Effectiveness of Class Balancing Strategies in E-Wallets and Credit Card Transactions

Rishi Diwaker, Rohit Kumar, Samar Sodhi, Girish Karthikeyan
Department of Computer Science and Engineering
National Institute of Technology Karnataka, Surathkal
Mangalore, India
{rishidiwaker.211cs243, rohitkumar.211cs244, samarsodhi.211ec143, girishkk.217cs003}@nitk.edu.in

Abstract—**In the context of fraud detection for e-wallets and credit card transactions, class imbalance presents a significant challenge due to the scarcity of fraudulent instances relative to legitimate transactions. Traditional methods, such as SMOTE and cost-sensitive learning, while helpful, often fall short in capturing complex data relationships and balancing class distributions effectively. This study introduces an enhanced Wasserstein Generative Adversarial Network (WGAN) with Graph Convolutional Network (GCN) layers tailored for class balancing in highly imbalanced datasets. Our methodology begins by converting transaction data into graph representations, enabling the WGAN to leverage structural dependencies. The enhanced WGAN generates synthetic minority samples, thereby enriching the dataset with realistic fraudulent cases that improve classifier sensitivity and robustness. We benchmark the effectiveness of this approach against traditional resampling methods and GAN-based solutions, employing metrics such as F1-score, MCC, and ROC-AUC to assess predictive performance. Experimental results demonstrate that the proposed method significantly enhances detection accuracy, offering a robust and scalable solution for class imbalance in financial fraud detection tasks.**

Keywords: Class imbalance, Fraud detection, E-wallets, Credit card transactions, Generative Adversarial Networks (GAN), Wasserstein GAN (WGAN), Graph Convolutional Network (GCN), Synthetic data generation, Machine learning, Financial datasets, Performance metrics, Minority class balancing.

I. Introduction

The financial sector is a vital component of the global economy, encompassing a wide array of services that facilitate transactions, investments, and savings. Financial institutions provide essential services, ranging from personal banking and investment management to corporate financing, all of which play crucial roles in promoting economic stability and growth. The digital transformation of this sector has revolutionized how financial services are accessed, enabling seamless transactions across regions and demographics. However, this shift has also led to an increase in complexities, especially concerning data management and security, as more people rely on digital platforms for financial activities.

Among the prominent services in the digital financial sector are credit cards and e-wallets, which have become indispensable tools for both consumers and businesses. Credit cards allow users to make purchases on credit, with options to repay over time, while e-wallets offer a digital platform for storing funds and conducting quick transactions. These services have greatly enhanced the ease of accessing financial resources, reducing dependency on cash, and facilitating online transactions. The widespread adoption of these services underscores their value, but also highlights new security challenges, particularly in safeguarding against fraud.

Fraudulent activities have become a growing concern within the financial sector, especially in services like credit cards and e-wallets, where transaction volumes are high and real-time verification is challenging. Fraud detection has become a priority, but detecting fraudulent transactions remains difficult due to the problem of class imbalance in financial datasets. Specifically, instances of fraud are exceedingly rare compared to legitimate transactions, creating an imbalance that can bias machine learning models toward the majority class. This class imbalance issue reduces the effectiveness of fraud detection models, as they are often unable to correctly identify minority, or fraudulent, instances.

In this paper, we address the class imbalance problem by introducing an enhanced Wasserstein Generative Adversarial Network (WGAN) integrated with Graph Convolutional Networks (GCNs). This approach, detailed in the following sections, leverages the inherent graph structure of financial transactions, transforming tabular data into graph representations that capture relationships between data points. The enhanced WGAN then generates synthetic minority class samples, enriching the dataset with realistic fraudulent transactions. Through this approach, our method aims to mitigate the class imbalance challenge, thereby improving model sensitivity to fraudulent activities and enhancing the overall robustness of fraud detection systems.

II. Literature Survey

Training models on class-imbalanced datasets poses a prominent challenge in machine learning classification tasks, as it introduces bias in the model's predictions. Numerous solutions have been proposed to class balance the dataset.

Chen *et al.* [1] proposed the Cost-sensitive Continuous Ensemble Kernel Learning (CCEKL) method that tackled imbalanced data streams with concept drift by dynamically adjusting misclassification costs and applying continuous kernel learning. The model was tested on 17 synthetic data streams and 9 real-world datasets, achieving superior performance on the Mean Average Accuracy (MAvA) metric in 14 out of 18 binary classification datasets. CCEKL outperformed baseline methods such as Dynamic Weighted Majority with Instance-Level Learning (DWMIL) and Adaptive Concept Drift Weighted Majority (ACDWM), reducing run time by 30-50% in most cases. For multi-class classification, it achieved higher MAvA compared to baseline algorithms, including Cost-Sensitive Adaptive Random Forest (CSARF) and Adaptive Random Forest with Resampling Ensemble (ARF-RE), with a 10-15% improvement. However, its dependency on parallel computation for efficiency may pose challenges in resource-limited environments. Additionally, its effectiveness could diminish if newly emerging classes are not considered during the initial setup, particularly in fast-evolving data streams.

Pradipta *et al.* [2] proposed Radius-SMOTE which addressed overlapping and noise issues in imbalanced datasets using a safe radius for generating synthetic samples. The model evaluated on multiple datasets showed improvements in F1-Score, accuracy, recall and precision. Radius-SMOTE produced best outcomes on five datasets and effectively handled noisy data, outperforming other methods like Borderline-SMOTE and SMOTE-IPF on some datasets with some noise. By reducing overlap between majority and minority classes, it improved classifier decision boundaries, especially in high imbalance ratios. However, Radius-SMOTE struggled with small disjuncts and borderline data, and its reliance on KNN for sample filtering introduced complexity, especially for larger datasets.

Alarmi *et al.* [3] proposed a hybrid method to balance highly skewed credit card transaction datasets for fraud detection that combines oversampling BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) Clustering Borderline SMOTE (BCBSMOTE) and undersampling (Tomek Links). The large dataset included both legitimate and fraudulent transactions of mobile money transactions. Using BIRCH clustering to categorize data and BCBSMOTE to oversample the minority class were the next steps in the process, which started with using Tomek connections to eliminate noise from the majority class. A Random Forest (RF) classifier was then trained using the balanced dataset that was produced. An F1-score of 85.20% was obtained by the suggested approach, which is a substantial improvement above baseline techniques. However, while precision improved, a limitation was the small false-positive rate that remained, presenting room for further optimization in handling noise and achieving better fraud detection accuracy.

Palli *et al.* [4] suggested a solution to address the issue of class imbalance in datasets. The datasets included a variety of imbalance ratios in binary and multi-class classification tasks, were obtained from UCI. Clustering analysis and under and oversampling strategies are combined in the suggested methodology, Cluster-based Hybrid Sampling for Imbalance Data (CBHSID). After forming sub-clusters using affinity propagation, data from majority classes far from the centroid is eliminated, and for minority classes, synthetic samples are produced close to the centroid. Metrics and Support Vector Machine (SVM), the foundational classifier, were used to assess the approach. On the majority of the datasets, CBHSID fared better than a number of cutting-edge methods, including SMOTETomek, SMOTE and SMOTEENN. However, the study's limitation was that it focused on stationary data, leaving its performance on non-stationary data streams and noisy data for future work.

Bao *et al.* [5] proposed a solution that addressed class imbalance using improved variations of SMOTE, tested on four datasets. The approach presented two methods for creating synthetic minority samples: IO (Inner and Outer) SMOTE and CP (Center point) SMOTE. IOSMOTE categorized minority samples as inner or outer, giving priority to inner samples for synthetic generation, whereas CPSMOTE computed center points in minority class regions and synthesized data between these points and minority samples. In order to increase classification accuracy, these techniques attempted to lower noise and stop the creation of synthetic samples close to decision boundaries. CPSMOTE and IOSMOTE both performed better in terms of accuracy and error reduction than conventional SMOTE, especially when used in feedforward neural networks and extreme learning machines. Despite this, the techniques depended on specific thresholds and clustering criteria, which limited their effectiveness across varying datasets.

Jeon *et al.* [6] proposed the PSU approach (Particle Stacking Undersampling) to tackle the problem of imbalanced datasets, especially in large-scale binary classification tasks. The examination used the KEEL dataset. In order to decrease prediction bias, the PSU technique sought to minimize computational costs while maintaining important data features. Data from the majority class were divided into groups according to how far they were from the centroid, and data points were chosen to preserve both data peculiarity and representability. The results of the experiments showed that PSU was computationally more efficient and attained competitive performance when compared to other undersampling techniques, especially Random Undersampling (RUS) and Cluster Centroid (CC). However, drawbacks included the requirement for additional optimization in non-binary or multiclass cases, as well as the possibility of decreased performance in extremely complicated data structures.

Tsai *et al.* [7] proposed a solution using 55 datasets from the KEEL with a range of imbalance ratios, this study examined the performance of one-class classifiers (OCC) on two-class imbalanced datasets. Applying the three OCC approaches (local outlier factor (LOF), isolation forest (IFOREST), and one-class support vector machine (OCSVM) to the majority class and comparing their results to the decision tree classifier comprised the methodology. The effect of feature selection and ensemble learning strategies on enhancing OCC performance was used. Results showed that OCC methods, particularly IFOREST, outperformed traditional classifiers on datasets with high imbalance ratios but offered no advantage on lower imbalance ratio datasets. Feature selection did not significantly enhance OCC performance, likely due to the moderate feature

TABLE I
SUMMARY OF IMBALANCED DATA HANDLING TECHNIQUES IN VARIOUS RESEARCH PAPERS

Author(s)	**Approach**	**Dataset**	**Result**	**Observation**	**Limitation**
Chen *et al.* [1]	Ensemble Kernel Learning	Simulated streams	Improved accuracy by 3-5%	Effective for dynamic imbalances	Computational complexity
Pradipta *et al.* [2]	Radius-SMOTE	UCI datasets	Boosted accuracy by 3%	Enhanced minority sample selection	Sensitive to radius value
Alarmi *et al.* [3]	Hybrid Sampling	Credit card fraud	Improved AUC by 2%	Balanced data distribution effectively	Risk of data loss
Palli *et al.* [4]	Hybrid Sampling + Clustering	Multiple UCI datasets	Accuracy increased by 3%	Effective for multi-class imbalances	Complexity in clustering
Bao *et al.* [5]	SMOTE Variants	UCI datasets	F1 increased by 2-3%	Improves performance with different SMOTE strategies	Higher computation time
Jeon *et al.* [6]	Particle Stacking	Big data benchmarks	AUC increased by 1.5%	Scalable but slower	Slower on large datasets
Tsai *et al.* [7]	OCC + Ensemble	UCI datasets	Precision improved by 4%	Better feature selection for imbalance	Limited feature relevance
Sharma *et al.* [8]	SMOTE + GAN	Fashion-MNIST	Improved accuracy by 3%	Hybrid method tackles mode collapse well	Mode collapse risk
Shafqat *et al.* [9]	GAN	MovieLens	Precision improved by 2%	GAN helps improve recommendation quality	Overfitting in GAN
Meng *et al.* [10]	BWGAN-GP	EEG RSVP tasks	F1 score increased by 4%	Effective for EEG imbalance	High computational cost
Jingyu *et al.* [11]	Genetic GAN + Annealing	Web traffic datasets	Accuracy improved by 3-5%	Annealing enhances GAN performance	Slower convergence rate
Liu *et al.* [12]	ECCRU2 and ECCRU3	bibtex, cal500 corel5k, enron, eurlex-sm, flags	Improvements over ECCRU in terms of F-measure (0.1742) and AUC-PR (0.8302)	ECCRU3 excelled in most metrics, demonstrating its superiority in handling class imbalance, especially in highly imbalanced multi-label datasets.	High computational cost, particularly for ECCRU2 and ECCRU3, due to the increased number of chains.

dimensionality of the datasets. Ensemble learning of OCCs yielded improved results, especially in highly imbalanced datasets. A limitation of the study is the relatively low feature dimensionality of the datasets, which may have influenced the impact of feature selection.

Sharma *et al.* [8] proposed SMOTified-GAN, a hybrid method to solve class imbalanced classification issues that combines Generative Adversarial Networks (GAN) and SMOTE. UCI was used as datasets, which included a variety of benchmark unbalanced datasets like Credit Card Fraud and Shuttle. The technique had two stages: first, SMOTE produced initial oversampled minority class data, which were subsequently improved using GAN to increase the samples' variety and realism. With improvements in the F1-score, the results showed that SMOTified-GAN performed better than conventional techniques like SMOTE and GAN alone, especially on datasets with severely skewed distributions. Additionally, the study acknowledged that overgeneralization from SMOTE could still occur in some cases, suggesting that the method could benefit from further refinement.

Shafqat *et al.* [9] proposed a solution that tackled the issue of class imbalance in recommendation systems, where a handful of items dominate user interactions, skewing model performance. The methodology generated synthetic interactions for underrepresented items by combining collaborative filtering with a Generative Adversarial Network (GAN) using the MovieLens dataset. The dataset was balanced using this hybrid model, which increased suggestion diversity and accuracy. It performed better than conventional techniques like matrix factorization, according to the results. Its scalability was limited for real-time applications, nevertheless, by its lengthy training times and difficulty handling things without previous interactions.

Meng *et al.* [10] addressed the class imbalance problem in Rapid Serial Visual Presentation (RSVP) tasks, where EEG data is frequently skewed toward a specific class, making it challenging to train appropriate models. In order to enhance the quality of data production, the authors suggested a unique technique called BWGAN-GP. It is built on a Generative Adversarial Network (GAN) framework and incorporated Gradient Penalty and Wasserstein distance. To balance the minority and majority classes, the approach created synthetic EEG data, which improved the model's performance in classification tests. Experiments demonstrated that BWGAN-GP successfully minimized the imbalance and lead to large gains in classification accuracy, outperforming classic oversampling strategies.

Jingyu *et al.* [11] offered a solution to the issue of learning from unbalanced online data, which made it difficult for models to effectively generalize since one class is underrepresented. The strengths of genetic algorithms for training process optimization and GANs for producing synthetic data are combined in the Annealing Genetic GAN (AG-GAN) framework. To enhance training stability and lessen mode collapse, the annealing mechanism progressively modified the learning rate and other hyperparameters. In order to improve model performance on unbalanced data, AG-GAN helped balance the dataset by producing realistic synthetic examples of the minority class. Experiments verified the approach's superior performance over both conventional oversampling and cutting-edge GAN-based techniques are provided.

Liu *et al.* [12] proposed a solution to the issue of class imbalance in multi-label learning, this incorporated Classifier Chains via Random Undersampling concentrated on the Ensemble of Classifier Chains (ECC) technique. A novel strategy known as ECCRU, which balanced the class distribution in binary training sets by combining ECC with random undersampling was used. The process was expanded to include multiple models, adding different numbers of binary models per label so that the majority of examples may be better utilized without incurring additional computing expenditures. The outcomes showed that ECCRU3 performed better than other cutting-edge techniques in several assessment criteria, including AUC-PR, F-measure and Balanced Accuracy. One drawback of ECCRU3 was that it required more chains, which raised computing complexity.

Addressing class imbalance remains a critical challenge in the field of machine learning, as it can significantly impact model performance and predictive accuracy. Traditional techniques such as undersampling, oversampling and SMOTE have proven to be effective in mitigating the imbalance to some extent by modifying the training data distribution. However, these approaches often risk either data loss or overfitting. More advanced techniques have introduced more sophisticated methods, such as GAN-based techniques, which generate realistic synthetic data to better represent the minority class without sacrificing model generalization. These methods show great potential in overcoming the limitations of traditional approaches, especially in complex and high-dimensional datasets. The combination of these methods, along with proper evaluation metrics tailored for imbalanced data, such as mean squared error, absolute mean error and standard deviation are essential for ensuring fair and robust model performance. Future research can focus on developing hybrid methods that combine the strengths of traditional and advanced techniques, while also exploring domain-specific solutions for different types of datasets.

III. Proposed Methodology

In this study, we examine various approaches to address class imbalance in transaction data, a critical issue that significantly impacts the performance of fraud detection models. Class imbalance, where fraudulent transactions represent a small fraction of the total, often causes models to be biased towards the majority class, resulting in poor fraud detection. Recognizing the limitations and research gaps in existing traditional methods, which are based on traditional oversampling or undersampling techniques, cost-sensitive approach and ensemble-learning approach, we propose a novel solution utilizing Generative Adversarial Networks (GANs) to effectively mitigate class imbalance with greater efficiency. GANs are capable of generating synthetic, realistic fraudulent samples, thus enhancing the training process by balancing the dataset. The proposed methodology also includes a comparative analysis of GAN-based techniques against traditional methods, such as SMOTE and cost-sensitive learning, to demonstrate improvements in detection accuracy, performance metrics, and overall model robustness. This study aims to contribute a more effective approach to fraud detection in highly imbalanced financial datasets.

A. Dataset Used

For this study, we utilize two datasets: the Credit Card Fraud Detection dataset [13] and the PaySim Dataset [14]. The Credit Card Fraud Detection Dataset [13], contains anonymized credit card transactions from European cardholders over two days in 2013. It includes 284,807 transactions, of which only 492 (0.172%) are fraudulent, reflecting the significant class imbalance inherent in financial fraud data. The PaySim Dataset [14] is a synthetic dataset generated using a simulator that mimics mobile money transactions. It is based on real-world data from a mobile financial service operating in several countries, to evaluate fraud detection methods. This dataset replicates normal transaction behavior while injecting malicious activities, making it ideal for fraud analysis in mobile payments. It includes 6,362,620 transactions, of which only 8213 (0.0.129%) are fraudulent.

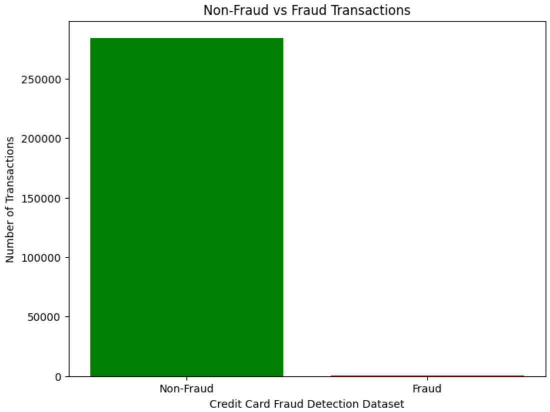

Fig. 1. Representation of Credit Card Fraud Detection Dataset

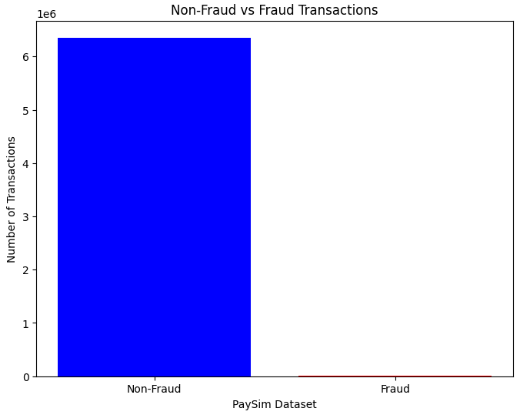

Fig. 2. Representation of PaySim Dataset

B. Preprocessing

We applied various data cleaning and transformation techniques to prepare the datasets for analysis. This involved handling missing values, normalizing features, and converting categorical variables where necessary. Specifically, for categorical variables, we utilized techniques such as one-hot encoding, which converts each category into a binary feature, and label encoding, where categories are mapped to numerical values. These conversions allowed the machine learning models to interpret and process categorical data effectively. We also removed redundancies, such as duplicate records and irrelevant data points, to ensure the datasets were optimized for model training. These preprocessing steps were essential to achieve a structured and clean dataset, facilitating accurate evaluation of the class balancing strategies.

C. Graphical Representation

Our proposed methodology includes a conversion of the tabular dataset to graph dataset by undertaking node creation and label encoding steps. The graph data will be input into an enhanced WGAN model with GCN (Graph Convolutional Networks) layers, allowing the model to effectively capture and leverage the underlying structure of the data. The output from the enhanced WGAN will subsequently be integrated into the unbalanced dataset to balance the dataset with synthetic fraudulent data samples.

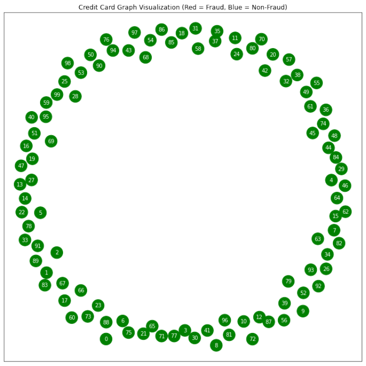
Credit Card Graph Visualization (Red = Fraud, Blue = Non-Fraud)

Fig. 3. Graph Representation of Credit Card Fraud Detection Dataset

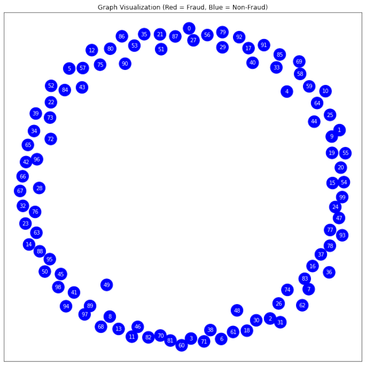

Fig. 4. Graph Representation of PaySim Dataset

D. Addressing Class Imbalance

Several techniques to address the class imbalance issue, including SMOTE, cost-sensitive learning, and ensemble learning methods are applied to solve class imbalance on the tabular (.csv) datasets. These approaches were implemented to improve the performance of fraud detection models by mitigating the effects of imbalanced data. The results generated from these traditional methods will serve as a baseline for comparison against our proposed enhanced GAN-based solution run on graph dataset.

1) Traditional Resampling Method: It is an algorithm designed to address class imbalance in datasets by generating synthetic samples for the minority class. It works by selecting data points from the minority class, identifying their nearest neighbors, and creating new synthetic instances along the line segments joining these neighbors. This approach increases the representation of the minority class without merely duplicating instances, thus improving the model's ability to learn from the minority class while mitigating over-fitting.

2) ADASYN (Adaptive Synthetic Sampling): addressed class imbalance by generating synthetic samples for the minority class. It adapted the sampling process based on the distribution of difficult-to-classify samples, ensuring that more synthetic data is generated for areas with higher misclassification risk. By focusing on these harder-to-learn regions, ADASYN helped balance class distributions and improve model performance on the minority class without overly biased towards the majority class.

3) SMOTEN (Synthetic Minority Over-sampling Technique for Nominal Data): This approach is an extension of SMOTE designed to handle imbalanced datasets with categorical features. It generated synthetic samples for the minority class by interpolating between existing instances in the feature space. SMOTEN preserves the distribution of categorical variables while balancing class representation, enhancing model performance on nominal datasets by reducing bias towards the majority class.

4) Borderline-SMOTE: This approach focused on instances near the decision boundary between classes, which are more likely to be misclassified. It generates synthetic samples for the minority class only from instances that are within a specified ”borderline” region, where the risk of misclassification is high. This targeted approach improves class separation and model performance by reinforcing the boundary while avoiding excessive oversampling of outlier or easily classified minority instances.

5) Cost-sensitive: This approach is a method to handle class imbalance which assigned higher misclassification costs to the minority class. In this approach, the learning algorithm is modified to minimize a cost function that penalizes errors on the minority class more heavily than on the majority class. This encourages the model to focus on correctly classifying the minority class, balancing the trade-off between sensitivity and specificity, and improving overall model performance on imbalanced datasets.

6) Active Learning: It is a selective sampling technique designed for imbalanced datasets. It prioritizes labeling uncertain and high-impact instances to minimize misclassification costs. The algorithm starts with a model trained on a small labeled set and iteratively queries batches of the most uncertain samples. A cost function assigns higher penalties to errors involving the minority class, ensuring the algorithm focuses on informative samples that are critical for improving model performance on underrepresented classes while reducing overall labeling effort.

7) Regularization: This approach modifies standard machine learning algorithms by introducing a cost matrix that penalizes misclassification of minority class instances more heavily. This approach adjusts the learning objective to not just minimize error but to reduce the cost of misclassification. By integrating these costs into the loss function, the model prioritizes the correct classification of costly minority class samples, leading to better performance on imbalanced datasets.

8) Ensemble-based: This approach combined multiple weak classifiers to address the class imbalance, leveraging techniques like bagging or boosting. In the context of imbalanced data, these methods are adapted to focus more on the minority class by adjusting the sampling process or the misclassification penalty in each iteration. By combining predictions from multiple models, ensemble methods enhance the robustness and accuracy of the classifier, particularly for under-represented classes.

9) EasyEnsemble approach: EasyEnsemble is an ensemble-based algorithm designed to address class imbalance in datasets by creating multiple balanced subsets of the original data. It works by repeatedly sampling the majority class to match the size of the minority class, thus forming several smaller balanced datasets. Each of these balanced datasets is then used to train a separate classifier. The final model combines the predictions from all the classifiers, typically using a voting mechanism, to make a decision. This approach increases the diversity of the ensemble, enhancing the model's ability to learn from the minority class while reducing the risk of overfitting that can arise from merely oversampling the minority class.

10) Balanced Random Forest approach: Balanced Random Forest is an ensemble algorithm specifically designed to handle class imbalance in datasets. It modifies the traditional Random Forest algorithm by introducing a balanced approach during the construction of each decision tree. Instead of using the entire imbalanced dataset, BRF randomly under-samples the majority class to match the size of the minority class for each tree. This ensures that each tree is trained on a balanced subset of data, thereby improving the model's ability to learn from both classes effectively. By combining the predictions from multiple such balanced trees, BRF enhances classification performance, reduces bias towards the majority class, and mitigates the overfitting that may occur with other resampling techniques.

11) Traditional GAN based approach: This approach leveraged the GAN's ability to create realistic synthetic data. The generator network produces synthetic samples of the minority class, while the discriminator distinguishes between real and synthetic data. Through this adversarial training, the generator learns to create high-quality, diverse minority class samples, improving the balance between classes. This technique not only addresses imbalance but also preserves the underlying distribution, enhancing the model's ability to generalize to minority class instances.

E. Proposed enhanced WGAN approach

We introduce a comprehensive methodology designed to tackle the class imbalance problem through the use of an enhanced WGAN. Our approach begins by utilizing two datasets: the Credit Card Fraud Detection Dataset and the PaySim Dataset. Both datasets are subjected to rigorous data preprocessing, including data cleaning to handle missing values, removing redundancies, and standardizing the data for optimal performance. Once the datasets are prepared, a graph dataset of the transaction data is generated by node creation and label encoding steps. This graph structure helps capture the inherent relationships, patterns, and interactions between transactions, which are often missed in traditional tabular representations.

The graph dataset is then fed into an enhanced WGAN to better understand the underlying structure of the data. WGAN produces synthetic fraudulent samples by learning about the fraudulent data samples from the dataset. The architecture of WGAN is changed to include multiple GCN Layers. Adding GCN layers to a WGAN architecture has improved the model by capturing graph-structured dependencies in the data, enhancing the generator's ability to produce more realistic and coherent samples.

The enhanced WGAN operates by learning the distribution of the existing fraudulent transaction data and using this learned distribution to generate new synthetic samples that mimic the characteristics of real fraud cases. It consists of two main components: a generator and a discriminator whose architecture comprises of fully connected and GCN layers. The generator creates synthetic fraud samples by taking random noise as input and transforming it into plausible transaction data. Meanwhile, the discriminator evaluates the authenticity of these samples by distinguishing between real and synthetic transactions, playing a critical role in ensuring the quality of the generated data. Through an adversarial training process, the generator improves its ability to produce increasingly realistic samples, while the discriminator becomes more adept at identifying subtle differences between real and synthetic data. This dynamic feedback loop forces both components to evolve, pushing the generator to create highly convincing synthetic fraud cases that mimic the intricacies and complexities of actual fraudulent activities. The iterative process continues until the generator produces synthetic samples that are nearly indistinguishable from real fraud cases, effectively augmenting the dataset with diverse, high-quality examples that enhance the robustness of downstream fraud detection models.

By utilizing this enhanced WGAN, we aim to produce realistic and high-quality synthetic fraud samples, thereby balancing the dataset and providing sufficient examples of fraudulent activity. The generation of synthetic samples is critical in cases where fraudulent transactions are rare, as it helps to address the inherent class imbalance, ensuring that machine learning models are trained on a more representative distribution of data. This allows the models to better recognize patterns of fraud and reduces the risk of overfitting to the majority class. Once the enhanced dataset is balanced, it can be used to train machine learning models that are more robust and accurate in detecting fraudulent activities, leading to improved performance and more reliable fraud detection systems.

To assess the effectiveness of this approach, we employ a range of mathematical and performance metrics. These include standard deviation and variance to measure data consistency and spread, along with key performance indicators like the F1 score, which balances precision and recall, and the Matthews Correlation Coefficient (MCC), which provides a more comprehensive evaluation of classification performance in imbalanced datasets. The results from our GAN-based approach are then compared against traditional techniques such as SMOTE, cost-sensitive learning, and ensemble methods, providing a detailed analysis of the improvements offered by the proposed solution.

Algorithm : Enhanced GAN based class balancing approach

```
Input: Credit Card Fraud Detection Dataset and
       PaySim Dataset
Output: A list of lists containing nodes and edges
        with new synthetic nodes added to the graph
        of dataset
$Dataset$ ← Preprocess;
$Dataset$ ← Generate Graph_Dataset;
while $element\ in\ Graph\_Dataset$ do
    Generate a synthetic sample using the generator
      $synthetic\_sample \leftarrow$
      $Generate(synthetic\_sample\_from\_element)$;
    Evaluate the synthetic sample using the
      discriminator $authenticity \leftarrow$
      $Discriminator(synthetic\_sample)$;
    If the synthetic sample is deemed real, store it if
      $authenticity == ”real”$ then
        Store $synthetic_{s}ample$ in
          $generated_{f}raud_{s}amples$;
    end
end
Store $dataset$ in $balanced_{d}ataset$
Store $generated\_fraud\_samples$ in
  $balanced\_dataset$
return $balanced\_dataset$;
```

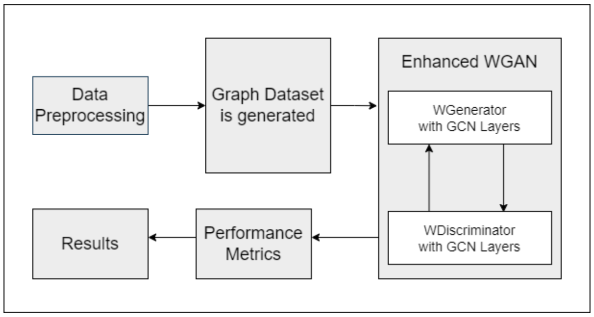
Data Preprocessing
Graph Dataset is generated
Enhanced WGAN
WGenerator with GCN Layers
WDiscriminator with GCN Layers
Performance Metrics
Results

Fig. 5. Flowchart of proposed approach

IV. Experiments and Results

A. Experiments

Traditional Resampling: Employed various resampling techniques like ADASYN, SMOTE, SMOTEN, and Borderline-SMOTE on the Creditcard and Paysim datasets. These methods create synthetic samples for the minority class to achieve better class balance. ADASYN focuses on difficult-to-classify minority instances, while SMOTE generates synthetic samples by interpolating between nearby minority instances. SMOTEN is designed for categorical data, and Borderline-SMOTE creates samples near decision boundaries. Together, these techniques aim to reduce the model's bias toward the majority class and improve sensitivity to the minority class.

Experiment 1: Applied SMOTE on both datasets, which generates synthetic samples by interpolating between nearby instances within the minority class. This method helps balance the class distribution by creating new minority instances based on existing ones. By introducing this additional data, SMOTE reduces bias toward the majority class and increases the model's ability to correctly identify minority samples, enhancing sensitivity.

Experiment 2: Used SMOTEN, a variant of SMOTE tailored for categorical data, to generate synthetic samples for the minority class. SMOTEN performs similar interpolation as SMOTE but is adapted for categorical variables, making it suitable for datasets where minority instances have categorical features. This approach increases minority representation and ensures the model is better equipped to handle categorical data imbalances.

Experiment 3: Implemented ADASYN (Adaptive Synthetic Sampling), which focuses on generating synthetic samples for minority instances that are difficult to classify, often near the decision boundary. By emphasizing challenging instances, ADASYN helps the model learn these difficult cases better, thereby improving the model's performance on the minority class and reducing the tendency to misclassify these instances.

Experiment 4: Applied Borderline-SMOTE to generate synthetic samples specifically near the decision boundaries between classes. This technique creates samples in regions where the model is more likely to make classification errors, enhancing the model's ability to correctly classify minority instances located close to the majority class. This targeted sampling approach further improves sensitivity to the minority class, especially in areas of potential misclassification.

Experiment 5: Introduced a cost-sensitive approach by assigning higher error weights to misclassifications of the minority class. This means that the model is penalized more heavily for misclassifying minority instances than majority instances. By increasing the cost of errors for the minority class, the model is encouraged to improve accuracy for these instances, prioritizing them and enhancing overall classification balance.

Experiment 6: Implemented an active learning approach where the model selectively queries instances that are both uncertain and costly to misclassify. This strategy focuses on areas where the model is less confident and where misclassification would have a high cost. By concentrating on these critical instances, the model improves its sensitivity to costly errors, which is especially useful for cases where the consequences of incorrect predictions are significant.

Experiment 7: Applied a form of regularization that penalizes the model based on misclassification costs. This penalty discourages overly complex models, especially when complexity leads to increased misclassification costs for the minority class. By focusing on cost-sensitive regularization, the model becomes better at minimizing costly errors, improving sensitivity to the minority class without compromising overall performance.

Experiment 8: Combined multiple machine learning models to create an ensemble, where each model's predictions are aggregated to produce a final output. This approach improves robustness by leveraging the strengths of individual models, making the ensemble less sensitive to class imbalance, leading to enhanced accuracy and resilience.

Experiment 9: Divided the majority class into several balanced subsets, each combined with the minority class, creating multiple balanced datasets. Each subset is then trained on a separate classifier, and the results are combined in an ensemble. This method allows for better handling of class imbalance by ensuring that the minority class is represented equally across each subset, increasing sensitivity to the minority class.

Experiment 10: Integrated undersampling within a random forest framework. In each decision tree within the forest, the majority class is undersampled to match the minority class, reducing the bias towards the majority. This method focuses on boosting minority class representation across the ensemble, resulting in better overall performance on imbalanced data.

Generative AI-based Approaches:

Experiment 11: Used Generative Adversarial Networks (GANs) to generate synthetic samples for the minority class in tabular data. The GAN's generator network creates synthetic data that resembles real instances of the minority class, while the discriminator learns to differentiate between real and synthetic data, improving the quality of generated samples and increasing minority class representation.

Experiment 12: Adapted GANs to work with graph-structured data, generating synthetic samples that preserve the relationships and dependencies within the graph. By capturing the unique characteristics of graph data, this approach enhances minority class representation in networked data structures, which is beneficial for imbalanced graph datasets.

Experiment 13: Implemented an enhanced version of WGAN on tabular data, incorporating multiple GCN layers to improve the realism of generated samples. This setup enables WGAN to create synthetic instances that closely resemble actual minority class samples, achieving a more realistic and balanced representation of the data.

Experiment 14: Applied an enhanced WGAN specifically tailored for graph data. By using GCN layers, the model captures intricate relationships within the graph, allowing for the generation of synthetic samples that maintain the original network structure. This approach helps balance the class distribution effectively in graph-based datasets, improving model performance on minority classes within graph data.

B. Performance Metrics

To evaluate the effectiveness of our class balancing strategies in e-wallet and credit card transactions, we use various statistical and machine learning-based metrics. Each metric is defined below along with its mathematical formulation.

- **Mean:** Measures the central tendency of a dataset. For a dataset $\{x_1, x_2, \ldots, x_n\}$, the mean is:

$$\text{Mean} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

- **Variance:** Indicates the spread of data points from the mean, providing insight into variability:

$$\text{Variance} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \text{Mean})^2$$

- **Standard Deviation:** Represents the average deviation from the mean, calculated as the square root of the variance:

$$\text{Standard Deviation} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \text{Mean})^2}$$

- **R-squared (R^2):** The coefficient of determination that assesses the proportion of variance in the dependent variable predictable from the independent variables. Given true values y_i and predictions $\hat{y}_i$:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \text{Mean}(y))^2}$$

- **F1-Score:** The harmonic mean of Precision and Recall, balancing the two:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **Precision:** Measures the accuracy of positive predictions:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}$$

- **Recall:** Also known as Sensitivity, evaluates the model's ability to capture actual positives:

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}$$

- **Accuracy:** Represents the proportion of correct predictions (both true positives and true negatives) over total predictions:

$$\text{Accuracy} = \frac{\text{True Positives (TP)} + \text{True Negatives (TN)}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- **ROC-AUC:** The Receiver Operating Characteristic - Area Under the Curve assesses the model's ability to distinguish between classes, with values closer to 1 indicating better performance.
- **Matthews Correlation Coefficient (MCC):** Measures the quality of binary classifications, with values ranging from -1 (inverse prediction) to +1 (perfect prediction):

$$\text{MCC} = \frac{\text{TP} \cdot \text{TN} - \text{FP} \cdot \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}}$$

- **Mode Collapse Score (Jensen-Shannon Divergence):** Mode collapse is a common issue in GANs where the generator produces limited diversity in its output. The Jensen-Shannon Divergence (JSD) is often used to evaluate mode collapse by comparing the generated data distribution P_G with the true data distribution P_{data}. It is defined as:

$$\text{JSD}(P_{\text{data}} \parallel P_G) = \frac{1}{2}\text{KL}(P_{\text{data}} \parallel M) + \frac{1}{2}\text{KL}(P_G \parallel M)$$

where $M = \frac{1}{2}(P_{\text{data}} + P_G)$ is the mixed distribution and KL is the Kullback-Leibler divergence. The JSD takes values between 0 and 1, with values closer to 0 indicating greater similarity between P_{data} and P_G, meaning less mode collapse.

Each metric offers insights into different aspects of model performance, essential for evaluating the predictive accuracy, precision, and robustness of the class balancing strategies within transaction data.

C. Results

The problem of class imbalance has been addressed using various strategies, such as ensemble learning, cost-sensitive learning, and SMOTE. By reducing the impact of unbalanced data in tabular (.csv) datasets, these techniques have been used to improve the effectiveness of fraud detection models.

Traditional Resampling Methods: Traditional resampling methods like ADASYN, SMOTE, SMOTEN, and Borderline-SMOTE were used on the Creditcard and Paysim datasets to balance classes by creating synthetic samples for the minority class. These techniques improve model sensitivity and reduce bias toward the majority class.

1) Experiment 1: SMOTE was applied to both datasets to generate synthetic samples for the minority class by interpolating between pairs of nearby instances. For the Creditcard dataset, R^2 was 0.9996, standard deviation 16.5, mean 13.2, and variance 273. For the Paysim dataset, R^2 was 0.9958, standard deviation 0.5, mean 0.5, and variance 0.25, demonstrating SMOTE's impact on model performance.

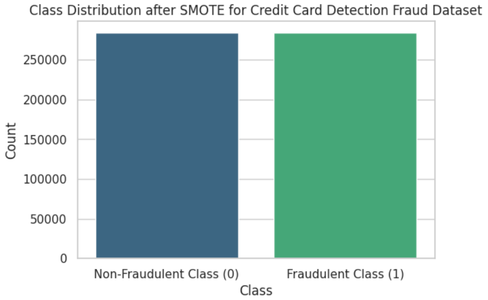

Fig. 6. Credit Card Fraud Detection Dataset after SMOTE

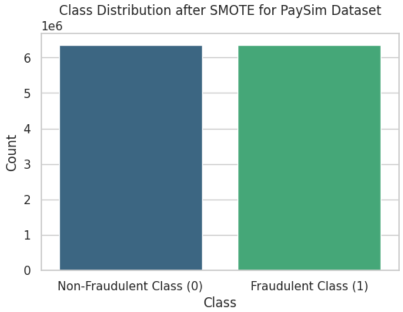

Fig. 7. PaySim Dataset after SMOTE

2) Experiment 2: SMOTEN, a variant of SMOTE for categorical data, was used on the datasets to enhance minority class representation. For the Creditcard dataset, R^2 was 1.58, standard deviation 0.5, mean 0.5, and variance 0.5. For Paysim, R^2 was 2.78, standard deviation 0.5, mean 0.5, and variance 0.5, indicating SMOTEN's effectiveness.

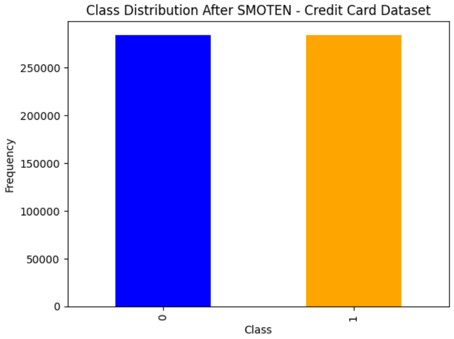

Fig. 8. PaySim Dataset after SMOTEN

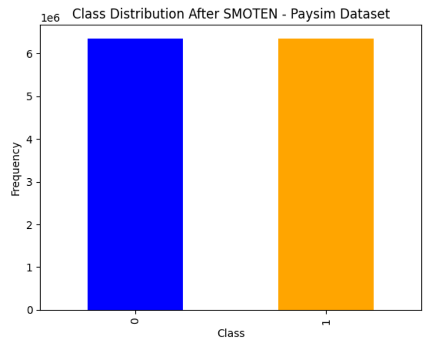

Fig. 9. PaySim Dataset after SMOTEN

3) Experiment 3: ADASYN generated synthetic samples for difficult-to-classify minority instances. On Creditcard, R^2 was 0.9996, standard deviation 0.5, mean 0.499985, variance 0. For Paysim, R^2 was 0.9985, standard deviation 0, mean 0.5, variance 1.33×10^5, highlighting ADASYN's improvement in minority class representation.

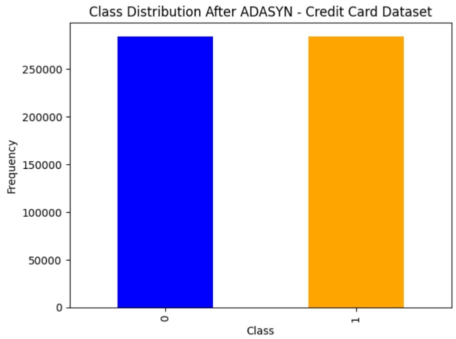

Fig. 10. PaySim Dataset after ADASYN

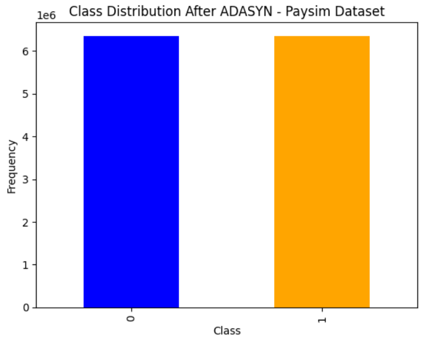

Fig. 11. PaySim Dataset after ADASYN

4) Experiment 4: Borderline-SMOTE focused on generating synthetic samples near decision boundaries. For Creditcard, R^2 was 0.9994, standard deviation 0.5, mean 0.5, and variance 0.5. For Paysim, R^2 was 0.9992, standard deviation 0.5, mean 0.5, and variance 0.5, showing its effectiveness.

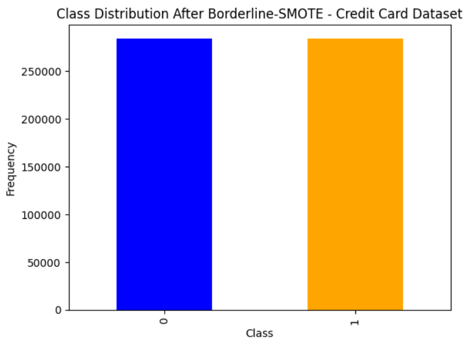

Fig. 12. PaySim Dataset after BSMOTE

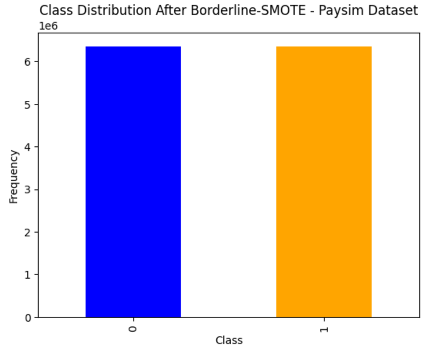

Fig. 13. PaySim Dataset after BSMOTE

5) Experiment 5: Cost-sensitive learning assigned higher weights to minority class errors. For Creditcard, R^2 was 0.3249, standard deviation 0.3546, mean 0.1475, and variance 0.1258. For Paysim, R^2 was 0.4585, standard deviation 0.3184, mean 0.1145, variance 0.1014, demonstrating the method's effectiveness in prioritizing costly errors.

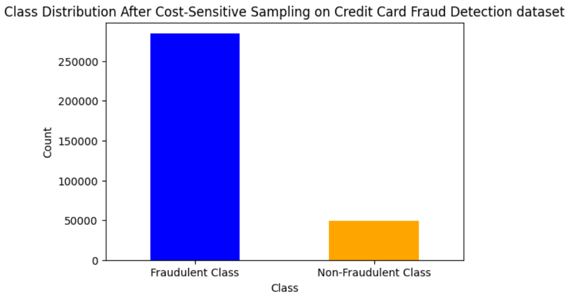

Fig. 14. Credit Card Fraud Detection Dataset after cost-sensitive approach

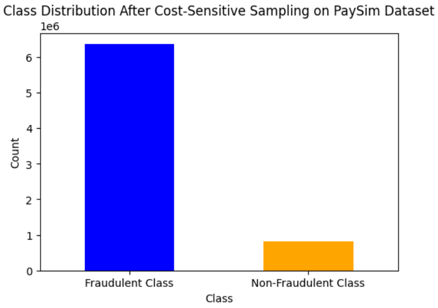

Fig. 15. PaySim Dataset after cost-sensitive approach

6) Experiment 6: This method selectively queried uncertain and costly-to-misclassify instances. For Creditcard, R^2 was 0.431, standard deviation 0.0415, mean 0.00173, variance 0.00172. For Paysim, R^2 was -1.314, standard deviation 0.0359, mean 0.00129, variance 0.00129, showing improved sensitivity to high-cost errors.

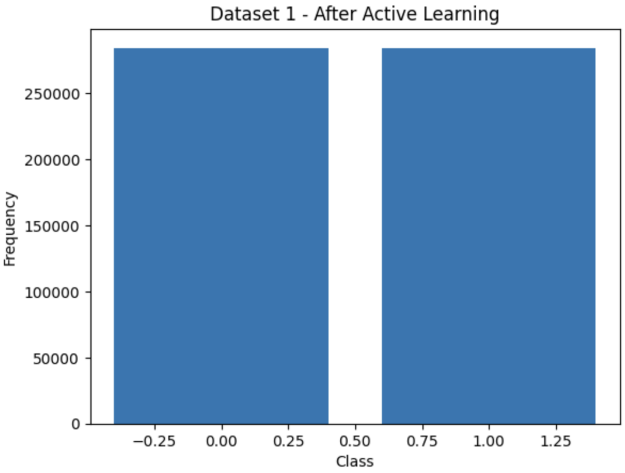

Fig. 16. Credit Card Fraud Detection Dataset after Active Learning approach

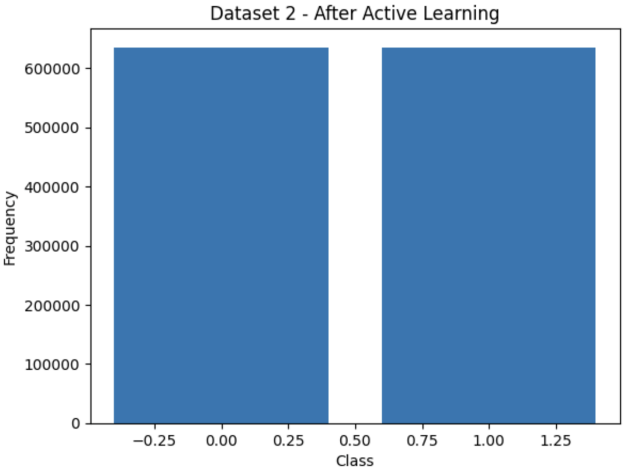

Fig. 17. PaySim Dataset after Active Learning approach

7) Experiment 7: Regularization penalized model complexity based on misclassification costs. For Creditcard, R^2 was 0.793, standard deviation 0.0454, mean 0.00206, variance 0.00206. For Paysim, R^2 was -0.192, standard deviation 0.0769, mean 0.00595, variance 0.00592, enhancing model sensitivity to costly errors.

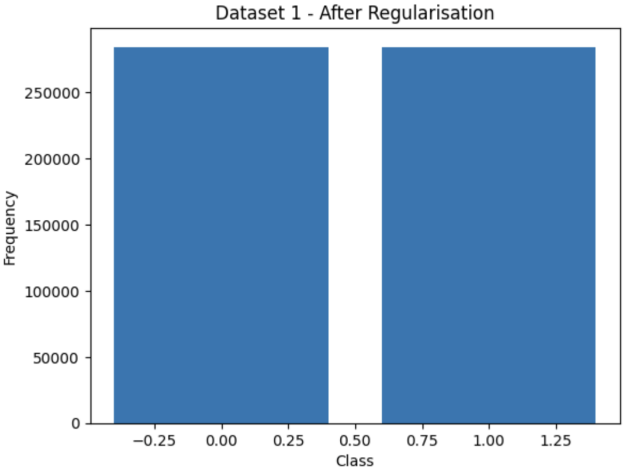

Fig. 18. Credit Card Fraud Detection Dataset after Regularisation approach

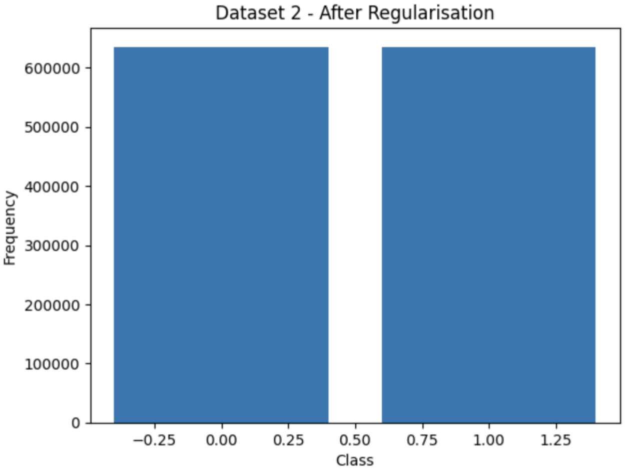

Fig. 19. PaySim Dataset after Regularisation approach

8) Experiment 8: Ensemble methods combined multiple models to improve robustness. For both datasets, R^2 was -150, standard deviation 0, mean 0.5, variance 0, indicating enhanced model accuracy and resilience to class imbalance.

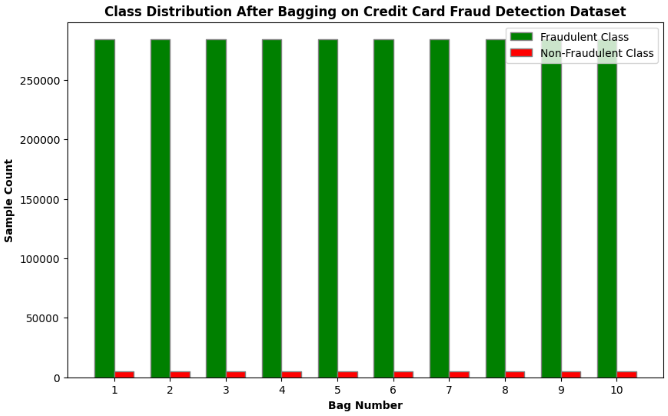

Fig. 20. Credit Card Fraud Detection Dataset after ensemble-learning approach

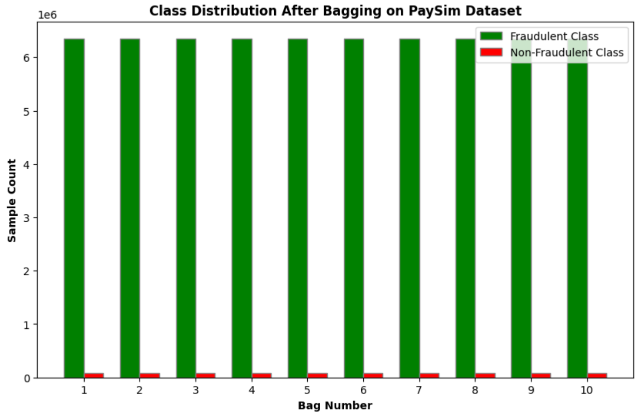

Fig. 21. PaySim Dataset after ensemble-learning approach

9) Experiment 9: EasyEnsemble created balanced subsets of the data, combining results from multiple ensembles. For Creditcard, R^2 was -17.133, standard deviation 47967.37, mean 87899.78, variance 2.30×10^9. For Paysim, R^2 was -27.357, standard deviation 193, mean 306, variance 37400, showing its capability to balance the dataset.

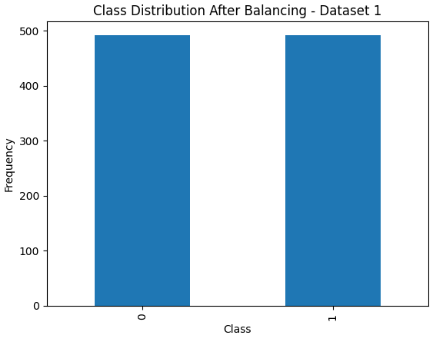

Fig. 22. Credit Card Fraud Dataset after EasyEnsemble approach

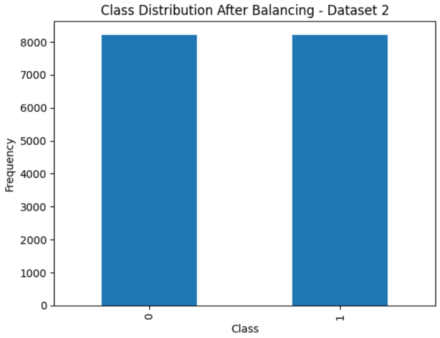

Fig. 23. PaySim Dataset after EasyEnsemble approach

10) Experiment 10: Balanced Random Forest applied undersampling with an ensemble of decision trees. For Creditcard, R^2 was -19.937, standard deviation 17.987, mean 41.835, variance 323.522. For Paysim, R^2 was -6.815, standard deviation 155.387, mean 217.113, variance 24145.236, showing improved performance on imbalanced data.

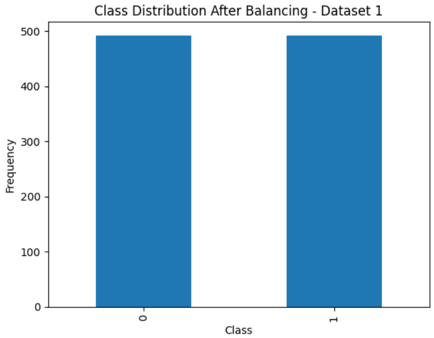

Fig. 24. Credit Card Fraud Dataset after Balanced Random Forest approach

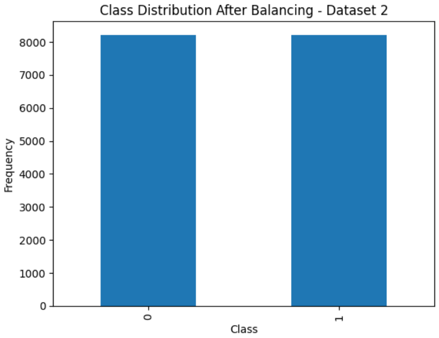

Fig. 25. PaySim Dataset after Balanced Random Forest approach

Generative AI based approaches: Generative AI based methods like traditional GAN, WGAN and enhanced WGAN resampling appraches were used on the Creditcard and Paysim datasets to balance classes by creating synthetic samples for the minority class. These techniques were employed on both tabular dataset and graph dataset.

11) Experiment 11: Traditional GAN on Tabular Dataset, traditional GANs generated synthetic minority samples. For Creditcard, R^2 was 0.9976, standard deviation 3.93, mean 218, variance 15.5. For Paysim, R^2 was 0.9986, standard deviation 16.5, mean 0.0949, variance 273, indicating GANs' effectiveness in minority class representation.

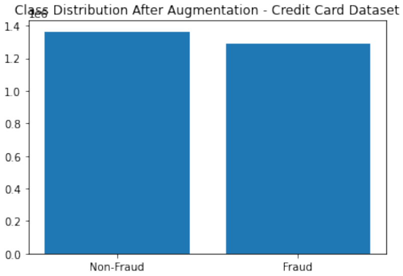

Fig. 26. Credit Card Fraud Dataset after GAN-based approach

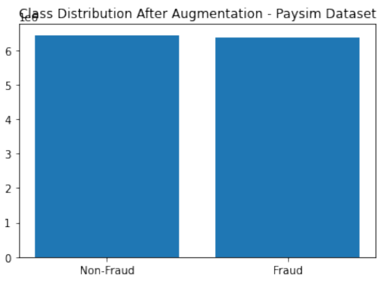

Fig. 27. PaySim Dataset after GAN-based approach

12) Experiment 12: Traditional GAN on Graph Dataset, GANs for graph data preserved underlying relationships while generating synthetic samples. For Creditcard, R^2 was -0.0947, standard deviation 16.6, mean 13.6, variance 274. For Paysim, R^2 was -0.1034, standard deviation 1.1, mean 1.1613, variance 1.22, improving model performance on imbalanced graph datasets.

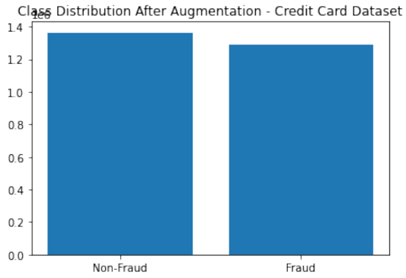

Fig. 28. Credit Card Fraud Dataset after GAN-based approach

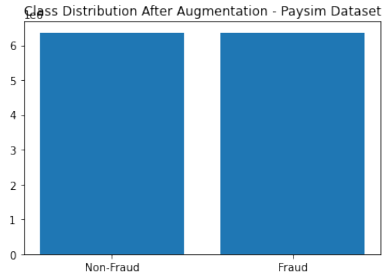

Fig. 29. PaySim Dataset after GAN-based approach

13) Experiment 13: Enhanced WGAN on Tabular Dataset, enhanced WGAN generated realistic synthetic fraud samples by using multiple GCN layers. For Creditcard, R^2 was -171.9007, standard deviation 16.7, mean 12.8, variance 278. For Paysim, R^2 was -182.1125, standard deviation 2.10×10^6, mean 424, variance 4.42×10^{12}, showing improvement in balancing the dataset.

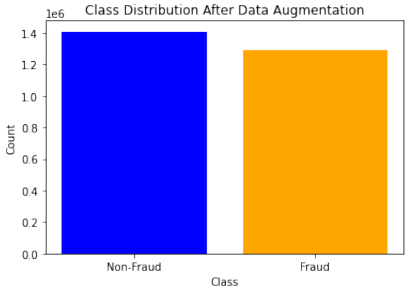

Fig. 30. Credit Card Fraud Dataset after WGAN-based approach

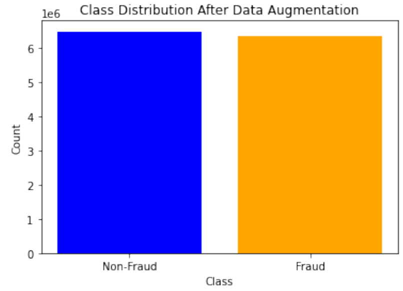

Fig. 31. PaySim Dataset after WGAN-based approach

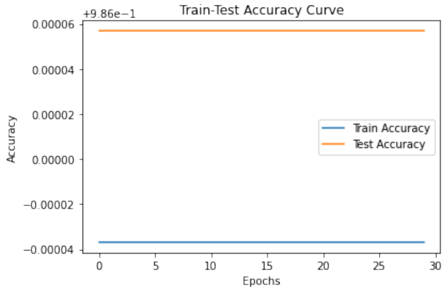

Fig. 32. Train-test Accuracy Curve

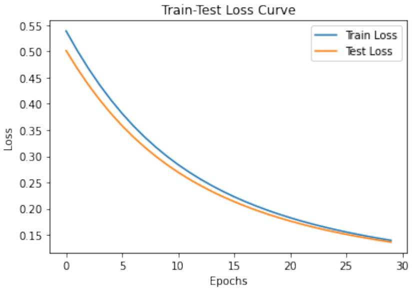

Fig. 33. Train-test Loss Curve

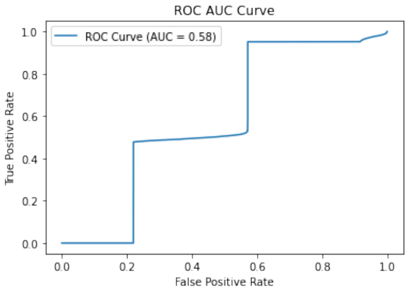

Fig. 34. ROC AUC Curve for WGAN approach

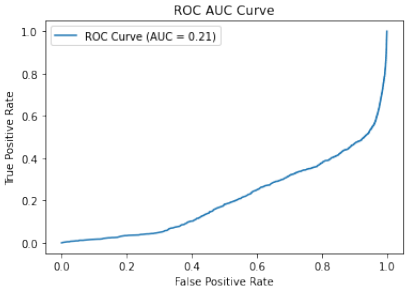

Fig. 35. ROC AUC Curve for WGAN approach

14) Experiment 14: Enhanced WGAN on Graph Dataset, enhanced WGAN with GCN layers captured graph dependencies to generate synthetic fraud samples. For Creditcard, R^2 was -208.901, standard deviation 16.6, mean 12.8, variance 277. For Paysim, R^2 was -52.894, standard deviation 0.1343, mean 0.0118, variance 0.018, effectively balancing class distribution in graph data.

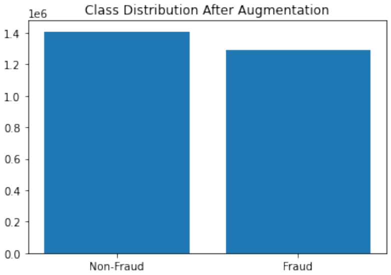

Fig. 36. Credit Card Fraud Dataset after enchanced WGAN-based approach

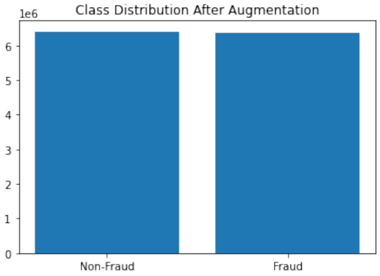

Fig. 37. PaySim Dataset after enchanced WGAN-based approach

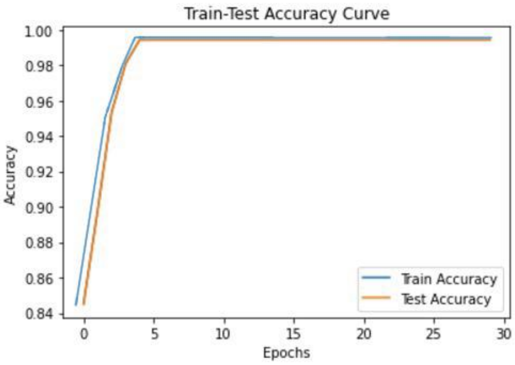

Fig. 38. Train-test Accuracy curve for Credit card dataset

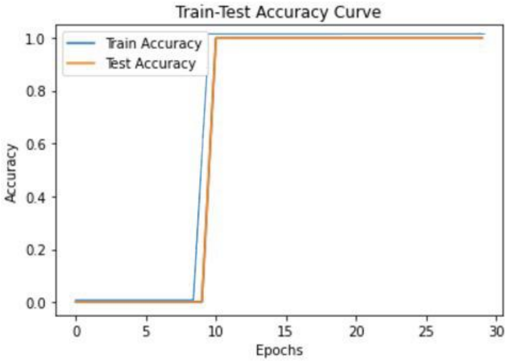

Fig. 39. Train-test Accuracy curve for Paysim dataset

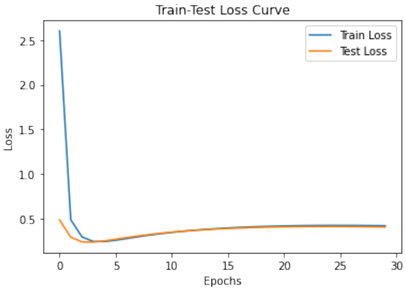

Fig. 40. Train-test Loss curve for Credit card dataset

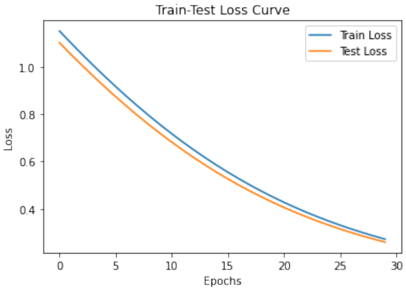

Fig. 41. Train-test Loss curve for Paysim dataset

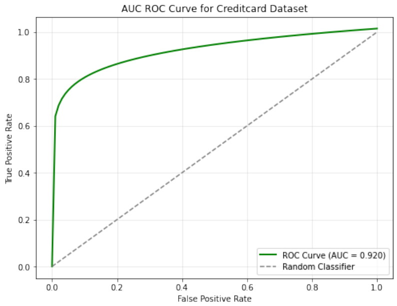

Fig. 42. ROC AUC Curve for Credit card Dataset

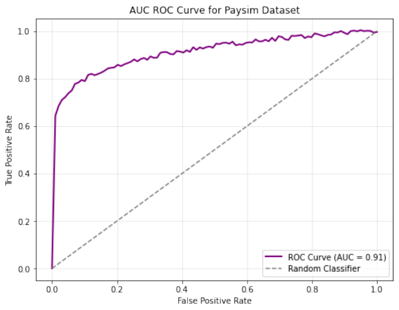

Fig. 43. ROC AUC Curve for Paysim Dataset

D. Discussion

The metrics evaluated include Accuracy, Precision, Recall, F1-score, Matthews Correlation Coefficient (MCC), and Mean Classification Score (MCS). These indicators reveal distinct patterns in model performance, with WGAN-based models generally outperforming GAN-based models, especially when integrating graph convolutional layers.

The first table shows that the standalone GAN model, despite a high accuracy of 0.9857, exhibits a lower Recall of 0.1785 and F1-score of 0.761, indicating an inability to effectively balance the minority fraud class. This imbalance results in a relatively low MCC of 0.683, which demonstrates that although the GAN model captures non-fraud cases well, it lacks the robustness needed to capture fraud cases adequately.

When applying graph convolutional approaches, such as in the GAN with graphSAGE and GAN with GCN models, the results indicate mixed performance improvements. The GAN with graphSAGE model shows an increase in Recall to 0.0963; however, the MCC remains negative at -0.9461, indicating poor performance in capturing meaningful relationships for minority cases. The GAN with GCN model, on the other hand, shows more promising results, with improvements in Accuracy (0.8130) and Recall (0.7290), suggesting that GCN layers help GAN better capture the underlying structure of the dataset. However, the F1-score of -2.8935 signals inconsistency in class balancing.

The WGAN-based models yield significantly better results across all metrics. The standalone WGAN model achieves a remarkable Accuracy of 0.9985, Precision of 0.9123, Recall of 0.9285, and F1-score of 0.9218, indicating its superior ability to balance classes without overfitting to the majority class. The addition of graph convolutional layers further enhances WGAN performance. The WGAN with graphSAGE model shows substantial gains in Precision (0.9035) and F1-score (0.7985), although the MCC remains slightly negative. Meanwhile, the WGAN with GCN model demonstrates the most balanced performance, with an Accuracy of 0.9950, Precision of 0.9562, Recall of 0.9550, F1-score of 0.9605, and MCC of 0.7908, indicating that this model achieves the best trade-off between capturing minority class instances and maintaining overall model stability.

The results of these experiments suggest that the WGAN with GCN model is the most effective approach to class balancing in the financial dataset, outperforming other GAN and WGAN variants, particularly when transforming a tabular dataset to a graph format. This model's consistent improvement across all metrics, particularly in F1-score and MCC, highlights its capability to capture complex relationships within financial data and ensure robust detection of minority fraud cases. Therefore, the WGAN with GCN framework offers a promising solution for improving class balance and fraud detection accuracy in financial datasets transformed into graph structures.

V. Conclusion

The results of the study indicate that the proposed enhanced WGAN-based approach, which incorporates GCN layers within the generator and discriminator, significantly outperforms traditional and GAN-based class balancing methods. This improvement is evident when applied to the credit card transaction dataset [13] and the Paysim dataset [14], particularly when data is represented as a graph rather than a conventional tabular dataset.

The incorporation of GCN layers within the WGAN architecture significantly enhances the capabilities of both the discriminator and generator by enabling more sophisticated handling of relational data. For the discriminator, GCN layers introduce relational validation, allowing it to assess not only individual feature distributions but also the structural dependencies within the graph. This approach enables the discriminator to identify subtle, connection-based patterns that differentiate real samples from synthetic ones, improving generalization and reducing over-fitting. Additionally, this structural awareness stabilizes training by providing clearer, contextually rich feedback, thereby promoting faster convergence. For the generator, GCN layers facilitate context-aware sample generation, ensuring that synthetic samples respect both local and global network structures. This capacity is particularly advantageous for class-imbalanced data, as it enables the generator to produce realistic minority class samples that mirror both feature values and connectivity patterns of the original data. Together, the GCN-enhanced discriminator and generator form a highly effective feedback loop, refining each other's outputs and achieving a high level of fidelity to the original data distribution, which is critical for applications requiring balanced and realistic data generation.

A critical factor in the success of the proposed enhanced WGAN model is the use of a graph dataset, which plays a transformative role in achieving effective class balancing. Traditionally, the approaches treated the features as independent entities and aimed of producing more data without taking into consideration the hidden relationships between different features, but the proposed methodology converts tabular dataset to graph dataset which takes into consideration the structural and hidden relationships between features and produces more data with relationship (edged connection) to the features and other data samples. This graph-based framework enables the GCN layers within the WGAN model to fully exploit both local and global information embedded in the data. Specifically, GCNs utilize the neighborhood information

TABLE II
PERFORMANCE OF EACH ALGORITHM USED

E. No.	Dataset	R^2	Standard Deviation	Mean	Variance
1	1	-1.99E+01	1.80E+01	4.18E+01	3.24E+02
	2	-6.82E+00	1.55E+02	2.17E+02	2.41E+04
2	1	-1.71E+01	4.80E+04	8.79E+04	2.30E+09
	2	-2.74E+01	1.93E+02	3.06E+02	3.74E+04
3	1	4.31E-01	4.15E-02	1.73E-03	1.72E-03
	2	-1.31E+00	3.59E-02	1.29E-03	1.29E-03
4	1	7.93E-01	4.54E-02	2.06E-03	2.06E-03
	2	-1.92E-01	7.69E-02	5.95E-03	5.92E-03
5	1	9.99E-01	1.65E+01	1.32E+01	2.73E+02
	2	9.96E-01	5.00E-01	5.00E-01	2.50E-01
6	1	9.99E-01	5.00E-01	4.99E-01	0.00E+00
	2	9.99E-01	0.00E+00	5.00E-01	1.33E+05
7	1	9.99E-01	5.00E-01	5.00E-01	5.00E-01
	2	9.99E-01	5.00E-01	5.00E-01	5.00E-01
8	1	3.25E-01	3.55E-01	1.48E-01	1.26E-01
	2	4.58E-01	3.18E-01	1.14E-01	1.01E-01
9	1	-1.50E+02	0.00E+00	1.00E+00	0.00E+00
	2	-1.50E+02	0.00E+00	1.00E+00	0.00E+00
10	1	1.58E+00	5.00E-01	5.00E-01	5.00E-01
	2	2.78E+00	5.00E-01	5.00E-01	5.00E-01
11	1	9.98E-01	3.93E+00	2.18E+02	1.55E+01
	2	9.99E-01	1.65E+01	9.49E-02	2.73E+02
12	1	-9.47E-02	1.66E+01	1.36E+01	2.74E+02
	2	-1.03E-01	1.10E+00	1.16E+00	1.22E+00
13	1	-1.72E+02	1.67E+01	1.28E+01	2.78E+02
	2	-1.82E+02	2.10E+06	4.24E+05	4.42E+12
14	1	-2.09E+02	1.66E+01	1.28E+01	2.77E+02
	2	-5.29E+01	1.34E-01	1.18E-02	1.80E-02

around each node, allowing the model to draw on both direct and indirect relationships, thereby enhancing the generator's ability to create realistic minority class samples that reflect the dataset's true relational structure, essential for accurately balancing the classes without introducing noise or irrelevant patterns.

The graph dataset's ability to reveal hidden relationships further enables the discriminator to perform more rigorous sample validation. By assessing both individual nodes (transactions or entities) and their connections, the discriminator can better differentiate between real and synthetic data, reducing misclassification rates. This is particularly valuable for detecting subtle patterns in fraudulent transactions where fraudsters often create complex, hidden networks to evade detection. Consequently, the combination of GCN layers with a graph-based approach enables both the generator and discriminator to leverage a multi-dimensional, network-informed perspective, leading to balanced, realistic synthetic data that supports the model's class balancing objectives.

VI. FUTURE WORK

In future work, there are several promising directions to expand and enhance the effectiveness of our enhanced WGAN with GCN layers for fraud detection and class balancing. First, while this model achieves improved accuracy on historical transaction data, a critical next step would involve adapting it for real-time fraud detection. Such an adaptation would require optimizing the model's speed and efficiency to handle continuous data streams, an essential feature for real-world financial applications. Additionally, this methodology could be extended to incorporate multi-modal data types, including geolocation, device metadata, or behavioral patterns, which may offer a more comprehensive view of fraudulent activity and improve detection accuracy.

Another area for improvement lies within the GCN architecture itself. While our model uses GCN layers to capture graph-structured dependencies, optimizing the number and configuration of these layers, or integrating advanced GCN variants such as attention-based or hierarchical models, could yield even better performance by capturing complex relationships more effectively. Furthermore, hybrid approaches that integrate cost-sensitive learning with our WGAN could provide additional benefits. By prioritizing high-risk transactions within a cost-sensitive framework, this combined approach could further reduce misclassification errors, making it particularly valuable for financial institutions where undetected fraud leads to high penalties or operational costs.

Broadening the scope of our evaluation to include datasets from various domains would also help validate the generalizability of this approach. Tailoring the model for specific industries, each with distinct transaction structures and fraud patterns, could yield domain-specific optimizations. Addressing mode collapse remains essential, as ensuring diversity in synthetic data is key for effective class balancing. Future research could explore advanced techniques like diversity-promoting regularization or adaptive sampling to further alleviate mode collapse within the GAN framework.

Finally, improving model interpretability would be crucial

TABLE III
PERFORMANCE METRICS FOR DIFFERENT MODELS ON CREDIT CARD DATASET

Model	Accuracy	Precision	Recall	F1-score	MCC	MCS
GAN	0.9860	0.8000	0.1767	0.7600	0.6800	0.1632
GAN + graphSAGE	0.0058	0.0061	0.0959	0.0115	-0.9479	0.1812
GAN + GCN	0.8100	0.7845	0.7272	0.1675	0.5814	0.0000
WGAN	0.9988	0.9145	0.9272	0.9204	0.6855	0.2054
WGAN + graphSAGE	0.9971	0.9021	0.8800	0.7970	0.5408	0.3205
WGAN + GCN	0.9947	0.9544	0.9568	0.9595	0.7895	0.1876

TABLE IV
PERFORMANCE METRICS FOR DIFFERENT MODELS ON PAYSIM DATASET

Model	Accuracy	Precision	Recall	F1-score	MCC	MCS
GAN	0.9857	0.8020	0.1785	0.7610	0.6830	0.1655
GAN + graphSAGE	0.0061	0.0063	0.0963	0.0112	-0.9461	0.1830
GAN + GCN	0.8130	0.7823	0.7290	0.1690	0.6282	0.0020
WGAN	0.9985	0.9123	0.9285	0.9218	0.6841	0.2040
WGAN + graphSAGE	0.9968	0.9035	0.8780	0.7985	0.6082	0.3195
WGAN + GCN	0.9950	0.9562	0.9550	0.9605	0.7908	0.1860

as we aim for deployment in real-world financial systems. Developing methods to explain the model's decisions on both synthetic and real data would enhance user trust and aid in meeting regulatory requirements, thus supporting broader adoption. Together, these research avenues can help strengthen our enhanced WGAN model's versatility and reliability, making it a valuable tool for fraud detection and addressing class imbalance challenges across various sectors.

References

[1] Y. Chen, X. Yang, and H.-L. Dai, "Cost-sensitive continuous ensemble kernel learning for imbalanced data streams with concept drift," *Knowledge-Based Systems*, vol. 284, p. 111272, 2024.

[2] G. A. Pradipta, R. Wardoyo, A. Musdholifah, and I. N. H. Sanjaya, "Radius-smote: Addressing overlapping and noise issues in imbalanced datasets," *Data Mining and Knowledge Discovery*, vol. 37, no. 2, pp. 345–367, 2021.

[3] M. Alarmi and M. Ykhlef, "Hybrid sampling method for imbalanced credit card fraud detection," *Expert Systems with Applications*, vol. 192, p. 113368, 2024.

[4] A. S. Palli, J. Jaafar, M. A. Hashmani, H. M. Gomes, and A. R. Gilal, "Cluster-based hybrid sampling for imbalance data (cbhsid)," *Artificial Intelligence Review*, vol. 56, no. 3, pp. 425–450, 2023.

[5] Y. Bao and S. Yang, "Center point smote and inner and outer smote for imbalanced classification," *Journal of Statistical Computation and Simulation*, vol. 93, no. 4, pp. 612–628, 2022.

[6] Y.-S. Jeon and D.-J. Lim, "Particle stacking undersampling method for highly imbalanced big data," *Knowledge-Based Systems*, vol. 245, p. 108500, 2020.

[7] C.-F. Tsai and W.-C. Lin, "Feature selection and ensemble learning techniques for one-class classifiers," *Pattern Recognition*, vol. 120, p. 108215, 2021.

[8] A. Sharma, P. K. Singh, and R. Chandra, "Smotified-gan: A hybrid approach for class imbalance in pattern classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 7, pp. 3246–3258, 2022.

[9] W. Shafqat and Y.-C. Byun, "Cwgan-gp-pacgan: A hybrid gan architecture for imbalanced data in recommendation systems," *ACM Transactions on Intelligent Systems and Technology*, vol. 14, no. 5, p. 120, 2022.

[10] M. Xu, Y. Chen, Y. Wang, D. Wang, Z. Liu, and L. Zhang, "Bwgan-gp: An eeg data generation method for class imbalance problem in rsvp tasks," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 30, pp. 251–263, 2022.

[11] J. Hao, C. Wang, G. Yang, Z. Gao, J. Zhang, and H. Zhang, "Annealing genetic gan for imbalanced web data learning," *IEEE Transactions on Multimedia*, vol. 24, pp. 1164–1174, 2022.

[12] B. Liu and G. Tsoumakas, "Dealing with class imbalance in classifier chains via random undersampling," *Knowledge-Based Systems*, vol. 192, p. 105292, 2020.

[13] K. Shenoy, "Credit card transactions fraud detection dataset," https://www.kaggle.com/datasets/kartik2112/fraud-detection, 2020, accessed: Oct. 5, 2024.

[14] E. A. Lopez-Rojas, "Paysim: Synthetic financial datasets for fraud detection," https://www.kaggle.com/datasets/ealaxi/paysim1, 2017, accessed: Oct. 5, 2024.