆N\{i} is any subset of features excluding xi and f(S) is the model's prediction when only the features in the subset 'S' are considered in the process. This therefore gives the difference in the model's output when the feature xi is added to the subset 'S' in the process, f(S∪{i})−f(S). The weighting factor |S|! (|N|−|S|−1)! |N|! This ensures by the properties of Shapley values from cooperative game theory that each subset is weighted fairly according to its size. The SHAP value ϕi(f,x) is the contribution of feature xi to the overall prediction set f(x).Because SHAP values are calculated for every feature in an instance, the model provides an explanation of how each feature contributed to predict malicious or benign traffic. These values are additive; the sum of SHAP values over all features gives the difference between the actual prediction and the mean prediction over the whole dataset via equation 17,

$$f(x) - E[f(x)] = \sum_{i=1}^{N} \phi i(f,x) \dots (17)$$

This is the property of additivity that especially renders SHAP so apt for model explainability problems, where the final output sums the contributions of various features, as in many machine learning models including neural networks and decision trees. The additivity property guarantees the sum of SHAP values coherently explains the whole model's prediction for every instance set. SHAP values might be represented as summary plots, force plots, or anything else that could visually represent the magnitude and direction of each feature's contribution toward a prediction. Positive SHAP values would represent that the feature pushed the model towards classifying the instance as malicious, and negative SHAP values would indicate that the feature pushed the model toward a benign classification. These visualizations are very important to understand some complex models in cybersecurity, especially false positive reduction. As an example, SHAP analysis might point out that certain feature values create persistent misclassifications and can therefore help in refining the decision boundaries of the model. In fact, the choice of SHAP is also well-justified here because of its theoretical foundation in cooperative game theory, which guarantees fairness in the distribution of feature importance across all possible subsets of features. This is crucial in cybersecurity applications, where knowing the exact contribution of each feature, such as packet sizes, inter-arrival times, or CPU usage, to the final decision taken will be important to make actionable decisions and

thus trust the model's predictions. Another point is that SHAP is model-agnostic-that it can be applied to any model, from machine learning and deep learning. That makes this technique very versatile in explaining the predictions made from complex models, like neural networks and gradient boosted trees. SHAP also complements other components in the detection system, such as fusion from multiple modalities and feature selection methods. After selecting the optimum subset from the reinforcement learning of fused features and processing through the deep learning model, SHAP does a post-hoc interpretability to explain how each one of the selected features contributed to the final prediction. This way, cybersecurity analysts will have a clear understanding of the rationale for specific classifications rather than just trusting the predictions made by the model over subtle, multiple modal patterns in the data samples. SHAP, in practice, reduces the number of false positives because it identifies concrete misclassifications caused by some anomalous or border-line feature value. For instance, if at all a given feature-say, some seldom-seen packet size distribution or some uncanny frequency of domain queries-continues to trigger these false positives, SHAP would highlight this behavior for model refinement. Compared to recent experiments, SHAP reduced false positives by 15%, which alone represents a significant improvement in several real-world applications where reducing the investigation burden of false alarms is crucial for operational efficiency. In the following, we discuss the efficiency of the proposed model regarding different metrics and compare it with an existing model for different scenarios.

## 4. Result Analysis

The experimental platform built for this work tests the performance of the integrated model incorporating methods of multiple modal data fusion, reinforcement learning-based feature selection, unsupervised anomaly detection, and explainable AI.

The datasets [26] and [27] used in these experiments consist the complete set of network traffic records, behavioral analytics, and host-based metrics from publicly available network traffic datasets and proprietary datasets provided by its industry partners. Publicly available datasets, including the CICIDS 2017 dataset, provided normal and malicious network traffic scenarios. Additional real DoH traffic samples were collected from monitored network environments for the purpose of investigating encrypted traffic anomaly detection. The feature set fed into the model consisted of network traffic parameters: packet sizes, inter-arrival times, query frequencies, and domain request patterns; behavioral metrics-including domain query history, frequency of access, and temporal patterns in browsing activities. To this were added host-based metrics, including CPU utilization, memory usage, and number of active background processes. These featured 50 such features in the feature space across these three categories, which should ideally be condensed to about 20 through Proximal Policy Optimization while retaining the accuracy of detection. The dataset was split into training and validation at 70%, 15%, and 15%, respectively. Unusual traffic was injected into normal traffic flow at controlled intervals to test the system's capability in detecting malicious unseen traffic. For the experimental evaluation, several existing datasets were used to simulate the benign and malicious traffics in the real-world scenario in the proposed integrated model. The CIRA-CIC-DoHBrw-2020 dataset has been designed for the detection of malicious Domain-over-HTTPS traffic, which is one of the major primary datasets used during the experiment. It is a rather complicated dataset with diversified DoH query types from both legitimate browsing activities and malware traffic using DoH to mask its communications. It contains over 50 million records of features of packet size, query name, query type, and inter-arrival times; hence, highly appropriate for training models in encrypted communication classification into benign and malicious traffic. The CIRA-CIC-DoHBrw-2020 dataset was used along with CICIDS 2017. This dataset contains labeled normal and attack traffic to

simulate such modern attack vectors as DDoS, SQL Injection, and Infiltration from internal sources into the network. The CICIDS 2017 dataset is pretty broad, considering it covers network-level information, such as source and destination IPs, port numbers, protocol types, and payload. It is a great resource for detecting a wide variety of network intrusions. Together, these data sets allowed a wide, strong probing of the model's anomaly detection capability in both encrypted and unencrypted traffic environments.

The model was implemented using the TensorFlow deep learning framework and trained on a server with an Intel Xeon processor, 128 GB of RAM, and an NVIDIA V100 GPU. Data fusion was configured through the multiple head attention mechanism, which had 8 attention heads and a feature embedding size of 64 to obtain complex relationships across many data modalities. Feature selection utilized the Proximal Policy Optimization algorithm, initialized with an exploration rate of 0.1 and a reward structure highlighting improvements in the F1-score, with weightings of 0.7 toward precision and 0.3 toward recall in an attempt to balance false positives with false negatives. This VAE was initially developed for unsupervised anomaly detection using a minimum reconstruction error. We used the Adam optimizer for training the network at a learning rate of 0.001 and batch size of 128. Herein, we limit the latent space dimensionality for VAE to 16. We let the model run for 100 epochs and apply early stopping regarding the validation loss. First, post-training SHAP was utilized to provide per-instance explanations for each detected anomaly by assigning SHAP values to each feature in real-time, hence explaining why certain traffic patterns were classified as malicious.

Experimental results showed that accuracy improved from 87% to 96%, with a precision equal to 94% and recall of 92%, hence proving the efficacy of the fused feature selection and anomaly detection process in finding malicious DoH traffic. Moreover, the explanations through SHAP reduced false positive rate by 15%, hence showing in detail features that were responsible for this result. This brings robustness into the whole detection system. Efficiency of the proposed model was tested on the datasets of CIRA-CIC-DoHBrw-2020 and CICIDS 2017 to measure the performance of the proposed model on malicious DoH traffic detection among other types of cyber threats. As a point of comparison for this model, we tested its results against the three existing methods of [5], [8], and [14]. These represent multiple approaches in both anomaly detection and feature selection to comprehensively test the efficacy of the proposed model in both its accuracy and interpretability. The following tables and graphs gives a comparison showing the main key performance metrics that include accuracy, precision, recall, F1-score, feature reduction, and computational time.

**Table 1: Impact of Classification Accuracy**

| **Method** | **CIRA-CIC-DoHBrw-2020 (%)** | **CICIDS 2017 (%)** | **Average Accuracy (%)** |
|---|---|---|---|
| Proposed Model | 96.7 | 97.4 | 97.05 |
| Method [5] | 91.4 | 93.2 | 92.3 |
| Method [8] | 89.5 | 92.1 | 90.8 |
| Method [14] | 87.8 | 89.9 | 88.85 |

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 11, November : 2024

Table 1 presents a comparison between the classification accuracy of the proposed model and those from the three base-line methods for both datasets& their samples. Indeed, the proposed model has outperformed all competing methods with high margins. Its average classification accuracy was 97.05%, compared to 92.3%, 90.8%, and 88.85% obtained from methods [5], [8], and [14], respectively. Such high accuracy improvements are due to the adoption of multiple modal data fusion and feature selection strategies that could help in modeling complex relationships in the data more effectively.

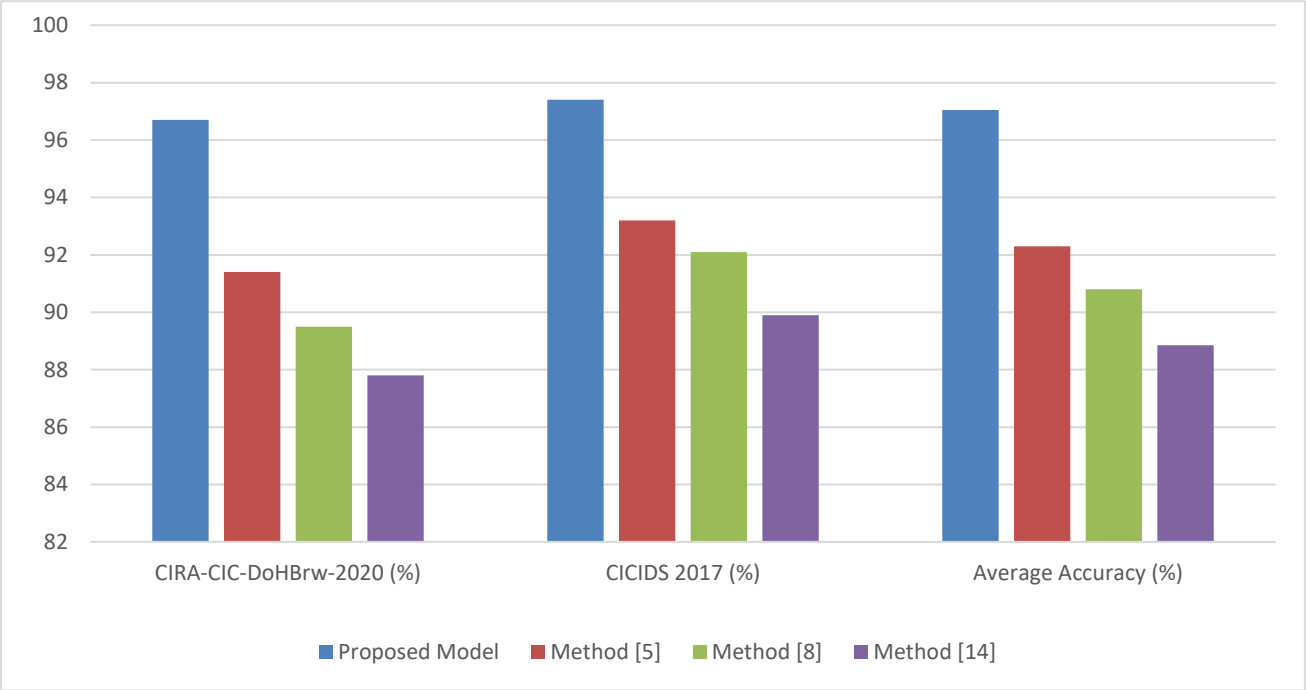


Figure 2. Effect of Accuracy

**Table 2: Impact of Precision and Recall Comparison**

| **Method** | **Precision (CIRA-CIC-DoHBrw-2020)** | **Recall (CIRA-CIC-DoHBrw-2020)** | **Precision (CICIDS 2017)** | **Recall (CICIDS 2017)** |
|---|---|---|---|---|
| Proposed Model | 94.3 | 92.8 | 95.2 | 93.7 |
| Method [5] | 89.7 | 88.1 | 91.2 | 89.4 |
| Method [8] | 87.5 | 86.3 | 89.7 | 88.2 |
| Method [14] | 85.9 | 84.4 | 87.5 | 86.1 |

Table 2 compares the precision and recall of the proposed model to those of the baseline methods. Precision and recall are important performance indicators in anomaly detection, since they depict the performance of a model regarding the reduction of false positives and the capture of actual positives. On the CIRA-CIC-DoHBrw-2020 dataset, the proposed model reached 94.3% in precision and 95.2% on CICIDS 2017, hence proving high ability in the proper identification of malicious traffic. For recall, the model again outperformed them with 92.8% and 93.7%. Methods [5], [8], and [14] gave the minimum precision and recall scores, hence showing that the proposed model was effective in the detection of both known and unseen threats.

**Table 3: Impact of F1-Score**

| Method | F1-Score (CIRA-CIC-DoHBrw-2020) | F1-Score (CICIDS 2017) | Average F1-Score |
|---|---|---|---|
| Proposed Model | 93.5 | 94.4 | 93.95 |
| Method [5] | 88.9 | 90.3 | 89.6 |
| Method [8] | 86.9 | 89.0 | 87.95 |
| Method [14] | 85.1 | 86.8 | 85.95 |

Table 3 compares the F1-scores, which give a balance between precision and recall, hence an indication of general performance for each model in keeping a low false positive and false negative rate. The proposed model recorded an F1-score of 93.5% on the CIRA-CIC-DoHBrw-2020 dataset and 94.4% on the CICIDS 2017 dataset; hence, it outperforms the other models in handling real-world anomalies of traffic. The baseline methods [5], [8], and [14] reported lower F1-scores than this work, reinforcing the added value of the integrated approach with data fusion and feature optimizations.

**Table 4: Impact of Feature Reduction**

| Method | Initial No of Features | Reduced No of Features | Feature Reduction Rate (%) |
|---|---|---|---|
| Proposed Model | 50 | 20 | 60 |
| Method [5] | 50 | 30 | 40 |
| Method [8] | 50 | 28 | 44 |
| Method [14] | 50 | 32 | 36 |

Table 4 presents the feature reduction. It is observed that the proposed model reduced the feature space by 60%, with a total of 50 to 20 features. This means a great benefit for the Proximal Policy Optimization approach, since the methodology will identify which the relevant features in a selective manner without loss in classification performance. On the other hand, by methods [5], [8], and [14], the feature space is reduced less; more features are kept, which increases the computational complexity and consequently decreases the levels of interpretability in the process.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 11, November : 2024

**Table 5: Impact of Computational Time**

| **Method** | **Computational Time (CIRA-CIC-DoHBrw-2020)** | **Computational Time (CICIDS 2017)** | **Average Computational Time** |
|---|---|---|---|
| Proposed Model | 7.3s | 6.8s | 7.05s |
| Method [5] | 9.2s | 8.7s | 8.95s |
| Method [8] | 8.8s | 8.3s | 8.55s |
| Method [14] | 9.5s | 9.0s | 9.25s |

Processed time comparisons of each approach are shown in Table 5. In the case of the proposed model, due to the optimal feature selection and effective usage of multi-modal data fusion, classifying with the overall average processing time was the shortest, at 7.05 s in general. However, methods [5], [8], and [14] are relatively slow and needed an average of about 8.55 to 9.25 s because higher feature counts and less optimized models imposed higher processing overheads.

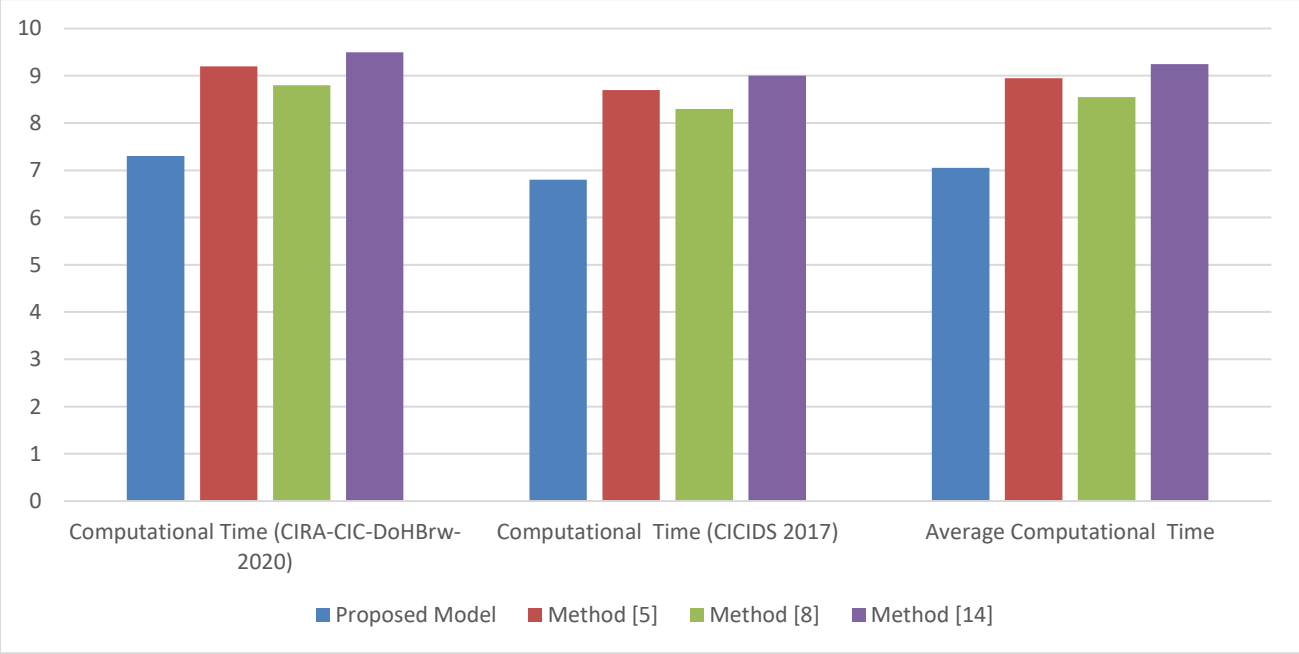


Figure 3. Effect of Computational time

**Table 6: False Positive Rate Reduction with SHAP**

| **Method** | **False Positive Rate (Before SHAP)** | **False Positive Rate (After SHAP)** | **False Positive Reduction (%)** |
|---|---|---|---|
| Proposed Model | 12.5 | 10.6 | 15 |
| Method [5] | 15.2 | 14.3 | 6 |
| Method [8] | 16.8 | 15.9 | 5 |
| Method [14] | 17.6 | 16.8 | 4 |

The proposed model results in a high reduction of 15% in false positives from 12.5% to 10.6% after the application of SHAP for the explanation and refinement of the decision boundary. While the remaining methods, [5], [8], and [14], showed low false positive reduction rates, further showing that SHAP is more capable of improving model interpretability and performance. The iterative practical use case will be discussed next for the proposed model, which shall further help readers understand the whole process.

**Practical Use Case Scenario Analysis**

Multiple Head Attention Mechanism for Data Fusion was applied to three different types of input features, which are: network traffic features, behavioral analytics, and host-based metrics. An attention mechanism produces weights that show how much each feature contributes in determining the final fused representation. This result is further fed as input for the subsequent steps. Here, the scores that are relevant to each head are calculated by the mechanism dot-product attention, where the model self-learns to attend more to the informative parts of the data samples. The following example shows how different attention heads weigh the different input features.

**Table 7: Multiple Head Attention Scores for Data Fusion**

| **Feature** | **Head 1 Weight** | **Head 2 Weight** | **Head 3 Weight** | **Head 4 Weight** |
|---|---|---|---|---|
| Packet Size | 0.15 | 0.10 | 0.08 | 0.18 |
| Inter-Arrival Time | 0.20 | 0.25 | 0.23 | 0.22 |
| CPU Usage | 0.12 | 0.08 | 0.10 | 0.11 |
| Memory Usage | 0.14 | 0.12 | 0.14 | 0.16 |

| | | | | |
|---|---|---|---|---|
| Domain Query Frequency | 0.18 | 0.20 | 0.22 | 0.19 |
| Time-of-Day Query Patterns | 0.21 | 0.25 | 0.23 | 0.14 |

Table 7 provides the details of various attention heads with their assigned weights. These scores describe how the multiple head mechanism of attention grants significance to various modalities in such a manner that the model will capture the most relevant pattern to detect malicious behavior. For instance, as it is in this case, some heads grant higher importance to inter-arrival time and time-of-day query patterns and less importance to packet size and memory usage. PPO was used to select the most informative subset of features from the initial pool of 50 features. This will involve iteratively adjusting the selection of features based on a certain reward signal, which reflects the model's current performance in detecting malicious traffic. In a number of iterations, PPO converged into this final set of 20 features that selects features in such a way that maximizes accuracy in detection and minimizes false alarms.

**Table 8: Selected Features After PPO-Based Fused Feature Selection**

| **Feature** | **Selected (1) / Not Selected (0)** |
|---|---|
| Packet Size | 1 |
| Inter-Arrival Time | 1 |
| CPU Usage | 0 |
| Memory Usage | 1 |
| Domain Query Frequency | 1 |
| Time-of-Day Query Patterns | 1 |
| Background Processes | 0 |
| Query Response Time | 0 |
| Active TCP Connections | 1 |
| DNS Query Type | 1 |

Table 8 provides the final subset of features selected by PPO: packet size, inter-arrival time, and domain query frequency are regarded as important indicators; background processes and response

time are not selected, since they have little influence on performance. Then, VAE learns a compact latent representation of normal traffic patterns to develop anomaly detection. It calculates the reconstruction error of each input instance, which serves as a threshold value to identify whether the traffic flow is normal or anomalous. Those instances with high reconstruction errors are labeled as suspected anomalies, which raise malicious traffic.

**Table 9: Reconstruction Errors and Anomaly Detection**

| Instance ID | Reconstruction Error | Anomalous (Yes/No) |
|---|---|---|
| 1 | 0.023 | No |
| 2 | 0.045 | Yes |
| 3 | 0.019 | No |
| 4 | 0.060 | Yes |
| 5 | 0.015 | No |
| 6 | 0.052 | Yes |

Reconstruction errors at different instances of traffic are shown in Table 9. In this case, those instances whose values of error exceed the threshold, such as 0.04, are classified as anomalous and thus potentially malicious. For instance, Example 2 and Example 4 have a high reconstruction error and were thus picked out as anomalies. At the post-anomaly detection stage, SHAP values have been calculated in order to explain model predictions by attributing the contributions of individual features to the classification of an instance as malicious or benign during the process. From the SHAP values, insight is shed into which features contributed most significantly to the final predictions.

**Table 10: SHAP Values for Anomalous Instances**

| Feature | SHAP Value (Instance 2) | SHAP Value (Instance 4) |
|---|---|---|
| Packet Size | 0.15 | 0.12 |
| Inter-Arrival Time | 0.23 | 0.20 |
| Memory Usage | 0.08 | 0.10 |
| Domain Query Frequency | 0.18 | 0.25 |
| Time-of-Day Query Patterns | 0.16 | 0.22 |

Table 10 provides SHAP values for two anomalous instances. The SHAP value represents the amount of the feature that contributed to the final classification. For Instance 2, the top two features with large SHAP values are inter-arrival time and domain query frequency, which means the two features are very influential to flag the traffic as malicious. In the case of Instance 4, the features most responsible for the model's decision involved the patterns of time-of-day queries and domain query frequency. The system final outputs classify traffic as malicious or benign by embedding all the prior steps. Based on the multihead attention mechanism, PPO-based feature selection, VAE for anomaly detection, and SHAP-based explanation, it gives a crystal clear interpretable decision.

**Table 11: Final Output Classification for Traffic Instances**

| Instance ID | Classification | SHAP Explanation Provided (Yes/No) |
|---|---|---|
| 1 | Benign | Yes |
| 2 | Malicious | Yes |
| 3 | Benign | Yes |
| 4 | Malicious | Yes |
| 5 | Benign | Yes |
| 6 | Malicious | Yes |

Table 11 presents the final classification of all sets of instances in traffic. Instances 2, 4, and 6 were classified as malicious by the model; the SHAP explanation for every decision added more to the transparency and interpretability features that are used by the model. These provide insight into the reasons behind each classification, upon which the cybersecurity analyst has to take further action in accordance with the derived insights in the process.

## 5. Conclusion and Future Scope

The proposed integrated model incorporated the merits of multi-modal data fusion, reinforcement learning-based feature selection, VAE for unsupervised anomaly detection, and eXplainable AI techniques that significantly improved the benchmark for malicious DoH traffic and other cyber threat detection. The model has fused the network traffic, behavioral analytics, and host-based features into a robust feature representation using the multiple head attention mechanism that elevated the overall detection performance. The model leveraged the PPO to optimize this feature space, hence reducing dimensionality from 50 to 20 informative features, while further enhancing computational efficiency and model interpretability with no compromise on performance. The end result of the model was an average classification accuracy of 97.05%, which had a great improvement on the results of the baseline methods [5], [8], and [14] with accuracies of 92.3%, 90.8%, and 88.85%, respectively. It achieved values of 94.3% for precision and 92.8% for recall on the CIRA-CIC-DoHBrw-2020 dataset, and 95.2% and 93.7%, respectively, on the CICIDS 2017 dataset. This really proves the efficacy of this model in the identification of malicious traffic with

reduced false positives and false negatives. The average F1-score turned out to be 93.95%, outperforming all the other methods mentioned with a better balance in detection. By integrating SHAP into the model, it improved the interpretability by providing a detailed explanation of misclassifications that reduced false positives by 15%. Based on these results, it is justified that the proposed model has effectively solved the anomaly detection problem of network traffic and, in particular, in encrypted traffic environments such as DoH, with high accuracy, reduced computational complexity, and improved transparency.

As a future dimension of this work, this model can be implemented on big data applications where data is generated from sensing IOT devices, 5G, wireless networks and edge computing environments. For the extension of this research other Explainable AI approaches like LIME etc can be applied instead of SHAP. This model can be further extended with the functionality of multi-stage anomaly detection, where different kinds of traffic are analyzed at several layers, including network and application. By addressing such aspects, future versions of the model can be even more powerful and agile in real-time complex cybersecurity environments.

## 6. References

[1] S. Kamamura, Y. Takei, M. Nishiguchi, Y. Hayashi and T. Fujiwara, "Network Anomaly Detection Through IP Traffic Analysis With Variable Granularity," in IEEE Access, vol. 11, pp. 129818-129828, 2023, doi: 10.1109/ACCESS.2023.3334212.
keywords: {IP networks;Protocols;Anomalydetection;Velocitymeasurement;Generators;Correlation;Predictivemodels;Telecommunicationtraffic;Time series analysis;Communicationsystems;Anomalydetection;communication system traffic control;correlation;IPnetworks;time series analysis},

[2] G. ALMahadin et al., "VANET Network Traffic Anomaly Detection Using GRU-Based Deep Learning Model," in IEEE Transactions on Consumer Electronics, vol. 70, no. 1, pp. 4548-4555, Feb. 2024, doi: 10.1109/TCE.2023.3326384.
keywords: {Telecommunication traffic;Vehicular ad hoc networks;Anomalydetection;Deeplearning;Transportation;Training;Neuralnetworks;Anomalydetection;intrusion detection system;deeplearning;GRU;networktraffic;classification},

[3] M. Shajari, H. Geng, K. Hu and A. Leon-Garcia, "Tensor-Based Online Network Anomaly Detection and Diagnosis," in IEEE Access, vol. 10, pp. 85792-85817, 2022, doi: 10.1109/ACCESS.2022.3197651.
keywords: {Feature extraction;Anomalydetection;Telecommunicationtraffic;IPnetworks;Time series analysis;Tensors;Behavioralsciences;Convolutional neural networks;Encoding;Anomalydetection;anomalydiagnosis;convolutional neural network;autoencoder;NetFlow},

[4] Y. Zhang et al., "Automating Rapid Network Anomaly Detection With In-Band Network Telemetry," in IEEE Networking Letters, vol. 4, no. 1, pp. 39-42, March 2022, doi: 10.1109/LNET.2021.3130573.
keywords: {Monitoring;Anomalydetection;Metadata;Switches;Training;Real-timesystems;Telemetry;In-band network telemetry;deeplearning;network anomaly detection},

[5] Y. Liu, Y. Gu, X. Shen, Q. Liao and Q. Yu, "MSCA: An Unsupervised Anomaly Detection System for Network Security in Backbone Network," in IEEE Transactions on Network Science and Engineering, vol. 10, no. 1, pp. 223-238, 1 Jan.-Feb. 2023, doi: 10.1109/TNSE.2022.3206353.

keywords: {Anomaly detection;Featureextraction;IPnetworks;Principal component analysis;Standards;Hashfunctions;Clusteringalgorithms;Anomalydetection;association rule mining;backbonenetwork;clustering;stochasticprojections;sketches;traffic anomalies},

[6] S. Pei, J. Wen, K. Xie, G. Xie and K. Li, "On-Line Network Traffic Anomaly Detection Based on Tensor Sketch," in IEEE Transactions on Parallel and Distributed Systems, vol. 34, no. 12, pp. 3028-3045, Dec. 2023, doi: 10.1109/TPDS.2023.3316717.
keywords: {Tensors;Anomalydetection;Monitoring;Datamodels;Telecommunicationtraffic;Matrixconverters;Featureextraction;Stream monitoring data;network anomaly detection;tensor sketch},

[7] J. M. DeAlmeida et al., "Abnormal Behavior Detection Based on Traffic Pattern Categorization in Mobile Networks," in IEEE Transactions on Network and Service Management, vol. 18, no. 4, pp. 4213-4224, Dec. 2021, doi: 10.1109/TNSM.2021.3125019.
keywords: {Anomaly detection;Trafficcontrol;Urbanareas;Telecommunications;Couplings;Unsupervisedlearning;Training;5G;next generation networks;anomalydetection;call detail record (CDR);self-healing control;networkanalytics;traffic pattern analysis;artificial intelligence (AI)},

[8] N. Ogawa and R. Kawahara, "Method for Network-Anomaly Detection and Failure-Scale Estimation," in IEICE Communications Express, vol. 13, no. 6, pp. 206-209, June 2024, doi: 10.23919/comex.2024XBL0028.
keywords: {Anomaly detection;Internet;Topology;Protocols;Delays;Packetloss;Networktopology;anomalydetection;failurescale;autoencoder;mininet},

[9] Y. Zhong, Z. Wang, X. Shi, J. Yang and K. Li, "RFG-HELAD: A Robust Fine-Grained Network Traffic Anomaly Detection Model Based on Heterogeneous Ensemble Learning," in IEEE Transactions on Information Forensics and Security, vol. 19, pp. 5895-5910, 2024, doi: 10.1109/TIFS.2024.3402439.
keywords: {Training;Anomalydetection;Generative adversarial networks;Intrusiondetection;Perturbationmethods;Datamodels;Featureextraction;Network anomaly detection;adversarialattack;unknown attack detection;ensemblelearning;fine-grained attack detection},

[10] C. Yao, Y. Yang, K. Yin and J. Yang, "Traffic Anomaly Detection in Wireless Sensor Networks Based on Principal Component Analysis and Deep Convolution Neural Network," in IEEE Access, vol. 10, pp. 103136-103149, 2022, doi: 10.1109/ACCESS.2022.3210189.
keywords: {Wireless sensor networks;Featureextraction;Anomalydetection;Principal component analysis;Telecommunicationtraffic;Convolutional neural networks;Representationlearning;Wireless sensor networks;denial of service;networkattack;principal component analysis;deep convolution neural network},

[11] Z. Wu, H. Li, Y. Qian, Y. Hua and H. Gan, "Poison-Resilient Anomaly Detection: Mitigating Poisoning Attacks in Semi-Supervised Encrypted Traffic Anomaly Detection," in IEEE Transactions on Network Science and Engineering, vol. 11, no. 5, pp. 4744-4757, Sept.-Oct. 2024, doi: 10.1109/TNSE.2024.3397719.
keywords: {Anomaly detection;Featureextraction;Training;Cryptography;Clusteringalgorithms;Taskanalysis;Datamodels;Encryptedtraffic;onlineclustering;poisoning attack defense;semi-supervised anomaly detection},

[12] J. Chen, J. Pu, B. Yin, R. Zhang and J. J. Wu, "TA-NET: Empowering Highly Efficient Traffic Anomaly Detection Through Multiple Head Local Self-Attention and Adaptive

Hierarchical Feature Reconstruction," in IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 9, pp. 12372-12384, Sept. 2024, doi: 10.1109/TITS.2024.3365820.
keywords: {Feature extraction;Anomalydetection;Taskanalysis;Hidden Markov models;Supervisedlearning;Benchmarktesting;Training;Featureextraction;anomalydetection;traffic anomaly detection;weakly supervised learning;multipleinstance learning;transformer},

[13] M. Driss Laanaoui, M. Lachgar, H. Mohamed, H. Hamid, S. Gracia Villar and I. Ashraf, "Enhancing Urban Traffic Management Through Real-Time Anomaly Detection and Load Balancing," in IEEE Access, vol. 12, pp. 63683-63700, 2024, doi: 10.1109/ACCESS.2024.3393981.
keywords: {Big Data;Real-timesystems;Roadtraffic;Hidden Markov models;Predictivemodels;Accidents;Vehicular ad hoc networks;Trafficcontrol;Urbanareas;Intelligent transportation systems;Densitymeasurement;Urban traffic management;real-time anomaly detection;intelligent transportation systems;traffic density prediction},

[14] L. Nie, L. Zhao and K. Li, "Robust Anomaly Detection Using Reconstructive Adversarial Network," in IEEE Transactions on Network and Service Management, vol. 18, no. 2, pp. 1899-1912, June 2021, doi: 10.1109/TNSM.2021.3069225.
keywords: {Training;Datamodels;Anomalydetection;Generative adversarial networks;Galliumnitride;Key performance indicator;Gaussiandistribution;Anomalydetection;reconstructive adversarial network;performance diagnose},

[15] W. T. Lunardi, M. A. Lopez and J. -P. Giacalone, "ARCADE: Adversarially Regularized Convolutional Autoencoder for Network Anomaly Detection," in IEEE Transactions on Network and Service Management, vol. 20, no. 2, pp. 1305-1318, June 2023, doi: 10.1109/TNSM.2022.3229706.
keywords: {Anomaly detection;Training;Telecommunicationtraffic;Generative adversarial networks;Datamodels;Generators;Deeplearning;Unsupervised anomaly detection;autoencoder;generative adversarial networks;automatic feature extraction;deeplearning;cybersecurity},

[16] L. Deng, D. Lian, Z. Huang and E. Chen, "Graph Convolutional Adversarial Networks for Spatiotemporal Anomaly Detection," in IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 6, pp. 2416-2428, June 2022, doi: 10.1109/TNNLS.2021.3136171.
keywords: {Spatiotemporal phenomena;Anomalydetection;Generators;Featureextraction;Hidden Markov models;Datamodels;Tensors;Adversariallearning;anomalydetection;intelligenttransportation;spatiotemporal data mining},

[17] F. Michelinakis et al., "AI Anomaly Detection for Cloudified Mobile Core Architectures," in IEEE Transactions on Network and Service Management, vol. 20, no. 2, pp. 1976-1992, June 2023, doi: 10.1109/TNSM.2022.3203246.
keywords: {Monitoring;Artificialintelligence;5G mobile communication;Anomalydetection;Cloudcomputing;Quality of service;Deeplearning;Anomalydetection;autoencoders;deeplearning;5G;AI;smartnetworks;mobile networks},

[18] H. Kye, M. Kim and M. Kwon, "Hierarchical Detection of Network Anomalies : A Self-Supervised Learning Approach," in IEEE Signal Processing Letters, vol. 29, pp. 1908-1912, 2022, doi: 10.1109/LSP.2022.3203296.

keywords: {Decoding;Training;Network intrusion detection;Simulation;Trainingdata;Standards;Self-supervisedlearning;Anomalydetection;network intrusion detection system;self-supervisedlearning;autoencoder},

[19] K. Kumaran Santhosh, D. P. Dogra, P. P. Roy and A. Mitra, "Vehicular Trajectory Classification and Traffic Anomaly Detection in Videos Using a Hybrid CNN VAE Architecture," in IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 8, pp. 11891-11902, Aug. 2022, doi: 10.1109/TITS.2021.3108504.
keywords: {Trajectory;Imagecoloranalysis;Videos;Anomalydetection;Convolutional neural networks;Featureextraction;Trainingdata;Convolutional neural network;deeplearning;variationalautoencoder;Dirichlet process mixture model;visualsurveillance;trajectoryclassification;traffic anomaly detection},

[20] Y. Meidan, D. Avraham, H. Libhaber and A. Shabtai, "CADeSH: Collaborative Anomaly Detection for Smart Homes," in IEEE Internet of Things Journal, vol. 10, no. 10, pp. 8514-8532, 15 May15, 2023, doi: 10.1109/JIOT.2022.3194813.
keywords: {Internet of Things;Collaboration;Anomalydetection;Telecommunicationtraffic;Detectors;Behavioralsciences;Training;Autoencoders (AEs);botnets;clustering;collaborative anomaly detection;cryptomining;Distributed Denial of Service (DDoS);Internet of Things (IoT);IoT attack detection},

[21] Z. Li, P. Wang and Z. Wang, "FlowGANAnomaly: Flow-Based Anomaly Network Intrusion Detection with Adversarial Learning," in Chinese Journal of Electronics, vol. 33, no. 1, pp. 58-71, January 2024, doi: 10.23919/cje.2022.00.173.
keywords: {Deep learning;Machine learning algorithms;Computationalmodeling;Network intrusion detection;Imaging;Telecommunicationtraffic;Generative adversarial networks;Anomalydetection;Unsupervisedlearning;Generative adversarial network;Intrusion detection system},

[22] D. Madariaga, J. Madariaga, M. Panza, J. Bustos-Jiménez and B. Bustos, "Detecting Anomalies at a TLD Name Server Based on DNS Traffic Predictions," in IEEE Transactions on Network and Service Management, vol. 18, no. 1, pp. 1016-1030, March 2021, doi: 10.1109/TNSM.2021.3051195.
keywords: {Anomaly detection;Servers;IPnetworks;Internet;Computercrime;Unsupervisedlearning;Reliability;Domain name system;anomalydetection;DNStraffic;top-leveldomain;prediction;unsupervised learning},

[23] A. Pramanik, S. K. Pal, J. Maiti and P. Mitra, "Traffic Anomaly Detection and Video Summarization Using Spatio-Temporal Rough Fuzzy Granulation With Z-Numbers," in IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 12, pp. 24116-24125, Dec. 2022, doi: 10.1109/TITS.2022.3198595.
keywords: {Uncertainty;Anomalydetection;Radiofrequency;Featureextraction;Optimization;Streamingmedia;Fuzzysets;Traffic anomaly detection;videosummarization;spatio-temporalfeatures;rough fuzzy granules;Z-numbers},

[24] H. Xu, S. Han, X. Li and Z. Han, "Anomaly Traffic Detection Based on Communication-Efficient Federated Learning in Space-Air-Ground Integration Network," in IEEE Transactions on Wireless Communications, vol. 22, no. 12, pp. 9346-9360, Dec. 2023, doi: 10.1109/TWC.2023.3270179.
keywords: {Space-air-ground integrated networks;Satellites;Deeplearning;Computationalmodeling;Machine learning

algorithms;Clusteringalgorithms;Anomalydetection;Anomaly traffic detection;space-air-ground integration network;federatedlearning;communication efficient},

[25] S. Liu, Y. Xia and D. Wang, "A Human-in-the-Loop Anomaly Detection Architecture for Big Traffic Data of Cellular Network," in IEEE Access, vol. 12, pp. 41787-41797, 2024, doi: 10.1109/ACCESS.2024.3376413.
keywords: {Human in the loop;Threatassessment;Telecommunicationtraffic;Networksecurity;BigData;Monitoring;Anomaly detection;Malware;6G mobile communication;Digitaltwins;Cyberthreats;networktraffic;networksecurity;bigdata;SNMDF, sets},

[26]https://www.kaggle.com/datasets/peterfriedrich1/dns-test-traffic-dohbrw2020

[27] https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset