Iraqi Journal of Intelligent Computing and Informatics (IJICI)
Vol.4, 1, June 2025, pp. 18~32
ISSN: 2791-2868, DOI: 10.52940/ijici.v4i1.65

A Review of Generative Adversarial Networks for Addressing Data Imbalance and Enhancing Model Performance

Lina Aziz Swadi[1], Haider AL-Mashhadi[2]
[1]Department of Computer Science, Basra University, Basra, Iraq
[2]Department of Cybersecurity, Basra University, Basra, Iraq

Article Info

Article history:

Received November 05, 2024
Revised December 27, 2024
Accepted February 06, 2024

Keywords:

Deep Learning
Generative Adversarial Network
Imbalanced Dataset
Neural Network
Synthetic Dataset

ABSTRACT

In current years, Generative Adversarial Networks (GANs) have grown into a popular and active research domain in artificial intelligence and machine learning. GANs provide an adequate data augmentation method, which helps improve model accuracy by generating realistic synthetic samples, especially for underrepresented classes. This paper reviews various techniques and methods used in training GANs, particularly focusing on their role in addressing dataset imbalance. The paper also discusses the implications of utilizing GANs for improving model generalization, mitigating bias, and reducing overfitting. This paper discusses the training methods for generative adversarial models, highlighting their significance. It overviews various model design strategies, algorithms, and recent approaches to enhance training.

Corresponding Author:

Lina Aziz Swadi
Department of Computer Science, Basra University, Basra, Iraq
Email:liaz68@yahoo.com

1. INTRODUCTION

Machine learning models face issues due to dataset instability, which leads to imbalances in dataset classes, negatively impacting performance and efficiency. This imbalance makes it difficult to extract meaningful insights from the data. AI-based classifiers face bias issues when dealing with imbalanced datasets, as they tend to focus more on the majority class due to the uneven distribution of training data. This makes it difficult to accurately identify rare minority samples, increasing the risk of misclassifying them as noise and incorrectly identifying them as valid data. [1] Various methods are proposed to address the imbalanced dataset, such as oversampling, a conventional data augmentation method employed to rectify dataset imbalance, especially in machine learning and statistical analysis. This includes balancing class distribution by creating more samples for the minority category in an imbalanced dataset, either by duplicating existing data or adding synthetic data until the number of samples in each class is balanced. The main aim of oversampling is to alleviate the bias towards the majority class, which can adversely impact the effectiveness of learning algorithms. [2] However, oversampling may result in overfitting, particularly when the same samples are duplicated multiple times. This causes the model to learn noise instead of the fundamental patterns in the data, culminating in inadequate generalization to novel data. Moreover, oversampling enlarges the dataset, perhaps resulting in elevated computational expenses regarding memory and processing duration. [3]

Generative Adversarial Networks (GANs) are algorithms developed to create new data instances that closely match a specific training dataset. In 2014, a deep learning researcher and his colleagues introduced GANs, marking a significant advancement in artificial intelligence. Gui et al. (2020) comprehensively reviewed GANs, discussing their motivations, mathematical representation, and structural details. They also highlighted the connections among different GAN variants and their evolution. [4] This groundbreaking study presented a novel approach to addressing the challenges of unsupervised learning. GANs operate within a deep learning framework characterized by a unique adversarial process. Specifically, two neural networks are simultaneously trained in competition: the generator, which creates data instances, while the discriminator evaluates these outputs. The discriminator provides critical feedback to the generator, indicating how well the generated instances mimic accurate data. Over time, the generator attempts to enhance its capability to create realistic data, and the discriminator becomes more skillful at differentiating between generated and actual instances. [5]

The structure of these networks consists of multiple layers, each contributing to the resolution of the issue at hand. In the framework of GANs, these layers are crucial in generating realistic images. The discriminator network is accountable for differentiating between original and generated images, and the generator network aims to produce images that can fool the discriminator. This dynamic creates a competitive process that drives the networks to improve their performance over time. [6]

Several studies have investigated various aspects of GANs, including their algorithms, theoretical foundations, and applications. A notable work by Sharma et al. (2014) delved into the taxonomy, variants, limitations, and application of GANs. Their study emphasized the growing demand for GAN-based utilizations in domains such as image-to-image translation, natural language processing (NLP), and architectural design. [7] Nayak et al. (2024) also systematically reviewed GANs, focusing on their challenges and future directions. They discussed GAN's potential in high-dimensional data analysis and their applications in computer vision, cybersecurity, and medical imaging. [8]

This paper reviews current research on GAN-based rebalancing methods, offering insights into their effectiveness and practical implementation. It comprehensively reviews methods used in training generative adversarial models to address dataset imbalance and improve machine learning and artificial intelligence performance. It details various methods and architectures developed for this purpose, emphasizing the theoretical foundations, algorithms, and practical applications of GANs in balancing datasets. The consistency of these methods proves their importance in training generative models. This work aims to enable those unfamiliar with the GAN model to grasp the model better by presenting and extensively discussing the existing methods for improving the GAN training process. This paper presents the different model design strategies and algorithms for the successful training of the GAN. Furthermore, the paper provides extensive information about the methods that come with multiple works to enhance the training of the generative adversarial network, and others proposed recently.

This review paper is arranged as follows: Section 1 includes an introduction. Section 2, Background, overviews foundational concepts in machine learning, deep learning, artificial neural networks, and GANs. Section 3 describes the application area of GANs in different fields and related work. Section 4 is a discussion that focuses on the utility of GANs, particularly in applications where synthetic data generation can enhance dataset quality. Section 5 includes the conclusion of the review.

2. BACKGROUND

This section provides context and definitions relevant to the rest of the paper.

2.1. Machine learning

Machine learning (ML) is a unique computational approach that permits machines to produce meaningful outputs from experience without human data or interaction. It is a highly interdisciplinary domain that merges concepts from computer science, mathematics, statistics, and more. ML operates by learning from data, with the data's quality and size influencing the learned model's accuracy. ML is utilized in various application areas, such as finance, marketing, healthcare, and more. [9] Learning method in ML involves the following:

2.1.1. Supervised learning

Supervised learning is a kind of ML where models are trained on labeled datasets to make predictions. Input features relate to output labels. A dataset is utilized to train the model and adjust features or add more data for accuracy. A separate dataset is required to test the model, and the model is evaluated using real-life data. Supervised learning includes classification and regression techniques, with classification for discrete values and regression for continuous predictions. [10] Support vector machine (SVM), naive Bayes, k-nearest neighbors, decision trees, and random forests are algorithms for classification and regression. Decision trees divide data based on characteristics. Linear regression, support vector machine regressors, and random forests are regression algorithms. Supervised learning provides accurate outcomes on unseen data and can predict and assess classifications. However, it requires acquiring information and can overfit if not appropriately considered—strong assumptions about the model needed for supervised learning methods. [11]

2.1.2. Unsupervised learning

In unsupervised learning, the focus shifts from a specific prediction task to letting the model uncover structures and patterns inherent in the input data. The crucial difference lies in the absence of explicitly labeled outputs, requiring the model to discern structure autonomously. As a result, unsupervised learning is often viewed as a more fundamental form of learning. [12] Cluster analysis is an unsupervised learning task that groups input data based on similarity or distance metrics. It requires a data point collection and a distance matrix. The goal is to make data within the same cluster comparable and dissimilar in different clusters, like dimensionality reduction. [13]

2.1.3. Semi-supervised learning

Another ML method is semi-supervised learning, which incorporates known and unknown data to optimize model performance and accuracy. This method is especially advantageous when labeled data is scarce and unlabeled data is plentiful. Some popular semi-supervised learning methods encompass co-training, self-training, and multi-view learning. Self-training involves iteratively labeling unlabeled data points using a classifier trained on the initial labeled data. Co-training involves training two classifiers on distinct perspectives of the data and exchanging confidently labeled data points between them. Multi-view learning leverages multiple representations of the data to improve performance. [14]

2.1.4. Reinforcement learning

Reinforcement learning is a unique approach within the ML field where an agent is taught to make decisions through interaction with an environment to optimize overall rewards. Key components of reinforcement learning include agents, environments, actions, states, and rewards. The agent, responsible for decision-making, observes states and receives rewards, following a learning policy to exhibit specific behaviors. The environment, which includes the initial state, time, action, and state transition model, is also known as the Markov Decision Process. [15]

2.2. DEEP LEARNING

Deep learning (DL), a subset of ML, utilizes artificial neural networks to extract complex attributes from raw data. It learns to detect and differentiate hierarchical features at multiple processing levels, enabling classification or decision-making on task-relevant information. DL's strong performance across various domains and applications is due to its ability to learn attributes directly. [16]

DL algorithms require a lot of data and computing power to optimize. Historically, building deep learning systems was impractical. Nevertheless, advancements in hardware and labeled datasets have made DL a promising tool in artificial intelligence, such as convolutional neural networks (CNNs) for image classification and computer vision and recurrent neural networks (RNNs) for NLP and speech recognition. [17]

2.3. ARTIFICIAL NEURAL NETWORKS (ANNs)

Artificial Neural Network (ANN) is integral to ML techniques in Artificial Intelligence, as they are engineered to emulate the human brain's information processing and learning capabilities. [18] A common variant of artificial neural networks is the feedforward neural network, characterized by interconnected nodes distributed across several layers. These networks are extensively employed in ML techniques within artificial intelligence for data analysis and processing. The ANN, inspired by the human brain, comprises interconnected nodes that process information. They are utilized in various domains, such as image identification, NLP, and predictive analytics. ANN architecture is shown in Figure 1. [19]

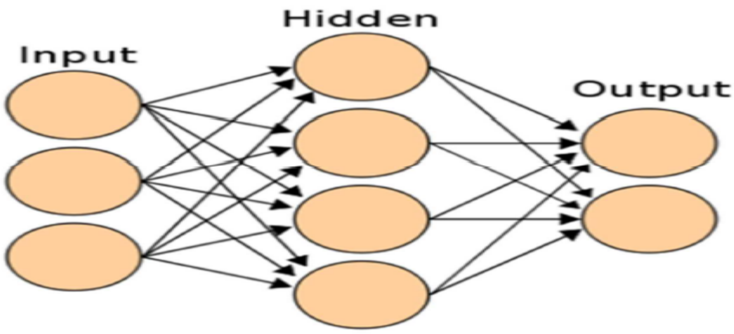

Figure 1. An architecture of artificial neural network

2.4. GENERATIVE ADVERSARIAL NETWORKS (GANs)

The Generative Adversarial Networks (GANs), presented by Goodfellow et al. (2014), serve as a novel option for Variational Autoencoders (VAE) to create substantial quantities of synthesized yet realistic data. [20] A framework for evaluating generative networks using an adversarial procedure, wherein two neural networks are trained: a generative network (G) that models the data distribution and a discriminative network (D) that determines the probability that a given instance is derived from the training data. The training methodology for G aims to optimize the likelihood of D's mistakes. This architecture relates to a minimax two-player game. [21] The objective function [22] is given by:

$$G_{min} D_{max} V(D,G) = E_{x\sim pdata(x)}\left[\log D(x)\right] + E_{z\sim pz(z)}\left[\log(1 - D(G(Z)))\right] \quad (1)$$

Where x represents real data, z denotes the latent vector, G(z) signifies the generated data, D(x) indicates the evaluation of real data by the discriminator, and D(G(z)) represents the evaluation of generated data by the discriminator. Typical varieties of GANs involve the following:

2.4.1. Vanilla (GANs)

The architecture of Vanilla GANs is straightforward, comprising two neural networks that work in an adversarial manner. The generator network produces data from an unknown noise pattern by learning to map from a latent space (Z). The aim is to create samples indistinguishable from real data, such as legitimate images. The second network, the discriminator (the classifier), takes inputs from real data and the generator's outputs, attempting to discern their origin. [23] The structure of Vanilla (GANs) is shown in Figure 2. [24]

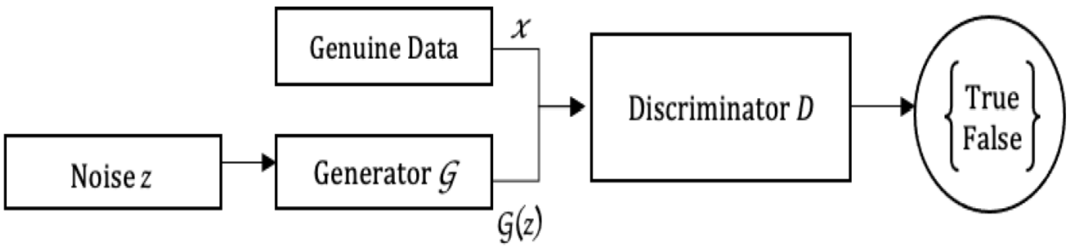

Figure 2. A structure of vanilla generative adversarial networks

The GAN framework can be analogized to a contest between a forger (the generator) and a police officer (the discriminator) trying to identify counterfeit items. In this minimax game, while the generator generates realistic images, the discriminator strives to precisely distinguish between actual and generated images. Both networks function collaboratively; as the generator improves in producing convincing samples from random noise, the discriminator simultaneously enhances its ability to detect these fake samples. This interplay drives mutual improvement, resulting in the generation of outputs that closely resemble real data. [25]

2.4.2. Conditional (cGANs)

The generator network initially introduced conditional generation in the seizure generation framework. This formulation created a discrete one-hot representation for class labels of real data, often utilizing networks like encoders or classifiers. cGANs gained significant recognition by incorporating label information within discriminator networks. Specifically, class labels are concatenated into the feature maps at various network layers. However, this concatenation is only feasible when using images from other GANs. Different architectures of GANs now employ discriminator networks using class label information, ultimately forming conditional GAN networks. The objective function is Equation 2. [26] The architecture of Conditional (cGANs) is shown in Figure 3. [27]

$$G_{min} D_{max} V(D,G) = E_{x\sim pdata(x)}\left[\log D(x|y)\right] + E_{z\sim pz(z)}\left[\log(1 - D(G(z|y)))\right] \quad (2)$$

Where y: acts as a class label (conditional information), D(x|y): is the discriminator's probability estimate that x is an original sample given the condition y, G(z|y): is the generator's output, conditioned on y, which attempts to produce samples resembling x given the label y.

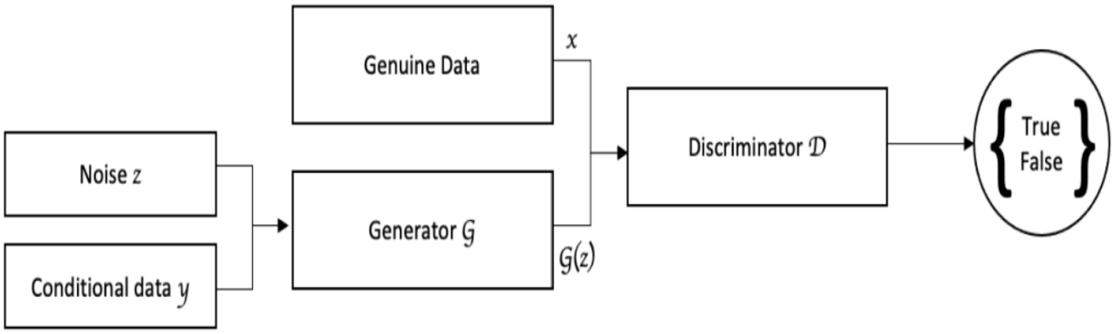

Figure 3. An architecture of conditional generative adversarial network

2.4.3. Wasserstein (GANs)

Wasserstein GANs (WGANs) represent a significant overhaul in GAN loss functions. The key theoretical advancement lies in aligning the generator objective with the earth mover's distance (Wasserstein distance) between the model distribution and the actual data distribution. In WGANs, the discriminator is replaced by a value critic, and the pointwise use of the critic's output is substituted with an integral. Weight clipping is employed to enforce the Lipschitz restriction on the critic. [28] WGANs are guided by the principle that increasing the divergence between the generator's distribution and actual data distribution amplifies the generator's loss. This approach stabilizes the training of GANs by utilizing the gradient of a 1-Lipschitz function as the critic. Unlike traditional GANs, which have a zero gradient almost everywhere, WGANs exhibit a non-vanishing gradient concerning the generator, provided that the critic and generator have sufficient capacity. [29]

Additionally, WGANs replace the pointwise use of the critic's output with an integral to ensure more accurate feedback on the generated images. This method addresses the limitations of one-sided label smoothing, often leading to nondeterministic convergence due to reduced gradient flow through the discriminator. The term "Wasserstein" is derived from the earth mover distance, emphasizing the algorithm's foundation in measuring distribution divergence. [30]

2.4.4. Deep convolutional (DCGANs)

Deep Convolutional GANs (DCGANs) represent a more stable iteration of the original GANs. In DCGANs, the generator and the discriminator are deep convolutional networks trained with batch normalization to maintain a systematic structure in parameter space. This training process enhances stability and enables the generation of high-resolution sample data by reaching deeper network layers. [31]

The optimal DCGAN architecture pools layers using strode and fractional stridden convolutions and batch normalizes both networks. Weights are set up with a zero-centered normal distribution and a standard deviation of 0.02. The generator uses the ReLU activation function for all layers except the output, and the discriminator uses the Leaky ReLU activation function for all layers. This results in high-quality, sharp images with a structured architecture. [32]

2.4.5. Cycle generative adversarial networks (CycleGANs)

CycleGAN consists of two GANs—GAN-A and GAN-B. GAN-A maps domain A to domain B, while GAN-B maps domain B to domain A. By training both GANs, CycleGAN learns the bidirectional mappings between the two domains. Adversarial loss and cycle-consistent loss are defined for both GANs, with the total loss being their sum. CycleGAN can be trained with unpaired data, unlike many GANs that require paired data. However, the non-entropy-based cycle-consistent loss can lead to noise reduction issues, potentially hindering proper one-way mapping. Despite these challenges, the continuous development of CycleGAN enables it to handle high-dimensional data across multiple modes, reducing mode collapse and generating more realistic data. [33] CycleGANs architecture is shown in Figure 4. [34]

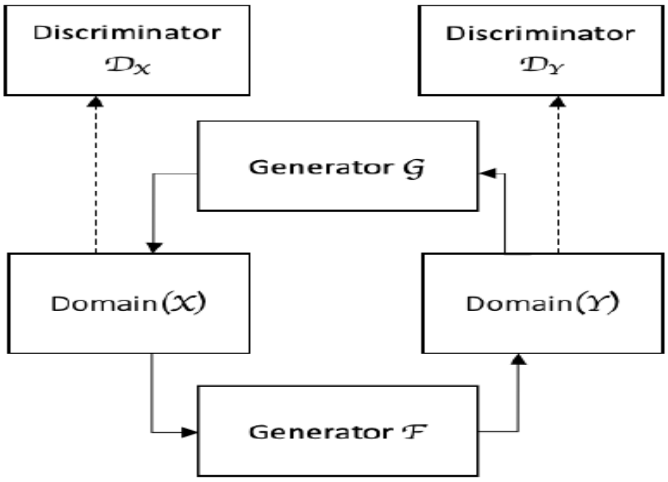

Figure 4. An architecture of cycle generative adversarial networks

Table 1. A comparison of GANs architecture, features, and application

GANs type	Architecture	Key characteristic	Application	Limitation
Vanilla GAN	The basic GANs structure compromises two networks: a generator and a discriminator.	Straightforward design uses random noise as input for the generator.	used for generating images, text, and other data types	Prone to mode collapse, it can suffer from instability issues.
Conditional GAN (cGAN)	Extends the Vanilla GAN by incorporating class labels in the training, enabling conditional generation.	Adds conditional information (e.g., labels) to control output characteristics, enabling targeted data generation.	Useful in scenarios where specific attributes or classes are desired in generated data.	It requires labeled data; it may not generalize well to unseen classes.
Wasserstein GAN (WGAN)	This improves GANs' training stability by reducing the Wasserstein distance (earth mover's distance) between generated and real data distributions.	Utilizes a "critic" instead of a discriminator; helps reduce mode collapse and makes training more stable.	Applicable in various domains requiring high-quality data generation.	It can be computationally intensive and requires careful tuning of parameters.
Deep Convolutional GAN (DCGAN)	A variant using deep convolutional layers in the generator and discriminator to enhance image quality.	Adds convolutional layers and batch normalization, avoids pooling, and achieves realistic and high-resolution outputs.	highly effective in generating high-resolution images	It may introduce additional complexity in architectural design.

CycleGAN	Consists of two GANs: GAN-A and GAN-B. GAN-A maps domain A to domain B, while GAN-B maps domain B to domain A.	It uses cycle-consistent loss to learn bi-directional mappings between domains; it is suitable for tasks without paired datasets.	Widely used for tasks such as style transfer and transforming images across different seasons.	Complexity in maintaining cycle consistency may produce artifacts.

3. AREA OF APPLICATION

This section provides foundational information on GANs, which are essential for understanding their applications and variations and will be discussed later in the paper.

A generative adversarial model was created to generate data from random noise. GAN has contributed significantly to various fields such as image production, genome data analysis, semantic image synthesis, high-energy nuclear physics generation models, computer game level generation, and machine learning prototype exploration, such as unintentional learning. [35] Several models have been created based on GAN. cGAN introduces conditions to improve image quality, while WGAN addresses mode collapse and stabilizes learning by modifying the loss function. Inception GAN evaluates images classification performance and enhances image synthesis precision with AutoEncoder version and Variant AutoEncoders and their combination with GAN (VAE-GAN). And other studies have utilized GAN series models to reduce experimental errors and improve application contributions. [36]

3.1. Computer vision

In [37] proposed Pix2PixHD and Pix2Pix, which are GAN-based models for solar image translation to translate solar magnetograms into UV images at high resolution. The Pix2PixHD performs better in capturing fine details and generating high-resolution solar images, with Pearson Correlation Coefficients of 0.99 for Pix2PixHD and 0.962 for Pix2Pix. Pix2PixHD is highly effective for solar image translation, outperforming the previous model's accuracy. The study employs a conditional GAN architecture designed for image-to-image translation tasks, specifically the Pix2Pix model. GANs methodology for high-resolution solar images has limitations like limited data range, manual data cleaning, model complexity, and training instability. However, it presents significant advancements in solar image generation, and continuous improvement in data handling, model training, evaluation, and applicability is crucial for future developments.

In [38] BalaGAN is a technique that improves cross-modal image translation by translating images between imbalanced domains and decomposing richer domains into multiple modes. It uses contrastive learning, modality identification, and a robust training framework to enhance translation quality across imbalanced domains, focusing on latent modes and structured training. This method has been successfully applied in highly imbalanced domains. Fréchet Inception Distance (FID) is a widely utilized metric for assessing the quality of created images, comparing the feature distributions of original and created images in the Inception feature space. Lower FID scores indicate higher quality and realism in the generated outputs. However, expanding the approach to broader tasks like object detection and classification could enhance its utility in computer vision.

In [39] the mGANprior framework is a significant advancement in GAN-based image processing, utilizing multiple latent codes to generate feature maps in intermediate layers. This approach improves image reconstruction quality for colorization, super-resolution, and inpainting tasks. The paper utilizes a Peak Signal-To-Noise Ratio (PSTNR) metric to measure the pixel-level similarity between the original and reconstructed images. Higher PSNR demonstrates better reconstruction quality. The mGANprior framework addresses the limitations of previous methods by utilizing pre-trained GANs for diverse tasks without retraining. However, challenges such as computational complexity, reliance on pre-trained models, and task-specific customization need further exploration. Future work could focus on improving scalability, expanding to diverse datasets, and automating task-specific adaptations to enhance its practical applicability.

In [40] the hybrid model uses DCGAN for image generation and CycleGAN for unpaired image-to-image translation. DCGAN works well with paired training data, while CycleGAN excels when no pairs are present. This technique prevents overfitting and accelerates convergence by normalizing the output of each layer, ensuring smoother image transformation between domains. Cycle-Consistency Loss metric employed to ensures that transformed images can return to their original domain, reducing artifacts and improving output reliability. The combination of DCGAN and CycleGAN introduces additional computational overhead. The requirement for high-powered GPUs and extended training periods may limit practical adoption, and incorporating metrics like FID or PSNR would enable more objective comparisons with other cutting-edge models. Future research should address scalability, efficiency, and robustness to enhance its practicality and generalizability across domains.

In [41] the method tackles the issue of mode collapse in GANs, where the generator produces a limited range of outputs. Self-Conditioned GANs address this by clustering the discriminator's feature space to identify data modes, which act as implicit classes automatically. This allows the generator to generate more diverse images by covering broader possibilities. Training alternates between clustering and refining the GAN model, evaluated using metrics like FID for image quality, Inception Score (IS) for quality and diversity, Reverse Kullback-Leibler (KL) divergence for distribution matching, and Learned Perceptual Image Patch Similarity (LPIPS) for perceptual similarity from original to reconstructed images. Despite improving diversity, the method falls short of the quality and diversity achieved by supervised class-conditional GANs that rely on labeled data.

In [42] design a Balancing GAN Gradient Penalty (BAGAN-GP), an enhancement of BAGAN, designed to produce minority-class images in imbalanced datasets. The original BAGAN used an autoencoder to initialize GAN training by giving the generator common knowledge of all classes. BAGAN-GP adds a gradient penalty to the GAN training process, which improves stability and ensures better image generation for minority classes (such as rare medical images). The gradient penalty helps ensure the generated image distribution is smooth and realistic. The primary metric to assess the model is FID, which measures the quality of produced images. Lower FID scores demonstrate better quality of target samples. Additionally, while BAGAN-GP shows improved stability with less hyperparameter tuning than the original model, there may be room for further optimization and enhancement of the model architecture and training process to handle more complex datasets.

3.2. Natural language processing (NLP)

Generative Pre-trained Transformers (GPT) is a generative model of NLP. It employs attention mechanisms to capture the dependencies of words in a passage. One of the intriguing capabilities of the model is its ability to complete sentences and even paragraphs in various styles. The autoencoders come in different forms, such as Variational Autoencoder, Denoising Autoencoder, and Contractive Autoencoder, capturing a latent space from the training data. They compromise an encoder and a decoder, which map the original data to a latent space and then map the latent space back to the original data, respectively. The Diffusion Models put randomness in the magnification of perceptual inputs and the blurring process of the compressed image, making the final prediction. [43]

In [44] the system focuses on two GAN models, catGAN and sentiGAN, to generate synthetic text to balance datasets; the paper addresses the issue of dataset imbalance in sentiment analysis within the educational domain. Experimental results indicate significant improvements in model performance when trained on synthetic balanced datasets compared to the original imbalanced datasets. Specifically, accuracy and F1-scores were enhanced, demonstrating the effectiveness of the synthetic data approach. However, The educational domain lacks structured, manually-labeled datasets, leading to data imbalance issues. Most research relies on private datasets, and models often fail on real-world imbalanced datasets. Traditional text generation approaches also struggle with long sentence dependencies. Future research must create benchmark datasets to properly test and compare model performance for sentiment classification on student feedback.

In [45] uses a combination of VAE-GAN and reinforcement learning, fine-tuning in latent space with GPT-2 for single-word generation. To address text generation, the work combines pre-trained models (VAE, GPT-2) with GANs. Introduces reinforcement learning fine-tuning to balance quality and diversity. The paper achieves cutting-edge performance in text generation benchmarks, outperforming standard GANs in high-level language modeling. Reinforcement learning fine-tuning yields significant quality improvements in generated text. Two primary metrics are used to evaluate text generation: the Bilingual Evaluation Understudy (BLEU) to measure precision-oriented overlap between generated text and ground truth and the recall-oriented Understudy for Gisting Evaluation (ROUGE) to measure recall-oriented overlap between generated text and ground truth. However, discriminators are specialized for the current generator distribution, leading to overestimating sequences outside that distribution and high variance in importance sampling estimators.

In [46] the paper explores GANs for low-resource language modeling. It is a new ML technology that can aid low-resource language preservation. GANs aim to model graphotactics and morphological inflection. The study utilized two main metrics, including accuracy, where the models trained on GAN-augmented data achieved less than 18% accuracy, and in most cases (10 out of 13 languages) had less than 10% accuracy. Models trained on GAN data performed only one-fifth as accurately as those trained on unaugmented data. Levenshtein Distance metric was utilized to measure the difference between predicted and actual inflections. While following similar patterns to accuracy measurements, this metric occasionally favored models trained on trigram-augmented data. However, models quickly exhausted their learning potential, and GAN models performed poorly, with no GAN-augmented model exceeding 18% accuracy. The study also highlighted the limitations of training on small datasets.

In [47] Bidirectional Encoder Representations from Transformers GAN (GAN-BERT) is designed to improve robustness in poor training conditions. Adversarial training enables semi-supervised learning for Transformer architectures. Fine-tuning with a few labeled examples leads to unstable models. It enhances accuracy in sentence classification tasks with few examples. GAN-BERT shows systematic improvement over BERT in all tested conditions. Although GAN-BERT shows improvements in scenarios with limited labeled data, the performance may still be less stable in tasks with many categories, where the classification becomes more complex.

3.3. Intrusion detection system (IDS)

Computer network security measures are essential to prevent unauthorized access, breaches, and cyber threats. Both firewalls and intrusion detection systems (IDS) are essential in ensuring network security. While firewalls filter and analyze network packets, they have limitations in analyzing packet content. On the other hand, IDS scrutinizes packet content to identify potential security issues. However, traditional IDS encounter challenges related to performance and accuracy, necessitating ongoing research and development. On the other hand, modern IDS leverages advanced technologies to enhance efficiency and reliability. [48]

The Internet of Things (IoT) is a network of interconnected devices communicating without human intervention. Sensors detect and transmit data to a central hub, enhancing human convenience by exchanging information through smart devices. [49]

Detecting unauthorized access and malicious behavior presents a primary obstacle in cybersecurity. Recent research has been concentrated on the application of artificial intelligence in Network Intrusion Detection Systems (NIDS). AI-based NIDS has exhibited exceptional effectiveness. Initially, efforts were directed toward integrating conventional ML techniques such as SVM and decision trees into existing intrusion detection systems. More recently, attention has shifted to incorporating deep learning techniques. [50]

In [51] the system uses GANs to produce synthetic data to train ML models in NIDS. The GAN's training termination criterion is based on the alignment of boxplot distributions between the synthetic and real data. The GAN's performance is measured through adversarial loss during training, which involves the discriminator's ability to differentiate between original and artificial data and the generator’s ability to create artificial data that the discriminator cannot differentiate from original data. The system has undergone testing on various datasets, including UNSW-NB15 and NSL-KDD datasets. It does propose a synthetic data generation method. It uses GANs to train NIDS, reducing reliance on real-world data and improving flexibility. It obtained an accuracy of 100% on the BoT-IoT dataset and an accuracy of 90% on the UNSW-NB15 dataset. Future work must improve generalizability across datasets, handle unseen threats, and reduce the computational overhead of GAN training.

In [52] the model utilizes One Class GAN (ocGAN) for data augmentation, considering the class imbalance in IoT networks, which aids in creating a more uniform distribution of normal and malicious profiles. The model is trained on algorithms for finding outliers that use feedforward networks and tested on different datasets. The Binary Class GAN (bcGAN) model is all about adding to data by making fake data to improve the training dataset and find anomalies better. This study improves the accuracy, precision, and F1-score of anomaly detection, getting better results than traditional models across several datasets. The cGAN framework proficiently mitigates data imbalance in IoT networks and improves detection precision. However, the ocGAN and bcGAN models showed low detection rates in less than 1000 samples but improved significantly with more samples. Concerns about generalization and effectiveness in diverse IoT environments remain unresolved.

In [53] the study uses GAN-based synthetic data generation and anomaly detection models to improve network security and address class imbalance in NIDS datasets. It achieves higher accuracy and reduces false alarm rates in the NSL-KDD and UNSW-NB15 datasets, enhancing detection performance for rare attacks and improving security in enterprise networks and critical infrastructure systems. However, the study's focus on familiar attack types may not adequately address emerging or sophisticated vectors, potentially leaving networks vulnerable to novel threats.

In [54] the CWVAEGAN-1DCNN model is a proposed method for generating minority class samples and solving class imbalance in NIDS datasets. It balances training data using GAN-generated samples and classifies attacks using 1DCNN. The model outperforms other class-balancing methods and improves detection accuracy, making it suitable for environments with significant class imbalance, such as healthcare systems or industrial IoT applications. However, there is a need for improvement in applying CWVAEGAN to complex datasets and developing automated parameter-tuning methods.

In [55] an IDS was developed using cGAN for unsupervised learning in Wireless Sensor Network (WSN) and integrated XGBoost for fast classification. Introduced a cGAN-based IDS model that reduces the need for extra sensors and generates synthetic data for more efficient intrusion detection. Reduced the false alarm rate by 1.827% and improved the accuracy of intrusion detection in WSN environments. The CGAN effectively reduces the need for additional hardware, improving the efficiency and accuracy of IDS in WSN. The study offers a cost-effective solution for securing WSNs in agriculture, military surveillance, and environmental monitoring. However, the CGAN model parameters were not extensively varied enough to study their full impact on performance, potentially limiting the optimization of the IDS. Optimizing hyperparameters and studying their effects could improve model performance.

Table 2. Summary of related work

study	**Improved performance of GANs model**	**Benefit**	**Challenges**	**Discussion**
[39]	mGANprior uses multi-code latent spaces to improve reconstruction quality and adaptability across tasks like colorization and inpainting.	Excels in inpainting, super-resolution, and colorization tasks	Heavy reliance on pre-trained models; computational complexity	Valuable for creative and restoration tasks, the reliance on pre-trained models and high computational costs limit accessibility in real-time applications.
[42]	BAGAN-GP uses gradient penalty to ensure stable training and effective generation of minority class data.	Balances datasets, enhances training stability and reduces overfitting.	Requires careful tuning; generalization across datasets is limited.	It is effective for addressing class imbalance in medical and security datasets. However, scaling to highly diverse or dynamic datasets requires further research.

[44]	SentiGAN employs multiple generators, each focusing on creating synthetic samples for a specific sentiment class. catGAN employs an entropy-based loss function to enhance the discriminator's ability to produce high-confidence predictions.	Addresses dataset imbalance and improves classification accuracy in sentiment analysis.	Struggles with long-term dependencies; relies on structured labeled datasets.	Helpful in improving sentiment classification; lack of benchmark datasets and poor performance on real-world imbalanced datasets remain limitations.
[45]	Combines reinforcement learning and adversarial learning to enhance sequence generation performance.	Generates diverse and high-quality text; balances quality and diversity.	Overestimation of sequences outside current generator distribution; high variance in estimators.	It is effective in NLP tasks requiring diversity and coherence in text generation; optimizing performance in low-resource settings remains challenging.
[52]	Combine ocGAN and bcGAN in a hybrid approach, Use regularization techniques (e.g., dropout, batch normalization) to reduce overfitting and improve the model's ability to generalize better across diverse IoT environments.	Reduces false alarm rates while improving detection accuracy.	Limited variation in model parameters; difficulty scaling to new scenarios.	Provides cost-effective security solutions; parameter tuning and scaling to diverse environments need improvement for broader applicability.
[54]	WGAN uses Wasserstein loss to stabilize training and reduce mode collapse, improving the quality of generated images.	Generates realistic minority class samples and improves classification performance.	Needs optimization for complex datasets; computationally intensive.	Suitable for industries like healthcare and IoT applications; applying CWVAEGAN to more complex datasets is necessary to expand its utility.

4. DISCUSSION

Generative Adversarial Networks are optimally utilized in scenarios where the creation of synthetic data can significantly enhance the quality of existing datasets, particularly in cases of data imbalance or scarcity.

1. Minority Class Augmentation and Imbalanced Data: GANs are highly useful for applications requiring class balancing in datasets. For example, in UV imaging or IDS, GANs like cGAN and BAGAN models make realistic samples that balance the dataset across all classes. This is called Detail-Specific Data Generation. GANs such as Pix2Pix and Pix2PixHD are adequate for tasks needing high-resolution data outputs. Pix2PixHD for solar imaging and translating magnetograms into UV images requires preserving fine details. This also applies

to applications like satellite imagery, where high resolution and accurate image translation are essential.

2. Low-resource languages and balance datasets for sentiment analysis and text classification: SentiGAN and catGAN aim to enhance sentiment analysis by creating artificial samples to address imbalanced sentiment classes, thus improving performance on datasets with uneven distribution. GAN-BERT integrates adversarial training with BERT, enabling semi-supervised learning in situations with limited data. This improves text classification in resource-limited scenarios, ensuring resilience across various text samples.
3. Generating synthetic data for intrusion detection: cGANs improve detection by creating labeled fake data, balancing unequal classes in security datasets, and making the models more accurate.

SYN-GAN and CWVAEGAN-1DCNN make samples to find rare intrusions in IoT environments. This improves IDS accuracy and adaptability by reducing the need to collect real-world data.

Table 3. Summary of training approaches applied to GANs

Training Method	Purpose	Applicable GAN Types	Benefits	Challenges
Wasserstein Distance	Stabilizes training by measuring distribution differences.	WGAN, WGAN-GP	Reduces mode collapse and provides consistent convergence.	Requires weight clipping and careful tuning.
Batch Normalization	Normalizes inputs across layers to prevent training issues.	DCGAN, CycleGAN	It improves image quality and supports deeper network layers.	It can create artifacts if not applied consistently.
Gradient Penalty	Enforces smooth training by limiting gradient magnitudes.	WGAN-GP, BAGAN-GP	Increases stability and enhances minority-class data generation.	The higher computation al cost adds complexity.
Feature Matching	Ensures synthetic data resembles real data distributions.	cGAN	Reduces overfitting and enhances realism in generated data.	Slower convergence requires parameter adjustments.

Notwithstanding its benefits, GANs have drawbacks like:

1. Mode Collapse: GANs occasionally produce a small number of outputs while ignoring the dataset's diversity. Methods such as Wasserstein GANs and self-conditioned GANs have been suggested to tackle this problem.
2. Training Instability: For best results, GANs need to be fine-tuned and can be sensitive to hyperparameter selection. The WGAN and DCGAN architectures address these stability issues in part.
3. Demand for Computation: The substantial computational resources needed to train GANs, particularly for high-resolution data, may not be practical for all applications.

5. CONCLUSION

To improve model generalization, address data imbalance, decrease overfitting, and augment data, GANs have proven to be an effective tool. GANs reduce bias and optimize ML models' accuracy, fairness, and resilience by generating synthetic data. Although stability, computational efficiency, and interpretability issues still exist, GAN utilizations in computer vision, natural language processing, and intrusion detection demonstrate their impact and versatility. GANs have the potential to revolutionize a variety of domains with further development, especially those where data quantity and quality are crucial. Each work offers distinct insights into addressing prevalent issues associated with GANs, including mode collapse, image quality in imbalanced domains, and efficient image translation without paired data. By resolving these challenges, these GAN models facilitate progress in high-resolution imaging, unpaired translation, and broader applications, including healthcare, cybersecurity, and natural language processing. This dialogue invites further investigation of these techniques and their possible applications in other data-intensive fields.

REFERENCES

[1] M. Khushi *et al.*, “A Comparative Performance Analysis of Data Resampling Methods on Imbalance Medical Data,” *IEEE Access*, vol. 9, pp. 109960–109975, 2021, doi: 10.1109/ACCESS.2021.3102399.

[2] S. Bej, N. Davtyan, M. Wolfien, M. Nassar, and O. Wolkenhauer, “LoRAS: an oversampling approach for imbalanced datasets,” *Mach Learn*, vol. 110, no. 2, pp. 279–301, Feb. 2021, doi: 10.1007/s10994-020-05913-4.

[3] M. T. García-Ordás, C. Benavides, J. A. Benítez-Andrades, H. Alaiz-Moretón, and I. García-Rodríguez, “Diabetes detection using deep learning techniques with oversampling and feature augmentation,” *Comput Methods Programs Biomed*, vol. 202, p. 105968, 2021, doi: https://doi.org/10.1016/j.cmpb.2021.105968.

[4] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, “A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications,” *IEEE Trans Knowl Data Eng*, vol. PP, p. 1, Oct. 2021, doi: 10.1109/TKDE.2021.3130191.

[5] A. Dash, J. Ye, and G. Wang, “A Review of Generative Adversarial Networks (GANs) and Its Applications in a Wide Variety of Disciplines: From Medical to Remote Sensing,” *IEEE Access*, vol. 12, pp. 18330–18357, 2024, doi: 10.1109/ACCESS.2023.3346273.

[6] Z. Huang, J. Zhang, Y. Zhang, and H. Shan, “DU-GAN: Generative Adversarial Networks with Dual-Domain U-Net-Based Discriminators for Low-Dose CT Denoising,” *IEEE Trans Instrum Meas*, vol. 71, 2022, doi: 10.1109/TIM.2021.3128703.

[7] P. Sharma, M. Kumar, H. K. Sharma, and S. M. Biju, “Generative adversarial networks (GANs): Introduction, Taxonomy, Variants, Limitations, and Applications,” *Multimed Tools Appl*, 2024, doi: 10.1007/s11042-024-18767-y.

[8] A. A. Nayak, P. S. Venugopala, and B. Ashwini, “A Systematic Review on Generative Adversarial Network (GAN): Challenges and Future Directions,” *Archives of Computational Methods in Engineering*, 2024, doi: 10.1007/s11831-024-10119-1.

[9] E. Mosqueira-Rey, E. Hernández-Pereira, D. Alonso-Ríos, J. Bobes-Bascarán, and Á. Fernández-Leal, “Human-in-the-loop machine learning: a state of the art,” *Artif Intell Rev*, vol. 56, no. 4, pp. 3005–3054, Apr. 2023, doi: 10.1007/s10462-022-10246-w.

[10] N. Nasir *et al.*, “Water quality classification using machine learning algorithms,” *Journal of Water Process Engineering*, vol. 48, Aug. 2022, doi: 10.1016/j.jwpe.2022.102920.

[11] V. H. Nhu *et al.*, “Comparison of support vector machine, bayesian logistic regression, and alternating decision tree algorithms for shallow landslide susceptibility mapping along a mountainous road in the west of Iran,” *Applied Sciences (Switzerland)*, vol. 10, no. 15, Aug. 2020, doi: 10.3390/app10155047.

[12] P. Dini, A. Elhanashi, A. Begni, S. Saponara, Q. Zheng, and K. Gasmi, “Overview on Intrusion Detection Systems Design Exploiting Machine Learning for Networking Cybersecurity,” Jul. 01, 2023, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/app13137507.

[13] A. E. Ezugwu *et al.*, “A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects,” Apr. 01, 2022, *Elsevier Ltd*. doi: 10.1016/j.engappai.2022.104743.

[14] I. Nassar, S. Herath, E. Abbasnejad, W. Buntine, and G. Haffari, “All Labels Are Not Created Equal: Enhancing Semi-supervision via Label Grouping and Co-training,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 7237–7246. doi: 10.1109/CVPR46437.2021.00716.

[15] V. Singh, S. S. Chen, M. Singhania, B. Nanavati, A. kumar kar, and A. Gupta, “How are reinforcement learning and deep learning algorithms used for big data based decision making in financial industries–A review and research agenda,” Nov. 01, 2022, *Elsevier B.V.* doi: 10.1016/j.jjimei.2022.100094.

[16] Z. K. Maseer, R. Yusof, N. Bahaman, S. A. Mostafa, and C. F. M. Foozy, “Benchmarking of Machine Learning for Anomaly Based Intrusion Detection Systems in the CICIDS2017 Dataset,” *IEEE Access*, vol. 9, pp. 22351–22370, 2021, doi: 10.1109/ACCESS.2021.3056614.

[17] R. Gupta, D. Srivastava, M. Sahu, S. Tiwari, R. K. Ambasta, and P. Kumar, “Artificial intelligence to deep learning: machine intelligence approach for drug discovery,” *Mol Divers*, vol. 25, no. 3, pp. 1315–1360, Aug. 2021, doi: 10.1007/s11030-021-10217-3.

[18] M. Islam, G. Chen, and S. Jin, “An Overview of Neural Network,” *American Journal of Neural Networks and Applications*, vol. 5, no. 1, p. 7, 2019, doi: 10.11648/j.ajnna.20190501.12.

[19] A. Shrestha and A. Mahmood, “Review of deep learning algorithms and architectures,” 2019, *Institute of Electrical and Electronics Engineers Inc.* doi: 10.1109/ACCESS.2019.2912200.

[20] A. Dunmore, J. Jang-Jaccard, F. Sabrina, and J. Kwak, “A Comprehensive Survey of Generative Adversarial Networks (GANs) in Cybersecurity Intrusion Detection,” *IEEE Access*, vol. 11, pp. 76071–76094, 2023, doi: 10.1109/ACCESS.2023.3296707.

[21] I. Goodfellow *et al.*, “Generative adversarial networks,” *Commun ACM*, vol. 63, no. 11, pp. 139–144, Oct. 2020, doi: 10.1145/3422622.

[22] Y. Shi, Q. Li, and X. X. Zhu, “Building Footprint Generation Using Improved Generative Adversarial Networks,” *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 4, pp. 603–607, 2019, doi: 10.1109/LGRS.2018.2878486.

[23] A. M. Shaker, M. Tantawi, H. A. Shedeed, and M. F. Tolba, “Generalization of Convolutional Neural Networks for ECG Classification Using Generative Adversarial Networks,” *IEEE Access*, vol. 8, pp. 35592–35605, 2020, doi: 10.1109/ACCESS.2020.2974712.

[24] S. Y. Shin, Y. W. Kang, and Y. G. Kim, “Android-GAN: Defending against android pattern attacks using multi-modal generative network as anomaly detector,” *Expert Syst Appl*, vol. 141, Mar. 2020, doi: 10.1016/j.eswa.2019.112964.

[25] A. Carnevale, C. F. Delgado, and P. Bisconti, “Hybrid Ethics for Generative AI: Some Philosophical Inquiries on GANs,” *HUMANA.MENTE Journal of Philosophical Studies*, vol. 44, pp. 33–56, 2023, [Online]. Available: https://www.who.int/health-topics/infodemic#tab=tab_1

[26] Y. Xu, J. Yang, and M. Sawan, “Multichannel Synthetic Preictal EEG Signals to Enhance the Prediction of Epileptic Seizures,” Apr. 2022, doi: 10.1109/TBME.2022.3171982.

[27] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, “Generative adversarial networks: introduction and outlook,” *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 588–598, 2017, doi: 10.1109/JAS.2017.7510583.

[28] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, Institute of Electrical and Electronics Engineers Inc., Nov. 2017, pp. 5967–5976. doi: 10.1109/CVPR.2017.632.

[29] X. Guo, J. Hong, T. Lin, and N. Yang, “Relaxed Wasserstein with Applications to GANs,” Oct. 2017, doi: 10.48550/arXiv.1705.07164.

[30] A. Bandi, P. V. S. R. Adapa, and Y. E. V. P. K. Kuchi, “The Power of Generative AI: A Review of Requirements, Models, Input–Output Formats, Evaluation Metrics, and Challenges,” Aug. 01, 2023, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/fi15080260.

[31] A. Roy, R. Fablet, and S. L. Bertrand, “Using generative adversarial networks (GAN) to simulate central-place foraging trajectories,” *Methods Ecol Evol*, vol. 13, no. 6, pp. 1275–1287, Jun. 2022, doi: 10.1111/2041-210X.13853.

[32] S. Xu, X. Li, C. Xie, H. Chen, C. Chen, and Z. Song, “A high-precision implementation of the sigmoid activation function for computing-in-memory architecture,” *Micromachines (Basel)*, vol. 12, no. 10, Oct. 2021, doi: 10.3390/mi12101183.

[33] L. Sun, D. Shen, and H. Feng, “Theoretical Insights into CycleGAN: Analyzing Approximation and Estimation Errors in Unpaired Data Generation,” Jul. 2024, [Online]. Available: http://arxiv.org/abs/2407.11678

[34] A. Kanagaki, M. Tanaka, T. Nose, R. Shimizu, A. Ito, and A. Ito, “CycleGAN-Based High-Quality Non-Parallel Voice Conversion with Spectrogram and WaveRNN,” in *2020 IEEE 9th Global Conference on Consumer Electronics (GCCE)*, 2020, pp. 356–357. doi: 10.1109/GCCE50665.2020.9291952.

[35] Prabhat, Nishant, and D. Kumar Vishwakarma, “Comparative Analysis of Deep Convolutional Generative Adversarial Network and Conditional Generative Adversarial Network using Hand Written Digits,” in *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 2020, pp. 1072–1075. doi: 10.1109/ICICCS48265.2020.9121178.

[36] M. A. Souibgui and Y. Kessentini, “DE-GAN: A Conditional Generative Adversarial Network for Document Enhancement,” *IEEE Trans Pattern Anal Mach Intell*, vol. 44, no. 3, pp. 1180–1191, 2022, doi: 10.1109/TPAMI.2020.3022406.

[37] A. Dash, J. Ye, G. Wang, and H. Jin, "High Resolution Solar Image Generation Using Generative Adversarial Networks," *Annals of Data Science*, vol. 11, no. 5, pp. 1545–1561, 2024, doi: 10.1007/s40745-022-00436-2.

[38] O. Patashnik, D. Danon, H. Zhang, and D. Cohen-Or, "BalaGAN: Cross-Modal Image Translation Between Imbalanced Domains," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021, pp. 2653–2661. doi: 10.1109/CVPRW53098.2021.00299.

[39] J. Gu, Y. Shen, and B. Zhou, "Image Processing Using Multi-Code GAN Prior," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 3009–3018. doi: 10.1109/CVPR42600.2020.00308.

[40] J. Chen and L. Zhao, "Image Transformation Based on Generative Adversarial Networks," *International Journal of Advanced Network, Monitoring and Controls*, vol. 4, no. 2, pp. 93–98, Jan. 2019, doi: 10.21307/ijanmc-2019-053.

[41] S. Liu, T. Wang, D. Bau, J.-Y. Zhu, and A. Torralba, "Diverse Image Generation via Self-Conditioned GANs," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 14274–14283. doi: 10.1109/CVPR42600.2020.01429.

[42] G. Huang and A. H. Jafari, "Enhanced balancing GAN: minority-class image generation," *Neural Comput Appl*, vol. 35, no. 7, pp. 5145–5154, Mar. 2023, doi: 10.1007/s00521-021-06163-8.

[43] A. M. Hasani *et al.*, "Evaluating the performance of Generative Pre-trained Transformer-4 (GPT-4) in standardizing radiology reports," *Eur Radiol*, vol. 34, no. 6, pp. 3566–3574, Jun. 2024, doi: 10.1007/s00330-023-10384-x.

[44] A. S. Imran, R. Yang, Z. Kastrati, S. M. Daudpota, and S. Shaikh, "The impact of synthetic text generation for sentiment analysis using GAN based models," *Egyptian Informatics Journal*, vol. 23, no. 3, pp. 547–557, Sep. 2022, doi: 10.1016/j.eij.2022.05.006.

[45] S. Lamprier *et al.*, "Generative Cooperative Networks for Natural Language Generation," Oct. 2022. doi: 10.48550/arXiv.2201.12320.

[46] I. Wasserman, "Modeling the Graphotactics of Low-Resource Languages Using Sequential GANs," Oct. 2022. doi: 10.48550/arXiv.2210.14409.

[47] D. Croce, G. Castellucci, and R. Basili, "GAN-BERT: Generative Adversarial Learning for Robust Text Classification with a Bunch of Labeled Examples," Oct. 2020, pp. 2114–2119. doi: 10.18653/v1/2020.acl-main.191.

[48] M. Ozkan-Okay, R. Samet, Ö. Aslan, and D. Gupta, "A Comprehensive Systematic Literature Review on Intrusion Detection Systems," *IEEE Access*, vol. 9, pp. 157727–157760, 2021, doi: 10.1109/ACCESS.2021.3129336.

[49] H. Landaluce, L. Arjona, A. Perallos, F. Falcone, I. Angulo, and F. Muralter, "A review of iot sensing applications and challenges using RFID and wireless sensor networks," May 01, 2020, *MDPI AG*. doi: 10.3390/s20092495.

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

[51] S. Rahman, S. Pal, S. Mittal, T. Chawla, and C. Karmakar, "SYN-GAN: A robust intrusion detection system using GAN-based synthetic data for IoT security," *Internet of Things (Netherlands)*, vol. 26, Jul. 2024, doi: 10.1016/j.iot.2024.101212.

[52] I. Ullah and Q. H. Mahmoud, "A Framework for Anomaly Detection in IoT Networks Using Conditional Generative Adversarial Networks," *IEEE Access*, vol. 9, pp. 165907–165931, 2021, doi: 10.1109/ACCESS.2021.3132127.

[53] C. Park, J. Lee, Y. Kim, J. G. Park, H. Kim, and D. Hong, "An Enhanced AI-Based Network Intrusion Detection System Using Generative Adversarial Networks," *IEEE Internet Things J*, vol. 10, no. 3, pp. 2330–2345, Feb. 2023, doi: 10.1109/JIOT.2022.3211346.

[54] J. He, X. Wang, Y. Song, Q. Xiang, and C. Chen, "Network intrusion detection based on conditional wasserstein variational autoencoder with generative adversarial network and one-dimensional convolutional neural networks," *Applied Intelligence*, vol. 53, no. 10, pp. 12416–12436, May 2023, doi: 10.1007/s10489-022-03995-2.

[55] T. Sood, S. Prakash, S. Sharma, A. Singh, and H. Choubey, "Intrusion Detection System in Wireless Sensor Network Using Conditional Generative Adversarial Network," *Wirel Pers Commun*, vol. 126, no. 1, pp. 911–931, Sep. 2022, doi: 10.1007/s11277-022-09776-x.