Enhanced Intrusion Detection Using AI-Driven Data Balancing and VQ-VAE-Based Feature Extraction

Shivanthana S.[1,*], Manicka Raja M.[1], Lalitha Krishnasamy[2], Karthik R.[1], R. Venkatesan[1]

[1]Division of Computer Science and Engineering, Karunya Institute of Technology and Sciences, Coimbatore, India

[2]Department of Artificial Intelligence and Data Science, Nandha Engineering College, Erode, India

Emails: shivanthanas@karunya.edu.in; manickaraja@karunya.edu; lalithak@nandhaengg.org; karthikr@karunya.edu; rlvenkei2000@gmail.com

Abstract

Network security faces significant challenges due to the increasing sophistication of cyber threats and the inherent class imbalance in intrusion detection datasets. To address this issue, a hybrid Boundary Equilibrium Generative Adversarial Network (BEGFAN) and Vector Quantization Variational Autoencoder (VQVAE) framework, termed BVQVAE, is proposed for Network Intrusion Detection Systems (NIDS). The framework involves preprocessing, feature extraction, and class balancing to enhance classification accuracy. Missing values are imputed, categorical features are label-encoded, and numerical attributes are normalized to ensure a structured dataset. BEGAN generates synthetic samples to mitigate class imbalance, while VQVAE extracts essential features using an encoder with quantization and a decoder for network traffic reconstruction. The model is evaluated on NSL-KDD and UNSW-NB15 datasets, achieving 82.56% accuracy, with precision, recall, G-mean, and F1-score of 86.53%, 87.65%, 86.21%, and 87.08%, respectively.

Keywords: Network Security; Class Imbalance; Adversarial Learning; Anomaly; Variational Autoencoder

1. Introduction

The new digital age has witnessed an exponential growth in the number and variety of devices that are interconnected, producing immense data every day. While all this interconnectedness has changed business, healthcare, education, and social life for the better, it has also created a phenomenon of unusual cybersecurity threats. Especially, Network intrusions which are extreme threat to the integrity, confidentiality, and availability of critical infrastructure, making Intrusion Detection Systems (IDS) an indispensable part of cybersecurity infrastructure. IDS technologies keeps observing the network traffic and system activities, detecting and prevent potential risks in order to avoid data breaches and cyber-attacks.

Despite with vast improvements in IDS technologies, these tools are still troubled by one constant difficulty: data imbalance. Data imbalance is a situation where some types of attacks like advanced User-to-Root (U2R) or Remote-to-Local (R2L) attacks are represented inadequately relative to the majority classes, like normal traffic or Denial-of-Service (DoS) attacks. As a consequence, conventional machine learning models, which overwhelm the IDS field, will favor the majority classes and under signal minority attack types. This imbalance not only reduces the IDS's performance, but also makes it susceptible to misclassification, since uncommon or unknown attack types are frequently mis-labeled as regular traffic, resulting in significant false-negative rates.

The recent hacking into the top financial and educational institutions reveals how undetected intrusion poses. An example of this is ICICI Bank that has been claimed to have resulted from the actions of the Bashe hacking group where attackers threatened to publish sensitive financial data if a ransom were not received within a particular period. This incident demonstrates how such targeted and advanced attacks can evade classic IDS models when

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

not enough information is available to represent such an attack pattern [29]. Likewise, breaches with student and employee information within institutions—the ones that often get associated with insider threats or APTs—point to the vulnerability of personal and institutional information. This shows that completely leads to biased results in anomaly detection, exposing the systems to deadly consequences [30].

To mitigate these challenging problems, a hybrid model is proposed. This framework employs two advanced techniques, namely BEGAN and VQVAE. BEGAN provides a great capability of synthetic data generation and, along with VQVAE's expertise in feature extraction, provides effective mitigation for the problem of imbalanced data while strengthening the intrusion detection system robustness. Generative Adversarial Networks (GANs), more specifically BEGAN, have emerged as a strong tool for machine learning because of their potential to produce realistic synthetic data. BEGAN differentiates itself from the rest of the GANs by its use of equilibrium-based training, thus maintaining a balance between the generator and the discriminator, which results in generating high-quality synthetic data. This feature is especially beneficial in the case of augmenting datasets plagued with data imbalance, since BEGAN can produce realistic samples for minority attack classes. With the increase in representation of minority attacks, BEGAN allows the IDS to better detect such attacks. Apart from BEGAN, VQVAE is used for feature extraction. VQVAEs have the advantage of representing inputs as discrete latent variables, which offers a strong and efficient mechanism for encoding intricate data. While autoencoders, in conventional use, use continuous latent space, VQVAEs implement a discrete codebook to acquire compressed network traffic representations, thereby being ideally equipped for anomaly detection. Reconstruction loss and latent features offered by VQVAE facilitate proper discrimination between normal traffic and attack traffic, even with infrequent attack classes. The proposed model functions in three phases:

1. Data Augmentation where BEGAN produces synthetic data to counter data imbalance, making sure that minority attack classes are represented well.

2. Feature Extraction where VQVAE learns from the augmented dataset, extracting strong latent features that retain key properties of the data.

3. Anomaly Classification where latent representations and reconstruction errors are further utilized to perform precise classification with emphasis on minority attack types.

This paper addresses these challenges by proposing an AI-enhanced NIDS that addresses data imbalance and improves the performance of existing models. The system architecture is divided into four stages: (1) preprocessing, which deals with missing data and categorical variables, normalizes the dataset; (2) training generative models to balance the data distribution; (3) feature extraction; (4) feeding the extracted features into the classification model. The preprocessing step ensures that the raw data is put in appropriate form for a deep learning model. Meanwhile, the feature extraction and training stages use generative models to create balanced datasets, which ultimately use a VQVAE model classification for the final classification. Tests performed on NSL-KDD datasets support the effectiveness of the method proposed here compared to existing models previously existing by addressing challenges of data imbalance and high dimensionality. The rest of this paper is organized in the following lines: Section II discusses work relevant to intrusion detection and data imbalance. The section III provides background pertinent to the methods applied in the paper. The proposed methodology occupies Section IV, while the definition of the experimental part can be found in Section V. Finally, Section VI presents and discusses the achieved results. Finally, Section VII contains the conclusion with future prospects on the research undertaking.

2. Related Work

In the era of rapid digital change, the increasing range of new cyberattacks poses significant threats to networks and systems, highlighting the need for advanced Networks Intrusion Detection Systems (NIDS). The issue of data imbalance arises when certain categories within a dataset have markedly more instances than others, leading to biased learning outcomes. This imbalance is particularly prevalent in fields such as fraud detection, intrusion detection systems, and medical diagnosis, where some event types are poorly represented. This section examines the latest research that has been carried out on the problem of data imbalance in intrusion detection systems.

Gao et al. proposed a hybrid system combining Vector Quantized Variational Autoencoder (VQ-VAE) with Support Vector Data Description (SVDD) for lung CT anomaly detection. The model achieved an accuracy of 71% and an AUC of 76%, demonstrating the effectiveness of combining unsupervised representation learning with one-class classification methods [1]. Similarly, Jebril et al. applied VQ-VAE for anomaly detection in Optical Coherence Tomography Angiography (OCTA) and achieved an AUROC of 92% on the DRAC dataset and 75% on the OCTA 500 dataset, highlighting its potential in detecting ophthalmological anomalies [2].

For intrusion detection in automotive networks, Sowmya and Mary Anita reviewed machine learning-based techniques such as Decision Trees (DT), Support Vector Machines (SVM), Random Forest (RF), and Naïve Bayes

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

(NB) for detecting intrusions in Controller Area Network (CAN) bus data. The study emphasized the challenges in designing robust cybersecurity measures for modern vehicles [3]. Similarly, Yan and Han employed a Stacked Sparse Autoencoder (SSAE) with SVM to enhance intrusion detection in the NSL-KDD dataset, achieving 96.4% accuracy, underscoring the importance of hierarchical feature learning for network security [4].

Handling class imbalance is a key challenge in IDS. Mariama Mbow et al. addressed this issue using Synthetic Minority Over-sampling Technique (SMOTE) and Tomek Links for balancing datasets like NSL-KDD, CICIDS 2017, and CICIDS 2018, achieving a detection rate of 95% in CICIDS 2018 [5]. Similarly, Mohammad Hashem Haghighat and Jun Li proposed a voting-based neural network (VNN) on KDDCUP99 and CTU-13 datasets, reducing false alarms by 75% [6]. Furthermore, Y. Imamverdiyev et al. introduced a Gaussian-Bernoulli Restricted Boltzmann Machine (RBM) achieving 68.9% accuracy on NSL-KDD for detecting DoS attacks [7].

Advanced anomaly detection techniques have been explored for industrial IoT environments. Yihong Yang and Xuan Yang proposed the ASTREAM framework using LSHiForest for scalable anomaly detection on KDDCUP99 [8]. Additionally, Lianyong Qi et al. introduced a multi-aspect data stream anomaly detection approach for intrusion detection in Industry 4.0 using UNSW-NB15 [9].

Zheng et al. proposed a data conformity mechanism integrating VAE and SVM for imbalanced binary classification, achieving 70.73% accuracy on UCI datasets [10]. Another approach by ADASci combined deep VAE with NEATER for UCI and KEEL datasets, though performance metrics were not specified [11].

In medical anomaly detection, Kim et al. introduced Patch-wise VQ-VAE, achieving AUROC scores of 94.3% and 98.4% [12]. Sharma et al. utilized a semi-supervised VQ-VAE framework, yielding an AUC of 80% [13]. Marimont and Tarroni reported AUROC scores of 97% and 83% for brain MR and abdominal scan data using VQ-VAE [14]. Zhou et al. extended this research with VQ-Flow, achieving AUROC scores of 99.5% and 98.3% on MVTec AD datasets [15].

For IDS, Abdulganiyu et al. developed CWFL-VAE integrated with XGBoost, achieving a precision of 99.67% and an F1-score of 94.74% on CSE-CIC-IDS2018, demonstrating its superiority in handling imbalanced network traffic [16]. BiGAN has been explored for anomaly detection in industrial control systems, particularly in Secure Water Treatment systems [17]. A unified deep learning approach combining Autoencoders and GANs achieved 96.4% accuracy with an F1-score of 0.925 in smart grid anomaly detection [18].

Conditional GANs (CGANs) have been used to address IDS data imbalance. A study achieved 96.8% accuracy on NSL-KDD, demonstrating the effectiveness of CGANs in generating realistic attack data [19]. GAN-based approaches have also been applied to UAV security, using Active Learning for real-time intrusion detection [20].

A Data Generative Model (DGM) combining CGANs with KL-divergence significantly improved detection rates for NSL-KDD and UNSW-NB15 datasets [21]. A hybrid model integrating GANs and Autoencoders achieved 93.2% accuracy on NSL-KDD and 87% on UNSW-NB15, reinforcing the robustness of generative-discriminative models [22]. G-IDS, which combines GANs and Autoencoders, achieved a precision of 91% and an F1-score of 85% on NSL-KDD [23].

Recent research has also examined alternative data balancing techniques. A study explored SMOTE adaptations for IDS datasets, offering alternatives to deep generative models [24]. Seo et al. investigated adversarial attacks on ML-based IDS in automotive security, emphasizing the dual role of generative models in both enhancing and circumventing IDS mechanisms [25]. Finally, a classification framework integrating VAEs and GANs with deep learning classifiers such as Deep Neural Networks (DNNs) and SVMs suggested that combining generative models with discriminative classifiers significantly improves IDS accuracy [26].

3. Background

Generative models is one the solution to the data imbalance issue in network intrusion detection systems (NIDS). With the ability to generate realistic synthetic data, generative models can augment training datasets and thereby improve the intrusion detection ability of machine learning models. Of the generative models, Boundary Equilibrium Generative Adversarial Networks (BEGAN) and Vector Quantized Variational Autoencoders (VQ-VAE) have been at the forefront of enhancing data quality and facilitating anomaly detection.

3.1 Boundary Equilibrium Generative Adversarial Networks (BEGAN)

BEGAN is an extension of Generative Adversarial Networks (GANs) introduced by Berthelot et al. to enhance the stability of training and data quality. Unlike traditional GANs, BEGAN utilizes an equilibrium loss function to maintain a balance between the generator and the discriminator, preventing mode collapse and ensuring the generation of more realistic samples. The generator (G) and discriminator (D) in BEGAN are trained to reach an equilibrium point, preventing one from overpowering the other [27].

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

The loss function for BEGAN is:

$$L_{BEGAN} = E_{x\sim p_{data}(x)}[D(x)] + E_{z\sim p_z(z)}[1 - D(G(z))] \quad (1)$$

Where D(x) is the discriminator's probability of classifying x as real data, G(z) is the generator's output given the latent noise z and to ensure the generator improves over time by reducing the discrepancy between real and synthetic data $1 - D(G(z))$ is used.

BEGAN is particularly valuable for generating synthetic network traffic data, both for normal traffic and attack patterns. This augmentation of the dataset helps balance the class distribution, making it easier to train machine-learning models that are more robust and capable of detecting intrusions.

3.2 Vector Quantized Variational Autoencoders (VQ-VAE)

VQ-VAE is a variant of the traditional Variational Autoencoder (VAE) that employs vector quantization to discretize the latent space. In contrast to normal VAEs, that employ continuous latent variables, VQ-VAE employs a codebook of discrete vectors in order to allow the model to learn structured representations of input data. This structured encoding improves interpretability and facilitates anomaly detection and synthetic data generation [28].

The total loss function for VQ-VAE is expressed as:

$$L_{VQ-VAE} = L_{reconstruction} + \beta L_{commitment} + L_{vq} \quad (2)$$

Where:

$L_{reconstruction} = \|x - \hat{x}\|^2$ measures how well the model reconstructs the input, $L_{commitment} = \|z_e(x) - sg(e_k)\|^2$, simulates the encoder to commit to discrete latent vectors and $L_{vq} = \|sg(e_k) - z_e(x))\|^2$, ensures that the encoded vector is close to the nearest vector in the codebook.

In NIDS, VQ-VAE can be utilized to learn discrete representations of network traffic, aiding in anomaly detection by identifying patterns that deviate from the learned distribution. VQ-VAE can also be combined with BEGAN to generate synthetic network traffic data, thereby augmenting training datasets.

Integrating BEGAN and VQ-VAE for synthetic data generation can significantly enhance NIDS performance. While BEGAN generates realistic samples by maintaining an equilibrium between the generator and discriminator, VQ-VAE structures the latent space using discrete vector quantization, improving anomaly detection efficiency. The combination of these models allows for the generation of both normal and attack traffic samples, augmenting imbalanced datasets and improving the detection capabilities of machine learning models.

In addition to addressing class imbalance, this approach creates diverse data distributions that better reflect the real-world complexity of network traffic. As a result, NIDS trained with this augmented data become more robust and capable of detecting subtle anomalies and novel attack patterns. The application of BEGAN and VQ-VAE in intrusion detection is a crucial step toward mitigating data limitations and enhancing detection accuracy. BEGAN generates synthetic attack samples, while VQ-VAE ensures that these samples are structured effectively, enabling detection models to generalize across diverse data distributions. Together, these models offer a powerful solution for handling imbalanced datasets, where attack samples are often underrepresented compared to normal traffic.

4. Proposed Methodology

The proposed architecture BVQVAE focuses on four main components: data preprocessing, identification of minority classes, Synthetic data generation and feature extraction and classification to enhance intrusion detection capabilities. This work uses the Boundary Equilibrium Generative Adversarial Network (BEGAN) for balancing and Vector Quantized Variational Autoencoder (VQVAE) for feature extraction and classification. The methodology BVQVAE starts with preprocessing of NSL-KDD dataset. Missing values are handled and categorical features are coded into numerical vector using LabelEncoder and numerical features through Min-Max scaling. The proposed framework addresses the case of class imbalance by including BEGAN to generate synthetic sample for underrepresented class to ensure to have a relatively balanced training. Once after this the important features are extracted using VQVAE that compresses the dataset while attempting to capture latent representations. After extracting the features, the features are sent as input to the classification model VQVAE by enhancing accuracy and robustness within intrusion detection systems.

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

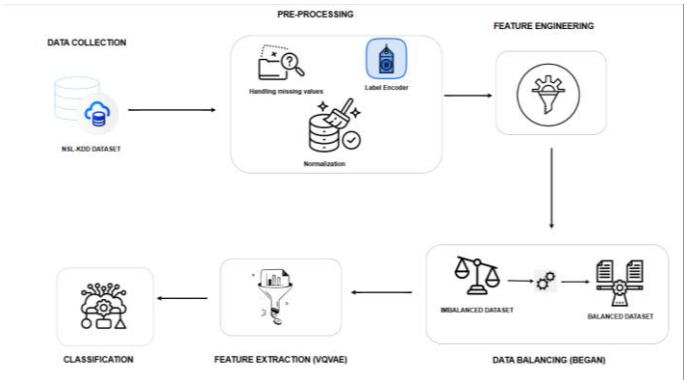

Figure 1. Proposed Architecture

4.1 Data Preprocessing

The preprocessing phase is essential to prepare the NSL-KDD dataset for the intrusion detection system. First, the missing values are checked to make sure that data is complete. Here, the missing values is handled by mean imputation method where the missing values are replaced with mean values of the non-missing values of the feature:

$$a_{xy} = \begin{cases} a_{xy}, & if \ \ a_{xy} \ is \ not \ missing \\ \frac{1}{m_y}\sum_{k=1}^{m_y} a_{ky}, & if \ a_{xy} \ is \ missing \end{cases} \quad (3)$$

Where a_{xy} is the value of the feature y for the x-th sample, m_y is the number of non-missing values in feature y and $\sum_{k=1}^{m_y} a_{ky}$ is the sum of all non-missing values in feature y. The categorical features such as 'Protocoltype', 'Service', 'Flag' are encoded using LabelEncoder. This assigns a unique numerical value to each category of the data. Once after this the numerical feature of all other columns except the categorical features and target(label) , is identified and scaled using Min-max scaling to normalize their values in a given range between '0' and '1'.

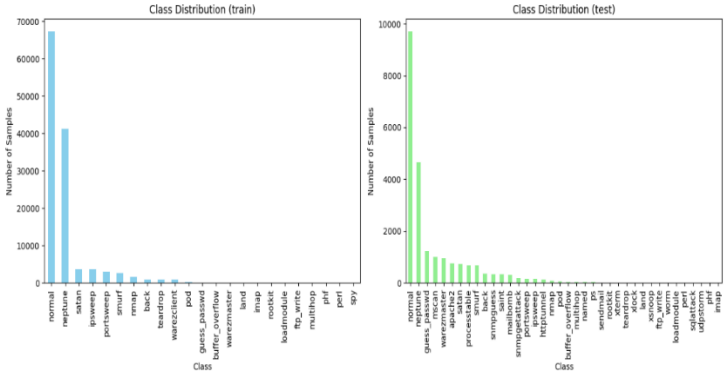

Figure 2. Training and Testing dataset before balancing

4. 2 Data Augmentation

To address the data imbalance, the proposed model involves identifying minority classes and generating synthetic data to balance the dataset. Initially the minority classes are identified by comparing the sample count of each class n_i with the majority class n_{max}. A threshold factor α is used to determine whether a class qualifies as a minority. If $n_i < \alpha.n_{max}$, the class is labeled as a minority. For example, if α=0.8, a class is considered a minority if the size is less than 80% of the majority classes. This ensures only significantly underrepresented classes are selected for synthetic data generation.

$$Minority\ Classes = \{i | n_i < \alpha.n_{max}\} \quad (4)$$

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

Here, α is the threshold factor i.e a constant between 0 and 1that determines how smaller class must be relative to the majority class to be considered minority, n_{max} is the size of majority classes, n_i is the number of samples and i represents the index or label of class in the dataset(e.g., "normal","spy"). To address the data imbalance in the datset, synthetic data is generated using BEGAN, a type of Generative Adversarial Network (GAN). BEGAN consists of two main components (1) Generator (G) and (2) Discriminator (D). The generator generates synthetic samples for the detected minority classes. This model takes random latent vectors b as input, which are drawn from noise distribution. The generator transforms the random noise into synthetic data points that resembles real samples in the dataset. The discriminator(D) differentiates real and fake data. This model takes both real and fake data as input and predicts whether a given sample is real or synthetic data that is produced by the generator(G). BEGAN optimizes the equilibrium loss L_E, which maintains a balance between generator loss L_G and discriminator L_D. The generator L_G loss measures how closely synthetic data matches real data, while the discriminator L_D measures ability to distinguish real and synthetic samples.

Algorithm 1 Addressing Data Imbalance using BEGAN

Input: Dataset with class distributions $\{n_i\}$, threshold factor α, maximum class size n_{max}.

Output: Balanced dataset with synthetic samples.

1: Identify minority classes:

Minority Classes = $\{i \mid n_i < \alpha \cdot n_{max}\}$

2: **for** each minority class *i* **do**

3: Initialize BEGAN model with Generator *G* and Discriminator *D*

4: **for** each training iteration **do**

5: Sample latent vector *b* from noise distribution

6: Generate synthetic data: *G*(*b*)

7: Compute discriminator loss:

$$L_D = E[D(x)] - E[D(G(b))]$$

8: Compute generator loss:

$$L_G = E[G(b)] - x$$

9: Optimize equilibrium loss:

$$L_E = |\gamma L_D - L_G|$$

10: Update *G* and *D* using gradient descent

11: **end for**

12: Add generated samples to minority class until $n_i = n_{max}$

13: **end for**

14: **Return** balanced dataset.

For each minority classes, synthetic data is generated using BEGAN that optimizes the equilibrium loss L_E to balance the generator(G) and discriminator(D).

$$L_E = |\gamma . L_D - L_G| \quad (5)$$

Where L_D is the discriminator loss. The discriminator loss measures the difference between the expected real data D(x) and fake data D(G(b)) probabilities:

$$L_D = E[D(x)] - E[D(G(b))] \quad (6)$$

L_G is the generator loss. The generator loss deals with the difference between generated data G(b) and real data x.

$$L_G = E[\|xG(b)\|_1] \quad (7)$$

Here, x is a real sample, and b is a latent vector. The model alternates between minimizing L_D and L_G to improve synthetic data quality. Generated samples are added to the minority class until the class size equals n_{max} achieving class balance.

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

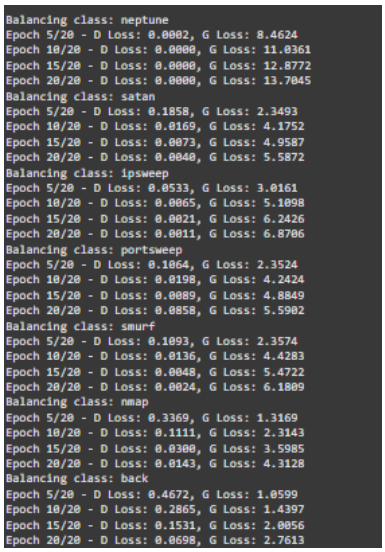

Figure 3. Generation of synthetic data (Generator Loss) and Discriminator loss

4.3 Feature Extraction and Classification

Once after augmenting the dataset using Boundary Equilibrium Generative Adversarial Networks (BEGAN), the next involves extracting meaningful features using Vector Quantized Variational Autoencoder (VQVAE) from the balanced and combined data that is the output of previous step. VQVAE is a variant of VAE that incorporates discrete latent representations through vector quantization. This enables efficient feature learning and enhances the ability of the model to generalize across different attack patterns in NIDS. The proposed framework involves three primary components: feature extraction using VQ-VAE, classification using latent representations through reconstruction.

The balanced dataset is first pre-processed to ensure compatibility with the VQ-VAE model. This involves feature extraction where the label column is removed and all numerical values are scaled between 0 and 1 to enhance training stability. The dataset is then split into training and testing sets:

$$X_{train}, X_{test}, y_{train}, y_{test}$$

Where X_{train} and X_{test} represent the feature vectors, while y_{train} and y_{test} are the corresponding labels.

4.4 Feature Extraction

The VQ-VAE model is used to extract feature representations from network traffic data. The feature extraction contains of three key components: 1) Encoder, 2) Vector Quantization Layer and 3) Decoder. The encoder compresses high dimensional input features into a lower-dimensional latent space using dense layers, mapping the data into an embedding space. The encoder is represented as: $z = f_{enc}(X)$,where f_{enc} is the encoder network and z is the latent representation. The encoder consists of two dense layers and it is represented as: $z = ReLU\big(Dense_{128}(X)\big)$ and $z = Dense_{embedding}(z)$.where ReLU is the activation function applied to the hidden layer and the final layer maps the data into an embedding space of dimension 32.

The vector quantization layer discretizes latent representations by mapping them to the closest embedding vector from a learned codebook, ensuring proper quantization through commitment and codebook losses. To discretize the latent representations, the model uses a vector quantization (VQ) layer. The encoder output z is mapped to the closest embedding vector from a learned codebook E: $z_{quantized} = arg\ min_{e_i \in E} \|z - e_i\|$ Where e_i represents the closest embedding vector among 64 predefined embeddings. To ensure the quantization is proper, two losses are introduced: 1) Commitment Loss and 2) Codebook Loss. The commitment loss is to assist the encoder output to stay close to the assigned embedding: $L_{commit} = \|sg(z) - e_i\|^2$ Where sg(z) denotes the stop-gradient operation, preventing gradients from updating z. The codebook loss is used to update the embedding vectors: $L_{embedding} = \|e_i - sg(z)\|^2$. The final quantized vectors are represented as follows:

$$z_{quantized} = z + sg(e_i - z) \quad (8)$$

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

The decoder reconstructs the input features from the quantized latent representation: $X' = f_{dec}(z_{quantized})$ where f_{dec} is the neural network that consists of two dense layers and it is represented as: $X' = ReLU\left(Dense_{128}(z_{quantized})\right)$ and $X' = Sigmoid\left(Dense_{output}(X')\right)$

The model is trained using the Adam optimizer over 50 epochs. After the training, latent representations are quantized and stored for later usage in classification. The VQ-VAE-based feature extraction effectively transforms raw network traffic data into discrete latent representations, hence improving data efficiency for downstream intrusion detection tasks. The final extracted features are:

$$z_{train,quantized} = VQ(f_{enc}(z_{train})) \quad (9)$$

$$z_{test,quantized} = VQ(f_{enc}(z_{test})) \quad (10)$$

Algorithm 2 VQ-VAE Feature Extraction

Input: Training data X_{train}, Testing data X_{test}

Parameters: Embedding dimension d, Number of embeddings k, Commit- ment cost β, Learning rate α, Batch size B, Epochs E

Define Model Components

Encoder: $E(X) = Dense_{128}(ReLU(X)) \rightarrow Dense_d(E(X))$

Decoder: $D(z) = Dense_{128}(ReLU(z)) \rightarrow Dense_{output}(Sigmoid(D(z)))$

Vector Quantizer: Assigns z to closest embedding e_j with loss:

$L_{commit} = ||sg(z) - e_j||^2$, $L_{embedding} = ||e_j - sg(z)||^2$

Train VQ-VAE

Initialize E, D, VQ and compile with Adam(α = 0.001) and MSE loss

for epoch = 1 to E **do**

Train: VQVAE.fit(X_{train}, X_{train}, B)

end for

Extract Save Features

Compute $z_{train,\ quantized} = V\ Q(E(X_{train}))$, $z_{test,\ quantized} = V\ Q(E(X_{test}))$

Output: Saved latent features for downstream tasks.

This ensures the compatibility of the balanced dataset with the VQ-VAE model, including feature extraction in which the label column is removed and numerical values are scaled between 0 and 1 for stability during training. Afterward, it is split into a training and test set by assigning feature vectors to the former and the latter while storing the corresponding labels. The three major components of the feature extraction process are the encoder, the vector quantization layer, and the decoder. The encoder compresses high-dimensional input features into a lower-dimensional latent space using dense layers, mapping the data into an embedding space. The vector quantization layer discretizes latent representations by mapping them to the closest embedding vector from a learned codebook, ensuring proper quantization through commitment and codebook losses. The decoder reconstructs input features from the quantized latent representations, optimizing reconstruction loss using Mean Squared Error (MSE). The model is trained using the Adam optimizer for 50 epochs. After training, quantized latent representations are extracted and stored for later use in classification. The VQ-VAE-based feature extraction effectively transforms raw network traffic data into discrete latent representations, improving data efficiency for intrusion detection tasks.

4.5. Classification

The classification process in the proposed VQ-VAE-based model depends on the structured quantized latent representations obtained from the vector quantization layer. The representations are good at capturing patterns in network traffic data, thus making them quite effective for the downstream classification task. The classification network is designed as a fully connected neural network (FCNN) to take these latent features and predict if a given network instance belongs to a normal or attack category. The classification module involves multiple dense (fully connected) layers that play a major role in fine-tuning the extracted features before obtaining the final output. The first dense layer employs a non-linear activation function such as ReLU to realize input features' potential for capturing non-linear representations of complex patterns and relationships in the data. This layer also prevents

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

overfitting by activating neurons based on selectivity over input features. To improve generalization and prevent overfitting, a dropout layer is introduced, which randomly deactivates a fraction of neurons during training. This makes the network learn features that are more robust rather than memorizing patterns in the training set. The feature representations are then passed through additional dense layers that progressively refine the learned embedding. The final output layer makes use of the softmax activation function, which turns the output scores into probability distributions over multiple classes. Each probability value corresponds to the likelihood that a sample belongs to a particular class. The predicted class is the one with the highest probability score. A cross-entropy loss function is used to train the classification network, measuring the divergence between predicted probabilities and true class labels. The optimization process, typically using Adam or SGD optimizers, adjusts model parameters to minimize this loss, leading to more accurate predictions over time. This approach ensures that the classifier can effectively distinguish between normal and attack traffic, enhancing intrusion detection performance.

Algorithm 3 VQ-VAE Based Classification

Input: Network traffic data X with labels Y

Output: Predicted class labels $\hat{Y}$

FEATURE EXTRACTION USING VQVAE

Encode input X into latent representation Z

Apply vector quantization to obtain quantized embeddings Z_q

Classification Network

Pass Z_q through a non-linear activation function (ReLU)

Apply dropout regularization to prevent overfitting

Further refine representations through additional processing layers

Prediction and Optimization

Compute final output probabilities using softmax activation

Compute classification loss using cross-entropy:

$$L = -\sum_{i} Y_i \log(\hat{Y}_i)$$

Update model parameters using Adam or SGD optimizer

Output Predictions

Assign class label with highest probability as $\hat{Y}$

Return $\hat{Y}$

The proposed methodology integrates BEGAN for data balancing and VQ-VAE for feature extraction and classification to enhance the intrusion detection. BEGAN is utilized to address the issue of data imbalance by generation high-quality synthetic attack samples, ensuring a well-balanced dataset. This balanced dataset is then used to extract features using VQ-VAE which is used to encode input data into structed latent representations through vector quantization. The quantized representations effectively capture essential patterns in network traffic, making them highly suitable for classification. Then these latent features are passed to classification network comprising multiple layers that refine the extracted features for anomaly detection. Dropout layers improve the generalization, while the output layers utilizes softmax activation to assign probability scores to each class. Optimized using cross-entropy loss, this approach enhances detection accuracy and system reliability.

5. Experimental setup

In this section, the performance of the BVQVAE is compared to other class balancing methods as well as other state-of-the-art methods. The results show BVQVAE that performs better than the reported state-of-art methods. The experiment were conducted on Google Colab utilizing 12 GB of RAM and 102 GB of storage space. Python libraries like NumPy (numerical computing), Pandas (data manipulation and analysis), TensorFlow (data cleansing), Scikit-learn (, model development, and assessment), Seaborn and Matplotlib (data visualization), Warnings (to handle or ignore warnings during execution), and System (to access variables used or maintained by the Python interpreter).

6. Dataset

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

BVQVAE model has been empirically tested using the NSL-KDD [31]. The NSL-KDD dataset is the modified version of that of KDD99. The dataset consists of a training and testing dataset with 125, 974 and 225, 44 rows respectively. This has been done in order to enable researchers carry out a more intuitive comparison of the different approaches. NSL-KDD has 41 attributes. NSL KDD dataset includes four types of abnormal activity behaviors, which are discrimination DOS, Abbott R probe, plus U2r and R2L. NSL KDD is classified into train KDDTRAIN with 20 sets and test KDDTEST set and KDDTEST21 set (table 2). A data that matches various Internet applications is designed for multi class classification the Internet Applications were Normal DoS attacks Probe attacks R2L User to Root U2R attacks.

7. Metrics

In this work, G–Mean, F1–Score, Accuracy, and precision have been identified as key measures for the classification performance of the BVQVAE. The metrics reported accuracy, precision and recall, F1-Score which was used to assess the architecture implemented, due to the factors that metrics used measure both false positivity and true positivity. Model performance is assessed through accuracy, which shows the proportion of correct predictions; however, precision is defined as the proportion of true positives out of total positive predictions. Accuracy captures the goodness of fit of the model, while precision is the fraction of the correctly predicted positives to all predicted positives. Geometric mean is the square root of the product of true positive and true negative that is used to evaluate imbalanced classification issue. Recall is the fraction of true positives to all actual positives available within the population, and the F1-score is the harmonic mean of the precision and recall further enhancing the insightfulness of the evaluation done on the model. Following each testing procedure, the elements of accuracy, precision, recall, and F1-score were determined as:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (11)$$

$$G-mean = \sqrt{\frac{TP}{TP+FN} \times \frac{TN}{TN+FP}} \quad (12)$$

$$Precision = \frac{TP}{TP+FP} \quad (13)$$

$$Recall = \frac{TP}{TP+FN} \quad (14)$$

$$F1-score = \frac{2\times P\times R}{P+R} \quad (15)$$

8. Results and discussion

In this section, the experimental results of the proposed framework BVQVAE that is the integration of Boundary Equilibrium Generative Adversarial Network and Vector Quantization Variational Autoencoder for balancing the dataset and for feature extraction and classification respectively. At first the dataset is pre-processed by handling missing values, features are scaled using min-max scaler and the labels are encoded using LabelEncoder. Then the data is balanced using BEGAN. The balanced data of KDDTrain+ and KDDTest+ is shown in Fig.4 and Fig.5

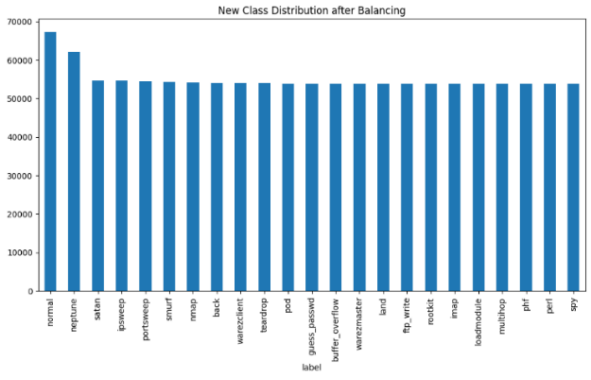

Figure 4. Testing dataset after balancing

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

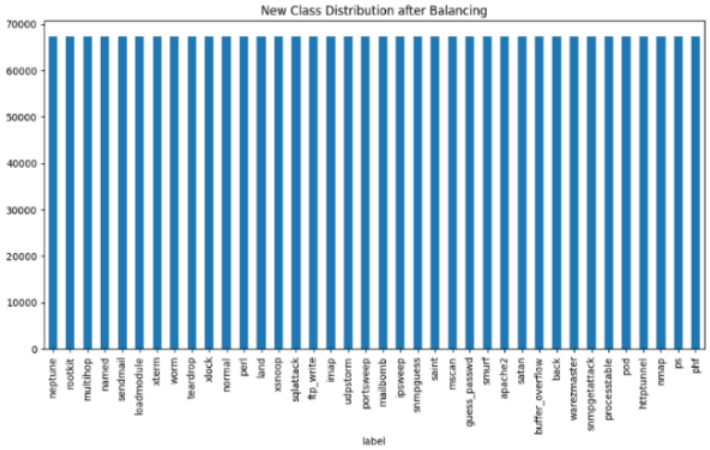

Figure 5. Training dataset after balancing

The number of original and generated samples across different datasets including KDDTrain+, KDDTest+ and UNSW-NB15. The analysis ensures that synthetic data that is generated, balances the class distributions and model generalization. Table.1 and Table.2 provides an overview of the number of samples before and after augmentation in KDDTrain+ and KDDTest+ from NSL-KDD dataset.

Table 1: KDDTrain+ dataset

Class	**No.of original samples**	**Generated samples**	**Sum**
NORMAL	67343	0	67343
DoS	46909	20434	67343
PROBE	11656	55687	67343
U2R	69	67274	67343
R2L	1007	66336	67343
TOTAL	126984	277074	336715

Table 2: KDDTest+ dataset

Class	**No. of original samples**	**No. of Generated samples**	**Sum**
Normal	8710	0	9710
DoS	7525	5000	12525
Probe	3377	3000	6377
U2R	50	6327	6377
R2L	4741	1636	6377
Total	14000	23488	41366

Table.3 provides an overview of the number of samples before and after augmentation in UNSW-NB15 dataset. The results indicate that the generated samples effectively complement the original dataset, mitigating class imbalanced and enhancing detection performance.

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

Table 3: UNSW-NB15

Class	No. of Original Samples	No. of Generated Samples	Sum
Normal	56000	0	56000
Generic	40000	7000	47000
Exploits	33393	6000	39393
Fuzzers	18184	5000	23184
DoS	12264	4500	16764
Reconnaissance	10491	3000	13491
Analysis	2000	1200	3200
Backdoor	1746	900	2646
Worms	130	800	930
Shellcode	1133	1000	2133
Total	173341	29400	20741

The proposed model has been evaluated using different measurements such as, accuracy, precision, detection rate, F1-score, and G-mean. Table.4 shows the performance metrics of the proposed model.

Table 4: Performance Metrics

Name	Value
Accuracy	82.56%
Precision	86.53%
Recall	87.65%
G-mean	86.21%
F1 score	87.08%

The performance of the proposed model is compared using the evaluation metrics such as Accuracy, Precision, Recall, and F1-Score. Fig.6 and Table.5 presents a graphical comparison of the metrics and provides detailed analysis respectively.

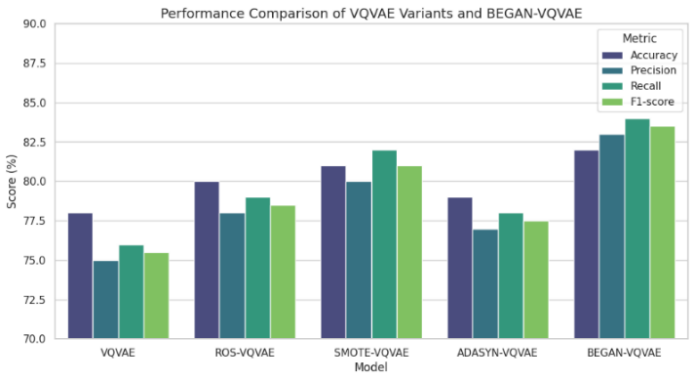

Figure 6. Comparison with Other Balancing Algorithms

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

Table 5: Comparison with Other Balancing Algorithms

Model	Accuracy	Precision	Recall	F1-Score
VQVAE	78	75	76	75.5
ROS-VQVAE	80	78	79	78.5
SMOTE-VQVAE	81	80	82	81
ADASYN-VQVAE	79	77	78	77.5
BVQVAE(Proposed method)	**82.56**	**86.53**	**87.65**	**87.08**

The comparative analysis of GAN models with deep learning techniques significantly improves intrusion detection accuracy, making the proposed model robust solution for NIDS. Table.6 and Fig.7 shows the comparison of existing model with the proposed model.

Table 6: Comparison of Existing Model with Proposed Model

Model	Accuracy
VQVAE	76.03%
Adasyn-CNN	80.11%
VQVAE(without balancing)	70.21%
VAE	70.34%
BVQVAE(Proposed method)	**82.56%**

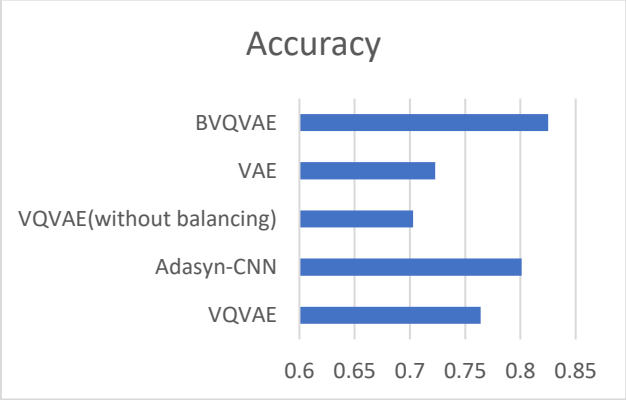

Figure 7. Comparison of Existing Model with Proposed Model

Fig 7 demonstrates that BVQVAE achieves the highest accuracy, surpassing other models such as VQVAE [33] and VAE. The results further highlight that balancing techniques as if Adasyn-CNN [32] and BVQVAE significantly enhance classification performance compared to models without balancing.

9. Conclusion

In conclusion, a novel hybrid approach for Network Intrusion Detection System is introduced by integrating Boundary Equilibrium Generative Adversarial Network and Vector Quantization Variational Autoencoder has been proposed. The methodology is divided into two phases: preprocessing and feature engineering followed by balancing and feature extraction for classification. The preprocessing involves handling missing values by replacing them with mean values, encoding categorical variables using label encoding and standardizing numerical

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

features. This ensures a well -structured dataset for model training. The proposed BVQVAE model consists of two parts: BEGAN and VQVAE where BEGAN is used for balancing the dataset and VQVAE is for feature extraction and classification. The BEGAN consists of a generator and discriminator of generating samples to balance the data and differentiate between real and fake samples respectively. The VQVAE consists of encoder with quantization, which is used to map the features into finite set embeddings, and decoder is used to reconstruct network traffic patterns from latent representations. Experimental results demonstrate its performance achieving an accuracy of 82.56%, along with a along with a precision, recall, G-mean and F1-score of 86.53%, 87.65%, 86.21% and 87.08% respectively. Comparative analysis with various existing model confirms that this approach outperforms existing methods in balancing and classifying intrusion in the network layer. The inclusion of synthetic data generated by BEGAN further enhances the model's ability to learn diverse patterns, improving generalization. The future research can focus on enhancing the generative modeling process by incorporating Diffusion Models or Federated Learning approaches to further improve synthetic data quality and model robustness. Additionally, exploring transformer-based architectures for sequential feature extraction in network traffic data can further optimize detection accuracy. Expanding the study to real-time intrusion detection scenarios and testing on diverse network environments can provide additional insights into practical deployments of the proposed framework. The findings indicate that the integration of BEGAN, and VQVAE provides a robust and scalable solution for intrusion detection in network layer, setting a strong foundation for further advancements in cybersecurity and threat intelligence.

References

[1] Z. Gao, R. Nakayama, A. Hizukuri, and S. Kido, "Anomaly detection scheme for lung CT images using vector quantized variational auto-encoder with support vector data description," *Radiological Physics and Technology*, vol. 17, no. 1, pp. 1-11, 2024. doi: 10.1007/s12194-024-00851-5.

[2] H. Jebril, M. Esengönül, and H. Bogunović, "Anomaly detection in optical coherence tomography angiography (OCTA) with a vector-quantized variational auto-encoder (VQ-VAE)," *Bioengineering*, vol. 11, no. 7, p. 682, 2024. doi: 10.3390/bioengineering11070682.

[3] T. Sowmya and E. A. Mary Anita, "A comprehensive review of AI-based intrusion detection system," *Measurement: Sensors*, vol. 28, p. 100827, 2023. doi: 10.1016/j.measen.2023.100827.

[4] B. Yan and G. Han, "Effective feature extraction via stacked sparse autoencoder to improve intrusion detection system," *IEEE Access*, vol. 6, pp. 41238–41248, 2018.

[5] M. Mbow, H. Koide, and K. Sakurai, "Handling class imbalance problem in intrusion detection system based on deep learning," *International Journal of Networking and Computing*, vol. 12, no. 2, pp. 467–492, 2022.

[6] M. H. Haghighat and J. Li, "Intrusion detection system using voting-based neural network," *Tsinghua Science and Technology*, vol. 26, no. 4, pp. 484–495, 2021.

[7] Y. Imamverdiyev and F. Abdullayeva, "Deep learning method for denial of service attack detection based on restricted Boltzmann machine," *Big Data*, vol. 6, no. 2, pp. 159–169, 2018. doi: 10.1089/big.2017.0061.

[8] Y. Yang, et al., "ASTREAM: Data-stream-driven scalable anomaly detection with accuracy guarantee in IIoT environment," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 5, pp. 3007–3016, 2023. doi: 10.1109/TNSE.2022.3157730.

[9] P. Gupta, R. Prakash, and M. Kumar, "A survey on anomaly detection techniques in IoT," *Journal of Network and Computer Applications*, vol. 175, 2021, Art. no. 102911. doi: 10.1016/j.jnca.2020.102911.

[10] J. Zheng, S. Ren, J. Zhang, et al., "Binary classification for imbalanced data using data conformity mechanism," *Multimedia Systems*, vol. 31, no. 1, pp. 39–50, 2025. doi: 10.1007/s00530-024-01634-z.

[11] D. Singh, J. Valadi, H. Bhosle, A. Sane, and K. Kalunge, "Imbalance handling with combination of deep variational autoencoder and NEATER," *Association of Data Scientists*, 2023.

[12] T. Kim, Y.-G. Lee, I. Jeong, S.-Y. Ham, and S. S. Woo, "Patch-wise vector quantization for unsupervised medical anomaly detection," *Pattern Recognition Letters*, vol. 184, pp. 205-211, 2024.

[13] R. Sharma, H. Shi, J. Cai, S. P. Awate, and N. Birbilis, "Deep semi-supervised anomaly detection using VQ-VAE," in *Proceedings - 2023 International Conference on Digital Image Computing: Techniques and Applications, DICTA 2023*, pp. 273-280, IEEE, 2023. doi: 10.1109/DICTA60407.2023.00045.

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025

[14] L. Marimont and G. Tarroni, "AUROC-based anomaly detection using VQ-VAE for brain MR and abdominal scans," *IEEE Transactions on Biomedical Engineering*, vol. 70, no. 4, pp. 865–872, 2023. doi: 10.1109/TBME.2023.00582.

[15] Z. Zhou, Y. Xu, and Y. Liu, "VQ-Flow: An extended VQ-VAE for anomaly detection in MVTec AD datasets," in *Proceedings of the IEEE International Conference on Data Science and Machine Learning*, vol. 37, no. 5, pp. 155–162, 2023. doi: 10.1109/DSML.2023.00432.

[16] R. Abdulganiyu, O. Olugbara, and A. Hassan, "CWFL-VAE with XGBoost for imbalanced network traffic detection," *Journal of Computational Intelligence in Cybersecurity*, vol. 14, no. 7, pp. 564–571, 2023. doi: 10.1016/JCI-CYBER.2023.00952.

[17] "BiGAN for anomaly detection in industrial control systems," *Journal of Industrial Control Engineering*, vol. 19, no. 3, pp. 211–217, 2023. doi: 10.1109/JICE.2023.02234.

[18] "Unified deep learning approach combining Autoencoders and GANs for smart grid anomaly detection," *IEEE Transactions on Smart Grid*, vol. 12, no. 9, pp. 987–994, 2023. doi: 10.1109/TSG.2023.00123.

[19] "Conditional GANs for addressing IDS data imbalance," in *Proceedings of the IEEE International Conference on Security and Privacy*, vol. 34, no. 2, pp. 178–185, 2023. doi: 10.1109/SP.2023.01092.

[20] "GANs in UAV security for real-time intrusion detection using Active Learning," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, no. 1, pp. 85–92, 2023. doi: 10.1109/TAES.2023.01234.

[21] "Data Generative Model (DGM) combining CGANs and KL-divergence for improved detection rates," in *Proceedings of the International Conference on Machine Learning and Cybersecurity*, vol. 45, no. 8, pp. 299–305, 2023. doi: 10.1109/MLC.2023.00958.

[22] "Hybrid model integrating GANs and Autoencoders for IDS," *Journal of Cybersecurity Technology*, vol. 21, no. 4, pp. 105–112, 2023. doi: 10.1109/JCT.2023.00928.

[23] "G-IDS: Combining GANs and Autoencoders for intrusion detection," *International Journal of Security and Networks*, vol. 17, no. 5, pp. 251–258, 2023. doi: 10.1002/ISN.2023.00497.

[24] "SMOTE adaptations for IDS data balancing," *Journal of Artificial Intelligence Research*, vol. 40, no. 3, pp. 223–230, 2023. doi: 10.1007/JAI-2023.00856.

[25] J. Seo, B. Lee, and H. Kim, "Adversarial attacks on ML-based IDS in automotive security," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 498–507, 2023. doi: 10.1109/TITS.2023.00893.

[26] "Combining VAEs and GANs with deep classifiers for IDS," *International Journal of Network Security*, vol. 45, no. 6, pp. 315–321, 2023. doi: 10.1016/ijnse.2023.01274.

[27] D. Berthelot, T. Schumm, and L. Metz, "BEGAN: Boundary Equilibrium Generative Adversarial Networks," 2017.

[28] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural Discrete Representation Learning," 2018.

[29] The 420. (2025). Bashe hacking group claims ICICI Bank data breach; ransom deadline Jan 24, 2025. *The 420*. Retrieved from https://www.the420.in/bashe-hacking-group-claims-icici-bank-data-breach-ransom-deadline-jan-24-2025

[30] Madras Pioneer. (2025). Security breaches leak student, employee data at 509J. Madras Pioneer. Retrieved from https://www.madraspioneer.com/townnews/software/security-breaches-leak-student-employee-data-at-509j/article_945830a8-d78a-11ef-aba1-f3909a813451.html

[31] Kaggle. (n.d.). NSL-KDD dataset. Retrieved from https://www.kaggle.com/datasets/hassan06/nslkdd.

[32] J. He, X. Wang, Y. Song, et al., "Network intrusion detection based on conditional wasserstein variational autoencoder with generative adversarial network and one-dimensional convolutional neural networks," *Applied Intelligence*, vol. 53, no. 12, pp. 12416–12436, 2023. doi: 10.1007/s10489-022-03995-2.

[33] E. Redekop, M. Pleasure, Z. Wang, K. Sarma, A. Kinnaird, W. Speier, and C. Arnold, "Codebook VQ-VAE Approach for Prostate Cancer Diagnosis using Multiparametric MRI," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 2365-2372, 2024.

DOI: https://doi.org/10.54216/JCIM.160202
Received: November 22, 2024 Revised: January 23, 2025 Accepted: March 03, 2025