National College of Ireland

Smart Synthetic Data as Solution to the Limitations of Conventional Anonymization Means in Big Data

MSc Research Project
Master of Science in Data Analytics

Bingwei Wang
Student ID: X21111596

School of Computing
National College of Ireland

Supervisor: Catherine Mulwa

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name:	Bingwei Wang
Student ID:	X21111596
Programme:	Master of Science in Data Analytics
Year:	2022
Module:	MSc Research Project
Supervisor:	Catherine Mulwa
Submission Due Date:	15/08/2022
Project Title:	Smart Synthetic Data as Solution to the Limitations of Conventional Anonymization Means in Big Data
Word Count:	11621
Page Count:	27

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	17th September 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Smart Synthetic Data as Solution to the Limitations of Conventional Anonymization Means in Big Data

Bingwei Wang
X21111596

Abstract

Ideally, the anonymized data method of protecting privacy and promoting data security ought to remove all personally identifiable information and simultaneously maintain the crucial information for the application of data without invading privacy. However, anonymized data neither offers data privacy nor does it retain the key useful information. As a tool, it is associated with several risks and limits, especially in Big Data applications. There should be a major trade-off between absolute privacy protection and actual data utility. The smart artificial data can be characterized by better or similar predictive power as real data, void of any privacy challenges present in the original data. In this study, conditional Generative Adversarial Network (cGAN) is used to generate anonymous data and verify whether the characteristics of the generated anonymous data are close to the real data. Results indicate that cGAN can generate artificial data that eliminates the risks of privacy and confidentiality violation in the use of big data while enabling shareability and hence maximization of big data. In the result, synthetic data generated by cGAN has not only the distribution is similar to the original data, but also in machine learning performance is close to real data. In the final results, the smart synthetic data generated by the method used in this paper were improved by 0.93%,0.39% and 1.6% respectively in the three machine learning algorithms, and the accuracy is sometimes improved by more than 5.0% after optimization in the isolated forest algorithm. The results show that the data synthesized by cGAN can replace anonymous data to protect user privacy.

1 Introduction

The aim of the present research is to foster data shareability while upholding privacy and confidentiality. The key contribution as a result of this project is a novel deep learning model that capitalizes on machine learning, artificial intelligence, and big data, to generate smart synthetic data. – data generated from original datasets, preserves all the key features while ensuring data privacy and confidentiality. In essence, the main contribution of this research is a framework that generates new data with all of the characteristics of the original data while preventing the identification of key information (from the real data) from the synthetic data or method. As a result, smart artificial data will have greater or similar predictive potential to actual data, with no issues regarding privacy as well as a utility that arise from the original data.

The current conventional anonymized data methods neither provide data privacy nor does it retain the key useful information. As a tool, it is associated with several risks and

limits, especially in Big Data applications; consequently, hindering wide shareability and utility. This AI-driven solution aims to generate smart synthetic data that retains the key qualities and elements of real-world sample data while eliminating the privacy, and security risks thus increasing big data utility and shareability.The method of synthetic intelligent data (ICT Solution) adopted in this paper is of commercial use value and will have very good commercial potential after continuous optimization.

1.1 Background

The contemporary world is driven and powered by data. As of 2020, each person generated 1.7 megabytes of data every second (BFM Quantum; 2019). And by 2025, the creation of data will exceed 180 zettabytes (Statista), as data interactions (creation, capturing, copying, and consumption) grew by 5000 percent over the 2010-2020 decade (United States Data Science Institute; 2022). Data security and privacy concerns abound despite major technological strides in data anonymization and security. Data shareability is of great importance to realize the optimal potential and obtain maximum benefits from voluminous amounts of data collected hourly. However, data privacy and security remain a volatile topic that often hinders shareability and hence all the potential benefits. Similarly, the value and importance of privacy and confidentiality can neither be overlooked nor under-emphasized. Therefore, the result is a dilemma between data privacy and data utility.

Anonymization is the most prevalent method applied to data to offer privacy. Ideally, anonymized data ought to have all the personally identifiable details stripped while maintaining the key usable information for utility. However, though it is theoretically sensible, it still raises major privacy concerns. Research shows that coupling basic data that is publicly available (for instance in social media) with anonymized or obscured data enables easy re-identification. It hence fails to strictly adhere to modern data protection laws such as the European General Data Protection Regulation (GDPR). In essence, anonymization is arguably a key threat to identity and privacy. Therefore, the need for a solution that leverages emerging technologies to fill this gap.

1.2 Research Questions, Objectives, and Contribution

Value: Data has become a ubiquitous element in the contemporary digital era. Markedly, the advent of big has radically changed all aspects of modern living. This cuts across all use cases ranging from improved production efficiency to providing customers more accurate services and enabling advancement in disease research among others. However, the use of big data is plagued with notable risks of infringing user privacy and confidentiality. Therefore, it is critical to develop and use approaches that would not only enable optimal utilization of big data but also protect users' privacy and confidentiality against violation.

RQ: "By generating data through cGAN model, can these smart synthesized data solve the limitations of traditional anonymous data, and have the same performance as real data to achieve the purpose of protecting user privacy?"

Sub-RQ1: Is the smart synthetic data close to the real data in terms of data characteristics, such as data distribution?

Sub-RQ2: Can synthetic data be used as real data? How well do they perform when they are used in machine learning?

Objectives: To address the research question, project objectives are clearly listed in Table 1.

Table 1: Project Objectives

Num.	Objective	Method
1.	To investigate and compare existing methods for synthesizing data	
2.	Select the appropriate data set as the original dataset	
3.	Design specification and methodology	
4.	Design and implementation of Smart Synthetic Data 4.1 Data Clean 4.2 Feature extraction 4.3 Data standardization 4.4 Design generator 4.5 Design discriminator 4.6 Design loss function, optimizer, activation function 4.7 Integrate cGAN for training 4.8 Generate synthetic data 4.9 Design the output of the generator and discriminator loss functions 4.10 Design the distribution of synthetic data output 4.11 A common machine learning model is designed to import synthetic data and generated data into the model for prediction	CGAN ONE-HOT Data Normalization Neural network PCA Isolation Forest Random forests KNN Gradient Descent Backpropagation
5.	Evaluation and Results 5.1 Loss function evaluation of generators and discriminators 5.2 Evaluation of data distribution characteristics for synthetic data 5.3 Evaluate the performance of synthetic and generated data in machine learning	Visual analysis PCA Isolation Forest Random forests KNN
6.	A comparison of developed models vs. Existing models.	

Contribution: Whether it be in the established or emerging markets, not a single business is totally immune to fraud. Research suggests that frauds of all kinds might cost businesses between 1% and 1.75 percent of their monthly sales, or more than $200 billion annually. One of the most common types of fraud, credit card transaction fraud affects over 127 million people and resulting in around $8 billion in attempted fraudulent charges on credit and debit cards used by Americans each year.

Credit card companies, researchers and industry experts must comprehend the characteristics of a fraudulent transaction in order to develop prediction algorithms that can spot potentially risky conduct and stop fraud. However, such transaction datasets are difficult to come across for researchers, learners, and experts interested in learning and developing solutions to curb credit card fraud. This is primarily attributable to the privacy and confidentiality risks associated with transaction data, which in turn hinders shareability and ultimately, maximum use of big data. Hence the importance of artificial data that has similar features of the real data, without the risk of violating privacy and confidentiality.

Therefore, the key contribution of the current work is to use a cGAN based model to produce synthetic fake transaction datasets that bare similar characteristics as the original data, but eliminates the threats of breach of users' privacy and confidentiality. The artificial data can be freely and widely shared, hence easily accessible to all interested parties including researchers, scholars and industry experts. Notably, the cGAN model can be used in other fields and use cases other than credit card fraud detection.

The rest of the project report is structured as follows. Chapter 2 discusses related works focused on synthetic data. The Chapter 3 focuses on research methodology. The Chapter 4 contains design and implement. Chapter 5 is experimental results and evaluation and Chapter 6 is a conclusion to culminate the research.

2 Related Work from 2014 to 2022

In this section, a literature on major synthetic data between 2014 and 2022 is summarized. It mainly includes the current research on generative adversarial network (GAN), comparison with other data generation models, and the current research on conditional generative adversarial network(cGAN).

Data is a critical element of modern open society. The community is adequately served when data is obtained through proper methods, widely shared, and critically analyzed from various perspectives. The outcomes are scrutinized to aid in informed decision-making. Multiple laws have been established to enforce the right to privacy and Confidentiality, and stiff punishments apply in case of violations. The right to privacy refers to a respondent's right to keep their information private and not to disclose it unless they want to (Raghunathan; 2021, pp.5.2). Confidentiality is defined as a pledge a data collector issues to a participant that the collected details will be kept confidential (Raghunathan; 2021, pp.5.2). Despite the efforts, upholding privacy. Confidentiality of data remains a long-term concern. Synthetic data follows the basic principle of synthetization. According to Raghunathan (2021, pp.5.3), synthetization is "a chemical process that, by human agency, emulates certain properties of a naturally occurring material". Such technology enables wide use of key elements or products while retaining their natural properties.

2.1 Critique of Existing Generative Adversarial Network Approach

Generative adversarial networks (GANs) is a deep learning model that comprises two distinct neural networks—a discriminator and a generator—that is concurrently competitively trained, as in a framework for zero-sum games (Vega-Márquez et al.; 2019). As shown in Figure 1., the generative network (G) learns how to associate components of a latent space (noise) with a certain data distribution by generating new data that closely matches the original data. The discriminator (D) differentiates between elements of the original distribution and those created by the generative network by calculating the likelihood of belonging to one group. In summary, the discriminator network can classify the examples fed to it, acting as a binomial classifier to identify cases as true or false. The generator is an inverted convolutional network since it samples a random noise vector into an instance rather than downsampling an example as a conventional convolutional classifier does. The second creates fresh data, while the first discards it

through down-sampling methods like max pooling.

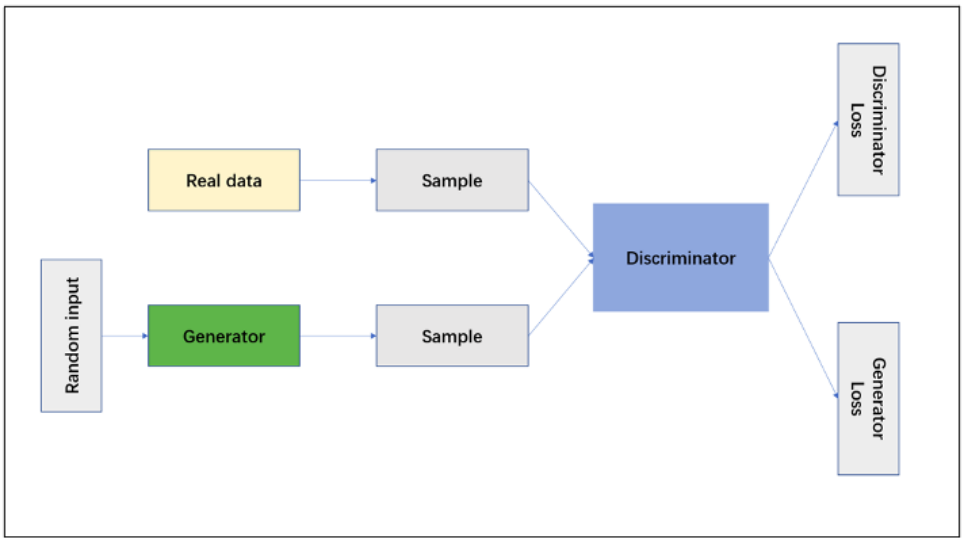

Figure 1: A simple architecture of a GAN network.

GAN provides a range of algorithms for data generation. According to Chen et al. (2021), GAN is a kind of generative model that "learn probability distributions of how high-dimensional data are likely to be distributed" through a double neural network framework that consists of "a generator and a discriminator that compete in a minimax game to fool each other" (pp. 493). These approaches are the most commonly used to generate high-quality data (Zheng et al.; 2019).

Markedly, the density of GAN-generated distribution focuses on the training data, especially when trained using semantically rich datasets (Liu et al.; 2019, pp. 1). The Privacy-preserving GAN, in particular, has garnered much interest as it couples differential privacy that has noise well-designed, with training gradients aimed at randomizing the distribution of the original data. The result is a synthetic data generation model that is privacy leakage proof. However, Liu et al.'s proposed model does not cater to the shareability and utility of data and entirely focuses on mitigating information leakage.

Deep learning models have rapidly advanced over the recent past. The Conditional Generative Adversarial Network (CGAN) is a variant of GAN that is characterized by networks that consider target class hence producing remarkable results for data sets that have this feature – target class - hence, new data proximally aligns to the data according to which each instance belongs to (Ramponi et al.; 2018, pp. 2). A typical GAN does not consider any kind of data condition. Generally, artificial data ought to have a feature that not only distinguishes it but is also applicable in ensuring the synthetic data mirrors the original ones as closely as possible. A study by Vega-Márquez et al. (2019) set out create synthetic data using a Conditional Generative Adversarial Network (CGAN). Although the results indicated high classification accuracy and good performance in artificial data synthetization, the study does not address the aspect of parameter adjustment to obtain maximally reliable results.

The Wasserstein GAN is a robust GAN variant that employs a detailed training method. A study by Zheng et al. (2019) employed WGAN on two variations of power systems. However, the researchers concluded that, though promising, the WGAN yielded synthetic phasor measurement unit (PMU) time series datasets that were inadequately

unrealistic. Another variant of WGAN proposed by Gulrajani et al. (2017) showed stability and firm modelling performance.

The imaging field has made significant strides through synthetic data, particularly restoring real-world low-resolution images characterized by complex and unknown degradations. A study by Wang et al. (2021) proved that Real-ESRGAN, trained on purely synthetic data, can tremendously enhance details and eliminate undesired artefacts for typical real-world degraded images (pp. 1905). This was achieved through an implicit approach based on data distribution learning using Generative Adversarial Networks (GAN).

2.2 Comparative Analysis

Notably, differential privacy solutions ought to ideally guarantee formal privacy. The GAN models and extensions have shown impressive performance in modelling the underlying data distribution to yield quality 'artificial' data that resemble real data samples (Mogren; 2016). Thus, Differential Privacy Generative Adversarial Network (DPGAN) models are advancing. GANObfusicator is a generative adversarial network that aims to mitigate information leakage by coupling differential privacy with introducing carefully developed noise into the gradients during training (Xu, Ren, Zhang, Zhang, Qin and Ren; 2019). Accordingly, this solution does not experience gradient vanishing or mode collapse over the training process, meaning it can maintain stability and scalability during training.

The Renyi-differentially private-GAN (RDP-GAN) is a differential privacy-based GAN proposed by Ma et al. (2020), inspired by Xu, Ren, Zhang, Zhang, Qin and Ren (2019) study. Like the GANObfusicator, RDP-GAN is based on cautiously injecting random noises into the loss function value to attain differential privacy to prevent information leakage and create a private GAN (Ma et al.; 2020, pp. 2). the authors also employed an adaptive noise tuning step to counter the negative effects of adding noise. The RDP-GAN model yielded improved privacy levels and high-quality datasets compared to the DP-GAN approach concerning noise perturbation on training gradients (Ma et al.; 2020). However, the GANObfusicator and Renyi-differentially private-GAN suffer from a relatively high privacy budget. Therefore, there is an overarching need to focus on cutting the privacy budget while maximizing utility. This can be achieved through, for instance, experimenting with different pruning methods.

The advantages of GAN over RBMs, DBMs, DBMs and VAE: RBMs, DBNs and DBMS all have the difficulties of intractable partition functions or intractable posterior distributions, which thus use the approximation methods to learn the models. Variational Autoencoders (VAE), a directed model, can be trained with gradient-based optimization methods. But VAEs are trained by maximizing the variational lower bound, which may lead to the blurry problem of generated images.

Reasons for GAN's attention: First, in theory, the model approximates the real data distribution, automatically forming traditional models such as the Bayes model and variational encoder. However, in the past ten years, these technologies could not get close to the real and high dimension of data distribution, and image generation was still a difficult task until the emergence of GAN. The second impact is to give researchers an updated way of thinking when studying AI. It has educated our group of researchers on whether to consider the introduction of adversarial ideas in the design of a series of deep learning algorithms, how to introduce them, how to introduce them appropriately, and

how to improve the performance of traditional AI algorithms on tasks. The third is the possibility of semi-supervised learning based on small data. It is well known that a large amount of data is needed to train neural networks. GAN provides a new idea – training GAN to simulate the real data distribution. In some cases, when the real data is not enough or difficult to obtain, researchers can try to train GAN to simulate the real data distribution. If the simulation is good enough, GAN can generate more training data, which helps solve the problem of small data volume in deep learning.

2.3 Utility Assessment Comparison

AI-generated synthetic data has gained much interest as a seminal approach to not only high utility, for instance, through safely sharing but also bolsters privacy levels compared to traditional methods such as data anonymization (Tiwald et al.; 2021, pp. 1). Machine learning algorithms can learn from training data to decipher patterns, rules, and associations (Dikici et al.; 2020). However, these training data are obtained from a biased society and history.

Ideally, synthetic data ought to align maximally with real-world data. The utility is among the top measures of determining and assuring data scientists, researchers, and other data-focused experts the proper confidence to use artificially generated data instead of original data. Thus, the importance of robust utility assessment methods. Notable utility evaluation models focused on every dataset include structural similarity, general utility metrics, and basic and stability assessment (El Emam; 2020, pp. 59). A prior study that focused on both general and specific utility measures for synthetic data by Snoke et al. (2018) offered remarkable but very different utility measures. However, neither of the approaches was satisfactory.

2.4 cGAN

The generative adversarial network model remains a seminal breakthrough with myriad applications and use cases. Nonetheless, the original, unaltered version of the GAN model arguably falls short in implementing certain tasks. Therefore, the advancement of numerous versions of GANs addresses key concerns for particular use cases. The conditional generative adversarial network is a GAN version that introduces additional inputs. According to Gong et al. (2019), cGAN is the singular solution that can generate controllable and certain images in text-to-image synthesis tasks. The researchers leverage the feature of cGAN to effectively control content through the additional conditions and inputs introduced in both the generator and the discriminator. Remarkably, the yielded output semantically aligns with the input text.

Conditional generative models have seen impressive improvements over the past few years. Gong et al. (2019) defined conditional generative adversarial networks as a GAN variant that uses additional conditions or information that have a wide application in the synthesis of class-conditioned images as well as text.

Aerodynamics is a complex field that demands the highest efficiency and optimization. It heavily relies on simulations to model real-world scenarios in a virtual space to ensure safety and save on costs. Nonetheless, the simulations remain a costly expenditure and computationally expensive. As such, it stands to benefit from the implementation of advanced machine learning models. Achour et al. (2020) researched to leverage the affordances of novel approaches such as GAN in optimizing Computational Fluid Dynamics

(CFD) tools and processes. Particularly, the researchers employ an elevated version of the Conditional GAN to optimize the airfoil's shape. The researchers note that Conditional GAN has major benefits, such as accepting labelled data inputs during training and synthesizing realistic samples for each data set class. Therefore, it was possible to define and influence the shape generation process, allowing remarkable shape maximization.

Generative Adversarial Networks (GANs) and their variants have exhibited great performance in generating synthetic data. They present an elegant and efficient solution that has drawn the attention of multiple researchers and experts (). Notably, the original variant of GAN is hampered by a lack of means to input class labels during the generation process. Vega-Márquez et al. (2019) utilize the conditional GAN to alleviate this classification problem as it considers the class label when generating new data. Conditional GAN also shows great imbalance improvements compared to raw GAN. Douzas and Bacao (2018) study showed that cGAN has great potential in approximating the real data distribution and the generation of minority class data for various imbalanced datasets. Xu, Skoularidou, Cuesta-Infante and Veeramachaneni (2019) conducted a similar study using a mildly different variant of conditional GAN – the Conditional Tabualar-GAN to help establish a balance between continuous and discontinuous columns in tabular collections. They created a benchmark using multiple Bayesian network baselines, seven simulated datasets, eight actual datasets, and five genuine datasets. On most real datasets, cGAN performed better than Bayesian approaches, whereas other deep learning approaches did not.

2.5 Conclusion

Finding the optimal balance between the risk of re-identification, privacy compromise, and data utility is increasingly critical. GAN-based solutions have proven to offer the most viable models for addressing the above conundrum (Goodfellow et al.; 2014; Radford et al.; 2015). Valuable data is characterized by personal details, raising privacy concerns around its usage and sharing (Lu et al.; 2019). Privacy is of paramount importance and hence guarded and governed by strict data privacy regulations, including General Data Protection Regulation (Intersoft Consulting; n.d.) and HIPAA (HHS.gov; n.d.). Based on the above-reviewed slew of studies and the proposed seminal models, the Conditional GAN-based approach for synthetic data generation is a feasible, affordable, and practical privacy-preserving solution. Generally, these works exhibit various strengths but are also hampered by multiple weaknesses, particularly regarding classification and acceptable privacy costs. Therefore, this paper is inspired by these works that utilize GAN and cGAN approaches and introduces additional steps to produce AI-generated smart synthetic data.

3 Design Methodology and Specification

The research method of this paper is shown in Figure 2. The purpose of this paper is to generate intelligent synthetic data so that it can have a distribution similar to the real data, and when these data are used as big data data sets, they will have a performance close to the real data in machine learning, and finally solve the problem of protecting user privacy. cGAN (Conditional generative adversarial network) was chosen for the synthesis of synthetic intelligent data, choose data set contained user information as the original data, through a series of data pre-processing, imported into cGAN model, generate synthetic data, finally through the model of the loss function to evaluate the

performance of the model, the distribution of synthetic data to validate the similarity of data, And its performance in machine learning can verify its utility in the application of big data.

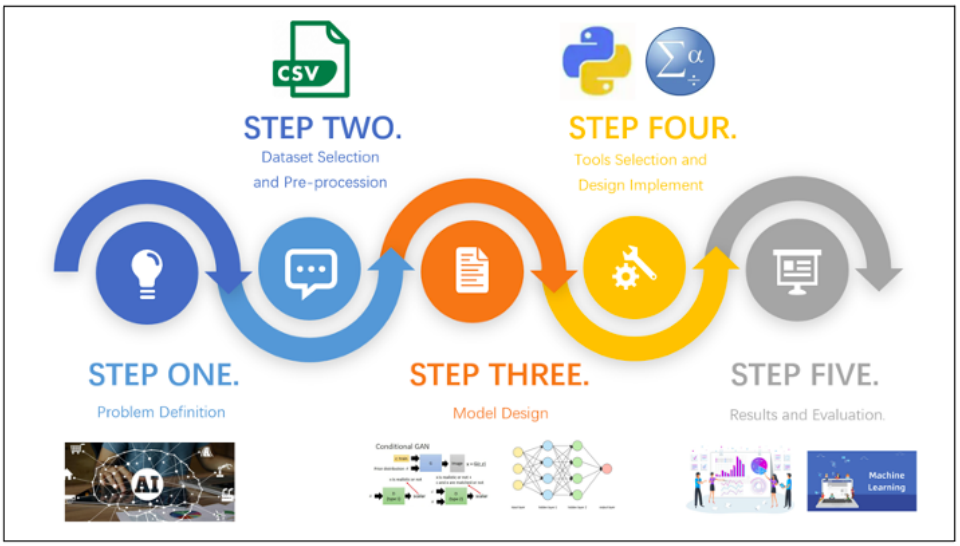

Figure 2: Methodology flow chat of Smart Synthetic Data.

3.1 Selection and Analysis of Data Sets

The selection of the original dataset must be open source and have data in a similar format to user information. At the same time, the dataset must be large enough to meet the purpose of training and final testing and ensure the stability and reliability of the results.

For this research, the credit card transactions fraud detection dataset that is readily available on Kaggle (Credit Card Fraud Analysis) is used, which is contained more than 550,000 records. The data contains both a training dataset and a testing dataset. The objective is to utilize the testing dataset to assess the model's performance after it has been trained on the training data set. The training dataset comprises 23 columns that include information about the credit card users' names, genders, localities, and birthdays as well as the merchant, spending category, transaction amount, and time of the credit card transaction.

3.2 Tools in the Study of Smart Synthetic Data

The main programming language used in this study is Python, and the software packages used include Keras, Pandas, Numpy, Matplot, Seaborn, Plotly, SkLearn, etc. Through these software packages, neural networks are built, machine learning models are built, and data visualization is performed. At the same time, Excel and Spss are used as data preprocessing and result analysis.

3.3 Reasons for the Methodology Used in the Study

Since 2014, generated against network (GAN) was put forward, it has always been a hot topic, many studies have shown that GAN in image processing and NLP(natural language

processing) has obtained the good result, synthesize GAN anonymous user information, therefore, cGAN should be a better method because of using the supervised learning, The resulting synthetic data will be much closer to the real data.

Compared with image recognition, text information cannot be evaluated by Inception Score. Therefore, Principal component analysis (PCA) is used in this paper to evaluate the gap between generated data and real data. PCA is essentially a kind of ranking analysis. The data after dimensionality reduction are displayed in the two-dimensional or three-dimensional plane by scatter diagram. The closer the distance between two sample points is, the more consistent the two samples are. PCA diagram is widely used in bioinformatics, and the algorithm is applicable to a wide range of data analysis, such as genome and transcriptome. At the same time, in order to verify the application of synthetic data in big data, the performance of synthetic data in three common machine learning (isolated forest, random forest, nearest neighbor algorithm) to verify whether synthetic data can be treated as real data and used in big data processing.

4 Design and Implement of Smart Synthetic Data

4.1 Introduction

The approach followed in this research is represented in Figure 3. The whole realization process of this study: i. Data clean for original dataset; ii. Feature Extraction; iii. Digital Coding; iv. Data Normalization; v. Training of Generator and Discriminator; vi. Integration and optimization of cGAN model; vii. Visual evaluation of the loss function; viii. Visual evaluation of data distribution; ix. Performance evaluation of synthetic data in Machine learning.

In the following sections, the core modules of each step are explained and implemented.

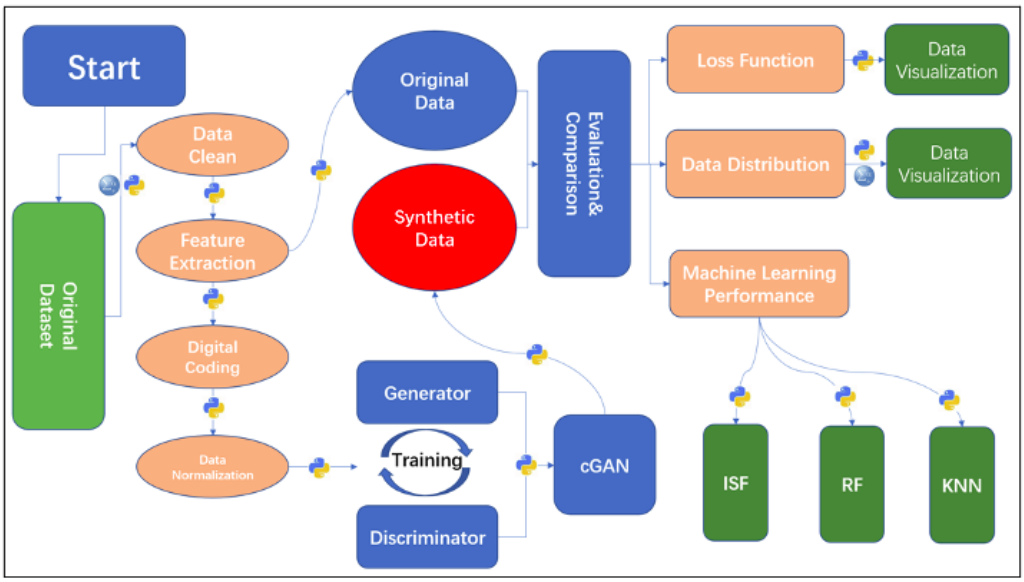

Figure 3: Flow chat of Data Preprocessing, Implement and Evaluation.

4.2 cGAN, the Core Model for Generating Intelligent Synthetic Data

The generative adversarial networks (GANs) have gained much popularity and research interest owing to their remarkable performance and promising results in multiple big data and computer vision jobs, including image translation and image generation (Goodfellow et al.; 2014). In essence, GANs (Generative Adversarial Networks) enable the generation of the novel picture, video, or audio data from a random input. Typically, the random input is drawn from a normal distribution and then subjected to a few changes to make it credible (image, video, audio, etc.). A basic DCGAN, on the other hand, does not allow us to modify the look (for instance, class) of the samples we generate. A basic DCGAN, for example, would not allow us to pick the class of digits generated by a GAN that creates MNIST handwritten digits. We must condition the GAN output on a semantic input, such as an image's class, to control what we produce.

Primarily, GANs are designed to produce distributions that have a high degree of resemblance to the distributions exhibited in real data. This is attained through a minimax game between the generator G and the discriminator D of GANs. The discriminator aims to learn and discern real samples from fake ones, while the generator learns and improves on generating fake samples to fool the discriminator. The game continues until a point of Nash equilibrium is established between both modules (Qu et al.; 2019; Nash Jr; 1950). Therefore, GANs can emulate real data distribution and generate believable fake data.

The objective function of GAN is defined as $V(G, D)$. In the process of the game, G hopes to reduce the value of V so that the distribution generated by itself cannot be identified, while D hopes to increase the value of V so that it can efficiently distinguish the true and false categories of data. The whole process is shown in Figure 4. Then, the expression of $V(G, D)$ is

$$\min_G \max_D V(D, G) = E_{x \sim p_{data(x)}}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$

In the formula above, E represents the mathematical expectation of real data X and noise data z.

G network is a generator, which can be a fully connected neural network, convolutional neural network, and so on. Through the noise distribution $P(Z)$, which is generally Gaussian distribution, a distribution $P_g(x)$ of generated data can be obtained. $P_g(x)$ is expected to be very close to $P_{data}(x)$ to fit and approximate the true distribution.

D network is a discriminant function, which needs to solve the traditional dichotomous problem. Its responsibility is to effectively distinguish the real distribution from the generated distribution, that is, to measure the gap between $P_g(x)$ and $P_{data}(x)$, and to train it through repeated iterations.

GAN is ultimately constrained by the absence of dimensionality reduction features (Achour et al.; 2020). As noted in the preceding section, GANs provide a robust approach for training generative models. In this research, conditional generative adversarial networks have been intensively studied. The conditional generative adversarial network (cGAN) is developed by inputting the data we want to condition on the generator and the discriminator (Mirza and Osindero; 2014). Therefore, the cGAN is characterized by extra variables or labels as inputs that have the potential to deterministically control the output of the generator. In contrast to the GAN model, the generator also requires additional input for category information y, which is connected to and inputted into the generator together with label information y.

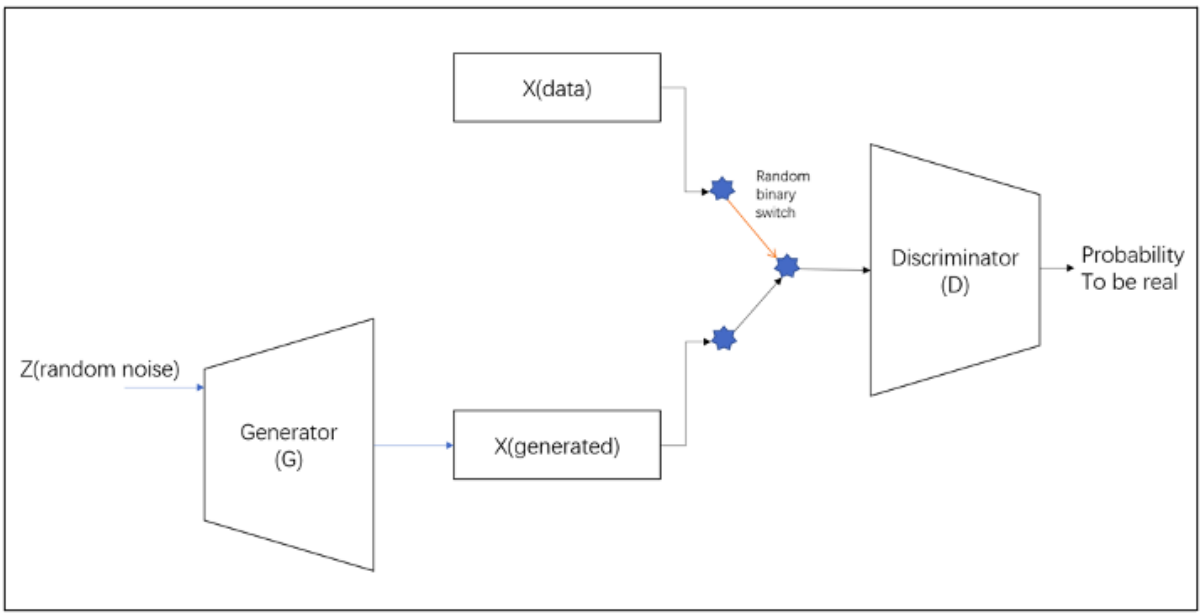

Figure 4: Generative Adversarial Network architecture.

The prior input and input noise in the generator, $P_z(z)$ and y, are coupled in a joint hidden depiction. The adversarial training model allows for flexibility in how the hidden details are produced. Within the discriminator, x and y form the inputs and a discriminative function.

The two-player min-max game's primary function can be represented as:

$$\min_G \max_D V(D,G) = E_{x\sim p_{data(x)}}[\log D(x \mid y)] + E_{z\sim p_z(z)}[\log(1 - D(G(z \mid y)))]$$

The central idea of cGAN is to control GAN's generated data rather than generate data randomly. Specifically, Conditional GAN adds additional information to the input of the generator and discriminator, and the data generated by the generator can pass the discriminator only if it is sufficiently true and consistent with the condition. In fact, in an unconstrained generative model, you have no control over the pattern of data generation. However, it is possible to guide the data generation process by constraining the model with additional information. As shown in Figure 5, cGAN adds the green Y part, which is the difference between cGAN and GAN. The constraint can be a class label, a piece of data patched or even data from a different mode. CGAN transforms unsupervised learning into supervised learning so that the network can get better training under our control.

4.3 Data Pre-processing

4.3.1 Data Clean

The original data set contains many missing values, invalid values, and data inconsistent with the data type. After getting the dataset, we first need to clean these "dirty data" and use the Pandas module in Python to remove invalid values and ensure data consistency. After that, the cleaned data can be correctly used by functions in the following program.

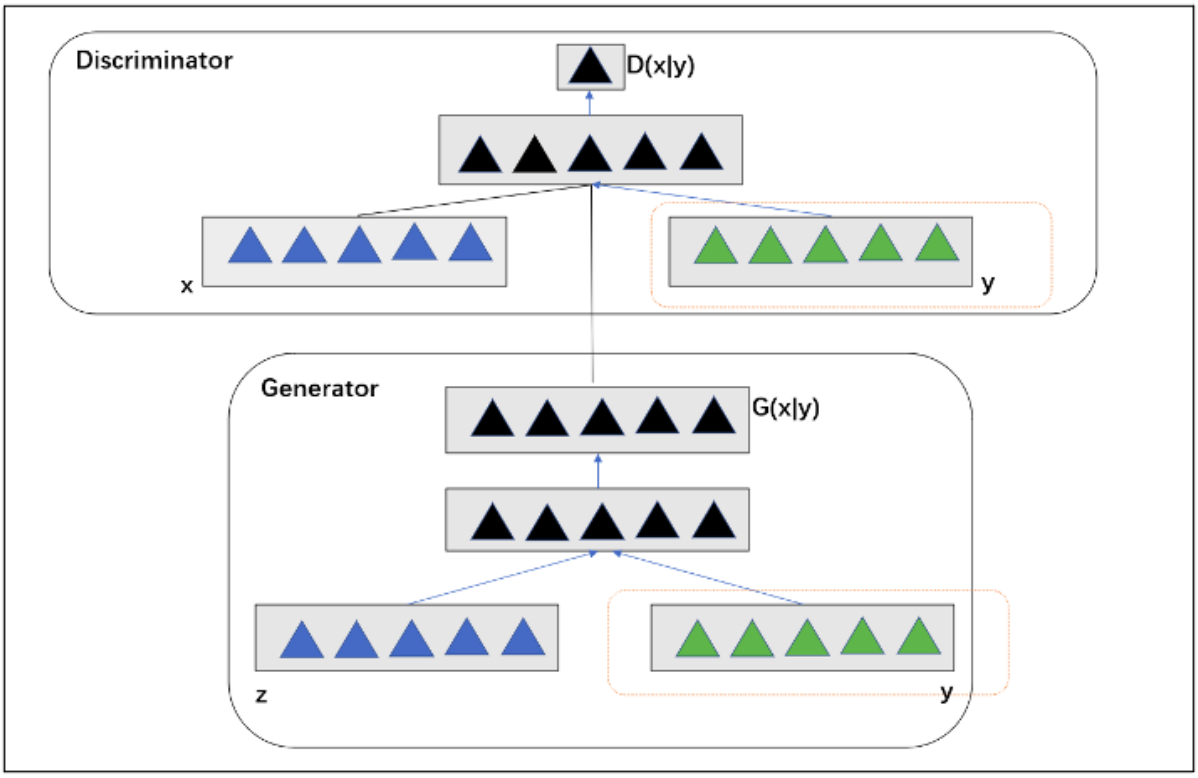

Figure 5: Conditional GAN Architecture.

4.3.2 Encoding of Strings

One-hot encoding is a process by which categorical data is converted into numerical form. This is done by creating a new column for each category and assigning a binary value (1 or 0) to indicate whether or not the observation belongs to that category. One-hot encoding is often used when working with machine learning algorithms since many of these algorithms require numerical data to function properly. There are a few things to keep in mind when using one-hot encoding. First, it can create many new columns (one for each category), making the data set more difficult to work with. Second, it can introduce a bias if some categories are over-represented in the data set. For example, if the data set contains more observations of males than females, then the resulting encoded data set will be biased in favour of males. Finally, one-hot encoding can create problems for some machine learning algorithms if the categories are not mutually exclusive (e.g. if there are multiple labels for a single observation). The string characteristics in the input must be transformed into numerical features since the deep learning model cannot recognize string information. This study processes the data using the one-hot approach since it is often utilized in numerical coding in this sector. Encoding with one-hot: The technique, often referred to as one-bit valid encoding, primarily uses N-bit state registers to encode N states, each of which has a unique register bit and only one of which is ever valid. Categorical variables are represented as binary vectors using one-hot encoding. To begin, categorical values must be converted to integer values. Each integer value is then represented as a binary vector, except the integer's index, which is denoted with a 1 and has zero value. If the students' ages [elementary, junior high, and high school] are encoded, then elementary school $> [1, 0, 0]$, junior high $> [0, 1, 0]$, and high school $> [0, 0, 1]$. After one-hot encoding, the data's strings may be transformed into numerical

values.

4.3.3 Normalization of Data

Normalization is scaling input data to fall within a specific range, like 0 to 1. The specific range you choose depends on the nature of your data and the model you are training. For example, if you were training a neural network to classify images, you would typically normalize the data so that all the pixel values fall between 0 and 1. Maximum and minimum normalization is a common type of normalization, which adjusts the data so that the maximum value is 1 and the minimum value is 0. This type of normalization is often used when the data is already well-scaled, and there is no need to adjust the mean. Normalization is often used in machine learning to improve the convergence of gradient-based optimization algorithms. This is because if the data is not properly scaled, the gradients given to the input layer during backpropagation can grow to very large or very small values, which can cause the learning rate to be too high or too low.

Normalization can also be used to prevent overfitting. If the data is not properly normalized, the model may be able to memorize the training data too well and not generalize well to new data. The gradients given to the input layer during backpropagation will grow in size if the input layer x is big. If the gradient is substantial, the learning rate must be extremely low to prevent overshooting the ideal. In this situation, choosing the learning rate must consider the value of the input layer, which may be done simply by immediately normalizing the data. Mean normalization, as well as maximum and minimum normalization, are different types of normalizing. The maximum and minimal normalization are used in this paper. The equation reads as follows:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

Among them, x' is the normalized data, x is the raw data, $X_{\min}$ is the feature's minimum value, and $X_{\max}$ is the feature's highest value. The features are equally scaled to $[0, 1]$ after this procedure.

In the GAN model, due to the training method, the activation function of the generator chooses Tanh, whose value range is $[-1, 1]$, resulting in the final generated data in $[-1, 1]$. Therefore, to keep the original data consistent with the generated data, the data need to be transformed into $[-1, 1]$ in this model.

4.4 Generator and Discriminator

A generative adversarial network consists of two main parts: I. Generator: a machine that generates data (mostly images) to "fool" the discriminator. II. Discriminator: Determines whether an data is real or machine-generated to find "fake data" made by the generator.

In GAN's system, the input is the original data x, and the random noise signal Z (such as Gaussian or uniform distribution), and the output is a probability value or a scalar value. The example of counterfeiters and police can explain generators and discriminators. Counterfeit money is made based on the appearance of real money, and then the police can determine the authenticity of the money. In the beginning, the police can immediately identify counterfeit money because the counterfeit money maker's technical ability is not good. After the failure, counterfeit money makers can identify a better counterfeit method to improve their technology and create a more realistic money. At the same

time, the ability of the police to identify counterfeit constantly improves as well; this is a process of confrontation. Finally, the counterfeiters produced money are so real that the police cannot identify the difference between real and counterfeit money. Therefore, the probability of the policeman guessing correctly becomes 0.5. The final counterfeiter is already a good description of the money, and he has mastered the various characteristics of the money. He is the generator we want to get.

This study utilizes cGAN as it offers more advantages compared to GAN. Unlike GAN, which does not offer control over the modes of generated data, Conditional GAN is characterized by additional labels, particularly the label y parameter, that acts as an extension to the latent space z utilized to generate and discriminate data better (as illustrated in Figure 6 below). The extra labels make cGAN more superior in terms of performance, yielding a more sensitive and biased model. Therefore, the data generated through cGAN is better. Additionally, cGAN has faster convergence and allows control over the generator's output during test time.

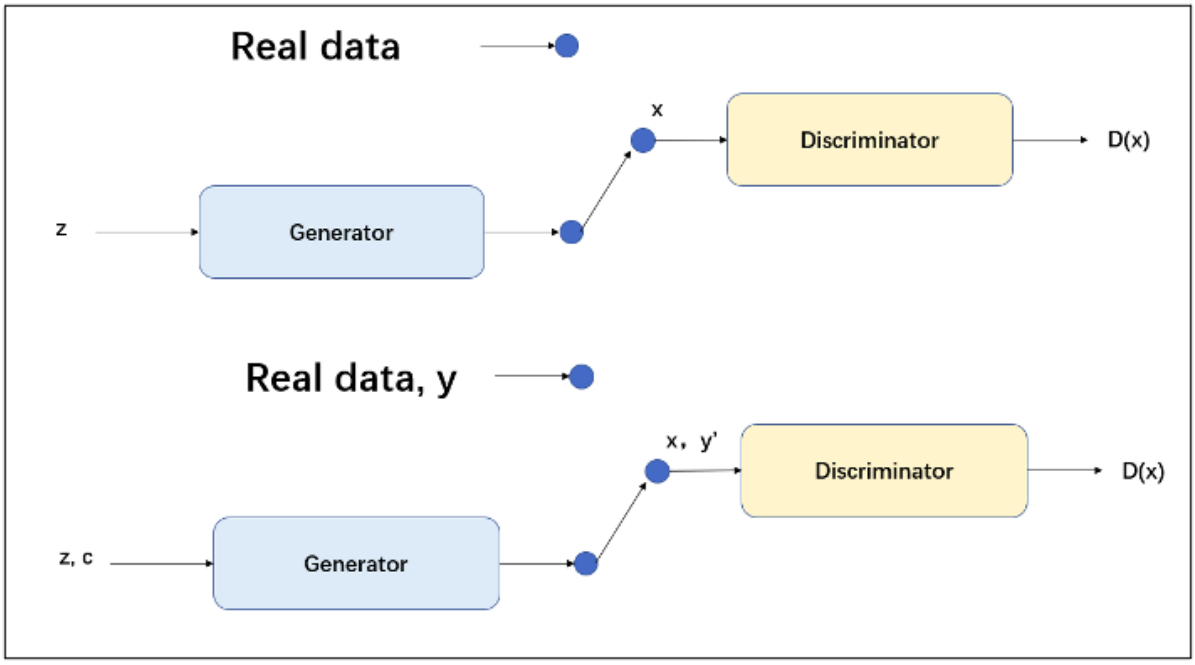

Figure 6: Additional label y in cGAN vs regular GAN without additional parameters.

Compared with GAN, cGAN is changed from unsupervised learning to supervised learning. Before pushing the data into the model, a label is added to the data, that is, condition y, which increases the controllability of the generated data.

In the paper introduction of Goodfellow et al. (2014), the author of GAN, it is theoretically proved the convergence of the GAN algorithm and the distribution of generated data with the same distribution as the real data when the model converges. According to the author's formula, the optimization objective function of cGAN can be derived as follows:

$$\min_G \max_D V(D,G)$$
$$V(D,G) = E_{x\sim p_{data(x)}}[\log D(x \mid y)] + E_{z\sim p_z(z)}[\log(1 - D(G(z \mid y)))]$$

D is trying to increase V, and G is trying to decrease V, so they're fighting each other.

The discriminator D and generator G pair are shown below:

$$\log(1 - D(G(z \mid y))]$$

The optimization goal is the opposite, which is reflected in the formula:

$$\min_G \max_D V(G, D)$$

As can be seen from the above formula, the calculation of loss function is generated in D(discriminator), because the output of D is generally TRUE or FAKE, so the binary cross-entropy function is adopted on the whole. The whole function can be split into two parts:

1. Discriminator:

$$\max_D L(D, G) = E_{x \sim p_{data(x)}}[\log D(x \mid y)] + E_{z \sim p_z(z)}[\log(1 - D(G(z \mid y)))]$$

2. The generator:

$$\min_G L(D, G) = E_{x \sim p_{data(x)}}[\log D(x \mid y)] + E_{z \sim p_z(z)}[\log(1 - D(G(z \mid y)))]$$

4.5 Parameter Selection and Optimization

When we use gradient descent in supervised learning, the learning rate is an important indicator because it determines the speed of the learning process (which can also be seen as the size of the stride). If the learning rate is too large, it is likely to exceed the optimal value; otherwise, if the learning rate is too low, the optimization efficiency may be very low, resulting in too long an operation time. Therefore, the learning rate is very important for the algorithm's performance. The optimizer Keras. Optimizers.Adam() is one solution to this problem. The general idea is to set the learning rate to a larger value at the beginning and then reduce the learning rate dynamically according to the time increase to achieve both efficiency and effect. So, we chose it as our optimizer.

Since the output of the discriminator in this project is true or false, it is a dichotomous problem, so sigmoid is chosen as the activation function, and binary_Crossentropy (BCE) is chosen as the loss function. The discriminator input is data with 24 features, so LeakyReLU is selected as the activation function. LeakyReLU can make the value of x in the training GAN have a small gradient when it is less than 0 instead of being directly judged as 0 like LeLU. This approach optimizes training.

For this research, we define a discriminator that accepts inputs, including the feature matrix and class vector, from which it predicts the probability of whether a certain transaction is fraudulent or not. Conversely, we define a generator that takes inputs of latent space and class labels and outputs a feature matrix 'X'. The defined discriminator constitutes two hidden layers, each with 200 nodes, while the generator has three hidden layers and $(150 \times 200 \times 150)$ nodes.

Lastly, we combine the generator and the discriminator to form a cGAN. At first, maintain a non-trainable discriminator to ensure that it does not update its weights while we train the cGAN.

Modifying it to 20 does not work well. After 6 rounds, the Nash balance disappears, and the d_loss decreases. This study's structure of the C-GAN model is ultimately established as generator 150-200-150-24 after several tests. The discriminator nodes are

200-200-1, respectively. The optimizer adopts Adam, and the learning rate is 0.0002, where the batch size is set to 512 and the noise dimension to 99.

Principle of optimization: With the loss function, the generation network and discriminant network can adjust the parameters by using the Backpropagation (BP) algorithm and optimization method (such as gradient descent method) based on their respective loss functions. The performance of generative network and discriminant network is constantly improved (the mature state of generative network and discriminant network is to learn a reasonable mapping function).

4.6 Network Training

Different from a single network of ordinary CNN classification or regression, GAN needs to train two networks: the parameters of the discriminator D are fixed when the generator G is trained, and the parameters of the discriminator D are fixed when the discriminator D is trained. The key to network training is to alternate the update generator and discriminator and correctly update the gradient. Finally, the two looped to get closer to the real data. The whole process is shown in Figure 7 below.

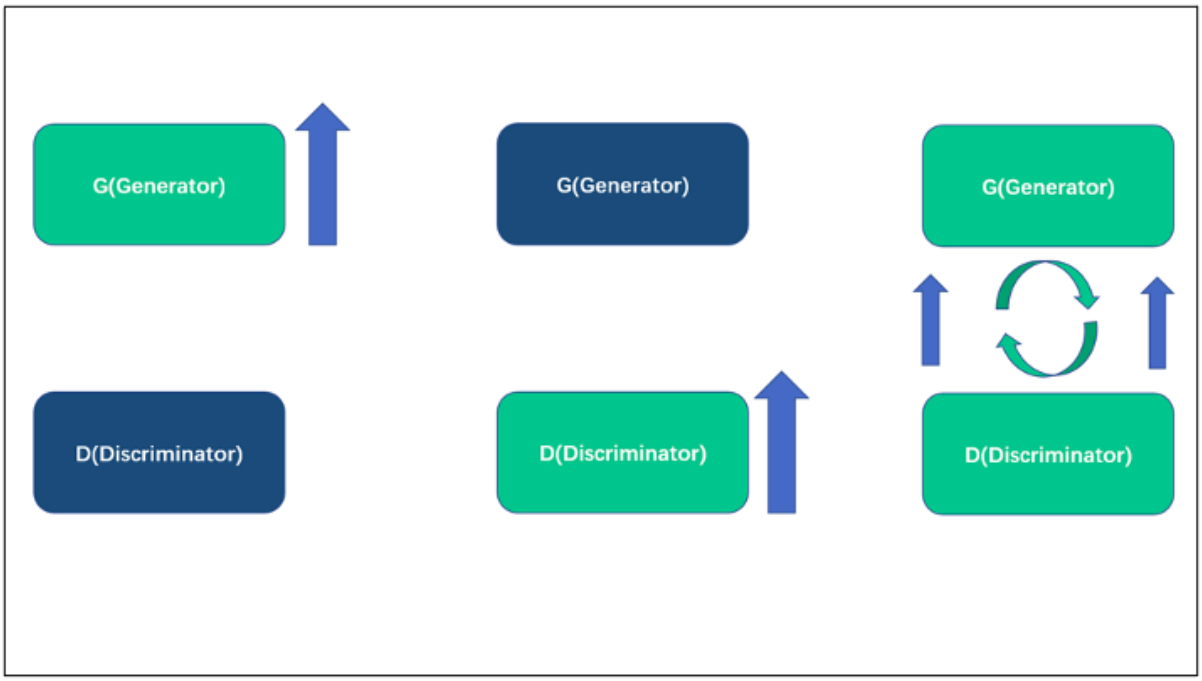

Figure 7: The training process of cGAN.

4.7 Evaluation Methods

This section shows the similarities between the generated data and the real data. Unlike image data, it does not have a unified metric standard, and the original data distribution is complex, so measuring the similarity of experimental data is challenging. Considering the above problems, this experiment will consider various methods to verify the validity of the generated data. To verify whether the synthetic data has similar utility to the real data, this paper verifies the distribution of the two kinds of data and the performance of the machine learning algorithm after data enhancement with the synthetic data.

4.7.1 Loss Function

The loss function is introduced for evaluation to determine whether the cGAN model gets the ideal result after training. Theoretically, the discriminator and generator in cGAN should be unstable at the beginning, and eventually stabilize to about 0.5 after repeated training.

The loss function is used to estimate the degree of inconsistency (i.e., error) between the predicted value and the real value of the model.

The loss function of generator network:

$$L_G = H(1, D(G(z)))$$

In the above formula, G stands for generation network, D stands for discriminative network, H stands for cross-entropy, and Z is random input data. It is the probability of judging the generated data, where 1 represents the absolute truth of the data and 0 represents the absolute falsehood of the data. Is the distance between the result and 1. If we want to do a good job of generating a network, we must let the discriminator identify the generated data as true (the distance between $D(G(z))$ and 1 is as small as possible).

The loss function of discriminative network:

$$L_D = H(1, D(x)) + H(0, D(G(z)))$$

The equation above is the real data. It should be noted that it represents the distance between the real data and 1 and between the generated data and 0. Suppose the recognition network wants to achieve good results. In that case, it must identify real data as real data and generated data as false data (that is, the distance between real data and 1 is small, and the distance between generated data and 0 is small).

4.7.2 Distribution of Data

The visualization method is used to distinguish the true and false data. Since the data in this paper belong to high-dimensional data and cannot be directly visualized, the PCA dimensionality reduction method is adopted to conduct the dimensionality reduction first and then the three-dimensional visualization. The difference between the raw and synthetic data can be seen visually in the generated 3D graphs. The PCA method is an eigenvalue decomposition of the covariance matrix. It transforms the original data into a lower dimensional space, thereby reducing the computational cost and facilitating the understanding of the data. A set of eigenvectors characterizes the projection of the data points onto a lower dimensional space. The eigenvectors with the biggest eigenvalues are chosen to form the projection. The synthetic data is generated by the GAN method. The generator generates synthetic samples that look like real data in this method. The discriminator is used to distinguish the synthetic data and real data. The training of GAN is an adversarial game between the generator and the discriminator. The objective function of the generator is to reduce the probability that the discriminator can distinguish the synthetic and real data. The objective function of the discriminator is to maximize the probability that the synthetic data is distinguished from the real data.

4.7.3 Machine Learning Performance

The performance of the machine learning model is the evaluation criterion in this paper. It is used to evaluate the performance of the machine learning models trained on synthetic

data. The synthetic data generated can be used to train machine learning models with the same performance as the real data.

The performance of machine learning models is evaluated by classification accuracy. The synthetic data generated by our method can train machine learning models with the same performance as the real data.

The classification based on machine learning is adopted in this paper. This study aims to generate samples with real data characteristics, expand the original data, and improve the generalization ability of the detection model. To avoid the error of individual classifiers due to the specific validity of generated data, three classifiers were used for verification, namely ISF(isolated forest), RF(random forest) and KNN (K-nearest neighbour algorithm). The accuracy was selected as the evaluation index.

ISF(isolated forest): The Isolation Forest technique, frequently employed by financial organizations to mine fraudulent behaviours, is chosen by the detection algorithm. The random forest concept is advantageous to the iForest algorithm. Similar to how the random forest is made up of several choice trees, the iForest forest comprises numerous binary trees. The algorithm employs a variety of tactics that are quite effective. Consider splitting the data space using a random hyperplane, which can result in two distinct subspaces (which is synonymous with a knife used to cut the cake into two). When each subspace has just one data point, we loop back and keep cutting each subspace with a random hyperplane. According to intuition, we may observe that dense clusters are chopped repeatedly before stopping, but sparse clusters can abruptly stop cutting and enter a subspace. The default settings are the same for both training models.

RF(random forest): It is to build a forest randomly. There are many decision trees in the forest, and each decision tree in the random forest does not correlate. After getting the forest, when a new input sample comes in, let each decision tree in the forest make a judgment separately to see which category the sample should belong to (for the classification algorithm), and then see which category is selected the most, and then predict the sample to be that category.

KNN((K-nearest neighbour algorithm): This algorithm can solve classification and regression problems. This method has a very simple principle: when classifying the test samples, the training sample set is scanned first to find the most similar training sample to the test sample, and the category of the test sample is determined by voting according to the category of the sample. A weighted vote can also be made by how similar a sample is to the test sample. Suppose the output needs to be in the probability of each class corresponding to the test sample. In that case. In that case, it can be estimated by distributing the number of samples of different classes in a sample.

5 Results and Discussion

5.1 Loss Function

While one-dimensional data lacks a standardized evaluation index, picture data may be compared using the Inception Score index. To assess, this article uses a multi-angle technique. The model loss function graph comes first. Based on the preceding evaluation, the training of the GAN network will ultimately achieve the Nash equilibrium, meaning that the produced data is near the real data distribution, and the discriminator cannot distinguish between true and false data. Based on close to 0.5, the likelihood that the forecast will come true for the provided data (equivalent to randomly guessing the class).

As can be seen in the image, the projected probability of the generator and discriminator eventually approaches 0.5 as the training epoch count rises. This implies that the discriminator in the model is unable to determine whether the data is accurate or not. Such results prove that a good cGAN model is obtained.

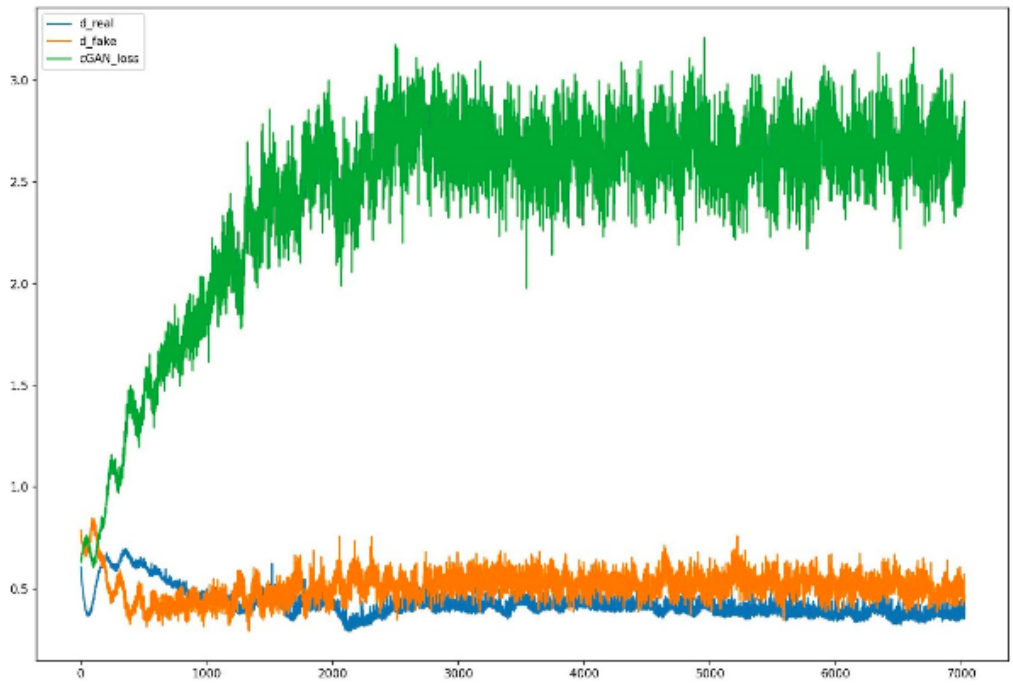

Figure 8: Loss function curve of the generator, discriminator and cGan.

Continuously improve the performance of generative and discriminant networks (the mature state of generative and discriminant networks is to learn a reasonable mapping function), which is an optimization process. By comparing the loss function in Figure 8,it is found that at the beginning, the loss values of the generator and discriminator fluctuated continuously. Finally, they were very close and tended to be balanced, which was in line with our expectations. With the loss function, the generation network and discriminant network can adjust the parameters using the Backpropagation (BP) algorithm and optimization method (such as gradient descent) based on their respective loss functions.

5.2 The Data Distribution

After dimensionality reduction by PCA, the actual data and the produced data may be viewed. There is no independent difference between the distribution of the created data and the distribution of the real data, as illustrated in Figure 9. The created data is comparable to the genuine data, according to the generated data outside the original distribution. The data's integrity is assessed using the visualization technique. The data in this article cannot be immediately viewed since it is high-dimensional data. As a result, the dimension is first reduced using the PCA dimension reduction method, and then three-dimensional visualization is done.

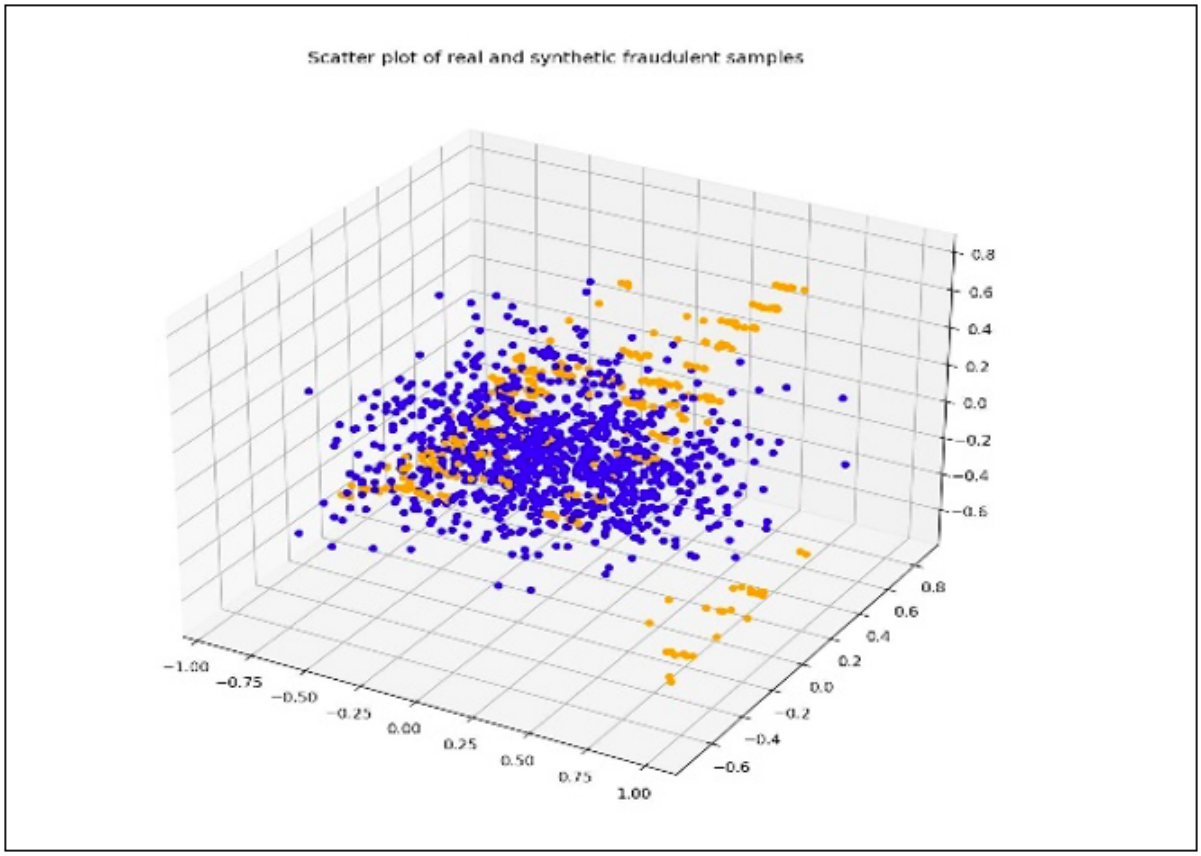

Figure 9: 3D visualization of real data vs generated synthetic data.

As it is shown in Figure 10, the feature distribution of the generated data, observing the real data and the feature distribution of the generated data. Obviously, the generated data distribution is roughly the same as the AMT (Amount in Bank) feature of the real data, but its distribution is not completely similar. The model in this paper has not fully learned the feature of the low frequency of the original data, which will be improved by the subsequent model, indicating that the quality of the generated data needs further improvement.

5.3 Machine Learning Performance

The verification process uses machine learning. Since fraud detection algorithms frequently require raw data, training detection methods are used to assess how good or terrible the generated data is. Combining the generated data with the initial data set creates an improved data set. The performance of the machine learning algorithms trained on both this data set and the original data is compared as they work together to identify the unified test set. The test data set must be rebuilt due to the severe imbalance between the original data categories, and the ratio between the rebuilt test set categories is 2:1. It is important to note that both algorithms will use this test set as their common test set.

The augmented dataset has a higher detection rate than the original data, with lower detection accuracy. It is also researched how the volume of data collected affects the outcomes. With the generated data, this is evident. The test set's accuracy is improved with the addition of. This demonstrates that the augmented dataset can increase the

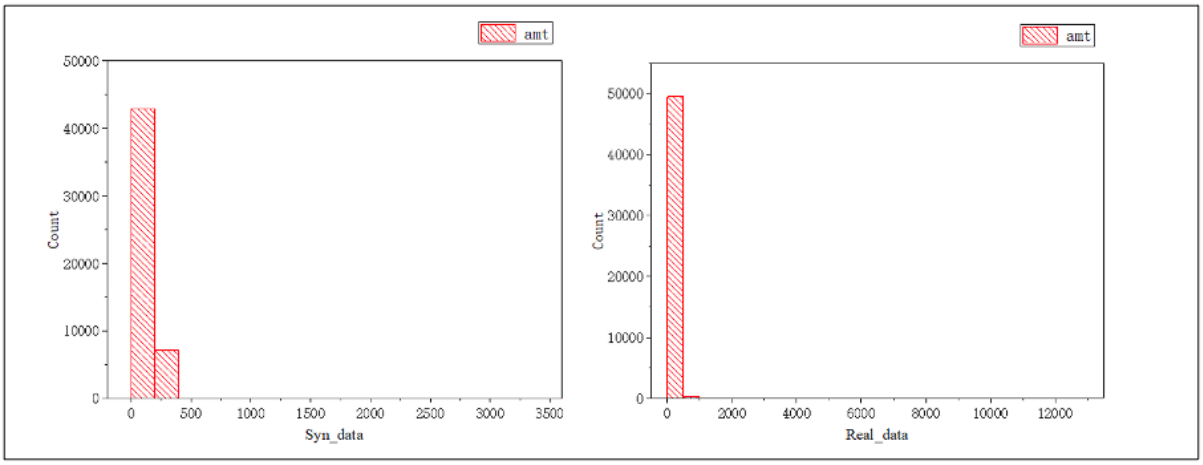

Figure 10: The distribution of amt (Amount in a bank) in original and synthetic data.

detection model's capacity to generalize because the created data is, in fact, quite close to the original data.

Three common machine learning models are used in this project: IsF, RF and KNN. Their accuracy performance is shown in Table 2.

Table 2: Accuracy of three machine learning models based on original data and the data augmented with synthetic data.

Accuracy in three machine learning arithmetic based on original and synthetic data										
	1	2	3	4	5	6	7	8	9	10
Ori_ISF	67.56%	67.48%	67.39%	68.10%	67.58%	67.76%	67.61%	67.21%	67.76%	67.76%
Ori_RF	67.56%	68.01%	67.43%	68.20%	67.72%	68.01%	68.20%	69.20%	67.30%	68.17%
Ori_KNN	68.10%	67.90%	68.20%	68.01%	67.03%	69.11%	68.31%	69.23%	69.10%	67.58%
Syn_ISF	69.21%	67.80%	69.40%	68.50%	67.64%	74.30%	71.20%	67.92%	67.90%	68.30%
Syn_RF	67.95%	67.89%	68.98%	68.21%	68.23%	68.90%	67.99%	68.21%	68.01%	69.35%
Syn_KNN	69.21%	69.13%	69.58%	69.21%	71.11%	68.21%	69.90%	68.81%	68.31%	68.47%

After ten times of repeated training and test prediction, Figure 11 shows the average accuracy of these ten times of results. The results show that the use of synthetic data for data enhancement has achieved an increase in accuracy in all three algorithms, with an average improvement of 0.93%,0.39% and 1.6% on KNN and RF. In the experiment, through the setting of optimization parameters, the accuracy of ISF algorithm can even be 74.3%. Due to the long training time and the instability of cGAN, there is still a lot of chance for optimization in the later stage of this model. If the deletion of abnormal results is increased, some unreasonable results can be eliminated.
Among them, ISF has the strongest generalization ability, and random forest also has a small improvement. However, KNN is slightly decreased, indicating that the generated data roughly has the characteristics of real data and can improve the generalization ability of the machine learning classification model.

5.4 Comparison of Developed Models vs Existing Models

According to the result of the above, the use of the methods used in this article build cGANs performance close to the real data on machine learning, can improve in data

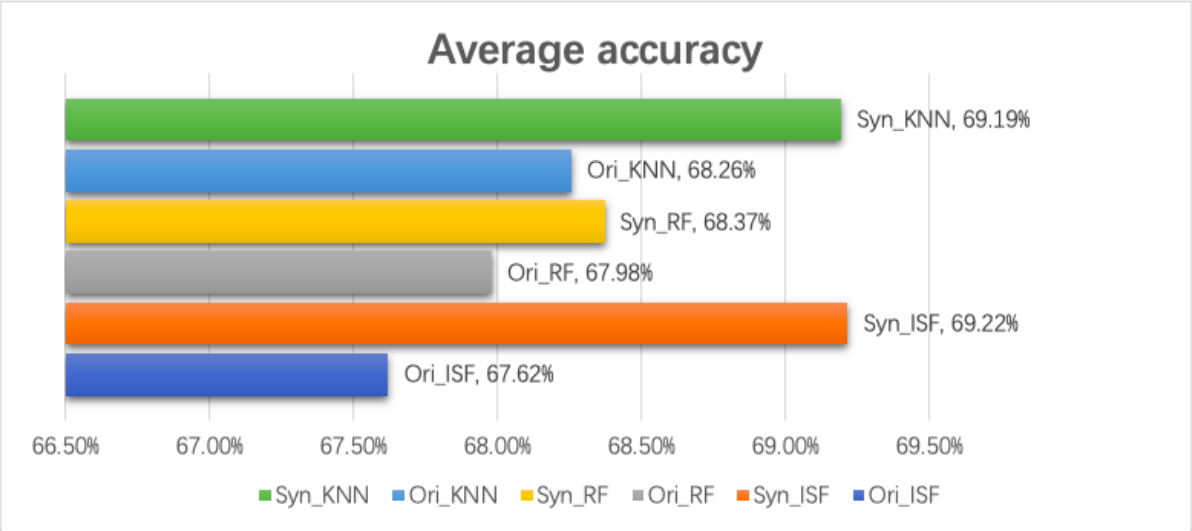

Figure 11: Average prediction accuracy of machine learning over 10 times.

set is limited in the number of samples, to increase the diversity of samples, after PCA dimension reduction of data distribution and the distribution of real data also very close, but due to extracting the characteristics of the reach of 24, There is a certain gap in the single data distribution, and optimization will be attempted in the future.

Through the research in this paper, it is proved that cGAN is a very effective method for synthesizing data, which can better model the data distribution. Theoretically, cGANs can train any kind of generator network. Other frameworks require the generator network to have some specific functional form, such that the output layer is Gaussian. There is no NEED for repeated sampling with Markov chains, no inference during learning, no complex variational lower bound, and avoiding the difficulty of approximate computation of tricky probabilities. The generated data can be used as data augmentation to meet most data research needs without violating user privacy. cGANs are a powerful tool for data synthesis, and cGAN is a very effective method for synthesizing data. cGAN can better model the data distribution and generate data close to the real data distribution. However, cGAN has some limitations, such as difficult training and instability. Good synchronization is required between the generator and the discriminator, but it is easy to D converge, and G diverge in practical training. One of the limitations of cGAN is that it is difficult to train. Good synchronization is required between the generator and the discriminator, but it is easy to D converge, and G diverge in practical training. D/G training requires careful design. Another limitation of cGAN is that it is prone to Mode Collapse.

One of the issues that can occur during the training of cGANs is called "mode collapse". Mode collapse is when the cGAN learns to generate only a subset of the possible outputs. This can happen for several reasons, but one of the main reasons is that the cGAN is only being trained on a limited data set. If the data set is too small, the cGAN will only learn to generate a small subset of the possible outputs. Another reason mode collapse can occur if the cGAN is not trained properly. If the cGAN is not trained correctly, it may only learn to generate a few different types of outputs. Several ways to prevent mode collapse include using a larger data set and a better training method. There are a few ways that Mode Collapse can be rectified. One way is to use a different loss function less susceptible to Mode Collapse. Another way is to use a different

generator architecture less susceptible to Mode Collapse. Finally, Mode Collapse can be rectified using a different training method, such as batch normalization or path normalization. The learning process of GANs may have a pattern loss, and the generator starts to degenerate, always generating the same sample points and unable to continue learning.

6 Conclusion and Future Work

In conclusion, this paper aims to use smart synthesis data to solve the limitations of traditional data anonymization. The data synthesized by cGAN is close to the data in distribution, and the performance in machine learning is close to the real data.

Synthetic data completely does not involve the user's privacy; they are completely by artificial intelligence created virtual data and using the study method of synthetic data in the data distribution and use in machine learning efficiency is close to real data, so this method is feasible, the research questions are solved and validated.

The model's output data are validated from two perspectives. The outcomes demonstrate that the original data distribution is followed by the data obtained using this approach. Due to this improvement, the classification model trained with the upgraded data is also better at generalization than previous approaches.

Synthetic data is one of the most popular research topics in AI. Banks, car manufacturers, unmanned aerial vehicles, factories, hospitals, retailers, robots and scientists are hoping to get more data due to the problem of privacy, even if the use of anonymous data will also be a variety of means to restore lost their privacy protection, and is not allowed by GDPR and other relevant laws, they tend to be the various restrictions. Finally, the speed of research is greatly reduced. Therefore, studying better synthetic data to replace the traditional anonymized data is a very valuable commercial approach.

In this study, an original data set containing user information was used and pushed into the generator and generator of cGAN after a series of data pre-processing. The cGAN model was trained and synthesized, and the data was generated and synthesized. Finally, the research topic is verified by comparing the model's loss function, data distribution, and machine learning performance. In this a few months of study, I used the relevant knowledge in the field of all kinds of data analysis due to GAN model is very difficult to train; it has a generator and the discriminator two parts, and data needs to be generated from multiple aspects in machine learning, statistics, modelling and optimization, there is a lot of knowledge in such aspects as programming. In the future, I will continue to strengthen my learning and knowledge reserve, overcome more challenges, and do more research and improvement in data analysis.

Although some of the work to produce artificial data with true data characteristics has been done in this study, there are still many areas that might use better in this line of inquiry. An in-depth study may be done from the following points in future work:

1. How to keep using the GAN model's data reconstruction capabilities, improve the model's structure, and add more top-notch deep learning models so that the combined model may benefit from various advantages to tackle more issues.

2. In the training of generator (G) and discriminator (D), if better sampling z is adopted, the training speed can be accelerated, and better data can be generated.

3. Because the synthetic data simulates the original data distribution, some unreasonable data are often generated. In future research, some restrictions can be added to

the generated data and those unreasonable or out of range can be removed, which can improve the authenticity of the generated data.

Acknowledgement

First, I would like to thank my supervisor, Catherine Mulwa. She was a very warm-hearted and responsible person who guided me in writing my thesis. Then I would like to thank the lecturers of NCI, who always helped me in machine learning and big data learning. Finally, I would also like to thank my family for their support in my study. Without this help, I don't think I can complete my research and thesis well.

References

Achour, G., Sung, W. J., Pinon-Fischer, O. J. and Mavris, D. N. (2020). Development of a conditional generative adversarial network for airfoil shape optimization, *AIAA Scitech 2020 Forum*, Orlando, FL, p. 2261.

BFM Quantum (2019). Data takes a quantum leap.
URL: *https://quantium.com/wp-content/uploads/2016/08/BFM_Quantium.pdf*

Chatterjee, S., Hazra, D., Byun, Y.-C. and Kim, Y.-W. (2022). Enhancement of image classification using transfer learning and gan-based synthetic data augmentation, *Mathematics* **10**(9): 1541.

Chen, R. J., Lu, M. Y., Chen, T. Y., Williamson, D. F. and Mahmood, F. (2021). Synthetic data in machine learning for medicine and healthcare, *Nature Biomedical Engineering* **5**(6): 493–497.

Dankar, F. K. and Ibrahim, M. (2021). Fake it till you make it: Guidelines for effective synthetic data generation, *Applied Sciences* **11**(5): 2158.

Deng, J., Pang, G., Zhang, Z., Pang, Z., Yang, H. and Yang, G. (2019). cGAN based facial expression recognition for human-robot interaction, *IEEE Access* **7**: 9848–9859.

Dikici, E., Prevedello, L. M., Bigelow, M., White, R. D. and Erdal, B. S. (2020). Constrained generative adversarial network ensembles for sharable synthetic data generation, *arXiv preprint arXiv:2003.00086* .

Douzas, G. and Bacao, F. (2018). Effective data generation for imbalanced learning using conditional generative adversarial networks, *Expert Systems with Applications* **91**: 464–471.

El Emam, K. (2020). Seven ways to evaluate the utility of synthetic data, *IEEE Security & Privacy* **18**(4): 56–59.

Gao, L., Chen, D., Zhao, Z., Shao, J. and Shen, H. T. (2021). Lightweight dynamic conditional gan with pyramid attention for text-to-image synthesis, *Pattern Recognition* **110**: 107384.

Gong, M., Xu, Y., Li, C., Zhang, K. and Batmanghelich, K. (2019). Twin auxilary classifiers gan, *Advances in Neural Information Processing Systems* **32**.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. (2014). Generative adversarial nets, *Advances in Neural Information Processing Systems* **27**.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V. and Courville, A. C. (2017). Improved training of wasserstein gans, *Advances in Neural Information Processing Systems* **30**.

HHS.gov (n.d.). Health information privacy.
URL: *https://www.hhs.gov/hipaa/index.html*

Intersoft Consulting (n.d.). General data protection regulation GDPR.
URL: *https://gdpr-info.eu*

Koenecke, A. and Varian, H. (2020). Synthetic data generation for economists, *arXiv preprint arXiv:2011.01374* .

Liu, Y., Peng, J., James, J. and Wu, Y. (2019). PPGAN: Privacy-preserving generative adversarial network, *2019 IEEE 25Th International Conference on Parallel and Distributed Systems (ICPADS)*, IEEE, Tianjin, China, pp. 985–989.

Lu, P.-H., Wang, P.-C. and Yu, C.-M. (2019). Empirical evaluation on synthetic data generation with generative adversarial network, *Proceedings of the 9th International Conference on Web Intelligence, Mining and Semantics*, Association for Computing Machinery, Seoul, Republic of Korea, pp. 1–6.

Ma, C., Li, J., Ding, M., Liu, B., Wei, K., Weng, J. and Poor, H. V. (2020). Rdp-gan: Ar\'enyi-differential privacy based generative adversarial network, *arXiv preprint arXiv:2007.02056* .

Mirza, M. and Osindero, S. (2014). Conditional generative adversarial nets, *arXiv preprint arXiv:1411.1784* .

Mogren, O. (2016). C-RNN-GAN: Continuous recurrent neural networks with adversarial training, *arXiv preprint arXiv:1611.09904* .

Nash Jr, J. F. (1950). Equilibrium points in n-person games, *Proceedings of the National Academy of Sciences* **36**(1): 48–49.

Patki, N., Wedge, R. and Veeramachaneni, K. (2016). The synthetic data vault, *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, IEEE, Montréal, Canada, pp. 399–410.

Qu, Y., Yu, S., Zhang, J., Binh, H. T. T., Gao, L. and Zhou, W. (2019). GAN-DP: Generative adversarial net driven differentially privacy-preserving big data publishing, *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, IEEE, Shanghai, China, pp. 1–6.

Radford, A., Metz, L. and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks, *arXiv preprint arXiv:1511.06434* .

Raghunathan, T. E. (2021). Synthetic data, *Annual Review of Statistics and Its Application* **8**: 129–140.

Ramponi, G., Protopapas, P., Brambilla, M. and Janssen, R. (2018). T-cgan: Conditional generative adversarial network for data augmentation in noisy time series with irregular sampling, *arXiv preprint arXiv:1811.08295* .

Snoke, J., Raab, G. M., Nowok, B., Dibben, C. and Slavkovic, A. (2018). General and specific utility measures for synthetic data, *Journal of the Royal Statistical Society: Series A (Statistics in Society)* **181**(3): 663–688.

Tiwald, P., Ebert, A. and Soukup, D. T. (2021). Representative & fair synthetic data, *arXiv preprint arXiv:2104.03007* .

United States Data Science Institute (2022). How to overcome challenges of data adoption in 2022.
URL: *https://www.usdsi.org/data-science-insights/how-to-overcome-challenges-of-data-adoption-in-2022*

Vega-Márquez, B., Rubio-Escudero, C., Riquelme, J. C. and Nepomuceno-Chamorro, I. (2019). Creation of synthetic data with conditional generative adversarial networks, *International Workshop on Soft Computing Models in Industrial and Environmental Applications*, Springer, Seville, Spain, pp. 231–240.

Wang, X., Xie, L., Dong, C. and Shan, Y. (2021). Real-esrgan: Training real-world blind super-resolution with pure synthetic data, *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Montreal, BC, Canada, pp. 1905–1914.

Xu, C., Ren, J., Zhang, D., Zhang, Y., Qin, Z. and Ren, K. (2019). GANobfuscator: Mitigating information leakage under GAN via differential privacy, *IEEE Transactions on Information Forensics and Security* **14**(9): 2358–2371.

Xu, L., Skoularidou, M., Cuesta-Infante, A. and Veeramachaneni, K. (2019). Modeling tabular data using conditional GAN, *arXiv preprint arXiv:1907.00503* **1**.

Zheng, X., Wang, B. and Xie, L. (2019). Synthetic dynamic PMU data generation: A generative adversarial network approach, *2019 International Conference on Smart Grid Synchronized Measurements and Analytics (SGSMA)*, IEEE, Texas A&M University, College Station, TX, pp. 1–6.