Volume 14, Issue 4
August 2025

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Mitigating Mode Collapse Problem in GANs using Data Augmentation techniques for Cloud anomaly detection

[1]Uma maheswari K and **[2]Shobana G**

[1]Department of Computer Science, Bharathi Women's College, Chennai - 600 108, india

[2]Department of Computer Science, Jeppiaar Engineering College, Chennai - 600 119, India

e-Mail: [1]uma.tvr1981@gmail.com, [2] drshobanagm@gmail.com

ABSTRACT

The Mode collapse problem is a common challenge for Generative Adversarial Networks (GANs), which hinders its effectiveness for anomaly detection types of tasks. In this work, the potential of data augmentation techniques is investigated for the mitigation of mode collapse problem and the performance improvement of GANs for anomaly detection in cloud environments. For this research CIC-IDS2018 dataset is leveraged that contains normal and anomalous cloud traffic data, both are labeled for learning. DCGAN architecture is implemented and trained in the beginning for cloud anomaly detection tasks. The performance of the system is recorded with the metrics of accuracy, precision, recall and F1 Score. Three types of data augmentation techniques on Gaussian noise injection, time series augmentation, and scaling are applied with the CIC-IDS2018 dataset in the model. The performance is again monitored to check the effectiveness of augmentation techniques in mitigating the mode collapse problem that is indirectly measured with the ROC curves to enhance the overall accuracy and precision of anomaly detection in the cloud. The system has proved the significance of data augmentation for the improvement of performance in DCGAN models applicable in cloud environments. Time-based augmentation techniques are proved to be effective for the dynamic nature of cloud network traffic data with results showing better accuracy than other augmentation techniques. MSE/MAE graphs have shown the mitigation of mode collapse problem in the anomaly detection application of DCGAN models.

Keywords: *Generative Adversarial Networks, Cloud Anomaly Detection, Mode collapse problem, Data augmentation techniques*.

1. INTRODUCTION

Generative Adversarial Networks (GANs) are showing the powerful performance in the generation of high-quality samples in various domains of image, video and text-based content. GANs have a great capability to learn difficult probability distributions in all kind of data spaces [31]. However, the challenge rises in handling the unbalanced datasets with insufficient faulty data or redundant data with the same condition. System can suffer from mode collapse in producing monotonous new signals and the results are further imbalanced [33]. Application of GANs in the area of anomaly detection in cloud environment is the crucial requirement at this time. This work aims to study the application of data augmentation techniques particularly tailored to cloud traffic data and its impact on mitigating mode collapse in GAN-based anomaly detection. As there is only limited research on applying data augmentation for cloud traffic data, the proposed system fills the gap by performing domain-specific augmentation for cloud anomaly detection. The system also quantifies the impact of data augmentation by drawing ROC curves for the assessment of diversity of generated data to indicate mode collapse mitigation. The insights and models are built using the dataset to simulate real-world cloud network traffic data from AWS environment.

1.1 DCGAN in Cloud Anomaly Detection

A DCGAN (Deep Convolutional Generative Adversarial Network) plays a vital role in cloud anomaly detection. It learns the patterns of normal cloud traffic data and flags any deviations from those patterns as potential anomalies. It consists of two parts: a generator and a discriminator [32]. Initially, the generator is trained on real-time and labeled cloud traffic data that contains information on packet size, resource usage metrics, and timestamps. It also attempts to generate new data points that closely resemble real-time cloud traffic. The discriminator must distinguish between the real cloud traffic and the data generated by the generator [34]. When the generator is successful, the discriminator will have difficulty to tell them apart. When the DCGAN encounters data points with significant deviation, the discriminator part will be able to identify the discrepancy with higher confidence [30]. DCGANs are well-suited for the identification of subtle anomalies in cloud traffic than any other methods because of their excellence in capturing complex data patterns [13].

1.2 Mode collapse problem

It is a common challenge addressed in GANs, where the generator part gets stuck inside a loop, by producing only the same kind of data points[12]. This will hinder the ability of the model to detect diverse anomalies. The generated data becomes homogeneous and unrealistic, which leads to failure in capturing the true diversity of the real-world data. Mode collapse essentially causes the computationally expensive training time of the DCGAN to be wasted. It leads to the model learning the full scope of the data it was supposed to learn from. Performance evaluation of DCGAN is based on the metrics that assess the diversity and realism of the data generated by the

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

model. Mode collapse makes this evaluation to be difficult to gauge the model's understanding of the data distribution [24].

1.3 Data Augmentation Techniques

Data augmentation is a method to artificially expand the size and diversity of training data without having to collect more data points. This can address the issues of overfitting and improve the model's performance on generalized, unseen data [1]. Various augmentation techniques are available based on the type of data as image, text, or other mixed data. Image augmentation can be done using Geometric transformations, color augmentation, or Noise injection. Text augmentation is performed with syntactic transformations and Back-translation methods. Other techniques involve mixing data and time warping etc. The proposed model has the potential to produce improved accuracy on results by using data augmentation techniques particularly tailored for cloud traffic data [16].

2. LITERATURE SURVEY

Among the research techniques for Data Augmentation, Mode collapse mitigation, and application of GANs for cloud anomaly detection, only limited works have been available to combine the techniques for mode collapse mitigation in a cloud environment. A Review work [26] on anomaly detection for cloud computing environments identifies the corresponding models in AI methodological areas of machine learning, deep learning and statistical approaches for the anomaly detection. The review also point out the concrete application areas which are addressed by the cloud computing environments and the related public datasets used in the evaluation process. A survey work performed on the data augmentation for Deep Learning [23] which is not specific to cloud data but provides a foundation for exploring the suitable augmentation techniques. The methodology of image augmentation technique with DCGAN from J,Maeda et al. [15] has shown the significance of augmentation in the context of DCGAN based applications. A model on the application of augmentation techniques for image classification in fruit recognition [4] is the base paper for domain-specific augmentation, for tailoring augmentation to the particular datatype, adaptable for cloud traffic data.

An Improved GAN Architecture [20] is proposed by the research for exploring alternative GAN architectures that may be less susceptible to mode collapse problem. The work has presented different architectural features and training procedures applicable to GANs framework. The high quality images generated by the model were confirmed by a visual Turing test. The work in reference 2 has discussed the application of latent variable models for learning both the probability distribution of the data and the identification of hidden structures in the data. The work has suggested the establishment of theoretical relationships among different methods of learning probability distributions in the data. A promised work on Spectral Normalization for the mitigation of mode collapse in GANs [17] paves the way for model formation and evaluation. The work has confirmed the capability of SN-GANs for the generation of better quality images relative to the previous training stabilization techniques.

The application of GANs for Anomaly Detection is introduced by an influential work named AnoGAN [21] however it doesn't address the mode collapse problem. The work has performed unsupervised learning for the identification of anomalies in imaging data as candidates for markers. CloudGAN [29] is one of the state-of-the-art model that employs an adaptive weighing technique for the optimization of training process of variational auto-encoders based on adaptive weights for the enhancement of different layer features and to avoid indiscrimination and biased learning. A survey work [7] for the research of deep anomaly detection with the advancements in 11 fine-grained categories has reviewed the objective functions, underlying assumptions, advantages and disadvantages of different methods. A work for combating Mode Collapse [19] in GAN training has proposed an optimization algorithm called nudged-Adam (NuGAN) that uses spectral information to overcome mode collapse. The work investigated the instabilities occurring during the training of GANs, focusing on the issue of mode collapse. The investigation of generalization properties of GANs by analyzing the flatness of the optima found during training suggests a promising approach to progress towards stable GAN training. The model also found a connection between the spectrum of the Generator and the mode collapse. An algorithm of manifold guided GAN (MGGAN) [5] was proposed to leverage a guidance network on existing GAN architecture to induce generator learning in all modes of data distribution. The generator avoids mode missing by getting feedback for the mode coverage of a data distribution from the guidance network. A type of generative diversity called uniform diversity relating to u-mode collapse was handled by an algorithm named UniGAN [35] with a normalizing flow based generator. A review work [27] on GANs which employs algorithms to reduce mode collapse problem finds out the capabilities and issues of GANs. A novel data augmentation technique [25] proposed for the class imbalance problem and mode collapse in DCGAN for the realistic tabular data. The method performs encoding of each column's data to produce a feature map for each record and then converted back to its original tabular form as an intermediate image format. A technique [18] of introducing an array of co-operative realness discriminators into the GAN framework to reduce mode collapse is introduced to generate realistic and diverse images. Synthesizing data samples by a data augmentation technique[14] in a tabular data environment has proposed to identify the characteristics of a dataset to provide a better performance. A technique [28] to adopt an in-depth exploration approach for the domain of generative machine learning integrated with cloud services is proposed with MNIST data. The method highlights the significance in addressing the challenges of data generation in GPU-enabled computational engines. An approach [22] was proposed to describe the tools facilitating data augmentation for the application of task-specific augmentations. The method proposed the ideas for AI-Gas (AI-Generating algorithms) for text data augmentation. 'Easy data augmentation techniques' [10] is the work to boost performance on text classification tasks that improves both convolutional and recurrent neural networks. Extensive ablation studies were performed that suggests parameters for the practical applications. An anomaly

Volume 14, Issue 4
August 2025

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

detection model [6] proposed in openstack. cloud environment has used Stacked and Bidirectional LSTM models to build the neural network.

3. PROPOSED METHODOLOGY

The proposed system employs a CICIDS2018 dataset for cloud anomaly detection as it contains labeled entries of both normal and anomalous cloud traffic data. The methodology involves four steps: 1. Data Acquisition and Preprocessing 2. GAN model selection and training 3. Data Augmentation techniques 4. Model performance evaluation. The steps involved in the model are shown in figure 1 and the following sections depict this in detail.

3.1 Data Acquisition and Preprocessing

A CICIDS2018 Dataset [9] is selected for Cloud Anomaly Detection which requires preprocessing to handle missing values, to scale numerical features and to consider creating new features with existing ones. Then it is required to split the data into training, validation and testing sets. Data Augmentation is to be applied only to the normal traffic data in order to avoid corrupting anomaly labels. The CICIDS2018 dataset encloses different attack scenarios on Brute-Force, Heartbleed, Botnet, DoS, DDoS, Web attacks and network infiltration from inside. The dataset contains the network traffic captures and system logs of about 420 machines and 30 servers and includes 80 features extracted traffic using CICFlowMeter-V3.

3.2 GAN model selection and Training

DCGAN architecture is selected for its effectiveness in performing anomaly detection task which is as shown in figure 2. The DCGAN model is implemented using TensorFlow framework. Hyperparameters on learning rate and optimizer settings are tuned for optimal model performance. Then the DCGAN model is trained on the preprocessed training data with separate loss functions suitable for generator and discriminator components.

3.2.1 Generator (G)

The DCGAN generator (G) creates realistic representations of benign cloud network traffic patterns on the basis of training data. Input layer takes as input a random noise vector (Z) and generates as output a synthetic network traffic sample G(Z) with the help of convolutional layers. The size of the noise vector is about 100 dimensions that determines the complexity of the features that generator can create. There will be a fully-connected layer after the input layer to transform Z into a format suitable for subsequent convolutional layers. The transposed Convolutional layers perform "upsampling" operations to gradually increase the spatial resolution of the maps to transform input noised vector into a representation of a network traffic sample. A Leaky feature ReLU activation function is applied to introduce non-linearity that allows the network to learn complex relationship between the input noise and the generated traffic sample. The final output layer uses a tanh activation function for the mapping of generated featuremaps to the range of -1 to 1 representing network traffic data.

3.2.2 Discriminator (D)

The discriminator (D) will distinguish the real network traffic samples from the dataset and the synthetic samples of generator. Input layer takes a network traffic sample from dataset or a synthetic generated by G as input. Convolutional layers perform downsampling operations will be used for the classification of sample as real or fake.The convolution layers apply filters to the input and extract local features to capture specific patterns in the network traffic. Leaky ReLU activation function is used to introduce non-linearity to allow the discriminator to learn complex relationship in features. Pooling layer is used to reduce the dimensionality in the feature maps by max pooling technique. There is a flatten layer that transforms the final feature maps from a multi-dimensional

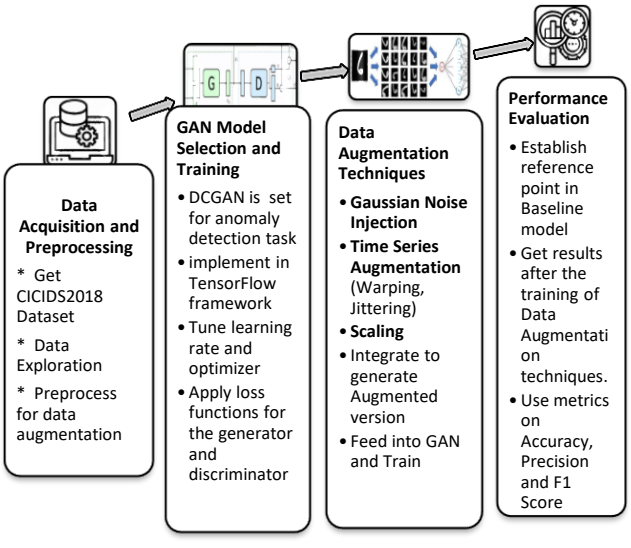

Figure. 1. Outlined Methodology

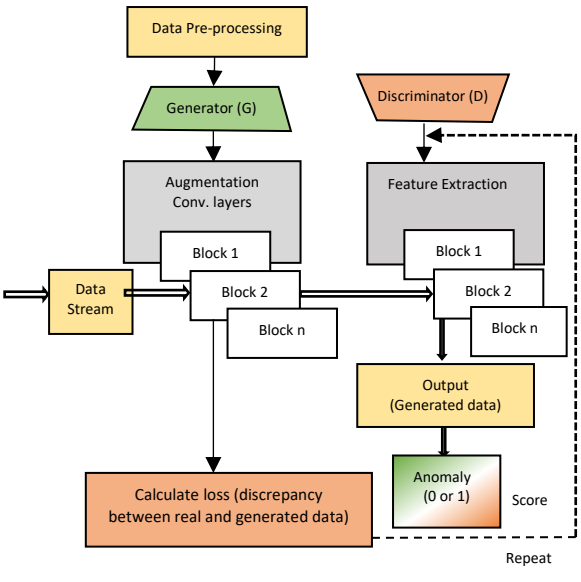

Figure. 2. Architecture of DCGAN Model

Volume 14, Issue 4
August 2025

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Table. 2. Hyper parameter tuning for the DCGAN anomaly detection model

Hyperparameter Configuration	Generator Loss (Avg.)	Discriminator Loss (Avg.)	Training time (Epochs)	Sample Quality	Anomaly Detection Metrics (Test Set)
Config 1 (Batch Size:32)	0.075	0.052	50	Moderate	Precision: 0.72 Recall: 0.68 F1: 0.70
Config 2 (Batch Size:128)	0.048	0.069	100	High	Precision: 0.78 Recall: 0.65 F1: 0.71
Config 3 (SGD optimizer)	0.082	0.061	75	Moderate	Precision: 0.70 Recall: 0.74 F1: 0.72
Config 4 (Number of layers increased from 5 to 7)	0.035	0.028	150	High	Precision: 0.78 Recall: 0.65 F1: 0.71

format to a one-dimensional vector. A Fully-connected layer is used to combine the extracted features from the convolution layers to a single output value. Finally a output layer is employed to output a probability between 0 and 1 by using sigmoid activation function. A value that lies closer to 1 indicates the discriminator assuming the sample is real or normal traffic, while a value closer to 0 indicates a fake or anomaly.

3.2.3 Anomaly Detection in DCGAN

In training phase, the generator and discriminator components are trained in adversarial manner. The generator will have trained on its ability to fool the discriminator by generating realistic samples, while the discriminator will have improved its ability in distinguishing real from fake traffic patterns. After the training process, discriminator will perform the anomaly detection task. The output of the discriminator identifies the likelihood of the sample being real i.e., normal traffic or fake i.e., potential anomaly. Those samples with a low chance of being real are considered to be the potential anomalies.

3.2.4 Model tuning for DCGAN

The training process is significantly affected by the tuning of hyper parameters like learning rates, batch sizes and optimizers. Grid search or random search is used to find the optimal hyper parameter values for the anomaly detection task. The above table 1 indicates the results of hyper parameter tuning for the DCGAN anomaly detection model built with TensorFlow.

3.3 Data Augmentation Techniques

Suitably three types of data augmentation techniques[1] are selected for cloud traffic data. They are 1. Gaussian Noise Injection – to add controlled noise to the features available in the dataset. 2. Time Series Augmentation – to apply warping, jittering for introducing variations in the temporal patterns. 3. Scaling – for specific features to simulate variations in resource usage or traffic volume. Then the system needs to be implemented and integrated with the augmentation results. The chosen data augmentation techniques are integrated into a data pipeline for the generation of augmented versions of normal traffic data in order to feed into the GAN for training purpose.

3.3.1 Gaussian Noise Injection

By means of adding Gaussian noise with a zero mean ($\mu=0$) to the features of a data point, small random variations around the original values can be introduced [3]. The magnitude of these variations are controlled by the standard deviation (σ) of the noise. Let X be the original data point denoted by a feature vector: X=[x1, x2, ..,xn]The noise injection process is indicated by:

$$X\_augmented = X + \varepsilon \quad (1)$$

Here ε is a noise vector of the same dimension of X, contains random values drawn from the Gaussian distribution N(O,σ^2).This can be modeled as the conditional probability distribution of the augmented data point (X_augmented) with the original data point X.

$$p(X\_augmented \mid X) = p(\varepsilon) * p(X\_augmented - \varepsilon \mid X) \quad (2)$$

Here $p(\varepsilon)$ is the probability density function of the Gaussian noise distribution. Since the noise is added independently, this term can be simplified to the original data distribution p(X) as:

$$p(X\_augmented \mid X) = p(\varepsilon) * p(X) \quad (3)$$

A convolution operation is applied between original data distribution p(X) and the noise distribution $p(\varepsilon)$ to obtain the overall distribution of the augmented data.

3.3.2 Time series augmentation

Time series data means the sequential data points representing a value or state over time. This kind of augmentation artificially creates new time series examples while preserving the essential temporal characteristics. Here 5 different techniques are applied to implement the time series augmentation [8].

(1) Time Shifting: This method shifts the entire time series to forward or backward by a specific number of time steps. By this kind of augmentation, the model can learn to recognize patterns regardless of their absolute position in the time series. The time-shifted series (Y) with a shift of k units on the original time series of length k units (X) is given by:

$$Y = [x(k+1), x(k+2), \ldots, xn, \ldots, x1, x2, \ldots xk] \text{ for } k>0, \text{ forward shift} \quad (4)$$

$$Y = [xn, x(n-1), \ldots, x(k+1), xk \ldots, x1] \quad (5)$$

(for k<0, backward shift)

(2) Time Stretching / Compression: This method stretches or compresses the time series through varying the interval between data points. Stretching increases the length of the series and the compression reduces it. Hence the model can learn patterns under different temporal scales.

Stretching with a factor s>1 is denoted as,

$$Yi = (1-alpha) * X[floor(i/s)] + alpha * X[ceil(i/s)] \quad (6)$$

Here, alpha = i/s – floor(i/s) is the interpolation weight between the neighbors.

Volume 14, Issue 4
August 2025

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

(3) Adding Noise: This method adds controlled noise to the time series data points that can make the model to learn robustness to noise present in real-world data.

(4) Random segment replacement: This method replaces a random segment of the time series with a segment from another time series to make the model learn to handle variations in data patterns within a single sequence.

(5) Frequency Masking: It focuses on the frequency domain representation of the time series data. This technique intends to masking out specific frequency bands and performing inverse transform to obtain a time series with altered frequency content. This can make the model to learn pattern recognition independent of specific frequency components.

3.3.3 Scaling

It is the method of simulating variations in resource usage or traffic volume or employing specific techniques for the creation of new data points to reflect these variations25. Here 4 different methods are employed for scaling type of data augmentation.

(1) Random Scaling: It scales the entire data point by a random factor within a predefined range for the resource usage of traffic volume. The randomly scaled data point (Y) is obtained using a scaling factor (s) drawn from uniform distribution is denoted as,

$$Yi = s * Xi \text{ (for all i=1 to n)} \quad (7)$$

Here s is a random scaling factor from a uniform distribution within predefined range of 0.8 to 1.2 to simulate variations in the range of 80% to 120% of the original resource usage or traffic volume.

(2) Feature-wise Scaling: It scales individual features within the data points by different random factors to gain more granular control over the variations in resource usage or traffic volume. It is denoted by,

$$Yi = si * Xi \text{ (for all i=1 to n)} \quad (8)$$

Here Yi is the i-th feature value in the scaled data point while s-I is the random scaling factor drawn from uniform distribution for simulating independent variations in different resource types.

(3) Time-warping (for time series data) : It involves stretching or compressing the time series in order to simulate variations in the temporal patterns of resource usage.

(4) Combined technique: The above techniques can be combined for more complex augmentation strategies. As an example, the entire data points are randomly scaled and then applied with feature-wise scaling for further variations.

3.4 Performance Evaluation

Initially a Baseline DCGAN model is trained without data augmentation for establishing a reference point towards performance comparison. Then separate DCGAN models are trained with different combinations of data augmentation techniques from the baseline architecture. The model performance is evaluated using the metrics on Accuracy, Precision, Recall and F1 score. The baseline DCGAN model is trained on CICIDS2018 dataset with 7 types of network traffic. The model is trained over 10 days of network traffic with 17% of the instances are attack traffic. According to the survey conducted by Joffrey L. Leevy [2], best performance scores for the models that were using CICIDS2018 dataset were unusually high, because of the consequence of model overfitting and also there were apparent lack of concern in for the class imbalance in the dataset. Here, in the proposed system, the two drawbacks are resolved by the application of data augmentation.

4. RESULTS

The baseline DCGAN model's performance is compared with the model after the application of different data augmentation techniques based on various output metrics are as shown in the following tables from 2 to 8. Table 2 shows the

Table 2. Baseline DCGAN model metrics

Input Parameters		Output Metrics	
Noise Dimension	100	Accuracy	72%
Generator Learning Rate	0.0002	Precision	65%
Discriminator Learning Rate	0.0008	Recall	55%
Batch Size	64	F1-Score	59%
Epochs	50	Training Time (hours)	3
Batch size = no. of samples processed together			

Table 3. Comparison of Baseline DCGAN and DCGAN with Gaussian Noise Injection

Input parameters			Performance			
Input Metrics	Baseline value	Gaussian Noise Injection Value	Output Metrics	Baseline DCGAN	DCGAN with Gaussian Noise Injection	Difference
Learning Rate	0.001	No Change	Accuracy	78%	89%	Potential slight increase
Batch Size	16	No Change	Precision	70%	82%	Potential slight increase
Number of Epochs	80	No Change	Recall	61%	71%	Potential slight increase
Noise Mean	N/A	0	F1-Score	65%	85%	Potential slight increase
Noise Standard Deviation	N/A	0.01	Training Time (hours)	4	5	Potential slight increase
Noise mean is Mean value of the injected noise; Noise Standard Deviation Controls the amount of noise						

Table 4. Comparison of Baseline DCGAN and DCGAN with Time Shifting

Input parameters			Performance			
Input Metrics	Baseline value	Time Shifting Value	Output Metrics	Baseline DCGAN	DCGAN with Time Shifting	Difference
Learning Rate	0.001	0.0008	Accuracy	77%	96%	Potential increase
Batch Size	32	64	Precision	69%	88%	Potential increase
Number of Epochs	120	150	Recall	58%	79%	Potential increase
Shifting Range	N/A	+/-5 units	F1-Score	63%	88%	Potential increase
Shifting range is set for network traffic analysis;			Training Time (hours)	5	7	Potential increase

Volume 14, Issue 4
August 2025

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Table 5. Comparison of Baseline DCGAN and DCGAN with Time Stretching / Compression

Input parameters				Performance		
Input Metrics	**Baseline value**	**Time Stretching / Compression Value**	**Output Metrics**	**Baseline DCGAN**	**DCGAN with Time Stretching / Compression**	**Difference**
Learning Rate	0.001	0.0009	**Accuracy**	76%	90%	Potential slight increase
Batch Size	16	48	**Precision**	67%	83%	Potential increase
Number of Epochs	100	120	**Recall**	57%	75%	Potential increase
Stretching /Compression Factor	N/A	1.2 (Stretch) / 0.8 (compress)	**F1-Score**	62%	86%	Potential increase
Batch size increased for efficiency; Stretching / Compression Factor denotes the ratio for scaling the time series			**Training Time (hours)**	6	8	Potential increase

Table 6. Comparison of Baseline DCGAN and DCGAN with Random Scaling

Input parameters				Performance		
Input Metrics	**Baseline value**	**Random Scaling Value**	**Output Metrics**	**Baseline DCGAN**	**DCGAN with Random Scaling**	**Difference**
Learning Rate	0.001	No Change	**Accuracy**	75%	82%	Potential decrease
Batch Size	32	No Change	**Precision**	68%	75%	Potential decrease
Number of Epochs	80	No Change	**Recall**	59%	72%	Potential increase
Scaling Range	N/A	0.8 – 1.2	**F1-Score**	63%	73%	Similar / slight decrease
Scaling Range defines the minimum and Maximum scaling factors			**Training Time (hours)**	4	5	Potential slight increase

Table 7. Comparison of Baseline DCGAN and DCGAN with Feature-wise Scaling

Input parameters				Performance		
Input Metrics	**Baseline value**	**Feature-Wise Scaling Value**	**Output Metrics**	**Baseline DCGAN**	**DCGAN with Feature-Wise Scaling**	**Difference**
Learning Rate	0.001	No Change	**Accuracy**	78%	92%	Similar
Batch Size	16	No Change	**Precision**	70%	81%	Potential Slight increase
Number of Epochs	80	No Change	**Recall**	62%	73%	Potential Slight increase
Scaler Type	N/A	Standard Scaler	**F1-Score**	66%	77%	Potential Slight increase
Scaler Type defines the scaling method StandardScaler, MinMaxScaler			**Training Time (hours)**	4	4	Similar

Table 8. Comparison of Baseline DCGAN and DCGAN with Time-Warping

Input parameters				Performance		
Input Metrics	**Baseline value**	**Time Warping Value**	**Output Metrics**	**Baseline DCGAN**	**DCGAN with Time Warping**	**Difference**
Learning Rate	0.001	0.0008	**Accuracy**	77%	95%	Potential Increase
Batch Size	32	64	**Precision**	69%	87%	Potential Increase
Number of Epochs	120	150	**Recall**	58%	80%	Potential Increase
Warping Degree	N/A	0.1	**F1-Score**	63%	92%	Potential Increase
Warping Degree defines the maximum allowed stretching/compression ratio			**Training Time (hours)**	5	7	Potential Increase

performance metrics of the DCGAN model in the beginning. In table 3, the effect of applying the Gaussian Noise Injection type of data augmentation can be found. Initially the noise mean is set to 0 then increased gradually with the noise standard deviation of 0.01. By applying different data augmentation techniques, thorough increase in the performance can be recognized by avoiding model overfitting. Table 4, shows the raise in accuracy after the application of Time shifting based data augmentation. In table 5, Time stretching / compression-based augmentation is applied to show the potential increase in precision and Recall. Table 6 shows the change in metrics after the application of random scaling in the range of 0.8 to 1.2. In table 7, the results after the application of Feature-wise scaling with the scaling methods on StandardScaler, MinMaxScaler are shown. Table 8 shows the potential increase in accuracy and Precision with the warping degree of 0.1 applied. Time warping and Time shifting kind of data augmentation required tremendous increase in the number of epochs and training time. The batch size also increased to 64 for the Time stretching and warping type of augmentation techniques.

5. DISCUSSION

The impact of Data Augmentation Techniques in the mitigation of mode collapse problem can be assessed by means of comparing the anomaly detection parameters on Learning Rate, Accuracy, Precision, Recall and F1 Score as shown in Figure 3. It can be seen that after the application of data augmentation, there is a notable improvement in the baseline model on the different parameters of accuracy, precision, recall and F1 score. In particular, Time shifting kind of data augmentation exhibits good results on 96% accuracy and Time warping results in 95% of accuracy. Feature-wise scaling type of augmentation gives 81% of precision as the highest value. Recall value of 80% and F1-score of 92% results from Time-

Volume 14, Issue 4
August 2025

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Figure 3. Anomaly Detection Performance Comparison of the Baseline DCGAN Model with the Data Augmentation Techniques applied models on Gaussian noise injection, Time shifting, Time Stretching/Compression, Random Scaling, Feature-wise Scaling and Time Warping.

Warping kind of data augmentation also this required the 150 number of Epochs as the maximum value. The anomaly detection results of 15 different attack types and their prediction percentage are shown in table 9. The table lists the prediction percentage after the application of data augmentation as a result of avoiding mode collapse in the listed attacks of CICIDS2018 dataset. The overall output metrics comparison for the baseline model with the applied data augmentation techniques are given in figure 4. Figure 5 shows the ROC curves with improvement in anomaly detection performance of baseline model after the application Time Shifting and Feature-wise scaling type of augmentation with the maximum AUC of 0.95. The system reveals the mitigation of Mode collapse by showing a decreasing trend in Mean Absolute Error (MAE) and Mean Squared Error (MSE) in Figure 6 for the Time Shifting and Time Warping type of data augmentation. As MSE is more sensitive to outliers than MAE, it is affected more than MAE with few outliers. The above results reveal that Time Shifting and Time Warping are effective augmentation techniques for DCGAN model in anomaly detection for the network traffic data. The methods capture temporal dependencies and add variations in the data patterns. Excessive shifting or warping kind of transformation distort the underlying data distribution and the system produces a degrade in performance. Gaussian Noise Injection improves the robustness of the model but also introduces noise that masks subtle anomalies. However, the Feature-wise Scaling and Random-Scaling are more suitable for image type of data and not seem to be effective in cloud network traffic data with numerical time series of values. The system's contribution in DCGAN based Anomaly Detection for Cloud Environment is listed as follows:

- The proposed system explores the effectiveness of data augmentation techniques on specific attack data available with CICIDS2018 dataset and scales resource usage metrics for anomaly detection in cloud.
- The robustness of the model is improved while mitigating the mode collapse problem in DCGAN as shown with MAE / MSE graphs.
- The proposed methodology is applicable to unseen data or the detection of new kind of anomalies in the CICIDS2018 dataset by avoiding Mode Collapse problem.

6. CONCLUSION

The proposed system has done the evaluation on the effectiveness of various data augmentation techniques for the mitigation of mode collapse problem occurring in DCGAN-based anomaly detection systems particularly tailored for the

Volume 14, Issue 4
August 2025

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

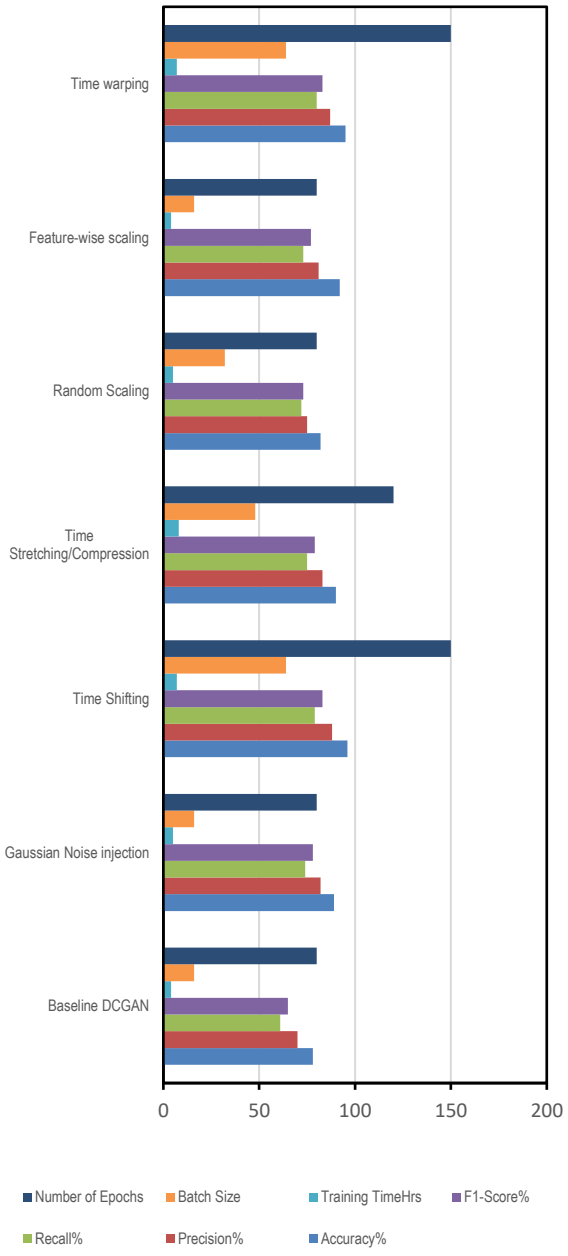

Figure 4. Output Metrics Comparison of the Baseline DCGAN with Data Augmentation applied models.

Table 9. Overall attack prediction percentage for the DCGAN model with Data Augmentation Techniques

Attack Name	Prediction Percentage %
FTP-BruteForce	99.56
SSH-Bruteforce	99.73
DoS-GoldenEye	99.02
DoS-Slowloris	99.86
DoS-SlowHTTPTest	99.76
DoS-Hulk	99.52
DDoS attacks-LOIC-HTTP	99.03
DDoS-LOIC-UDP	99.15
DDOS-LOIC-UDP	99.25
DDOS-HOIC	99.54
Brute Force -Web	86.26
Brute Force -XSS	78.34
SQL Injection	89.57
Infiltration	89.63
Bot	100

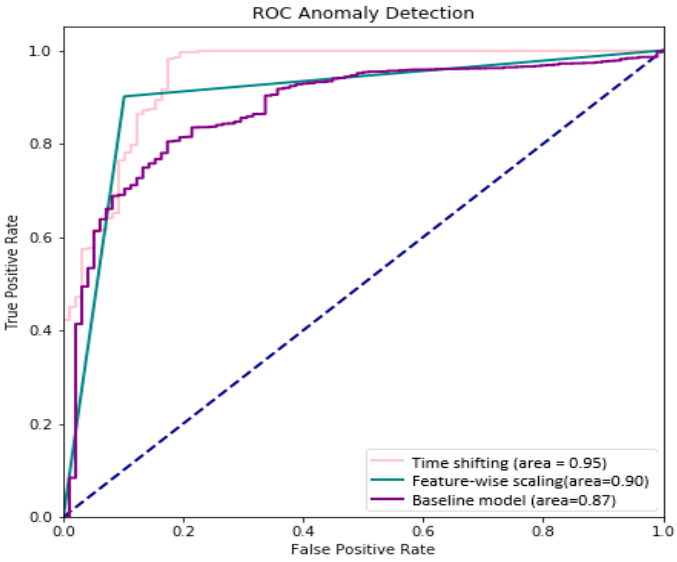

Figure 5. ROC Anomaly Detection Comparison of the 2 models.

cloud environment. With the utilization of CICIDS2018 dataset, it has found that Time-based augmentations on Time Stretching/Compression, Time Shifting and Time Warping has led to better performance than noise injection techniques on Gaussian Noise Injection. Scaling type of data augmentation on Random scaling and Feature-wise scaling are not effective in the particular dataset and performance degradation observed. The lower and more stable error values in the MAE/MSE graphs revealed the better anomaly detection performance of the model after the application of data augmentation techniques. ROC curves have provided more valuable insights into the trade-off between true positive and false positive rate for the data augmentation techniques applied model. By the introduction of diversity with augmentation of training data, the methodology has significantly improved the model's ability in capturing complex distribution of cloud network traffic patterns and resource usages. The proposed methodology has DCGAN to generate more realistic and diverse synthetic data, hence resulting in improved anomaly precision, recall and F1-score than the base-line model. The findings have highlighted the potential of data augmentation as an effective tool for the enhancement of efficacy of anomaly detection in the diversified data of cloud computing environment. Further research could explore the application of combined application of different augmentation techniques, developing adaptive augmentation strategies for the specific attack types, and enhancing the work to other types of network traffic data. By the systematic investigation on the impact of different augmentation techniques, the methodology has proved the development of robust and effective anomaly detection solution for cloud environment.

REFERENCES

[1] Alhassan Mumuni & Fuseini Mumuni, "Data augmentation: A comprehensive survey of modern approaches", Array. 16 (2022) 100258.

[2] Arjovsky.M et al. "Towards practical probabilistic approaches to variational autoencoders", (2016) arXiv preprint arXiv:1606.04934.

[3] Arslan.M et al. "SMOTE and Gaussian Noise Based Sensor Data Augmentation" in 4th International Conference on Computer Science and Engineering (UBMK), Samsun, Turkey, (2019) 1-5.

[4] Bustelo.A et al. "Augmentation techniques for deep learning-based image classification in fruit recognition, Sensors", 20(14) (2020) 3953

[5] Duhyeon Bang &Hyunjung Shim, "MGGAN: Solving Mode Collapse Using Manifold-Guided Training" in Proceedings - IEEE/CVF International Conference on Computer Vision Workshops, ICCVW, IEEE. (2021) 2347-2356

[6] Girish, L., Rao., S.K.N, "Anomaly detection in cloud environment using artificial intelligence techniques", Computing. 105 (2023) 675–688.

[7] Guansong et al, "Deep Learning for Anomaly Detection: A review", ACM Computing Surveys (CSUR). 54(2) (2021) 1-38.

[8] Guillermo et al. "Data Augmentation techniques in time series domain: a survey and taxonomy", Neural Computing and Application. 35 (2023) 10123–10145.

[9] https://www.unb.ca/cic/datasets/ids-2018.html

[10] Jason Wei, Kai Zou, "EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks", (2019) arXiv preprint arXiv:1901.11196.

[11] Jeong J, Jeong H and Kim J, "BAMTGAN: A Balanced Augmentation Technique for Tabular Data" in 9th International Conference on Applied System Innovation (ICASI). Chiba, Japan, (2023) 205-207, doi: 10.1109/ICASI57738.2023.10179533

[12] Kossale.Y, Airaj.M & Darouichi.A, "Mode Collapse in Generative Adversarial Networks: An Overview" in 8th International Conference on Optimization and Applications (ICOA), Genoa, Italy (2022)1-6.

[13] Li.W. et al. "Tackling mode collapse in multi-generator GANs with orthogonal vectors. Pattern Recognition", 110 (2021) 107646.

[14] Machado, P., Fernandes, B., Novais, P. "Benchmarking Data Augmentation Techniques for Tabular Data in Intelligent Data Engineering and Automated Learning – IDEAL", Lecture Notes in Computer Science. (2022) 13756, Springer, Cham.

[15] Maeda.J et al., "Image augmentation using deep convolutional generative adversarial networks", (2018) arXiv preprint arXiv:1801.07873

[16] Mikołajczyk, Agnieszka, & Michał Grochowski, "Data augmentation for improving deep learning in image classification problem", 2018 international interdisciplinary PhD workshop (IIPhDW), IEEE. (2018) 117-122.

[17] Miyatoet.T et al., "Spectral normalization for generative adversarial networks" in Proceedings of the 34th International Conference on Machine Learning. 70 (2018) 2967-2975.

[18] Mu J, Chen C, Zhu W, Li S, Zhou Y. "Taming mode collapse in generative adversarial networks using cooperative realness discriminators", IET Image Process, (2022) 16: 2240-2262. doi:10.1049/ipr2.12487

[19] Richard Durall et al. "Combating Mode Collapse in GAN training: An Empirical Analysis using Hessian Eigenvalues", (2020) arXiv:2012.09673v1

[20] Salimans.T et al. "Improved techniques for training GANs. Advances in Neural Information Processing Systems", (2016) 2234-2242.

[21] Schlegl et al. "Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery.Information Processing in Medical Imaging", IPMI 2017, Lecture Notes in Computer Science (2017) 10265, Springer.

[22] Shorten, C., Khoshgoftaar, T.M. & Furht, B, "Text Data Augmentation for Deep Learning. J Big Data", 8 (2021)101 https://doi.org/10.1186/s40537-021-00492-0.

[23] Shorten.A. & Khoshgoftaar.J, "A survey on image data augmentation for deep learning", Journal of Big Data, 6(1) (2019) 1-48.

[24] Srivastava A. et al., "Veegan: Reducing mode collapse in GANS using implicit variational learning. Advances in neural information processing systems", (2017) 30.

[25] Sumeyra et al. "Data augmentation for time series regression: Applying transformations, autoencoders and adversarial networks to electricity price forecasting", Applied Energy, 304 (2021) 117695.

[26] Tanja Hagemann and Katerina Katsarou. "A Systematic Review on Anomaly Detection for Cloud Computing Environments" in 3rd Artificial Intelligence and Cloud Computing Conference (AICCC 2020), Kyoto, Japan (2020) 18–20.

[27] Tomar, S., Gupta, A. "A Review on Mode Collapse Reducing GANs with GAN's Algorithm and Theory", In: Solanki, A., Naved, M. (eds) GANs for Data Augmentation in Healthcare. Springer, Cham. (2023) https://doi.org/10.1007/978-3-031-43205-7_2

[28] Vyas, P., Ragothaman, K.M., Chauhan, A. et al. "Data augmentation and generative machine learning on the cloud platform", Int. j. inf. tecnol. (2024) https://doi.org/10.1007/s41870-024-02104-5.

[29] Xianhua et al. "Cloud-GAN: Cloud Generation Adversarial Networks for anomaly detection", Pattern Recognition. 157 (2024) 110866.

[30] Zhang K. et al., "A multi-module generative adversarial network augmented with adaptive decoupling strategy for intelligent fault diagnosis of machines with small sample", Knowl. Based Syst. 239 (2022) 107980.

[31] Zhang, K., "On Mode Collapse in Generative Adversarial Networks", Farkaš, I., Masulli, P., Otte, S., Wermter, S. (eds) Artificial Neural Networks and Machine Learning – ICANN 2021. Lecture Notes in Computer Science, Springer, Cham. (2021) 12892.

[32] Zhao.B. et al., "Improved generative adversarial network for vibration-based fault diagnosis with imbalanced data", Measurement. 169 (2021) 108522.
[33] Zhenglin Dai et al., "Mode standardization: A practical countermeasure against mode collapse of GAN-based signal synthesis", Applied Soft Computing. 150 (2024) 111089.
[34] Zhenglin Dai et al., "Mode standardization: A practical countermeasure against mode collapse of GAN-based signal synthesis", Applied Soft Computing. 150 (2024) 111089.
[35] Ziqi Pan et al. "UniGAN: Reducing Mode Collapse in GANs using a Uniform Generator", 36th Conference on Neural Information Processing Systems (NeurIPS). 35 Pan (2022) 37690-37703.

AUTHOR PROFILES

Dr. K. Uma maheswari is currently working as Assistant Professor in the Department of Computer Science, Bharathi Women's College(A), Chennai. She completed M.C.A., from Bharathidasan University, Tituchirappalli,. She received her doctorate degree in the area of "Cloud Security and Digital Forensics" from Bharathiar University, Coimbatore. She has more than 13 years of teaching and research and 3 years of industrial experience. She has published more than 15 papers in reputed journals and conferences and book chapters. Her research interests include Cloud Computing, Artificial Intelligence, Cyber Security and Digital Forensics.

Dr. Shobana G is currently working as Professor in the Department of MCA, Jeppiaar Engineering College, Chennai. She received her Master's in Bioinformatics from Bharathiar University with distinction. She obtained her Ph.D from University of Madras. She has more than 15 years of teaching experience and has published more than 30 research articles, presented research articles in more than 20 International Conferences, published more than 8 patents and authored 2 books. Her research areas include Machine Learning, Drug Discovery, Computational Biology, Bioinformatics, Cheminformatics and Cloud Computing.