IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 26 June 2023, accepted 4 August 2023, date of publication 9 August 2023, date of current version 16 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3303509

APPLIED RESEARCH

CTGAN-MOS: Conditional Generative Adversarial Network Based Minority-Class-Augmented Oversampling Scheme for Imbalanced Problems

ABDUL MAJEED AND SEONG OUN HWANG, (Senior Member, IEEE)
Department of Computer Engineering, Gachon University, Seongnam 13120, South Korea

Corresponding authors: Seong Oun Hwang (sohwang@gachon.ac.kr) and Abdul Majeed (ab09@gachon.ac.kr)

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) funded by the Korean Government [Ministry of Science and ICT (MSIT)] under Grant 2021-0-00540.

ABSTRACT This paper proposes a novel data augmentation scheme called the conditional generative adversarial network minority-class-augmented oversampling scheme (CTGAN-MOS) for solving class imbalance problems. Our methodology encompassed six key steps: data engineering using sophisticated pre-processing techniques, identifying the type of vulnerabilities present in the data, curating good quality synthetic data using the CTGAN model, the intelligent fusion of real and synthetic data, noise removal from the augmented data using coin-throwing algorithm, and building classifiers with the high-quality augmented data. Our scheme maintains higher structural similarity (data truthfulness) between the original and the resampled data by intelligently adding high-quality samples only to the minority class, whereas some augmentation techniques add records to the majority class, leading to poor-quality resampled data. Our scheme removes noisy samples from the data, which has remained unexplored in the CTGAN-based data augmentation. Furthermore, it augments data by adding fewer records compared to existing schemes, while offering comparable performance. Experiments are conducted on benchmark datasets to prove the feasibility of the proposed CTGAN-MOS in realistic scenarios. Results prove the improvement by CTGAN-MOS over existing state-of-the-art (SOTA) techniques in terms of accuracy, recall, precision, F_1 score, and G-mean score. Specifically, the CTGAN-MOS has yielded accuracy values of 100% and 99.83% on two datasets which are higher than recent SOTA techniques. On average, it has yielded the 22.58% and 29.47% improvements w.r.t. G-mean score on two different datasets. On average, it adds 8.26% and 26.01% fewer records than the existing SOTA methods in the two datasets. Lastly, our scheme yields highly balanced confusion matrices compared to recent SOTA data augmentation techniques.

INDEX TERMS Imbalanced problem, data augmentation, machine learning, classifiers, noise, majority class, minority class, model training, samples, intelligent fusion, data truthfulness, data engineering.

I. INTRODUCTION

Machine learning (ML) models have contributed to seamlessly solving many real-world problems in medical diagnoses [1], fault detection in machines [2], credit card fraud [3], survival predictions [4], pattern recognition [5], event classification [6], and thermal image analysis [7], to name just a few. During the COVID-19 pandemic, ML models have played a significant role in the detection of infection, prediction of outbreaks, severity analysis, etc. [8]. In most of these ML applications, two elements are vital: the code of the ML model and the data. The code has vastly improved from multiple perspectives, but data quality and availability are still the main barriers to the development and adoption of ML models at a scale suitable for many real-world applications. In most ML applications, classifiers are mainly used to separate instances into one of two groups (yes/no, faulty/non-faulty, fraudulent/normal,

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Asif.

normal/abnormal, etc.), which therefore creates a binary classification problem. In such a problem, one class might have more samples than the other. The class with the higher number of samples is the majority class, and the class with the least samples is the minority class. The classifiers trained on such skewed data only learn information well enough from the majority class, and the minority class remains unlearned. Consequently, ML classifiers can only correctly predict the majority class, and the minority class cannot be predicted (not even a single sample in some cases) [9]. The performance of ML models when using unbalanced data with significant skews in the class distribution is called an *imbalanced learning problem*. A remedy for this problem is imperative because ML models are increasingly used in many safety-critical applications.

Despite the recent proliferation of data curation and availability tools, small-sized and poor-quality datasets are common in the ML domain, leading to imbalanced learning in the classifiers. As stated earlier, a remedy for this problem is vital to prevent negative consequences for society. For example, poorly learned classifiers for cancer detection can incorrectly identify unhealthy patients as healthy (and vice versa), leading to a wide range of negative consequences. Similarly, a classifier that was developed on poor-quality data for COVID-19 detection may misclassify an uninfected person, leading to social stigma or financial loss. Hence, it is necessary to develop reliable ML models for real-life scenarios involving high risk, particularly from binary problems [10]. Data augmentation techniques (DATs) are most widely used to address imbalances in real-world datasets. To date, many DATs have been proposed to increase data size, to balance class distributions, and to improve the generalization of ML models [11], [12], [13]. Most DATs are based on one of two approaches.

1) Improving the distribution of the minority class by resampling the data (e.g., getting more samples by utilizing information from the available samples).
2) Improving the distribution of the minority or majority class by curating synthetic data that mimic the properties of the real data by using generative AI models (e.g., generative adversarial networks and statistical methods).

In the first approach, there is the possibility of losing statistical information from either the majority or minority class, and the classifier might yield unsatisfactory performance when data are sparse (or cannot easily be separated) [10]. In the latter approach, it is hard to identify a sufficient amount of data, and the fusion of synthetic and real data is tricky. Furthermore, synthetic data generation is challenging when real data are of poor quality and most attributes have skewed distributions. Although both approaches assist in improving data size, there are few standardized data augmentation processes that can be applied to most applications using tabular data. Furthermore, it is hard to produce synthetic data with enough diversity to compensate for a deficiency of samples in the original data [14]. Therefore, this study develops a practical data augmentation scheme that explores the problems in the original data and fixes them (e.g., size, distribution, fusion), providing better performance than state-of-the-art (SOTA) DATs in a tabular data environment. The major contributions of this work are as follows.

- We explore major performance bottlenecks (induction of noise in training data, lower accuracy, imbalanced confusion matrices, and unnecessary record addition) in the existing DATs when used to supplement ML classifiers. We identify opportunities to develop a new conditional generative adversarial network-based minority-class-augmented oversampling scheme (CTGAN-MOS), which effectively resolves the performance bottlenecks of existing DATs, and yields better performance.
- The proposed CTGAN-MOS identifies data-quality problems in training data, and fixes them by using sophisticated data engineering techniques, leading to higher generalization, superior accuracy, and balanced learning in ML classifiers, compared to previous DATs.
- The proposed CTGAN-MOS curates more data of good quality by using a CTGAN and, with fewer records, augments only the minority class, whereas some of the existing DATs augment both majority and minority classes, leading to extra overhead and low accuracy [15].
- The proposed scheme removes noisy samples from the data by using a coin-throwing algorithm, which has remained unexplored in CTGAN-based data augmentation.
- To the best of our knowledge, this is the first scheme that adds fewer synthetic records to the data but still yields accuracy that is close to optimal ($\sim$100%).

Experiments were performed on poor-quality datasets to prove the effectiveness of the CTGAN-MOS. Experiment analysis based on six evaluation metrics indicates better performance from the CTGAN-MOS than from its counterparts.

The rest of this paper is organized as follows. Section II presents the background and a survey of the literature concerning DATs, and then analyzes data augmentation techniques. Section III presents the preliminaries and formulates the problem to be solved. Section IV explains the proposed CTGAN-MOS and highlights its key components. Section V presents the results obtained from experimentation on real-life datasets and offers comparisons with the existing SOTA DATs. Section VI summarizes the important findings of this work and explains other aspects related to this work. We conclude the paper in Section VII, and list directions for our future work.

II. BACKGROUND AND LITERATURE SURVEY

In this section, we present background information and related studies that have addressed the imbalanced learning problem.

A. CLASS IMBALANCE PROBLEM AND ITS REMEDY

The class imbalance problem can be perceived as a causal effect when one class distribution is highly skewed in one direction and the ML classifier is unable to correctly learn information (a.k.a. hidden patterns) in the minority class, resulting in a deficient performance with examples of the minority class [16]. Specifically, in some cases, the representation of one class in the data is so high that the classifier only learns the maximal information on this class and ignores others. In real-world datasets, class imbalance emerges due to flaws in the data collection process. For example, a researcher or domain expert can be interested in collecting data from only unhealthy persons rather than healthy ones to validate/verify some hypothesis. Similarly, an algorithm employed to collect data from machines in operation can collect them at regular intervals, but the machinery encounters faults only on some occasions. Hence, the data representation for non-faulty parts can be significantly higher than for faulty parts. Furthermore, in some cases, data collection for multiple classes is not possible owing to restrictions or privacy concerns. In some cases, the feature values of two classes can be identical, and it is hard to separate them with a clear boundary, so the ML classifier perceives them as one class. An example of a class imbalance problem is given in Fig. 1(a), where the round black data points belong to the majority class, and the square red data points are from the minority class.

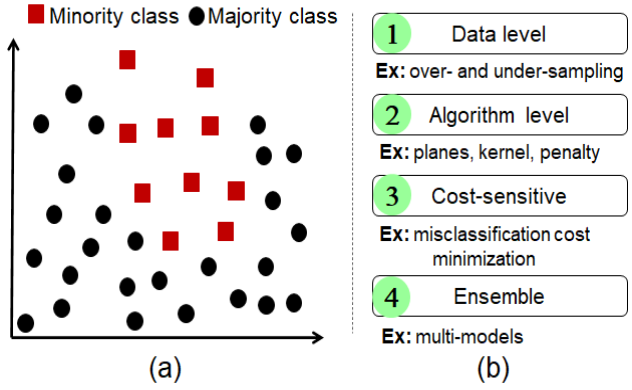


FIGURE 1. Overview of the class imbalance problem and its potential solutions.

The imbalance in the data can be determined using the imbalance ratio ($\mathcal{IR}$) and Fischer's ratio ($\mathcal{FR}$), as expressed in Eqs. 1 and 2, respectively:

$$\mathcal{IR} = \frac{\#neg}{\#pos} = \frac{|majority\ \ class|}{|minority\ \ class|} \tag{1}$$

where *#pos* denotes the # of samples in the minority class, and *#neg* represents the # of samples in the majority class (a high value for $\mathcal{IR}$ indicates a higher imbalance in the data) and

$$\mathcal{FR} = MAX(f), \quad where\ f = \frac{(\mu_{c_i} - \mu_{c_j})^2}{\sigma_{c_i}^2 + \sigma_{c_j}^2} \tag{2}$$

where μ_{c_i} and μ_{c_j} denote the mean values of features that belong to c_i and c_j, respectively, with $\sigma_{c_i}^2$ and $\sigma_{c_j}^2$ representing variances of features in those classes. A higher value for $\mathcal{FR}$ indicates better separability of the two classes.

To balance the class distribution, four mainstream methods are used, as shown in Fig. 1(b). These methods address the imbalance problem by tuning either the data or the ML model. In the *data-level* methods, the distribution of classes is balanced by either deleting some samples from the majority class or adding new samples to the minority class. Deleting samples from the majority class is called undersampling, and inserting more records to the minority class is oversampling. A conceptual overview of the data-level methods used to balance the distribution of classes is shown in Fig. 2. In undersampling methods, there is a risk of losing statistical information and reducing the data size. In contrast, oversampling methods may introduce noise into the data, and the learning ability of the ML classifiers can be impacted. In some cases, both methods can be used jointly to solve the class imbalance problem [17]. Both methods have been rigorously upgraded to improve the learning ability and generalization of the classifiers.

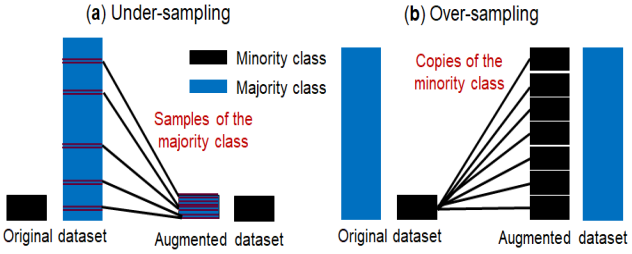


FIGURE 2. Overview of data-level techniques used to balance data in ML.

In *algorithm-level* methods, the workflow of the ML model is modified to address the imbalance problem. For example, support vector machine (SVM) hyperplanes can be guided with bias term b to separate two classes with a maximum margin. Similarly, penalty constants can be introduced for each class, and misclassifying minority class instances can be more expensive than misclassifying the majority class. In some cases, objective functions of ML models are designed in such a way that regions located close to the decision boundary are given more attention than other regions to prevent misclassification [18]. In addition, identifying hyperparameters that can distinguish majority and minority classes (e.g., kernel size) constitutes an algorithm-level method for an ML model.

In *cost-sensitive* methods, the total cost of misclassification is minimized by incorporating both of the previous methods (data-level and algorithm-level). Specifically, cost-sensitive methods assign a higher cost to the minority class than the majority class to prevent negative consequences [19]. In these methods, it is necessary to correctly recognize positive samples/instances, rather than negative instances. For example, a distinct cost depending upon the outcome can be assigned to classifiers during learning in the medical domain—the

cost of misclassifying a patient as noncancerous requires more tests. In contrast, the cost of misclassifying cancerous patients (from the diagnosis point of view) can be fatal if they are classified as healthy. In practice, the misclassification cost of positive samples is higher than for negative samples: $c(+,-) > c(-,+)$. These methods have been widely used in diverse domains to address imbalanced learning problems [20].

In *ensamble* methods, multiple ML classifiers/regressors are trained to improve accuracy from imbalanced learning problems [21]. The ensemble approach is more stable and reliable than the single-classifier approach. In these methods, multiple classifiers are employed to boost accuracy by combining the results of classifiers. In some cases, multiple classifiers are used to identify the best ML model for the given problem/task. In other cases, the code of base learners is modified to yield superior results involving imbalanced data.

All of the above-cited methods help in addressing the imbalanced learning problem in ML classifiers. This work amalgamates data and the ensemble method to resolve the imbalanced learning problem in real-world scenarios.

B. ANALYSIS OF DATA AUGMENTATION TECHNIQUES

Thus far, many DATs have been developed to address the imbalance problem in different applications, such as medical diagnosis, email classification, prediction of breast cancer, etc. The existing techniques aim to address the following data complexity factors to enhance ML model performance [22].

- Reduce the overlap between different classes.
- Address $N1$ complexity in the datasets.
- Avoid imbalance in the majority and minority class samples.
- Reduce noise by removing less important samples.
- Ensure a variable spread in a chunk of the class from the dataset.
- Apply a non-linear boundary among classes.

The two most widely used data-level techniques to address the imbalance problem are oversampling and undersampling. Undersampling techniques downsize the majority class and can have limited applicability in real-world scenarios owing to the small amount of data [23]. In contrast, oversampling techniques upsize the minority class by adding more samples, and have been widely used to balance class distributions in a dataset. There is less risk from using oversampling techniques than from undersampling techniques in terms of information loss and data overfitting [24]. The oversampling techniques pay attention to minority classes and balance the distribution by adding more records, leading to better performance of classifiers, regardless of data size [25], [26].

Before the inception of the synthetic minority oversampling technique (SMOTE), the balancing of classes was usually performed randomly, leading to overfitting issues in most cases [27]. However, with noisy data, SMOTE has shown better results than random sampling techniques in terms of robustness. SMOTE generates new data points by utilizing the k-nearest neighbor concept and information from minority data points. The formalization of SMOTE is in Eq. 3:

$$p_{new} = p_i + (p_l - p_i) \times w \tag{3}$$

where p_{new} is the new synthetic data point, p_i is a data point from the minority class, p_l is the k-nearest neighbor of p_i, and w is a random random where $w \in [0, 1]$.

By adding new synthetic data points, the class imbalance problem can be resolved effectively, and information loss can be restrained. The newly generated data points are well separated, and therefore, the possibility of noisy data generation is low, leading to better performance in most cases. Despite the many SMOTE benefits, it is prone to technical challenges owing to higher variations in real-world datasets, and therefore, SMOTE has undergone extensive improvements since its inception.

Douzas and Bacao [28] proposed an enhancement to the conventional SMOTE to generate data points of high quality, leading to better performance. Their proposed G-SMOTE can prevent class overlap and redundant data point generation. Camacho et al. [29] extended *G*-SMOTE to regression tasks, and improved the data generation methodology using a geometric region concept rather than line segments. Moutaouakil et al. [30] developed OEGFCM-SMOTE to reduce noise while balancing the data using optimization techniques. Their version of SMOTE performs a series of steps, including grouping, filtering, and interpolation, to reduce noise in the data. Zhang et al. [31] developed an improved variant of SMOTE called SMOTE-*RkNN*. Their method identifies and removes noisy samples by extracting the probability density information of each data point using *RkNN*. Zhang and colleagues [32] developed a SMOTE variant called IW-SMOTE that is more universal and robust compared to the original SMOTE. That technique extracts location information (safe, borderline, noise, etc.) of each data point and filters noisy data points.

Liu [33] developed SMOTE-RD to smooth the decision boundary in imbalanced classification problems. The method has three advantages (noise reduction, fewer parameters, and generation of additional points in sparse regions as well as close-to-class boundaries). Xie et al. [34] developed IH-MGD-SMOTE to generate additional data points that strictly follow the distribution of the original minority class instances, leading to better performance than many existing oversamplers. Zhai et al. [35] developed two SMOTE algorithms named BIDC1 and BIDC2 that are based on ELMAE and GAN. The proposed methods increase the diversity in newly generated data to improve the performance of classifiers. Li et al. [36] developed SP-SMOTE to deal with imbalanced classification problems by using density and space partitioning techniques. Sharma et al. [37] developed SMOTified-GAN to solve the imbalance problem by amalgamating a GAN with SMOTE. The proposed approach showed improvements of 9% over SOTA methods. Dou et al. [38] developed a new SMOTE technique called NSS-SMOTE

TABLE 1. Summary and comparisons of the recently developed state-of-the-art data augmentation schemes for solving class imbalanced problems.

Ref.	Advantages	Disadvantages	Method used	# of new records	Noise removal	CTGAN used
Dina et al. [15]	Improve learning dynamics of classifiers	Destroys data truthfulness by adding more records	CTGANSamp SMOTE	High	No	Yes
Douzas et al. [28]	Better quality of data when data size is small	High class overlap and poor separability	Geometric SMOTE	High	No	No
Camacho et al. [29]	Better prediction of rare & extreme values	Limited adoption to classification tasks	EG-SMOTE	High	No	No
Moutaouakil et al. [30]	Generate sample in safe regions only	Poor separability when samples are close	OEGFCM-SMOTE	High	Yes	No
Zhang et al. [31]	Better quality of synthetic data	Higher computing cost if noisy samples are large	SMOTE-*RkNN*	High	Yes	No
Zhang et al. [32]	Better separation of classes	Vulnerable to class overlap & small adjuncts	IW-SMOTE	High	Yes	No
Liu [33]	The complexity of method is low	Can only applied to sparse data	SMOTE-RD	High	Yes	No
Xie et al. [34]	Higher truthfulness in data	Poor separability when data is noisy	IH-MGD-SMOTE	High	No	No
Zhai et al. [35]	Augment classifiers' generalization	Cannot be applied to high-dimensional data	BIDC1 & BIDC2	High	No	No
Li et al. [36]	Detection of minority class with ease	Vulnerable to wrong sample selection	SP-SMOTE	High	No	No
Sharma et al. [37]	Improve the minor class representation	Introduce noise in data as diversity is not less	SMOTified-GAN	High	No	Yes
Dou et al. [38]	Can be applied to incomplete data	Vulnerable to class overlap and small adjuncts	NSS-SMOTE	High	Yes	No
Semenoglou et al. [41]	Improve data size for ML classifiers	Destroy data semantics by adding more records	ROS	High	No	No
Maldonado et al. [42]	Applicable to high-dimensional problems	Poor separability when data is dense	FW-SMOTE	High	No	No
Douzas et al. [43]	Better balance of data in complex datasets	Poor performance when data is low-quality	k-means SMOTE	High	No	No
Chakraborty et al. [44]	Preserves semantics of real data	Improve some parts of data only	Fair-SMOTE	High	No	No
This study (CTGAN-MOS)	Add less # of records and yield higher *Acc*	The complexity can increase if data is very noisy	CTGAN-MOS	Low	Yes	Yes

to deal with imbalanced problems involving incomplete datasets. The proposed technique is applied to poor-quality datasets and is effective at reducing noise in newly curated data. Further information about SMOTE and its latest variants can be obtained from recent surveys [39], [40].

Apart from the variants cited above, other recent and SOTA SMOTE techniques are random oversampling (ROS) [41], FW-SMOTE [42], k-means SMOTE [43], Fair-SMOTE [44], and CTGANSamp SMOTE [15]. The concrete analysis of recently developed SOTA augmentation techniques is given in Table 1. We contrast existing schemes based on six features (e.g., advantages, disadvantages, the method used, # of records added at the augmentation time, noise removal applied, and whether CTGAN was used to curate synthetic data) to visualize their relationship and differences. From the analysis given in Table 1, it can be observed that most techniques have not explored ways to reduce # of records to be added to the data while improving accuracy. Furthermore, the noise removal mechanism has not been applied to the CTGAN-based data thus far. Lastly, most existing techniques have applied fewer evaluation metrics than CTGAN-MOS while evaluating the performance. The ROS technique randomly adds more data points in the minority class by following either uniform or normal distributions. FW-SMOTE curates more samples in high-dimensional settings by using weighted Minkowski distance rather than Euclidian distance, and the resulting data points are more relevant for classification purposes. The k-means SMOTE variant uses the k-means algorithm to divide data into various clusters, and then balances the distribution of minority and majority classes in each cluster. Fair-SMOTE balances the distributions of data internally in such a way that the total number of samples is equal in both classes w.r.t. the target class. CTGANSamp SMOTE curates more data using the CTGAN model, and adds some new data points in both majority and minority classes. We affirm the contributions of the above-cited SMOTE variants, but there are four technical problems with the above-cited DATs.

- Some DATs add new records to both majority and minority classes, leading to loss of truthfulness and failure to handle the imbalanced problem in a fine-grained manner, particularly when the data size is large [15].
- Most DATs tend to add more records to the minority class without identifying a suitable region, leading to excessive noise in the data and poor separability [16].
- Most SMOTE-based methods try to resolve the imbalance between different classes while ignoring within-class imbalance that can lead to lower accuracy and over-generalization issues [16].
- Most DATs often ignore the quality of data being generated with GAN models from a diversity perspective, which can lead to performance issues while minimally improving the accuracy and other related metrics [35].

The proposed CTGAN-MOS resolves the above-cited problems in prior work without compromising performance.

III. PRELIMINARIES AND PROBLEM FORMULATION

A. DATA MODEL

In this work, we focus on a binary classification problem in R^n, where R^n is n-dimensional real space. A real-world dataset, D, encompassing input features and a corresponding target class is the input. Mathematically, $D = \{(x_i, y_i)|i = 1 \sim n\}$, where $x_i \in R^n$ denotes the input, and $y_i \in \{c_1, c_2\} = \{+, -\}$ shows the output. I, where $I = 1, 2, \ldots, n$, can be used to denote the indices of both $c_1/+$ and $c_2/-$ samples. Let D be a data table with $r \times c$ dimensions, where c represents the number of columns and r is the number of rows. D encompasses mixed attribute types, i.e., numerical and categorical. Each row in D contains complete information of a sample, $r = x_i \cup y_i$, whereas a column contains one item of the sample (e.g., age or gender). A common structure of D for a sample of 6,000 instances is shown in Eq. 4, as shown at the bottom of the next page.

In Eq. 4, all columns except the last are called input features, whereas the last column is called the target class. In this work, our focus is on the binary classification problem, and therefore, the cardinality of the last column is 2, which can be denoted with c_1 (where $c_1 =$ Rhinitis) and c_2 (where $c_2 =$ Cancer). If $|c_1| > |c_2|$, then c_1 is regarded as the majority class, denoted c_M. The minority class is denoted c_m. Ideally, $|c_M| == |c_m|$ is preferred to ensure balanced learning in the classifiers.

B. PROBLEM FORMULATION

Let $y = c_i \cup c_j$, where c_i and c_j represent the values of target class y. In any real-world D, the distribution of samples

in both c_i and c_j classes can be imbalanced, meaning that the total number of samples in c_i can be greater than c_j, or vice versa. To balance the distribution of c_i and c_j, data-level method $\mathcal{M}$ is mostly used. $\mathcal{M}$ can be an oversampling method (e.g., SMOTE) or an undersampling method to alter the number of samples in either c_i or c_j to yield a balanced y. Afterward, a classifier, $\mathcal{K}$, such as a random forest, an SVM, a decision tree, or a neural network, can be trained on a balanced subset y. However, the drawback with $\mathcal{M}$ is either a reduction in data size or induction of noise in D. In recent years, many variants of SMOTE have been developed to add more records to an underrepresented class, to improve the classification boundary, and/or prevent class overlap. However, addressing class imbalance issues in D is challenging for multiple reasons: (a) how to generate more data of good quality, (b) how to ensure that the distribution of values in newly generated data and the original data are alike, (c) how to ensure that newly generated samples only lie in a safe region, (d) how to guarantee that only a small portion of data can solve the class imbalance problem, (e) how to ensure that augmented data improve multiple aspects of classifiers (sampling quality, accuracy, learning ability, fair training, etc.), and (f) how to ensure that data augmentation significantly improves the results with the least possible computing overhead. Furthermore, restraining the possibility of bias in, and inappropriate conclusions from, the augmented D when given as input to $\mathcal{M}$ is also tricky.

The chosen problem to be formulated is as follows. *Given a real-world tabular dataset, D, that includes various features (age, gender, race, residence type, etc.) and a target class (disease, salary, stroke occurrence, etc.), there can be multiple vulnerabilities in D such as an inadequate number of records, big gaps in some sample values, inconsistent values for some features, records with missing values, outliers, duplicate records, etc. In addition, there can be advanced vulnerabilities (e.g., skewed distributions of the target class that can seriously impact the learning capability of classifiers, and classifiers that only learn information about the majority class). Consequently, the minority class remains unlearned during the training process, and inference can be very low. How do we produce a good-quality, augmented dataset D where (a)* $\mathcal{D} \subseteq D$*; (b)* $\forall\, c_i, c_j \in D$, $|c_i| == |c_j|$ *(i.e.,* $\mathcal{IR} \sim 1$*), where the distribution of classes is balanced; (c)* $\mathcal{M}$ *learns all classes fairly by using D during training; (d)* $\mathcal{D} = D + D_{new}$*, and augmentation is performed with as few external records as possible (e.g.,* $|D_{new}|$ *is a very small amount of data, but the distributions are accurately modeled); (e) the diversity of newly generated data (*D_{new}*) is reasonably high; (f) the new records preserve truthfulness from D; (g)* $\mathcal{D}$ *has the best quality for training classifiers and downstream tasks; and (h)* $\mathcal{D}$ *improves multiple performance metrics (i.e., has significantly high accuracy, recall, precision,* F_1*, and G-mean score)?*

In many data-driven solutions, the quality of the underlying data is imperative for conducting analytics (drawing pictures out of the data) as well as training classifiers. To this end, the $\mathcal{D}$ produced with our scheme is expected to meet the demand of analysts and can compensate for the deficiency of good data in futuristic, data-hungry AI applications.

IV. THE PROPOSED CTGAN-MOS FOR SOLVING CLASS IMBALANCE PROBLEM

In this section, we introduce our CTGAN-MOS for solving the class imbalance problem, and we describe its key components. CTGAN-MOS addresses the performance bottlenecks of ML classifiers when working with poor-quality data. The problem of class imbalance is resolved using this new scheme, which integrates data engineering techniques, applying vulnerability analysis of the datasets w.r.t. distribution/sizes, curating high-quality data using a CTGAN, using an intelligent fusion of newly curated and existing data, removing noisy samples from augmented data, and building classifiers with the final high-quality data. Table 2 presents details on the notations used in our proposed scheme.

TABLE 2. Notations used in the proposed CTGAN-MOS.

Symbol	Description
D, $\mathcal{D}$, D_{new}	Orignal data, augmented data, newly curated data
x_i, y_i, c	ith feature vector, ith target class, class label
$\mathcal{M}$, $\mathcal{K}$	Data-level augmentation method, ML classifier
N, n	# of users in a dataset, # of elements in x_i
$\mathcal{S}(x_i, x_j)$	Similarity between two records x_i and x_j
$\mathcal{Z}$	Similarity matrix at $N \times N$ (diagonal entries are 1)
c_M, c_m	Majority class, minority class

Balancing the distribution of classes is necessary, because many real-world datasets are skewed, messy, small-scale, and/or noisy, leading to imbalanced learning from clas-

$$D_{U,A} = \begin{pmatrix} x_i & x_1 & x_2 & x_3 \cdots & x_p = y_i \\ x_1 & v_{x_1} & v_{x_2} & v_{x_3} \cdots & v_{x_p} \\ x_2 & v_{x_1} & v_{x_2} & v_{x_3} \cdots & v_{x_p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_n & v_{x_1} & v_{x_2} & v_{x_3} \cdots & v_{x_p} \end{pmatrix}$$
$$= \begin{pmatrix} x_i & x_1 = age & x_2 = sex & x_3 = race & x_p = y_i = disease \\ 1 & 19 & M & White \cdots & Rhinitis \\ 2 & 34 & M & Black \cdots & Rhinitis \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 6,000 & 35 & F & Black \cdots & Cancer \end{pmatrix} \quad (4)$$

sifiers. Although some work recently used the CTGAN for data augmentation, in those methods, augmentation is performed on both majority and minority classes, which can significantly increase data size without significant improvement in results. In addition, the quality of newly generated data is not improved before augmentation, which can degrade the truthfulness of the overall data. To the best of our knowledge, the noise elimination method has not been used with CTGAN-based augmentation methods. Can we offer a generic data augmentation solution that is robust against noise, adds a minimal number of records to balance distributions, and significantly enhance the performance of classifiers? To answer that question, we devised CTGAN-MOS, depicted in Fig. 3. The proposed scheme encompasses six key steps that assist in addressing the imbalance problem in real-world datasets. Concise details of each step are explained in the following subsections.

1) In the first step, data engineering techniques such as cleaning, wrangling, and quality enhancement are applied to remove basic problems from the data. With the help of data engineering techniques, a clean dataset is obtained for further processing. Technical details are in subsection IV-A.
2) In the second step, vulnerability analysis of the original dataset is performed to determine the type and nature of vulnerabilities in the data. For example, in some cases, the size of the dataset can be insufficient for training models. Similarly, the distribution of classes in the dataset can be skewed, which leads to imbalanced learning. We call such problems vulnerabilities, and devise algorithms to determine them in an automated way. Technical details are in subsection IV-B.
3) In the third step, more data are curated to compensate for deficiencies as well as to address the class skewness problem. In this work, we leverage the CTGAN model (a SOTA generative model for synthetic data generation) to curate more data of superb quality for augmentation purposes. Details on this step are in subsection IV-C.
4) We then fuse the already available data and newly curated data. Specifically, we increase the data size in the minority class only and preserve data truthfulness. Furthermore, we implement an optimization algorithm to reduce the number of records to be included in the data, whereas existing schemes simply balance the number of samples in both classes, leading to a loss in truthfulness and structural similarity. Details are in subsection IV-D.
5) Through experiments, we found that data augmentation using the CTGAN model can introduce noise, meaning some of the newly added data can lie in regions of the majority class that can subsequently decrease the performance of classifiers. To the best of our knowledge, noise removal methods have not been integrated with CTGAN-based augmentation. We apply a noise removal method to further improve the quality of fused data. Details are in subsection IV-E.

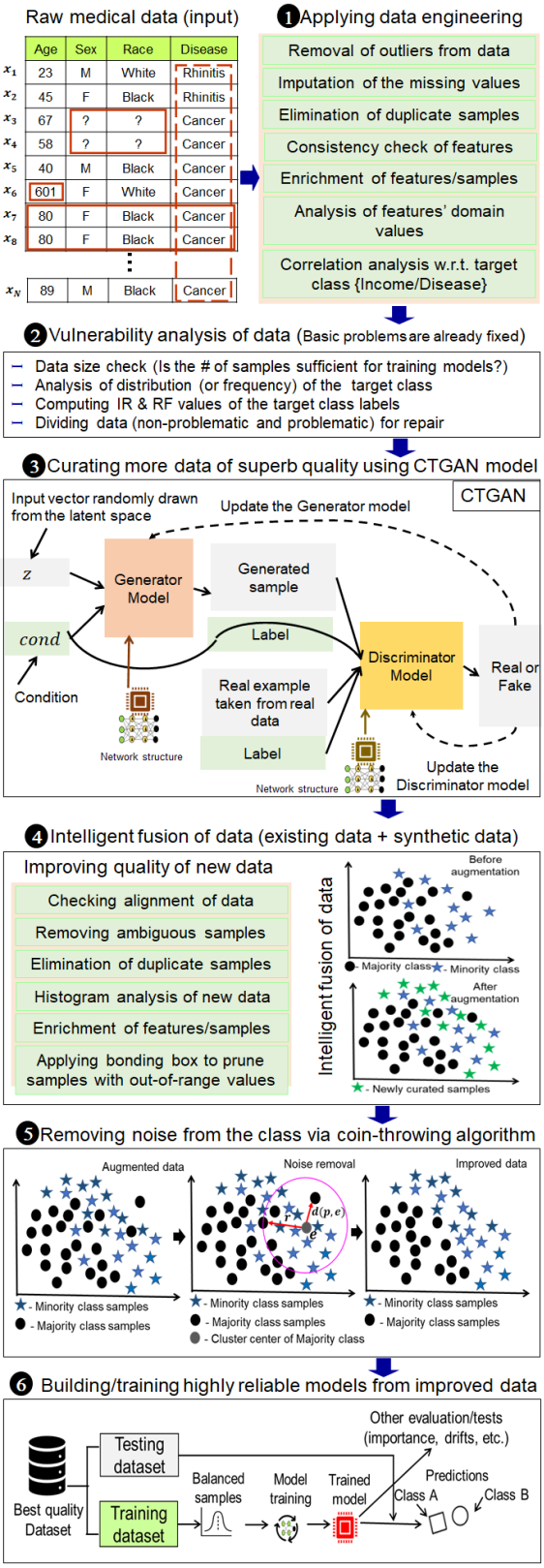


FIGURE 3. Conceptual overview of CTGAN-MOS.

6) In the last step, classifiers are built upon the improved data to verify the efficacy of CTGAN-MOS. We chose the ensemble method to verify our scheme in terms of accuracy and four other metrics. It is important to note that any ML model can be used for testing. Details are in subsection IV-F.

A. APPLICATION OF DATA ENGINEERING

In the initial step, essential data engineering techniques are used to clean the data. Specifically, problems related to outliers, missing values, duplicate records, value consistency, formats, etc., are resolved. The rigorous use of these techniques is vital to improving data quality, particularly when data are noisy, or the collection process is not reliable. In the beginning, we remove outliers from the data using a low-cost approach expressed in Algorithm 1.

Algorithm 1 Handling Outliers in D

Require: D, N
Ensure: $\mathcal{D}$, where $\mathcal{D}$ has no outliers.

$D \leftarrow 0$ ▷ initializing D as empty set.
$x_i \leftarrow \{x_1, x_2, x_3, \ldots x_n\}$
for $i = 1$ *to* n **do**
  if $(x_i == numeric)$ **then**
    $Max_{x_i} \leftarrow Max$ ▷ Acquire feasible max. value of x_i.
    $Min_{x_i} \leftarrow Min$ ▷ Acquire feasible min. value of x_i.
    for $j = 1$ *to* N **do**
      $v_{x_i}(j) \leftarrow if\,(v_{x_i} > Max_{x_i} || v_{x_i} < Min_{x_i})$
      $v_{x_i}(j) \leftarrow 0.0$ ▷ Replacing outliers with 0.0.
      $v_{x_i}(j) \leftarrow x_i \cup v_{x_i}(j)$
    end for
  else if $(x_i == discrete)$ **then**
    $U_{x_i} \leftarrow unique(D(x_i))$ ▷ Find unique values in x_i.
    $\forall v_{x_i} \in U_{x_i} \leftarrow f(v_{x_i})$
    Sort $f(v_{x_i})$ in ascending order.
    $\sigma_1 \leftarrow max(f(D_{x_i}))$ ▷ Dominant values.
    $\sigma_2 \leftarrow min(f(D_{x_i}))$ ▷ Less dominant values.
    if $(C_{x_i} == |U_{x_i}|)$ **then**
      No outliers in x_i
    else if $(C_{x_i} \neq |U_{x_i}|)$ **then**
      for $j = 1$ *to* N **do**
        $v'_{x_i}(j) \leftarrow x_i - v'_{x_i}$ ▷ Find wrong values.
        $v'_{x_i} \leftarrow \sigma_2$ ▷ Correcting wrong values.
        $x_i \leftarrow x_i \cup v'_{x_i}$
      end for
    end if
  end if
end for
While (D has no outliers in both columns) do
$\mathcal{D} \leftarrow \mathcal{D} \cup D$
End While
Return $\mathcal{D}$

In Algorithm 1, D and N are input, and $\mathcal{D}$ (where $\mathcal{D}$ has no outliers) is returned as output. We identify the outliers from a numerical column using $min-max$ analysis, whereas outliers from non-numeric columns are identified using the cardinality information of the columns. For example, if gender has cardinality 2 in the description of D given by data owners, then gender should have two unique values. If this is not the case, and the cardinality of gender is 3 (or > 2), then out-of-distribution values are replaced with feasible minority values from that column. It is worth noting that most of the existing work deletes records having outliers, which can reduce data size significantly if the number of outliers is large. In contrast, we do not delete the outliers and instead find suitable values to replace them. By doing so, data size is maintained, and an informative analysis of D can be performed. By applying both these concepts, the outliers from D are effectively handled, and the possibility of incorrect conclusions is restrained.

In the next step, issues related to missing data/values are addressed in a fine-grained manner. In the literature, there are two commonly used approaches to deal with missing values: deletion or imputation [45]. We implement the latter approach in order to maintain the size as well as the structure of the real data. We impute missing values in both numerical and categorical columns using Algorithm 2. In Algorithm 2, $\mathcal{D}$ with missing values is input, and $\mathcal{D}$ with no missing values is returned as output. We substitute the missing values or values with 0.0 in numerical columns with the mean of the respective column. The missing values in categorical columns are replaced with underrepresented values from that column. By applying Algorithm 2, all missing values, as well as outlier-related issues in numerical columns, are addressed. After these steps, the size of the data is not reduced, and the quality is significantly improved.

Next, we devised a method to filter duplicate records. A record at index i is considered a duplicate if it contains identical values of all features, including the target class to some other record located at index j in $\mathcal{D}$, and when i and j are next to each other.[1] We assume that records that are not located next to each other are not duplicates, even though they have identical values (e.g., they can be different records but with identical features). In some cases, the duplicates can also be removed using count information if emails/names are given in D. Algorithm 3 is applied to remove duplicate records from $\mathcal{D}$. In Algorithm 3, the similarity ($\mathcal{S}$) is computed among records in a pair-wise fashion, and a similarity matrix is generated. $\mathcal{S}$ between records can be calculated using Eq. 5:

$$\mathcal{S}(x_i, x_j) = \frac{\sum_{k=1}^{n} x_i \times x_j}{\sqrt{\sum_{k=1}^{n} x_i^2} \times \sqrt{\sum_{k=1}^{n} x_j^2}} \tag{5}$$

After computing $\mathcal{S}$ and storing values in $\mathcal{Z}$, duplicate records are deleted from the data by exploiting information in the similarity matrix. With the help of Algorithm 3,

[1] https://www.mysqltutorial.org/mysql-find-duplicate-values/

Algorithm 2 Handling Missing Values in D

Require: $\mathcal{D}$
Ensure: $\mathcal{D}$, where $\mathcal{D}$ has no missing values.
$\mathcal{D} \leftarrow 0$ ▷ initializing $\mathcal{D}$ as empty set.
$N \leftarrow |\mathcal{D}|$
$X \leftarrow \{x_1, x_2, x_3, \ldots x_n\}$
for $i = 1$ *to* n **do**
  if ($x_i ==$ *numeric*) **then**
    for $j = 1$ *to* N **do**
      $v_{x_i}(j) \leftarrow v_{x_i}$, where $v_{x_i} \neq 0.0 \;||\; '\;'$
      $\bar{x_i} \leftarrow \sum_1^N x_i/N$ ▷ Compute mean of values.
      Impute missing numerical values in x_i with $\bar{x_i}$
      Impute values having 0.0 in x_i with $\bar{x_i}$
      $\mathcal{D} \leftarrow \mathcal{D} \cup x_i$
    end for
  else if ($x_i ==$ *discrete*) **then**
    Impute missing discrete values via Alg. 1 (20-25).
    $\mathcal{D} \leftarrow \mathcal{D} \cup x_i$
  end if
end for
While ($\mathcal{D}$ has no missing values in both columns) do
$\mathcal{D} \leftarrow \mathcal{D} \cup D$
End While
Return $\mathcal{D}$

duplicate records are removed from $\mathcal{D}$, leading to a reduction in computing overhead.

In the next step, various built-in functions of the *R* programming language are employed to enhance the reliability of $\mathcal{D}$. To make sure that numerical columns have consistent values w.r.t. type, we use the *str* function. Similarly, we ensure that categorical columns also have consistent values. In some cases, the features/attributes can be enriched using normalization and scaling techniques. In this work, we assume data are mixed types (e.g., numeric and categorical), and therefore, are used as they are. Later, we analyze the domain values of each feature to determine data diversity. In the last step, we analyze the relationship of each feature with the target class to pay ample attention to significant features at the time of augmentation. With the help of the above-cited methods, a clean $\mathcal{D}$ is obtained for further processing.

B. VULNERABILITY ANALYSIS OF THE REAL DATASET

In this step, vulnerability analysis of $\mathcal{D}$ is performed w.r.t. distribution skew and data size by exploiting valuable knowledge enclosed in the data. In many real-world cases, the size of $\mathcal{D}$ can be very small, and $\mathcal{D}$ cannot be used in training classifiers. For example, a drug dataset[2] available at the Kaggle repository has sufficient features (e.g., age, blood pressure, sex, sodium-to-potassium ratio, cholesterol levels, and drug type or target class), but the number of records is just 200.

[2]https://www.kaggle.com/datasets/prathamtripathi/drug-classification

Algorithm 3 Removing Duplicate Records From D

Require: $\mathcal{D}$
Ensure: $\mathcal{D}$, where $\mathcal{D}$ has no duplicate records.
$\mathcal{D} \leftarrow 0$ ▷ initializing $\mathcal{D}$ as empty set.
$N \leftarrow |\mathcal{D}|$
$\mathcal{Z} \leftarrow \emptyset$ ▷ $\mathcal{Z}$ is a similarity matrix
for $i = 1$ *to* n **do**
  for $i = 1$ *to* n **do**
    $\mathcal{S} \leftarrow sim(x_i, x_j)$ ▷ Similarity computation.
  end for
  $\mathcal{Z} \leftarrow \mathcal{Z} \cup \mathcal{S}$ ▷ Similarity matrix of size $N \times N$
end for
Analyze ($\mathcal{Z}$ and find records having similarity 1 and located next to each other) do
Remove one of the duplicate records.
While ($\mathcal{D}$ has no duplicates) do
$\mathcal{D} \leftarrow \mathcal{D} \cup D$
End While
Return $\mathcal{D}$

Such a small dataset cannot be used for training classifiers, and therefore, curation of more data is necessary. In some cases, the distribution can also be imbalanced, leading to poor performance from classifiers. Thus, analyzing vulnerabilities in the data is a vital step toward enhancing the performance of classifiers. In this work, we develop an algorithm to identify the vulnerabilities in $\mathcal{D}$ to prevent performance bottlenecks in ML classifiers. In Algorithm 4, $\mathcal{D}$ is the input, and the sizes of $\mathcal{D}$ and ζ are returned as output. The critical step in this algorithm is determining T_ζ (a threshold) for deciding a reasonable data size. The value of T_ζ can be decided based on domain knowledge, or the data size needed to solve the given problem. When the data size is very small, more data can be curated as much as needed to solve the problem at hand using ML. After identifying the vulnerability w.r.t. data size, $\mathcal{D}$ is further analyzed for class imbalance.

Algorithm 4 Checking Data Size Vulnerability in $\mathcal{D}$

Require: $\mathcal{D}$
Ensure: $|D|$, where $|D|$ is an integer, and ζ, where $\zeta = 1/0$
$N \leftarrow |D|$ ▷ Determining data size.
if $N > T_\zeta$ **then**
  $\zeta \leftarrow 0$ ▷ No vulnerability related to size.
else if $N \leq T_\zeta$ **then** ▷ T_ζ (a threshold for decision)
  $\zeta \leftarrow 1$ ▷ Vulnerability related to size.
end if
Return ζ and $|\mathcal{D}|$

Algorithm 5 is the pseudocode to check for class imbalance in $\mathcal{D}$. Specifically, this algorithm computes $\mathcal{IR}$ of the available classes to determine whether a vulnerability w.r.t. class imbalance exists or not. Furthermore, size information on each class is also extracted for later use. With the help of the above algorithms, vulnerabilities related to data size

and class imbalance are determined and fixed to yield better performance from a classifier.

Algorithm 5 Checking for Class Imbalance Vulnerability in $\mathcal{D}$

Require: $\mathcal{D}$, $index[y_i]$, N
Ensure: Ω, where $\Omega = 1/0$, $|c_M|$ (size of majority class), and $|c_m|$ (size of the minority class)

$\Omega \leftarrow 0$ ▷ Assuming that no imbalance exists in $\mathcal{D}$.
$V_u \leftarrow unique(index[y_i])$ ▷ Unique values of y_i.
$\lambda_1 \leftarrow V_u[0]$ ▷ Get 1st unique value of y_i
$\lambda_2 \leftarrow V_u[1]$ ▷ Get 2nd unique value of y_i
$f_{\lambda_1} \leftarrow |\mathcal{D}|$, where $index[y_i] == V_u[0]$
$f_{\lambda_2} \leftarrow |\mathcal{D}|$, where $index[y_i] == V_u[1]$
if $(f_{\lambda_1} > f_{\lambda_2})$ **then**
    $c_M \leftarrow V_u[0]$
else if $(f_{\lambda_1} < f_{\lambda_2})$ **then**
    $c_m \leftarrow V_u[1]$
end if
$\mathcal{IR} \leftarrow$ using Eq. 1.
if $(\mathcal{IR} > 1)$ **then**
    $\Omega \leftarrow 1$
else if $(\mathcal{IR} == 1)$ **then**
    $\Omega \leftarrow 0$
end if
Return Ω, $|c_m|$, and $|c_M|$

C. CURATING MORE DATA BY USING THE CTGAN MODEL

In this step, high-quality data, denoted with D_{new}, is curated to compensate for a deficiency of data as well as to balance the distribution of classes. To curate more data, we use the CTGAN model which is a state-of-the-art synthetic data generation model [46]. We chose this model to generate D_{new} with higher diversity. Furthermore, it has a condition that enables balanced learning from $\mathcal{D}$ and ensures that all categories and their values are transformed correctly in the synthetic data. It has two neural networks called the generator (G) and the discriminator/critic (C). We use the open-source implementation of CTGAN with modifications to generate new data (the original implementation does not improve data quality before use, leading to unnecessary generation of various modes and performance overhead). The following key steps are applied while implementing the CTGAN to generate synthetic data in tabular form.

1) *Data representation and conversion*: In the first step, mixed columns are put into their respective categories, and their values are transformed. Categorical columns are represented as one-hot encoding vectors, whereas numerical columns are transformed using a variational Gaussian mixture model [15].
2) *Setting up the condition*: After representing data in a unified form, a conditional vector is specified to learn the distributions of the real data fairly. In the absence of a condition, the generator can only create samples for dominant values, leading to poor diversity in the generated data. To prevent the issue of imbalanced learning, we construct conditional vectors for categorical columns. Let v' be the value from the ith column and row r_i in a categorical column to be linked to a new curated sample denoted as r_s. In this situation, G can be regarded as a conditional distribution of r_i, which is stated formally in Eq. 6:

$$r_s \sim P_G(r_i|C_i = v') = P(r_i|C_i = v') \tag{6}$$

The use of the condition enables G to generate data of high quality. By applying the condition, original distributions can be reconstructed as given in Eq. 7:

$$P(r_i) \sum_{v \in C_i} P_G(r_i|C_i = v')P(C_i = v) \tag{7}$$

The condition is integrated with the CTGAN model via hot vectors.
3) *Designing a network structure*: To produce data of excellent quality, two fully connected networks are used in both the generator and critic. The activation function and normalization are also used in the generator. To prevent overfitting during training, optimized dropout, leaky ReLU, Adam optimizer values, etc., are used. With the help of fully connected networks, optimized parameters, and conditional vectors, a stable training process is guaranteed.
4) *Training networks and curating synthetic data*: In this step, the networks are simultaneously trained to yield data. To prevent the vanishing gradient problem and guarantee stable training, WGAN loss with a gradient penalty is used [47]. The loss weights are updated till the convergence of the models. The training process continues until the Nash equilibrium is achieved between the generator and discriminator.
5) *Data synthesis after training*: Once training is finished, the data can be gathered by applying minimal post-processing. Since the data are encoded into hot vectors and VGM models, data conversion to the original form is performed. Specifically, the ReLU is used for producing numerical columns, and Gumbel softmax is used for producing categorical columns. In the end, the rows in a simplified structure synthesize the data.

With the help of the above procedure, D_{new} is curated and can be used to address class imbalance problems. In some cases, it can be used to compensate for a deficiency of good data. In recent years, synthetic data generation tools have contributed extensively to generating data that can be used in various data-driven solutions [48], [49], [50]. In the coming years, generative AI can contribute greatly to addressing the problem of data shortages in many industrial and data-driven applications for social good.

In the beginning, both the generator and discriminator yield very poor performance because the discriminator cannot easily differentiate the real and synthetic samples [51].

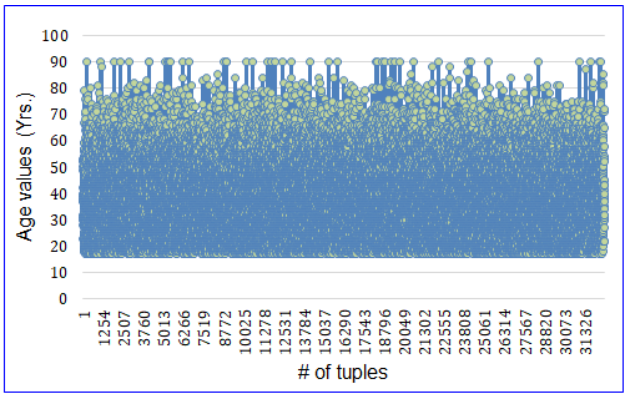
As a result, the generator cannot get desired feedback from the discriminator in the initial stages of training, leading to consistently high errors. As the training time progresses, the differences between real and synthetic samples improve the discernment ability, which in turn guides the generator to yield samples that are very similar to the real ones. The fluctuation in the error reflects the game process of the discriminator and generator, and the descent trend of the error implies that the generator can learn the distribution of the actual samples. After a reasonable iteration, the error reduced significantly, and it did not show much fluctuation. Further, the reduction in the magnitude of error does not differ much under various condition vectors, which implies the universality of the GAN-based models in data generation. To simultaneous use of the condition vector and WGAN-GP [47] assisted in achieving a narrow gap between bias and variance in the CTGAN model. The use of condition assisted in exploring all values regardless of their status (e.g., major/minor) with equal probability, and therefore, underfitting and overfitting issues were resolved, leading to lower bias and variance of CTGAN. Lastly, the training data size if sufficient, and therefore, the variance-bias trade-off was effectively resolved.

D. INTELLIGENT FUSION OF DATA

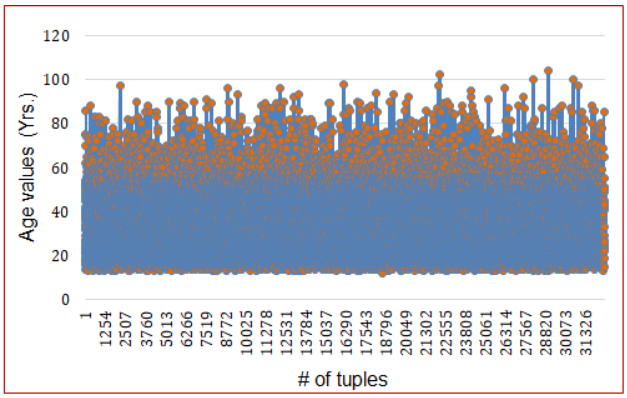
In this step, data fusion is performed intelligently, meaning fewer but good-quality records are added to the minority class of the original data. We also improve data quality before fusion to prevent truthfulness loss (none of the previous studies considered this important problem). By not improving the quality of the synthetic data, truthfulness and structural similarity losses can be very high, impacting the performance of classifiers and the correctness of the conclusions. Fig. 4 highlights the need for improving the quality of synthetic data. In this example, although the quality of synthetic data is good, some values are beyond the desirable range, compared to the original data. For example, in the real data illustrated in Fig. 4(a), none of the age values is > 90, but in the synthetic data in Fig. 4(b), some age values are ≥ 100. Therefore, rigorous improvements are needed before fusion to yield consistent results. Based on this analysis, it is fair to say that synthetic data quality enhancement before fusion is imperative. It is even mandatory when synthetic data are used in some safety-critical applications.

1) IMPROVING THE QUALITY OF SYNTHETIC DATA BEFORE FUSION

This work is a maiden attempt to enhance the synthetic data quality before fusion to prevent the possibility of wrong conclusions/inferences from the augmented data. In contrast, most of the existing DATs fuse data without paying due attention to the quality of the data generated, leading to wrong conclusions from mining results. We apply a set of sophisticated techniques to remove inconsistent samples from the data. In the first step, D_{new} is aligned with the already available D. Specifically, we only perform alignment



(a) Age distribution in real data (D).



(b) Age distribution in synthetic data (D_{new}).

FIGURE 4. Comparison of statistical information in D and D_{new}.

for c'_m (the minority class in D_{new}) rather than all classes. The pseudocode to align the numerical feature values is Algorithm 6, for which $\mathcal{D}$ and D_{new} are the input, and D'_{new} with aligned values of the numerical features is returned as output. This algorithm creates a bounding box for each numerical feature by using *Min* and *Max* information from the original data, and values that lie outside the box are removed from D_{new}. By applying this concept, values are brought into the desired range to lower the possibility of incorrect conclusions or structural similarity loss during fusion.

The values of the categorical columns are aligned using frequency information from $\mathcal{D}$. For instance, if a categorical feature has higher frequencies for some of its values, then some of the records are removed to balance the distribution without losing statistical information. We also supplement minor values with major values to increase data diversity. By aligning the values of numerical and categorical features, the possibility of propagating noise in the training data is restrained, leading to high-quality augmented data generation. We also apply a set of techniques for cleaning real data to clean the newly curated data. The application of different techniques to synthetic data significantly improves the data structure and quality.

Algorithm 6 Alignment of Numerical Features in D_{new} and $\mathcal{D}$

Require: $\mathcal{D}$, D_{new}
Ensure: D'_{new}, where D'_{new} has aligned numerical features

```
D'_new ← 0                                   ▷ Initialization with empty set.
C_m ← using D and Alg. 5                     ▷ Minority class in D.
C'_m ← using D_new and Alg. 5                ▷ Minority class in D_new.
X ← {x_1, x_2, ... x_t}
X' ← {x'_1, x'_2, ... x'_t}
for i = 1 to t do                            ▷ Removing values from Max end.
    Max(x_i) ← Max(D(x_i))
    for j = 1 to N do
        if (v'_xj > Max(x_i)) then
            D_new ← D_new − v'_xj
            D'_new ← D'_new ∪ D_new
        else if (v'_xj < Max(x_i)) then
            Do nothing
        end if
    end for
end for
for i = 1 to t do                            ▷ Removing values from Min end.
    Min(x_i) ← Min(D(x_i))
    for j = 1 to N do
        if (v'_xj < Min(x_i)) then
            D_new ← D_new − v'_xj
            D'_new ← D'_new ∪ D_new
        else if (v'_xj > Min(x_i)) then
            Do nothing
        end if
    end for
end for
Return D'_new                                ▷ Highly aligned data with D
```

2) INTELLIGENT FUSION

In the literature, data fusion is performed in an ad-hoc manner, meaning if $|c_M - c_m| = 500$, then 500 records will be curated and added to the data. In this work, we found that if the quality of newly generated data is good then the number of records to be added can be restrained, leading to a significant reduction in noise without losing guarantees on performance metrics (accuracy, precision, recall, etc.). In this example, 280 or 300 records can be sufficient to address the class imbalance problem. We call the fusion method intelligent because the new records are added only to c_m, and the number of records is also less, compared to existing SOTA DATs. To reduce the number of records to be added to the data, we formulate an optimization problem that can be solved by finding candidate solutions. The overall process has four key steps: (i) identifying important features from real data, (ii) drawing samples from D_{new} of reasonable size, (iii) calculating the entropy of important features in samples, and (iv) choosing samples having entropy of important features close to the entropy of important samples in the real data (e.g., the minority class) for augmentation. The value of

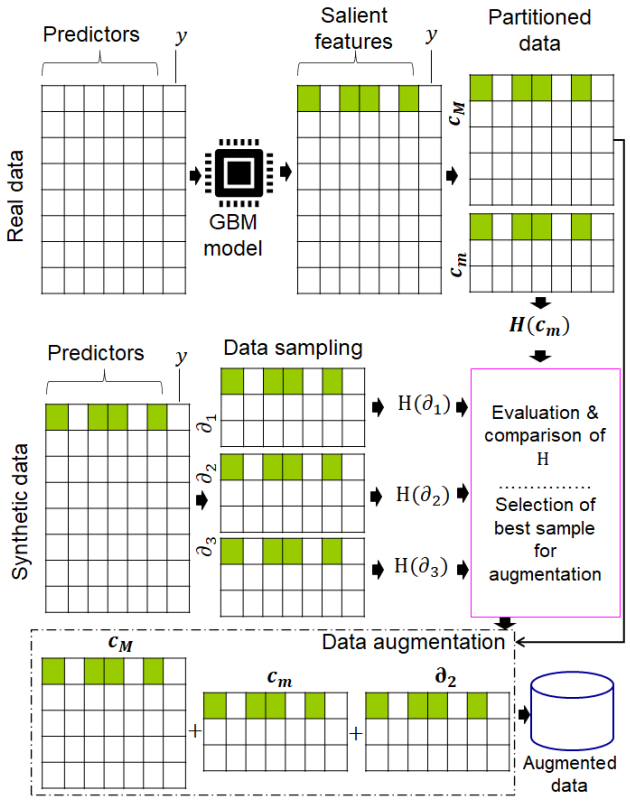


FIGURE 5. Workflow of the proposed intelligent data fusion process.

entropy H can be calculated using Eq. 8:

$$H(\zeta) = -\sum_{i=1}^{|\zeta|}[(p_i) \times ln(p_i)] \tag{8}$$

With the help of the above process, the number of records to be added to the data is lower, and quality in terms of diversity is higher. The schematic of the process employed to fuse data intelligently is depicted in Fig. 5, showing data augmentation performed considering the closeness of distribution between $\mathcal{D}$ and D_{new} in c_m. With the help of the improved c_m and fair samples, good-quality augmented data are obtained for further processing (e.g., noise removal).

E. NOISE REMOVAL FROM THE AUGMENTED DATA

After data fusion, there is a possibility that some data points may lie in unsafe regions (i.e., regions belonging to the majority class), and therefore, noise removal is imperative. To the best of our knowledge, noise removal methods have not been used with CTGAN-based DATs thus far. This paper is a maiden attempt to remove noise from CTGAN-augmented data. To remove noise from data (especially in the minority class), we employ a coin-throwing algorithm [52], [53], which can also contribute to preventing small adjuncts and class overlap problems. This algorithm amalgamates the k-means algorithm, Euclidean distance, and probability concepts in order to identify noisy samples. A sphere with radius r is drawn by using a randomly chosen majority class sample

Algorithm 7 Removal of Noisy Samples From c_m

Require: c_M, c_m
Ensure: $\mathcal{D}$, where $\mathcal{D}$ has no noisy samples in c_m.

```
D ← 0                                   ▷ Initialization with empty set.
Divide C_M into different clusters via k-means algorithm.
Store all clusters into cluster set R
for i = 1 to |R| do
    q(R_i) ← x_i ∈ c_M
    r_i(R_i) ← d(q_i, x)   ▷ x is the farthest sample from q_i.
    ∀x ∈ c_m, where R_i ⊆ c_m
    Calculate d between q_i and all other relevant samples
    Calculate p between q_i and all other relevant samples
    P ← {p_x1, p_x2, ..., p_xj}          ▷ j denotes minority samples
    P' ← {p_x1 = 0/1, ..., p_xj = 0/1}   ▷ [0 − 1] vector
    for j = 1 to |P'| do
        if (P'_i == 1) then
            c_m ← c_m − x_i
        else if (P'_i ≠ 1) then
            Do nothing
        end if
    end for
    D ← D ∪ c_m
    D ← D ∪ c_M
end for
Return D                                ▷ Final D of excellent quality
```

as the center (q). The length between any two samples or data points can be computed using Euclidean distance [54]. We assume that when $r \geq d(x, q)$, the minority samples are located within the sphere. Given an n-dimensional vector, $x = \{x_1, x_2, \ldots, x_n\}$, the distance from sphere center q, where $q = \{q_1, q_2, \ldots, q_n\}$, can be determined using Eq. 9:

$$d(x, q) = \sqrt{\sum_{i=1}^{n} (x_i - q_i)^2} \tag{9}$$

After computing the distance between the center and the minority samples located within the region of the majority class, we compute their probability of being noisy by using Eq. 10:

$$p = 1 - \frac{d(x, q)}{r} \tag{10}$$

where r and q are the radius and center of the sphere, respectively, and p denotes x's probability value. If x is located closer to q, the p value decreases, and vice versa. $\forall\ x_i \in S$ where p exhibits the form $0 \leq p \leq 1$. The higher the value of p, the higher the possibility of the x being noise. Afterward, a threshold is established to differentiate noisy samples from those that are not noisy. For the sake of simplicity, each sample is transformed to either 0 or 1 using the coin-throwing method. If $p \sim 1$, then the algorithm generates 1, and it yields 0 if $p \sim 0$. Based on the analysis of d and p, the samples located within the majority class are removed if they turn out to be noisy. With the help of three simple steps, noise is removed from the data, which can augment the performance of ML classifiers built upon them [55]. The pseudocode of the algorithm to remove noise is Algorithm 7 in which c_M, c_m are the input, and augmented data $\mathcal{D}$ with no noisy samples in c_m is returned as output. In this algorithm, clusters are

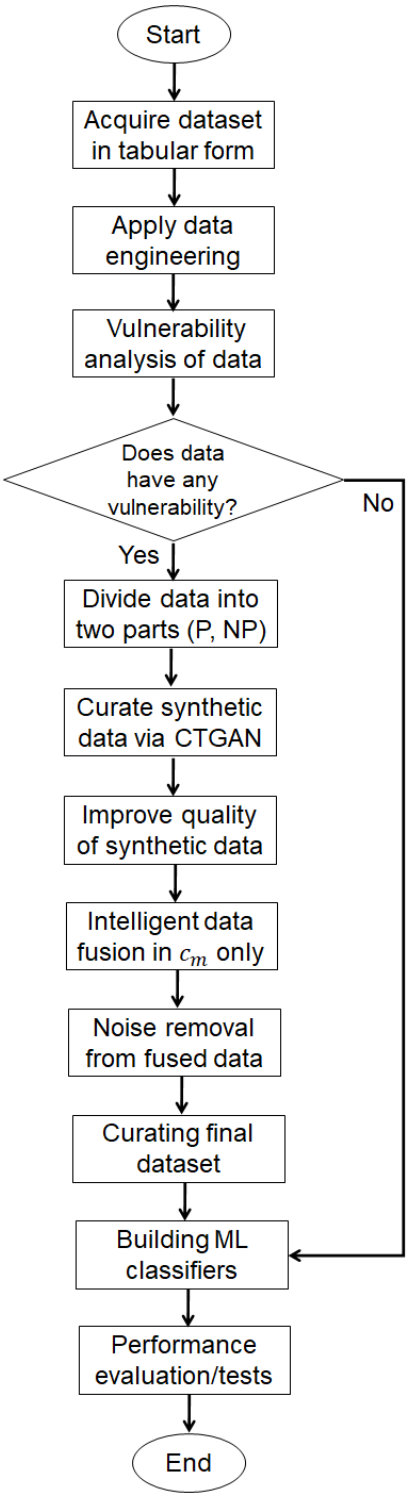


FIGURE 6. Flowchart of the proposed CTGAN-MOS (P = problematic, NP = non problematic).

derived from c_M, and each cluster is processed independently to remove noise.

F. BUILDING CLASSIFIERS WITH IMPROVED DATA

In the last step, classifiers are trained from the augmented data to yield better performance in real-world scenarios. To validate the effectiveness of the augmented data, we trained with the ensemble method (i.e., random forest). The main reason to use the ensemble method is to ensure balanced learning via fair samples drawn from the training data. We partition the data for training (d_{train}) and testing (d_{test}). In addition, we verify the sampling quality by analyzing feature distributions from the samples. We also reduce the parameter size to get similar accuracy in most tests. Experimental analysis proves that when data quality is excellent, some parameters' values can be reduced. We hope our analysis can help reduce computing overhead while building ML classifiers involving big data. The flowchart of the CTGAN-MOS along with the key steps is given in Figure 6. The workflow depicted in Figure 6 can pave the way to understanding the proposed CTGAN-MOS from a technical perspective. Details of supportive techniques used in some steps of the CTGAN-MOS are given in Figure 3.

V. EXPERIMENTAL RESULT

This section presents the output of the concepts discussed in this paper. To prove the significance of our scheme, we conducted thorough experiments on real-life datasets and compared the results with various SOTA DATs. In the next subsections, we present details of the datasets, the implementation setup, the evaluation criteria, and compare results with existing SOTA techniques by using five different metrics.

A. DESCRIPTIONS OF THE DATASETS

To evaluate the performance of the proposed CTGAN-MOS, we conducted detailed experiments on two real-life benchmark datasets: the adult dataset [56] and the stroke prediction dataset [57]. Both datasets are publicly available at UCI ML and Kaggle repositories and have been widely used for assessing the performance of ML models. Both datasets contain information about real-world persons with mixed feature types (i.e., categorical and numerical). The number of features in these datasets are 15 and 12, respectively. In addition, both datasets encompass many quality-related issues, which makes them more suitable for the evaluation of our scheme. Details of these datasets are in Table 3. Within the parentheses, C and N refer to the type of feature (categorical or numerical, respectively), and the numbers indicate cardinality.

It is worth noting that both datasets have a binary target class, and the distribution skew is very high, as shown in Table 4. In addition, there are many missing values as well as redundant records. From the results, we can see that most samples belong to <=50 K in the adult dataset. In contrast, a large number of samples in the stroke prediction dataset show no indication of a stroke.

TABLE 3. Details of the real-life datasets used in the experiments.

Feature category	Details of the datasets	
Predictors/class	Feature details of [56]	Feature details of [57]
Predictors	Gender (2, C) Age (74, N) Country (41, C) Race (5, C) Occupation (14, C) Work class (8, C) Education (16, C) Education # (16, N) Relationship (7, C) Marital status (6, C) Capital loss (118, N) Capital gain (91, N) Hours-per-week (93, N) fnlwgt (21, 648, N)	Gender (3, C) Hypertension(2, N) Work type (5, C) Ever-married (2, C) Residence type (2, C) Smoking status (4, C) Age (104, N) Heart disease (2, N) Glucose level (3,978, N) BMI (418, N) Patient id (5,510, N) - - -
Target class	Income (2, C)	Stroke (2, N)

TABLE 4. Unique values of the target class, and their frequencies in *D*.

Dataset	Total records	Value of target class	Frequency in D
Adult [56]	32,561	>50K <=50K	7, 841 24, 720
Stroke [57]	5,510	1 (stroke) 0 (No stroke)	249 5,261

To perform rigorous experiments and comparisons, we systematically applied our scheme to these datasets. It is important to note that our scheme is general, and can be applied to any dataset with minor modifications.

B. IMPLEMENTATION SETUP

The experiments were performed on a notebook with an Intel Core i5-3320M CPU with 8GB RAM clocked at 2.60GHz running Windows 10 Pro. The experiments were performed using R ver. 4.0.0 ($x64$) and RTools with the help of custom packages. An open-source CTGAN implementation with reasonable modifications was employed to curate a D_{new} of superb quality. The ensemble method was implemented with the help of two main libraries: RF and ranger[3] (a fast and customized RF code). Descriptions of some main parameters and other necessary variables, along with a snapshot of their values, are presented in Table 5. We used two different sets of values for non-augmented and augmented data during the experiments.

Apart from the details in Table 5, we used default values for some variables. For example, node size was set to 1, the value of the sample fraction was 0.8, and bootstrapping was the default sampling scheme. We found these values with the help of rigorous experiments under different conditions. Training data was about two-thirds of the records from D, and testing data comprised one-third of the records. Through detailed experiments with the RF model and minimal post-processing, we identified relationship, capital gain, marital status, occupation, and age as salient features in the adult dataset. In these features, there are many diverse values, and

[3]https://cran.r-project.org/web/packages/ranger/

TABLE 5. Parameters used for building ML classifiers on augmented data.

Parameter/variable	Values	
	Numerical (A;S)	Non-numerical
Size of training data	28,962; 5,643	-
Size of testing data	14,919; 2,910	-
ntree value	680; 358	-
Type of RF model	-	Classification (A) & Regression (S)
Splitting rule	-	Impurity
Features importance	-	True
Value of *mtry*	14;11	-
Keep forest	-	True
Predictors' label	-	All features of the respective D
Target class label	-	Target class of the respective D

Note: A = Adult dataset [56]; S = Stroke Prediction dataset [57]

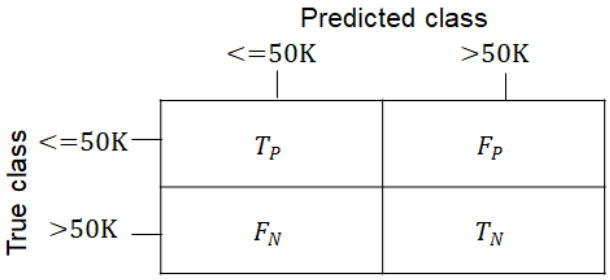


FIGURE 7. Overview of the confusion matrix structure. T_P = true positive, F_P=false positive, T_n=true negative, F_n=false negative.

instances with just one value were very few. In the stroke prediction dataset, age, BMI, glucose level, work type, and gender were identified as the most influential features. (One of those values did not occur with relatively high frequency.) Although gender usually has two values, this dataset has three distinct values for gender, which is thus regarded as a salient feature. Our experimental analysis encompassed the following three approaches.

1) Experiments with real data: We used real data without applying any modifications.
2) Experiments with augmented data (CTGAN-MOS): We increased the data size by applying the proposed scheme. Afterward, we computed and compared the performance against existing DATs.
3) Experiments with augmented data (existing SOTA DATs): We increased the data size by applying five existing data augmentation techniques. Afterward, we computed and compared the performance of each DAT against our scheme.

C. EVALUATION METRICS

To gauge the performance of our scheme, we used six different evaluation metrics: accuracy (*Acc*), precision (*Pre*), recall (*Rec*), specificity (*Spe*), F_1 score, and G-mean score. We obtained confusion matrices and analyzed them to determine the values of these metrics. The confusion matrix provides information on misclassification as well as correct classification/identification. It has four compartments denoted T_p, F_p, T_N, and F_N, for true positive, false positive, true negative, and false negative, respectively. In Fig. 7, we provide a concrete overview of these four confusion matrix parameters from the adult dataset.

Examples of the parameters in the confusion matrix are given below.

1) T_p: Income value <=50K marked as <=50K.
2) F_p: True income value <=50K marked as >50K.
3) T_N: Income value >50K marked as >50K.
4) F_p: True income value >50K marked as <=50K.

The formalization of evaluation metrics derived from the above four parameters is expressed in Eqs. 11-16.

$$Acc = \frac{T_p + T_N}{T_p + F_p + T_N + F_N} \tag{11}$$

$$Pre = \frac{T_p}{T_p + F_p} \tag{12}$$

$$Rec = \frac{T_p}{T_p + F_N} \tag{13}$$

$$Spe = \frac{T_N}{T_N + F_P} \tag{14}$$

$$F_1 = 2.\frac{Pre \times Rec}{Pre + Rec} \tag{15}$$

$$G - mean = \sqrt{Spe \times Rec} \tag{16}$$

We computed the values of each evaluation metric in three experimental settings (with real data, with our augmented data, and with other augmented data), and compared the results with existing DATs. Apart from these metrics, we also made some general comparisons, such as the number of records to be used in augmentation and the balancedness of the confusion matrices with existing DATs.

D. BASELINE DATs

To verify the superiority of our scheme over its peers, we compared the results with the recently developed SOTA oversampling DATs, namely, random oversampling (ROS) [41], FW-SMOTE [42], k-means SMOTE [43], Fair-SMOTE [44], and CTGANSamp SMOTE [15]. All these techniques are data-level methods used to address the class imbalance problem. ROS balances the number of samples by randomly adding more records until the number of samples in both classes is equal. FW-SMOTE improves the neighborhood selection procedure of the traditional SMOTE and yields better results than SMOTE. The k-means SMOTE scheme balances the distributions of classes by dividing data into clusters of size k. Fair-SMOTE improves the distribution of some features of the target class. However, Fair-SMOTE only leads to marginal improvement in the results. CTGANSamp SMOTE is the latest technique that employs a CTGAN to curate more data for balancing the class distribution. However, it does not improve the quality of newly curated data, which may lead to poor separability of classes as well as marginal improvements in results. Furthermore, it adds records to both majority and minority classes, which may lead to performance bottlenecks when the data size is large. Our proposed approach addresses the limitations of the

existing DATs and significantly improves the results without losing truthfulness in the data.

E. NUMERICAL COMPARISONS WITH SOTA DATs

In this section, we compare our scheme with existing techniques by using different evaluation metrics. We used real and augmented data to evaluate and compare the performance of our scheme with existing SOTA techniques.

1) ACCURACY COMPARISONS

In the first set of experiments, we computed and compared the performance of our scheme from the perspective of *Acc*. In these experiments, we used the ensemble method on the augmented data generated with the help of the different DATs, including the original data. Specifically, we divided the data for training and testing and developed the models. In the end, we analyzed the confusion matrix and determined *Acc* using the four parameters (T_p, F_p, T_n, F_n) as arranged in Eq. 11. Fig. 8 presents the comparisons of *Acc* values from our scheme against the existing DATs and the original data. From the results, note that our scheme outperformed the recent DATs. The *Acc* values from our scheme are close to 100% for both datasets. Our scheme yielded an accuracy of 100% on the adult dataset (e.g., $F_n = 0$ and $F_p = 0$), which makes it more suitable for safety-critical applications. In contrast, accuracy with the stroke dataset (99.83%) was only marginally lower than 100% with $F_p = 1$ and $F_n = 4$. Based on these results, it can be concluded that wrong predictions were very low, and the confusion matrix was more balanced compared to the previous techniques.

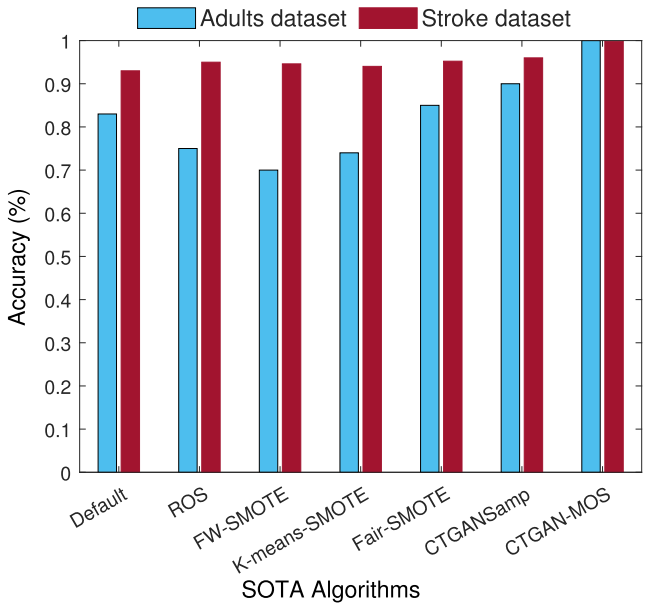


FIGURE 8. *Acc* comparisons: our scheme versus SOTA DATs and real data.

These results verify the significance of our scheme in terms of achieving better results from the perspective of data mining and pattern extraction.

TABLE 6. Confusion matrix balance comparisons: our scheme versus SOTA DATs.

	Details of the confusion matrix			
SOTA Algorithms	T_p	F_p	T_n	F_n
Default (Adult)	8,008	553	1,778	1,058
Default (Stroke)	1,690	5	3	91
ROS (Adult)	4,550	1,651	2,750	896
ROS (Stroke)	961	7	25	48
FW-SMOTE (Adult)	6,470	2,481	3,554	1,892
FW-SMOTE (Stroke)	965	3	2	52
k-mean SMOTE (Adult)	13,690	2,871	6,554	4,282
k-mean SMOTE (Stroke)	8,141	378	21	140
Fair SMOTE (Adult)	8,208	353	1,618	1,227
Fair SMOTE (Stroke)	957	11	31	42
CTGANSamp (Adult)	7,999	1,100	6,610	449
CTGANSamp (Stroke)	2,230	9	1,480	130
CTGAN-MOS (Adult)	8,699	0	6,659	0
CTGAN-MOS (Stroke)	1,749	1	1,240	4

The proposed scheme rigorously improves data quality, and therefore, the results are more accurate and highly reliable. Table 6 compares the composition of each confusion matrix in terms of balance. From the results, note that each technique yielded different values for T_p, F_p, T_n, and F_n. However, the confusion matrix from our scheme is more balanced than the SOTA techniques. It is worth noting that *Acc* values from some of the other schemes are high ($\geq 95\%$) with the stroke dataset, but the confusion matrices are not balanced, leading to poor classification/predictions in the minority class. The comparisons of *Acc* and balance in the confusion matrix prove the superiority of our scheme over existing SOTA techniques.

2) PRECISION AND RECALL COMPARISONS

In the second set of experiments, we computed and compared the precision and recall values with the existing SOTA DATs. The precision metric determines how frequently the ML model correctly predicts a positive attribute. In our experiments, the positive attributes are <= 50 K for the adult dataset, and 0 (no probability of stroke) in the stroke prediction dataset. For the recall metric, the proportion of correct predictions for positive attributes was analyzed. Both these metrics have been widely used to assess the performance of ML models. These metrics capture more information concerning the performance of ML models than *Acc* alone. We performed repetitive experiments and analyzed the values of relevant parameters to compute *Pre* and *Rec*. Subsequently, we compared the results with the existing SOTA augmentation techniques. Table 7 presents the results for *Pre* and *Rec* and comparisons with existing techniques.

From the results in Table 7, most schemes yielded better values for *Pre* and *Rec*. Through fair analysis and comparisons, our scheme yielded much higher values for both *Pre* and *Rec* than other schemes. These results verify the significance of the proposed scheme from the perspective of accurate predictions when applying ML models to noisy data. These results prove our scheme is a better choice when highly accurate predictions are desirable.

TABLE 7. *Pre* and *Rec* comparisons: our scheme versus previous SOTA DATs.

	Precision		Recall	
DAT	Adult [56]	Stroke [57]	Adult [56]	Stroke [57]
Default	0.9354	0.9970	0.8833	0.9489
ROS	0.7337	0.9927	0.8354	0.9524
FW-SMOTE	0.7228	0.9969	0.7737	0.9488
k-mean SMOTE	0.8266	0.9556	0.7617	0.9830
Fair SMOTE	0.9587	0.9886	0.8699	0.9579
CTGANSamp	0.8791	0.9959	0.9468	0.9449
CTGAN-MOS	1.0000	0.9988	1.0000	0.9977

3) F_1 SCORE COMPARISONS

In the third set of experiments, we computed and compared the F_1 score with existing SOTA DATs. F_1 score is basically the harmonic mean of *Pre* and *Rec*. In other words, it is the reciprocal of the arithmetic mean of two parameters. Since it takes into account both false positives and false negatives, it is more suitable for the evaluation of an ML model's performance. An F_1 score close to 1 is regarded as ideal. Table 8 presents the results and comparisons of F_1 scores from the existing SOTA DATs. The F_1 score for most techniques is fine for the adult dataset. With the stroke dataset, by contrast, contributions to F_1 score in the other techniques come mostly from the majority class only, leading to higher misclassification for some minority classes in real-world scenarios. Through comparison and analysis, we found our scheme had a higher F_1 score compared to most techniques when classifying both datasets. These results verify the significance of our scheme in terms of better performance from the F_1 score metric. In some applications (e.g., medical diagnosis, fraud detection, email classification), a higher F_1 score is desirable, and therefore, the proposed scheme is ideal for such scenarios.

TABLE 8. F_1 score: our scheme versus previous SOTA DATs.

SOTA DATs	Adults dataset [56]	Stroke dataset [57]
Default	0.9086	0.9724
ROS	0.7813	0.9721
FW-SMOTE	0.7474	0.9723
k-mean SMOTE	0.7928	0.9691
Fair SMOTE	0.9121	0.9730
CTGANSamp	0.9117	0.9697
CTGAN-MOS	1.0000	0.9982

4) *G*-MEAN SCORE COMPARISONS

In the fourth set of experiments, we compared the performance of our scheme against the existing techniques by using *G*-mean score, which is the geometric mean of *Spe* and *Rec*. *G*-mean score has been widely used to measure the performance of ML models in imbalanced learning problems [58]. In simple terms, to maximize the *G*-mean we minimize the total training errors of each target class during imbalanced data learning. If a *G*-mean value for any given model is higher, the model is regarded better in terms of performance. To calculate the *G*-mean score, we first compute the *Spe* value by utilizing information from the confusion matrix; the corresponding results are in Table 9.

TABLE 9. *Spe* comparisons: our scheme versus previous SOTA DATs.

SOTA DATs	Adults dataset [56]	Stroke dataset [57]
Default	0.7627	0.0375
ROS	0.6248	0.7812
FW-SMOTE	0.5888	0.40000
k-mean SMOTE	0.6953	0.0520
Fair SMOTE	0.8209	0.7380
CTGANSamp	0.8573	0.9939
CTGAN-MOS	1.0000	0.9983

From Table 9, we can see that *Spe* for some of the existing DATs was lower, and therefore, classifier performance is substandard. In contrast, our scheme yielded better values for *Spe*, and correct predictions for both classes were higher than the other techniques.

By leveraging the values of *Spe* and *Rec*, we computed the *G*-mean score, and the results are illustrated in Fig. 9. Default is the *G*-mean when the data used are without augmentation. For the stroke dataset, the misclassification rate for the minority class was extremely high, and therefore, the *G*-mean score is very low compared to the adult dataset. The stroke dataset was highly skewed, and only three instances existed in the minority class, so most contributions to *Acc* come from the majority class. Therefore, good-quality augmentation schemes are vital.

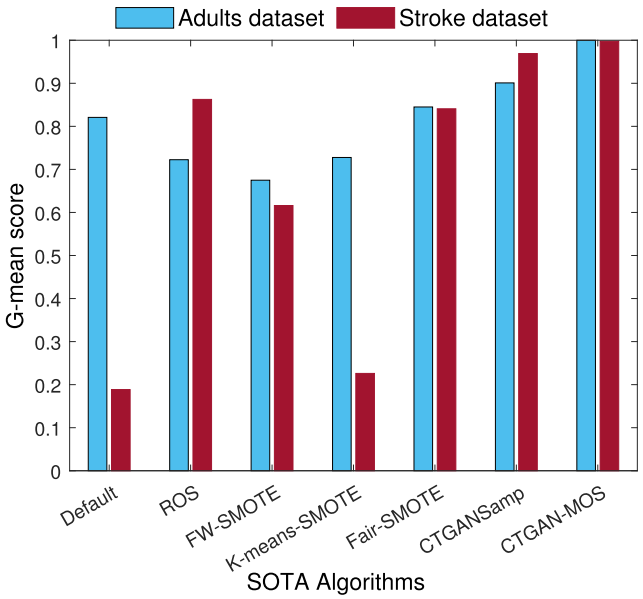


FIGURE 9. *G*-mean scores: our scheme versus SOTA DATs and real data.

Similarly, the *G*-mean score of the k-means-SMOTE scheme was also low owing to local balancing in each cluster and the introduction of noise. Since our scheme improves the quality of both real and synthetic data and removes noise using the coin-throwing algorithm, the *G*-mean scores are therefore very high with both datasets,

compared to the other techniques and the original data. These results validate the effectiveness of our scheme in terms of achieving a higher *G*-mean, leading to wide use in realistic scenarios when a higher *G*-mean value is expected.

5) NUMBERS OF NEWLY ADDED RECORDS

In the last set of experiments, we evaluated the effectiveness of our scheme in terms of the number of records used in augmentation. Specifically, we determined and compared the number of records added by our approach and by the other techniques during data augmentation. In the proposed scheme, the quality of data is meticulously enhanced, and therefore, small but good-quality records help the cause. In contrast, existing schemes do not improve the quality of data (in particular newly curated data), and therefore, more records are added. Although adding more records is handy to increase training data size, in some cases, the addition of more records marginally improves the results, but greatly increases computing overhead. How to compensate for the deficiency of data without losing guarantees on various metrics is a challenging task. In our scheme, we intend to lower the number of records during data augmentation while sustaining the values of most metrics. Some DATs add records to the majority class as well [15]. This can be handy when the data size is very small. In large datasets, adding even 2% more records to the majority class can introduce performance bottlenecks. To address this, our scheme provides a new perspective and performs augmentation with the least number of records possible. Fig. 10 presents a comparative analysis of the records added to datasets by the proposed scheme and existing schemes.

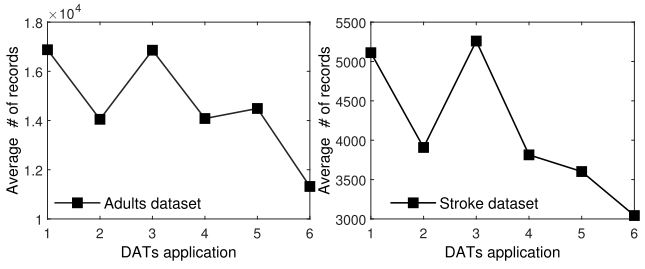


FIGURE 10. Number of records added. 1 = ROS, 2 = FW-SMOTE, 3 = *k*-means-SMOTE, 4 = Fair SMOTE, 5 = CTGANSamp, 6 = CTGAN-MOS (ours).

From the results, we can see that the proposed scheme adds fewer records than the existing techniques without losing guarantees on the performance objectives. These results verify the significance of our scheme in terms of robustness (i.e., adding fewer records to the data while yielding higher values for *Acc*, *Pre*, *Rec*, *Spe*, F_1 score, and *G*-mean score). Table 10 presents a holistic overview of the data sizes used in the experiments.

From Table 10, we can see that our scheme has a smaller data size than the other schemes. By adding fewer records (but of good quality), the possibility of adding noise is reduced, and classifier performance can be enhanced. Furthermore, our scheme removes noisy samples from the data, and therefore, the possibility of advanced problems such as a small adjunct or class overlap is also lower. Based on these results, we conclude that the proposed scheme can vastly contribute to augmenting the performance of ML classifiers, and can be used in safety-critical applications.

TABLE 10. Data size comparisons: our scheme versus previous SOTA DATs.

DAT	Adults dataset [56]	Stroke dataset [57]
Default	32,561	5,510
ROS	32,561 + 16,878	5,510 + 5,112
FW-SMOTE	32,561 + 14,050	5,510 + 3,908
k-mean SMOTE	32,561 + 16,859	5,510 + 5,260
Fair SMOTE	32,561 + 14,082	5,510 + 3,814
CTGANSamp	32,561 + 14,490	5,510 + 3,602
CTGAN-MOS	32,561 + 11,320	5,510 + 3,043

VI. DISCUSSION

In this section, we explain various key aspects in terms of significance, scope, novelty, etc. related to CTGAN-MOS.

Significance of the proposed scheme to the field of study: Our paper aimed to solve the class imbalance problem that is very common in ML applications by proposing a set of sophisticated operations that might serve as a baseline for future research. In the recent past, much of the work in data augmentation investigated adding more records without paying due attention to the quality of data. We affirm the idea that adding more records is an intuitive concept to augment the performance of classifiers. However, our work takes the opposite approach. We introduce six sophisticated operations, the result of asking the basic question, *how can we perform augmentation with as few records as possible while significantly enhancing the performance and generalization of ML classifiers on diverse evaluation matrices?* We solved the problem by identifying vulnerabilities in data that might hurt the performance of the ML models and curated additional data of high quality for augmentation. We also introduced new concepts that prevent the reduction of data size by handling outliers and missing data whereas existing work mostly removes such parts, which can lower the possibility of informed decisions. We applied noise removal techniques and improved the alignment of data before fusion for the first time in GAN-based augmentation. The sequential application of six steps significantly improved the performance of classifiers and accuracy reached optimal limits in both datasets. The numerical results determined through various tests proved the technical significance and validity of the proposed work. We uncovered the structure of the confusion matrix which can pose serious risks in high-stakes ML applications when underlying data used in training is skewed. Our solution is generic and can be applied to similar data modalities in multiple domains. We expect that our ideas and the corresponding solution might inspire new techniques for task-specific or universal data augmentation in the big data era.

Through experiments, we found that ML classifiers trained with imbalanced data cannot give equal importance to each class during inference, and the misclassification rate for some minority classes is always higher than for majority classes. The main reason for this problem is that the classifier only explores and learns features related to the majority class during the learning/training process. However, the CTGAN-MOS addresses this problem by increasing the representation of the minority class in the data, which makes some highly informative but difficult-to-learn features likely to be perceived during the training process [59]. In simple words, GAN-powered augmentation can change the importance score of some features and affect the learning dynamics of the respective classifiers, and therefore, the classifiers trained with augmented data can give equal importance (e.g., confusion matrix becomes more balanced) to each class during inference time.

Objectives of the paper: There are six main objectives of the paper: (i) To extend the applicability of the ML techniques to noisy, messy, and imbalanced datasets which are very common in realistic scenarios nowadays, (ii) To uncover the invisible problems with the confusion matrix in imbalanced learning context where even an extremely high accuracy can be problemtic becuase most contribution in accuracy can come from one class (e.g., majority class) while other classes do not contribute or minimally contribute, (iii) To yield better performance on diverse evaluation metrics (e.g., accuracy, precision, recall, F_1 score, and G-mean score.) by applying minimal changes to data (e.g., adding as minimal records as possible), (iv) To identify and fix the hidden risks (e.g., class overlap, small adjunct, noise, etc.) associated with data augmentation techniques which often remain undetected, (v) To improve the learning dynamics/ability of ML classifiers and to make features learning easier in ML classifiers, and (vi) To prevent negative consequences of ML classifiers in high stake applications (e.g., medical diagnosis, resource planning, loan granting, etc.) where binary outcomes are involved.

Scope of the work: The augmentation scheme proposed and implemented in this work can be applied to any dataset encompassed in tabular modality. Specifically, it can process datasets of mixed attribute type without any restriction on # of records and features. Furthermore, our scheme can process datasets that can have only one type of feature e.g., numerical or categorical. It was primarily proposed for binary class datasets in which the possibility of imbalance is very high. However, it can be applied to multi-class problems with minor modifications. It can work with both types of synthetic data generated with either generative AI tools (e.g., GANs) or statistical methods. The augmented data curated with the proposed scheme can be used in any supervised learning ML model for both classification and regression tasks. Also, our scheme proposed sophisticated pre-processing techniques that can be widely applied to tabular data used in unsupervised learning tasks (e.g., clustering) as well. It can also reduce the complexity of building ML classifiers as only minimal records are added to data at augmentation time, leading to applicability in resource-constrained environments. Lastly, our scheme can contribute to balancing data which can be used for general data ming tasks where diversity, equity, and inclusion are imperative.

Novel aspects of the proposed work: The novel aspects of the proposed work are explained below.

1) Maintaining the size of the real data by applying sophisticated data engineering techniques rather than deleting problematic parts of data. In an adult dataset, there are 38.24% records with null values, and most of the existing methods often delete records with such values [60]. However, deleting such a large amount of data not only reduces data size but also introduces many other problems like deleting a minor population's data. In this work, ample attention is paid to repairing the problematic portion of data rather than deleting it.
2) Adaptive augmentation rather than fixed ones. In the literature, augmentation is performed in a fixed manner which can lead to performance bottlenecks while minimally improving the accuracy and other matrices results. For example, if $|c_M| = 100$ and $|c_m| = 10$, then 90 more records will be added to balance c_m. However, we experimentally proved that if synthetic data has better quality and diversity, then 50/60 more records can do the same job as 90 records. Hence, this is the novel aspect of our proposed scheme.
3) The existing approaches add more records using GAN models and somehow yield better accuracy. However, through experiments, we observed that augmentation techniques can only compensate for the deficiency of data, leading to other severe problems such as noise, class overlap, and small adjunct problems in some cases. Unfortunately, the quality of the augmented data is rarely inspected from the perspective of noise and alignment. To the best of our knowledge, this work makes a maiden attempt to remove noise and align synthetic data before fusion in CTGAN-based augmentation.
4) Our scheme improved the learning ability of ML classifiers to the maximal level via balanced sampling, and optimal accuracy limits (i.e., ~ 100 %) were achieved with significantly reduced model size (e.g., less # of trees) than SOTA techniques. To the best of the authors' knowledge, our scheme yielded significantly higher performance than most recent augmentation techniques. This work also lowers the complications in the CTGAN model by pre-processing the real data which can prevent the unnecessary mode initialization and packing/unpacking of incomplete data while training the CTGAN model.
5) This work amalgamates data-centric techniques with the ML classifiers which is a hot research trend in the

big data and AI era [61]. Lastly, we analyzed the black box nature of the confusion matrix and ensured equal contribution from c_M and c_m to accuracy as opposed to existing practice where most contributions to accuracy come from c_M only.

The proposed approach yielded better results than its peers owing to the following enhancements: (i) sophisticated pre-processing of data, (ii) curating more data of good quality, (iii) intelligent fusion of data (minority class only, and in safe regions), and (iv) noise removal from the augmented data. The proposed scheme can be used in realistic scenarios when either good data are not available to scale, or the quality of the existing data is poor. Our approach can be used in medical applications involving binary outcomes (e.g., disease resistance or not). It can also be used in financial applications (classification of legitimate and fraudulent transactions) to prevent financial loss. Lastly, our scheme is a vital step towards the realization of responsible data science[4] where the aim is to provide accurate and fair decisions. Furthermore, our scheme can be used in scenarios where diversity, inclusion, and equity are imperative.

The proposed CTGAN-MOS can enhance the overall problem in five different ways. Foremost, it provides a new perspective to inspect and improve synthetic data quality before fusion which is rarely inspected/improved in the recent techniques. By not improving the synthetic data quality before fusion, the conclusion/results quality can have a higher deviation from real data. Second, it provides a new insight related to the composition of test and train data, which are imperative for ML classifiers' robustness. Specifically, CTGAN-MOS yields better performance with unseen data of varied sizes and compositions whereas existing methods often verify performance by using one fixed-size test data. Third, it introduces a new concept of vertical contractions (e.g., fewer record addition) to significantly reduce the performance bottlenecks contrary to the common concept of feature selection (e.g., horizontal contractions) applied in many AI applications. Fourth, it suggests a new method for fixing missing values under categorical features with under-representative values which have not been previously explored and are vital to reduce bias issues in real scenarios. Lastly, it suggests ensuring balancedness in the confusion matrix from all classes which is often overlooked as most preference is given to accuracy value only, leading to poor generalization/inference in high-stakes applications.

VII. CONCLUSION AND FUTURE WORK

In this paper, CTGAN-MOS is introduced to address the class imbalance problem for ML classifiers involving binary datasets. The proposed scheme amalgamates basic as well as advanced data engineering techniques to augment poor-quality data so they no longer create performance bottlenecks when training ML classifiers. CTGAN-MOS encompasses six key steps that are applied sequentially to produce data that are error-free, complete, dependable, and representative of the problem under investigation. To the best of our knowledge, this is the first pipeline that addresses the class imbalance problem with fewer (but good-quality) samples and that removes noise from CTGAN-powered augmentation, leading to better performance than its peers. The experiments were conducted to prove the validity of our scheme in realistic scenarios using real-life benchmark datasets. The experiment results and comparisons indicate better performance from our scheme from six different aspects compared to other SOTA DATs. In the future, we intend to apply our scheme to class imbalance problems involving multiple classes. We also intend to test the efficacy of our scheme using multiple classifiers. Lastly, we intend to optimize the synthetic data generation process to curate data only for the problematic portions of the original data. Finally, we intend to integrate a feature selection strategy with our proposal to prune the less important features from the data to enhance the accuracy of classifiers and to lower computing overhead.

[4]https://redasci.org/

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

REFERENCES

[1] G. Mathur, A. Pandey, and S. Goyal, "Applications of machine learning in healthcare," in *The Internet of Medical Things (IoMT) and Telemedicine Frameworks and Applications*. Pennsylvania, PA, USA: IGI Global, 2023, pp. 177–195.

[2] G. Lv, S. Guo, D. Chen, H. Feng, K. Zhang, Y. Liu, and W. Feng, "Laser ultrasonics and machine learning for automatic defect detection in metallic components," *NDT E Int.*, vol. 133, Jan. 2023, Art. no. 102752.

[3] J. K. Afriyie, K. Tawiah, W. A. Pels, S. Addai-Henne, H. A. Dwamena, E. O. Owiredu, S. A. Ayeh, and J. Eshun, "A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions," *Decis. Anal. J.*, vol. 6, Mar. 2023, Art. no. 100163.

[4] Y.-J. Zhai, Y. Zhang, H.-Z. Liu, and Z.-R. Zhang, "Multi-angle support vector survival analysis with neural tangent kernel study," *Arabian J. Sci. Eng.*, vol. 48, pp. 10267–10284, Jan. 2023.

[5] A. Salim, Juliandry, L. Raymond, and J. V. Moniaga, "General pattern recognition using machine learning in the cloud," *Proc. Comput. Sci.*, vol. 216, pp. 565–570, Jan. 2023.

[6] R. Jiao, C. Li, G. Xun, T. Zhang, B. B. Gupta, and G. Yan, "A context-aware multi-event identification method for nonintrusive load monitoring," *IEEE Trans. Consum. Electron.*, vol. 69, no. 2, pp. 194–204, May 2023.

[7] A. N. Wilson, K. A. Gupta, B. H. Koduru, A. Kumar, A. Jha, and L. R. Cenkeramaddi, "Recent advances in thermal imaging and its applications using machine learning: A review," *IEEE Sensors J.*, vol. 23, no. 4, pp. 3395–3407, Feb. 2023.

[8] F. Kamalov, A. K. Cherukuri, H. Sulieman, F. Thabtah, and A. Hossain, "Machine learning applications for COVID-19: A state-of-the-art review," *Data Sci. Genomics*, pp. 277–289, Jan. 2023, doi: 10.1016/B978-0-323-98352-5.00010-0.

[9] A. Farshidvard, F. Hooshmand, and S. A. MirHassani, "A novel two-phase clustering-based under-sampling method for imbalanced classification problems," *Exp. Syst. Appl.*, vol. 213, Mar. 2023, Art. no. 119003.

[10] Q. Dai, J.-W. Liu, and J.-P. Yang, "SWSEL: Sliding window-based selective ensemble learning for class-imbalance problems," *Eng. Appl. Artif. Intell.*, vol. 121, May 2023, Art. no. 105959.

[11] O. Habibi, M. Chemmakha, and M. Lazaar, "Imbalanced tabular data modelization using CTGAN and machine learning to improve IoT botnet attacks detection," *Eng. Appl. Artif. Intell.*, vol. 118, Feb. 2023, Art. no. 105669.

[12] Y. Zhang, G. Wang, X. Huang, and W. Ding, "TSK fuzzy system fusion at sensitivity-ensemble-level for imbalanced data classification," *Inf. Fusion*, vol. 92, pp. 350–362, Apr. 2023.

[13] Q. Dai, J.-W. Liu, and Y.-H. Shi, "Class-overlap undersampling based on Schur decomposition for class-imbalance problems," *Exp. Syst. Appl.*, vol. 221, Jul. 2023, Art. no. 119735.

[14] T. Kumar, A. Mileo, R. Brennan, and M. Bendechache, "Image data augmentation approaches: A comprehensive survey and future directions," 2023, *arXiv:2301.02830*.

[15] A. S. Dina, A. B. Siddique, and D. Manivannan, "Effect of balancing data using synthetic data on the performance of machine learning classifiers for intrusion detection in computer networks," *IEEE Access*, vol. 10, pp. 96731–96747, 2022.

[16] N. A. Azhar, M. S. Mohd Pozi, A. Mohamed Din, and A. Jatowt, "An investigation of SMOTE based methods for imbalanced datasets with data complexity analysis," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 7, pp. 6651–6672, Jul. 2023.

[17] B. Mirza, W. Wang, J. Wang, H. Choi, N. C. Chung, and P. Ping, "Machine learning and integrative analysis of biomedical big data," *Genes*, vol. 10, no. 2, p. 87, Jan. 2019.

[18] P. Vuttipittayamongkol, E. Elyan, A. Petrovski, and C. Jayne, "Overlap-based undersampling for improving imbalanced data classification," in *Intelligent Data Engineering and Automated Learning—IDEAL*. Madrid, Spain: Springer, 2018, pp. 689–697.

[19] Z. Huang, X. Gao, W. Chen, Y. Cheng, B. Xue, Z. Meng, G. Zhang, and S. Fu, "An imbalanced binary classification method via space mapping using normalizing flows with class discrepancy constraints," *Inf. Sci.*, vol. 623, pp. 493–523, Apr. 2023.

[20] S. Fu, Y. Tian, J. Tang, and X. Liu, "Cost-sensitive learning with modified stein loss function," *Neurocomputing*, vol. 525, pp. 57–75, Mar. 2023.

[21] Y. Song, Y. Wang, X. Ye, R. Zaretzki, and C. Liu, "Loan default prediction using a credit rating-specific and multi-objective ensemble learning scheme," *Inf. Sci.*, vol. 629, pp. 599–617, Jun. 2023.

[22] S. B. M. Gyanchandani, R. Wadhvani, and S. Shukla, "Data complexity-based dynamic ensembling of SVMs in classification," *Exp. Syst. Appl.*, vol. 216, Apr. 2023, Art. no. 119437.

[23] V. S. Spelmen and R. Porkodi, "A review on handling imbalanced data," in *Proc. Int. Conf. Current Trends towards Converging Technol. (ICCTCT)*, Mar. 2018, pp. 1–11.

[24] K. Cheng, C. Zhang, H. Yu, X. Yang, H. Zou, and S. Gao, "Grouped SMOTE with noise filtering mechanism for classifying imbalanced data," *IEEE Access*, vol. 7, pp. 170668–170681, 2019.

[25] S. S. Mullick, S. Datta, and S. Das, "Generative adversarial minority oversampling," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1695–1704.

[26] X. Wang, B. Yu, A. Ma, C. Chen, B. Liu, and Q. Ma, "Protein–protein interaction sites prediction by ensemble random forests with synthetic minority oversampling technique," *Bioinformatics*, vol. 35, no. 14, pp. 2395–2402, Jul. 2019.

[27] X. W. Liang, A. P. Jiang, T. Li, Y. Y. Xue, and G. T. Wang, "LR-SMOTE—An improved unbalanced data set oversampling based on K-means and SVM," *Knowl.-Based Syst.*, vol. 196, May 2020, Art. no. 105845.

[28] G. Douzas and F. Bacao, "Geometric SMOTE a geometrically enhanced drop-in replacement for SMOTE," *Inf. Sci.*, vol. 501, pp. 118–135, Oct. 2019.

[29] L. Camacho, G. Douzas, and F. Bacao, "Geometric SMOTE for regression," *Exp. Syst. Appl.*, vol. 193, May 2022, Art. no. 116387.

[30] K. El Moutaouakil, M. Roudani, and A. El Ouissari, "Optimal entropy genetic fuzzy-C-means SMOTE (OEGFCM-SMOTE)," *Knowl.-Based Syst.*, vol. 262, Feb. 2023, Art. no. 110235.

[31] A. Zhang, H. Yu, Z. Huan, X. Yang, S. Zheng, and S. Gao, "SMOTE-RkNN: A hybrid re-sampling method based on SMOTE and reverse k-nearest neighbors," *Inf. Sci.*, vol. 595, pp. 70–88, May 2022.

[32] A. Zhang, H. Yu, S. Zhou, Z. Huan, and X. Yang, "Instance weighted SMOTE by indirectly exploring the data distribution," *Knowl.-Based Syst.*, vol. 249, Aug. 2022, Art. no. 108919.

[33] R. Liu, "A novel synthetic minority oversampling technique based on relative and absolute densities for imbalanced classification," *Int. J. Speech Technol.*, vol. 53, no. 1, pp. 786–803, Jan. 2023.

[34] J. Xie, M. Zhu, K. Hu, and J. Zhang, "Instance hardness and multivariate Gaussian distribution-based oversampling technique for imbalance classification," *Pattern Anal. Appl.*, vol. 26, no. 2, pp. 735–749, 2023.

[35] J. Zhai, J. Qi, and C. Shen, "Binary imbalanced data classification based on diversity oversampling by generative models," *Inf. Sci.*, vol. 585, pp. 313–343, Mar. 2022.

[36] Y. Li, Y. Wang, T. Li, B. Li, and X. Lan, "SP-SMOTE: A novel space partitioning based synthetic minority oversampling technique," *Knowl.-Based Syst.*, vol. 228, Sep. 2021, Art. no. 107269.

[37] A. Sharma, P. K. Singh, and R. Chandra, "SMOTified-GAN for class imbalanced pattern classification problems," *IEEE Access*, vol. 10, pp. 30655–30665, 2022.

[38] J. Dou, G. Wei, Y. Song, D. Zhou, and M. Li, "Switching triple-weight-SMOTE in empirical feature space for imbalanced and incomplete data," *IEEE Trans. Autom. Sci. Eng.*, early access, Jan. 31, 2023, doi: 10.1109/TASE.2023.3240759.

[39] D. Elreedy, A. F. Atiya, and F. Kamalov, "A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning," *Mach. Learn.*, pp. 1–21, Jan. 2023, doi: 10.1007/s10994-022-06296-4.

[40] W. Pei, B. Xue, M. Zhang, L. Shang, X. Yao, and Q. Zhang, "A survey on unbalanced classification: How can evolutionary computation help?" *IEEE Trans. Evol. Comput.*, early access, Mar. 14, 2023, doi: 10.1109/TEVC.2023.3257230.

[41] A.-A. Semenoglou, E. Spiliotis, and V. Assimakopoulos, "Data augmentation for univariate time series forecasting with neural networks," *Pattern Recognit.*, vol. 134, Feb. 2023, Art. no. 109132.

[42] S. Maldonado, C. Vairetti, A. Fernandez, and F. Herrera, "FW-SMOTE: A feature-weighted oversampling approach for imbalanced classification," *Pattern Recognit.*, vol. 124, Apr. 2022, Art. no. 108511.

[43] G. Douzas, F. Bacao, and F. Last, "Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE," *Inf. Sci.*, vol. 465, pp. 1–20, Oct. 2018.

[44] J. Chakraborty, S. Majumder, and T. Menzies, "Bias in machine learning software: Why? How? What to do?" in *Proc. 29th ACM Joint Meeting Eur. Softw. Eng. Conf. Symp. Found. Softw. Eng.*, Aug. 2021, pp. 429–440.

[45] L. Yang, X. Chen, Y. Luo, X. Lan, and W. Wang, "IDEA: A utility-enhanced approach to incomplete data stream anonymization," *Tsinghua Sci. Technol.*, vol. 27, no. 1, pp. 127–140, Feb. 2022.

[46] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional GAN," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–11.

[47] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of Wasserstein GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[48] C. García-Vicente, D. Chushig-Muzo, I. Mora-Jiménez, H. Fabelo, I. T. Gram, M.-L. Løchen, C. Granja, and C. Soguero-Ruiz, "Evaluation of synthetic categorical data generation techniques for predicting cardiovascular diseases and post-hoc interpretability of the risk factors," *Appl. Sci.*, vol. 13, no. 7, p. 4119, Mar. 2023.

[49] C. García-Vicente, D. Chushig-Muzo, I. Mora-Jiménez, H. Fabelo, I. T. Gram, M.-L. Løchen, C. Granja, and C. Soguero-Ruiz, "Clinical synthetic data generation to predict and identify risk factors for cardiovascular diseases," in *Heterogeneous Data Management, Polystores, and Analytics for Healthcare*. Berlin, Germany: Springer, 2023, pp. 75–91.

[50] R. Alqudah, A. A. Al-Mousa, Y. Abu Hashyeh, and O. Z. Alzaibaq, "A systemic comparison between using augmented data and synthetic data as means of enhancing wafermap defect classification," *Comput. Ind.*, vol. 145, Feb. 2023, Art. no. 103809.

[51] C. Zou, F. Yang, J. Song, and Z. Han, "Generative adversarial network for wireless communication: Principle, application, and trends," *IEEE Commun. Mag.*, early access, Jun. 19, 2023, doi: 10.1109/MCOM.011.2200731.

[52] N. A. Frost and T. B. Hassan, "Serious eye injuries caused by coin throwing," *Eye*, vol. 7, no. 5, p. 714, Sep. 1993.

[53] H. Zhu, M. Zhou, G. Liu, Y. Xie, S. Liu, and C. Guo, "NUS: Noisy-sample-removed undersampling scheme for imbalanced classification and application to credit card fraud detection," *IEEE Trans. Computat. Social Syst.*, early access, Mar. 7, 2023, doi: 10.1109/TCSS.2023.3243925.

[54] N. S. Halvaiee and M. K. Akbari, "A novel model for credit card fraud detection using artificial immune systems," *Appl. Soft Comput.*, vol. 24, pp. 40–49, Nov. 2014.

[55] H. Zhu, G. Liu, M. Zhou, Y. Xie, and Q. Kang, "A noisy-sample-removed under-sampling scheme for imbalanced classification of public datasets," *IFAC-PapersOnLine*, vol. 53, no. 5, pp. 624–629, 2020.

[56] P. M. Murphy, "UCI repository of machine learning databases," Dept. Inf. Comput. Sci., Univ. California, USA, Tech. Rep., 1992. [Online]. Available: http://www.ics.uci.edu/AI/ML/MLDBRepository.html, doi: 10.24432/C5XW20.
[57] J. S. Díaz and Á. L. García, "A Python library to check the level of anonymity of a dataset," *Sci. Data*, vol. 9, no. 1, p. 785, Dec. 2022.
[58] J. Ri and H. Kim, "G-mean based extreme learning machine for imbalance learning," *Digit. Signal Process.*, vol. 98, Mar. 2020, Art. no. 102637.
[59] R. Shen, S. Bubeck, and S. Gunasekar, "Data augmentation as feature manipulation," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 19773–19808.
[60] L. Chen, L. Zeng, Y. Mu, and L. Chen, "Global combination and clustering based differential privacy mixed data publishing," *IEEE Trans. Knowl. Data Eng.*, early access, Jan. 17, 2023, doi: 10.1109/TKDE.2023.3237822.
[61] E. Strickland, "Andrew Ng, AI minimalist: The machine-learning pioneer says small is the new big," *IEEE Spectr.*, vol. 59, no. 4, pp. 22–50, Apr. 2022.

ABDUL MAJEED received the B.S. degree in information technology from UIIT, PMAS-UAAR, Rawalpindi, Pakistan, in 2013, the M.S. degree in information security from COMSATS University, Islamabad, Pakistan, in 2016, and the Ph.D. degree in computer information systems and networks from Korea Aerospace University, South Korea, in 2021. He was a Security Analyst with Trillium Information Security Systems (TISS), Rawalpindi, from 2015 to 2016. He is currently an Assistant Professor with the Department of Computer Engineering, Gachon University, South Korea. His research interests include privacy-preserving data publishing, statistical disclosure control, privacy-aware analytics, data-centric artificial intelligence, and machine learning.

SEONG OUN HWANG (Senior Member, IEEE) received the B.S. degree in mathematics from Seoul National University, in 1993, the M.S. degree in information and communications engineering from the Pohang University of Science and Technology, in 1998, and the Ph.D. degree in computer science from the Korea Advanced Institute of Science and Technology, South Korea, in 2004. He was a Software Engineer with LG-CNS Systems Inc., from 1994 to 1996. He was also a Senior Researcher with the Electronics and Telecommunications Research Institute (ETRI), from 1998 to 2007. He was also a Professor with the Department of Software and Communications Engineering, Hongik University, from 2008 to 2019. He is currently a Full Professor with the Department of Computer Engineering, Gachon University, South Korea. His research interests include cryptography, data-centric artificial intelligence, cybersecurity, and machine learning.