scientific reports

Check for updates

OPEN

Tabular transformer generative adversarial network for heterogeneous distribution in healthcare

Ha Ye Jin Kang[1,2], Minsam Ko[1] & Kwang Sun Ryu[2,3]✉

In healthcare, the most common type of data is tabular data, which holds high significance and potential in the field of medical AI. However, privacy concerns have hindered their widespread use. Despite the emergence of synthetic data as a viable solution, the generation of healthcare tabular data (HTD) is complex owing to the extensive interdependencies between the variables within each record that incorporate diverse clinical characteristics, including sensitive information. To overcome these issues, this study proposed a tabular transformer generative adversarial network (TT-GAN) to generate synthetic data that can effectively consider the relationships between variables potentially present in the HTD dataset. Transformers can consider the relationships between the columns in each record using a multi-attention mechanism. In addition, to address the potential risk of restoring sensitive data in patient information, a Transformer was employed in a generative adversarial network (GAN) architecture, to ensure an implicit-based algorithm. To consider the heterogeneous characteristics of the continuous variables in the HTD dataset, the discretization and converter methodology were applied. The experimental results confirmed the superior performance of the TT-GAN than the Conditional Tabular GAN (CTGAN) and copula GAN. Discretization and converters were proven to be effective using our proposed Transformer algorithm. However, the application of the same methodology to Transformer-based models without discretization and converters exhibited a significantly inferior performance. The CTGAN and copula GAN indicated minimal effectiveness with discretization and converter methodologies. Thus, the TT-GAN exhibited considerable potential in healthcare, demonstrating its ability to generate artificial data that closely resembled real healthcare datasets. The ability of the algorithm to handle different types of mixed variables efficiently, including polynomial, discrete, and continuous variables, demonstrated its versatility and practicality in health care research and data synthesis.

Keywords Tabular transformer generative adversarial network (TT-GAN), Heterogeneous distribution, Healthcare tabular data (HTD)

Abbreviations

HTD	Healthcare tabular data
STD	Synthetic tabular data
GAN	Generative adversarial network
CTGAN	Conditional tabular GAN
RF	Random forest
CatBoost	Category boosting
XGBoost	Extreme gradient boosting
LightGBM	Light gradient boosting machine
AUC	Area under the curve
LLM	Large language model

[1]Department of Applied Artificial Intelligence, Hanyang University, Seoul, Republic of Korea. [2]Department of Public Health & AI, Graduate School of Cancer Science and Policy, National Cancer Center, Goyang, Republic of Korea. [3]National Cancer Data Center, National Cancer Center, Goyang, Republic of Korea. ✉email: niceplay13@ncc.re.kr

Tabular data, which are organized in rows and columns, are the most common type of data across various real-world applications. Its prevalence in various domains underlines its importance in practical machine learning applications and research environments[1]. In particular, in healthcare, wherein structured information such as patient demographics, diagnoses, and treatments is critical, tabular data play an important role. Tabular data in healthcare have tremendous potential for artificial intelligence (AI) research, which is less utilized in this field than in other areas owing to various issues related to privacy and data sharing[2,3]. To overcome these issues, tabular synthetic data have been proposed, which have shown reliable research results[4,5]. However, minimal research has been conducted on the generation of synthetic data that considers the relationships between columns and the handling of continuous variables with various distributions. The complexity of tabular data, particularly in healthcare, is characterized by the relationships between columns, rendering the existence of a single distribution rare. This complexity presents challenges in generating synthetic tabular data (STD) in the context of healthcare tabular data (HTD). The addressal of these challenges necessitates an approach that accurately captures and replicates the inherent relationships in tabular healthcare datasets. Recently, the Transformer method has been demonstrated to be a good method for generating STDs using multi-attention to consider the relationships between columns. However, it inherently encounters difficulty in handling continuous attributes on the right side. In addition, the models proposed thus far based on Transformer algorithms are mostly intended for prediction tasks. Only a few transformer algorithms have been used to generate STDs; however, they are not suited to tabular data in healthcare and, in particular, lack deep thought about handling continuous variables. To overcome these challenges, we propose the tabular transformer generative adversarial network (TT-GAN) algorithm. The TT-GAN comprises three stages. The first stage was the discretization stage, where continuous variables were converted to discrete variables in order to apply the Transformer method. The second stage is the generation stage, wherein synthetic data were generated using a generator and discriminator based on a GAN architecture with a transformer. The third stage was the converter stage, wherein the discretized columns were translated into continuous variables in the STD.

The contributions of this study are as follows:

1. TT-GAN for HTD generation: We propose the TT-GAN, which aimed to capture the intricate dependencies, irregular patterns, and varied distributions present in real-world healthcare datasets, while ensuring that the synthesized data closely aligned with the statistical characteristics of authentic healthcare information. TT-GAN demonstrated superior performance with HTD compared with previously proposed generative adversarial network (GAN) algorithms designed for the synthesis of STD. The performance improvement was attributed to the efficient architecture design to apply column-to-column relationships, which are characteristic of healthcare data, to the Transformer algorithm. Moreover, our architecture was designed to avoid explicit density-based methods, thereby overcoming the issue of data privacy, and consequently propose a reasonable method for effectively handling continuous variables.
2. Synth Health Discovery Network: The constructed generative model and its corresponding code on shared platforms provided opportunities for transparency and reproducibility. This active contribution is expected to help advance overall progress in healthcare research. Sharing our generative model facilitates a fair evaluation and contributes to field advancement. Further, sharing codes and models for educational purposes would also be valuable. Students, researchers, and practitioners can gain a deeper understanding of optimal techniques and methodologies in the healthcare generative modeling domain. The Synth Health Discovery Network will play an important role in enabling the research ecosystem to promote the effective use of healthcare data and contribute to future healthcare breakthroughs.

The remainder of this paper is organized as follows. The "Related works" section presents the relevant studies explored to lay the foundation for our contributions. The "Method" section provides detailed information about the TT-GAN model. The "Results" presents the model's performance analysis and highlights the findings. Further, the "Discussion" section offers an analysis and explores the implications, comparisons, and future avenues. Finally, the "Conclusion" section summarizes our core findings.

Related works

Conditional tabular GAN (CTGAN) and copula GAN were developed by Xu et al.[6]. Copula GAN is an extension or variant of the CTGAN and is a type of GAN designed to generate synthetic tabular data. copula GAN incorporates copula functions based on the CTGAN to enhance the learning process. Juan Carlos Quirz et al.[7] proposed a machine-learning framework for automating the severity assessment of COVID-19 using clinical and imaging data. To address the imbalance problem in tabular clinical data, they employed the CTGAN model, along with various oversampling techniques. Ultimately, logistic regression models with balanced synthetic data effectively distinguished between mild and severe cases. Syde et al.[8] developed a fundamental tumor type classification model for decision support. An approach involving the utilization of a CTGAN was implemented to address imbalances in clinical data. All evaluation metrics demonstrated an improvement as when increasing the sample size through the application of the CTGAN. Kang et al.[9] demonstrated the preservation of data with logical relationships while generating STD using a CTGAN. They implemented a divide-and-conquer approach to mitigate the risk of information loss caused by dependence on condition columns. This DC-based strategy facilitated the creation of an STD that accurately reflects the inherent patterns and relationships within each subset of the Original Data.

Although much research has been conducted on STD, its implementation in real-world healthcare datasets remains challenging. This is because the HTD has the following unique characteristics. (1) Different types of columns: In general, HTD, which are non-standardized patient records, are heterogeneous and voluminous, that is, the data contain different column types such as numeric, float, integer, and character. (2) Non-Gaussian

distributions: In healthcare, data may exhibit skewness and kurtosis values that deviate significantly from the normal distribution, contain outliers, combine multiple distributions (bimodal or multimodal), or contain insufficient data. Therefore, generating synthetic HTD involves the addressing of the complex dependencies, irregularities, and diverse distributions found in real-world datasets. Moreover, privacy must be protected carefully. The inconsistent and diverse distribution and characteristics of such tabular data, and privacy concerns hinder the application and diffusion of effective algorithms representative of AI. To address the distributional issues of these data, we applied transformers with discretization and converter to consider the complex relationships between the data columns and validate our proposed research framework.

Method

Tabular transformer generative adversarial network (TT-GAN)

The proposed TT-GAN should efficiently process HTD with different distributions and generate approximately good synthetic data. The relationship between the columns is based on the Transformer and follows three steps, as shown in Fig. 1. In the discretization stage, a clustering algorithm was used to preprocess continuous variables, thereby transforming them into categorical variables. In the generation stage, categorical features were transformed into discretized data using ordinal encoding. We incorporated a Transformer encoder that used a multi-headed attention mechanism to capture the relational nature of each column and learn categorical features, to be fed into the generator. The generator approximated the probability distribution of real data within a high-dimensional latent space and used multi-head attention from the Transformer encoder to exploit contextual embeddings that capture the relational properties of each column and learn categorical features. The discriminator evaluated the data and output probabilities. It minimized binary cross-entropy by aiming for low probabilities for fake data and high probabilities for real data. Moreover, it was verified that the generated data satisfied the specified conditions from the condition vector. In the converter stage, we used a prediction model to convert categorical data that were originally continuous into continuous data. The TT-GAN successfully generated mixed variable types (multinomial, discrete, and continuous) similar to real tabular data (Algorithm 1).

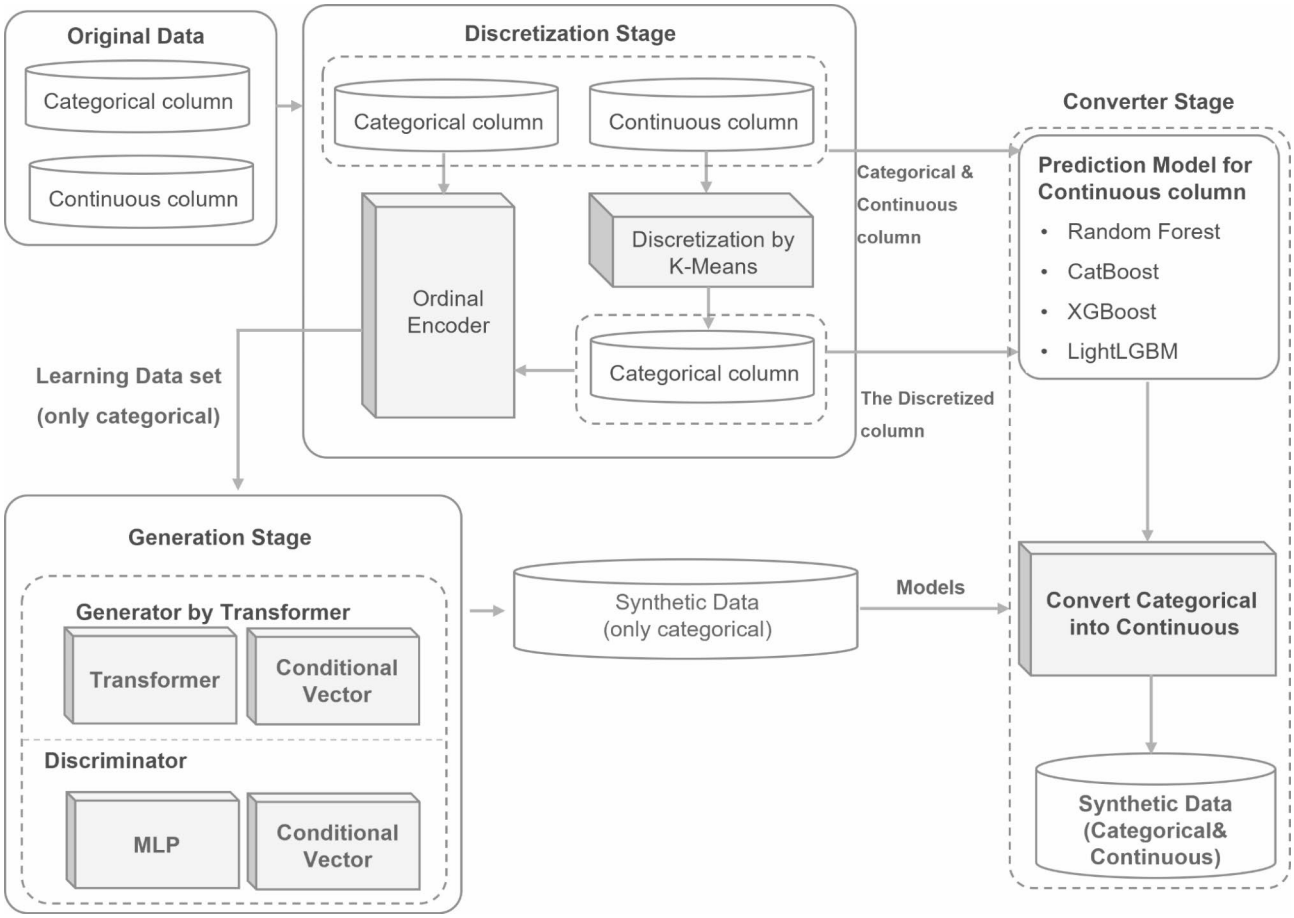

Fig. 1. Architecture of tabular transformer generative adversarial network (TT-GAN).

Input: real mixed type (continuous and categorical) data D; list of continuous variables *VContinuous*; list of categorical variables *VCategorical*; number of synthetic data N
Output: synthetic mixed type (continuous and categorical) data D'

$D_{categorical} \leftarrow$ select $V_{categorical}$ from D
$D_{continuous} \leftarrow$ select $V_{continuous}$ from D
$D_{discretized} \leftarrow Discretization\ (D_{continuous})$
$D_{all\_categorical} \leftarrow D_{discretized} + D_{categorical}$
for each i in $V_{continuous}$ *do*
    $R_i \leftarrow Train\ Regressor\ (D_{all\_categorical}, i)$
end for
$G \leftarrow Train\ generator\ (D_{all\_categorical})$
$D' \leftarrow sample\ (G, N)$
for each i in $Ddiscretized$ *do*
    $D'_i \leftarrow Converter\ (D', R_i)$
end for

Algorithm 1. TT-GAN algorithms

Discretization stage

The first stage was the preprocessing stage. Herein, discretization is the process of transforming continuous properties into discrete properties by forming a group of adjacent intervals that extend the range of the property (Algorithm 2). Data discretization with k-means clustering[10] for the numerical variables was performed before applying the transformer module. K-means clustering is a popular method for calculating continuous distance-based similarity measures to cluster data points, rendering it suitable for the discretization of continuous-valued variables. The k-means clustering algorithm divides the input data into clusters by first assigning k random data points as centroids. Each data point was then assigned to its nearest center to form the initial cluster distribution. This process discretized the data using min-max values, calculated clusters, and distances between clusters. The algorithm iteratively created clusters by recalculating the cluster centers as the averages of the values in each cluster and reassigning the data points to the nearest center[11].

function *Discretization* ($D_{continuous}$):
    $D_{discretized} = []$
    for each i in $V_{continuous}$ ***do***
        Discretize i from $D_{continuous}$
        Append $i_{discretized}$ into $D_{discretized}$
    end for
    return $D_{discretized}$

Algorithm 2. Discretization

Generation stage

The second stage was the generation stage. A combination of the CTGAN architecture, comprising a generator and discriminator, and a Transformer encoder was used to learn the real data distribution and produce an optimal generative model (Algorithm 3). The transformer encoder had the following order. The Transformer input is an embedded vector from the column embedding. Embedding techniques were used to represent the data as a dense vector to place highly similar data at similar positions in the vector space to calculate similarity. We set a categorical variable $x_i = \{x_1^{cat}, ?, x_m^{cat}\}$, for $i \in \{1, ?, m\}$. Using each of the xi categorical features in a parametric embedding of dimension d by column embedding yields $e_{\phi i}\ (x_i) \in R^d$. Column embedding shared optimal dimension of parameters ci in column i. The input to the encoder first passed through a self-

attention layer, which examined the relationship between all the vectors of the columns in the input for the encoder to encode one particular column. After the input passed through the self-attention layer, the output was returned to the feed-forward neural network. The same feedforward neural network was applied independently to each vector of the columns at each position to create the output. The attention layer facilitates the creation of multiple "representation spaces." After training, each set was multiplied by the input vectors to project the vectors for each purpose. The fact that there are several such sets implies that each vector was represented in a different space. The final encoder output a representation of the input columns. To capture all possible correlations between columns, fully connected networks were used in both the generator and critic because the columns in a row do not exhibit a local structure. A synthetic row representation was generated using a mix of activation functions after two hidden layers. The scalar value αi was generated by tanh, while the mode indicator βi and discrete values di were generated by Gumbel-SoftMax. The critic used the LeakyReLU function and dropped each hidden layer to ensure accuracy. Both the generator and critic used two fully connected hidden layers. Batch normalization and ReLU activation functions were used in the generator. Through training based on these steps, the synthetic at a generation model generated synthetic data (Algorithm 4).

function *train_generator* ($D_{all\_categorical}$):
    Initialize generative model G
    Initialize Transformer Encoder
        Initialize transformer encoder parameters
    for number of training iterations **do**
        for k steps **do**
            Sample mini-batch of noise samples from noise prior
            Sample mini-batch of examples from data generating distribution
            Update the discriminator by ascending its stochastic gradient
        end for
        Sample mini-batch of noise samples from noise prior
        Update G model parameters by descending its stochastic gradient
    end for
    return G

Algorithm 3. Model generation

function *sample* (G, N):
    Sample N noise vectors from noise prior distribution
    Generate samples from the generator model G with the noise vectors
    return D'

Algorithm 4. Synthetic generation

Converter stage

In the converter stage, we developed models to predict the original value of the discretized continuous variable based on the original data (Algorithm 5). This model was applied to the synthetic data to convert the categorical variables into continuous variables (Algorithm 6). We applied various tree-based ensemble models, including Random Forest (RF), Categorical Boosting (CatBoost), Extreme Gradient Boosting (XGBoost), and light gradient boosting machine (LightGBM). RF, which was originally developed in 1995 by Tin Kam Ho[12], utilizes an ensemble of decision trees, randomly selects subsets of features and samples during training, and provides robustness against overfitting. CatBoost[13] is designed to seamlessly support categorical features and automatically handle their encoding complexity. Known for its speed, performance, and scalability, XGBoost[14] supports regularization, efficiently handles missing data, and facilitates parallel processing. LightGBM[15] was optimized for large and high-dimensional datasets. It efficiently handles categorical features without the need for one-hot encoding, and uses a histogram-based approach for faster training.

function *train_regressor* ($D_{all\_categorical}$, i):
Define ensemble regressors (R_{i1}, R_{i2},..., R_{iM})
for number of regressors **do**
Initialize a new regressor
Train R_i regressors
end for
For a given M ensemble regressor (R_{i1}, R_{i2},..., R_{iM}) predicts the final value
return R_i

Algorithm 5. Train regressor

function *Converter* (D', R_i):
Set empty list of predicted values
for number of regressors **do**
Predict using each regressor in R_i
Append predicted value to the list
end for
Combine predictions from all regressors in the ensemble into D'_i
return D'_i

Algorithm 6. Converter

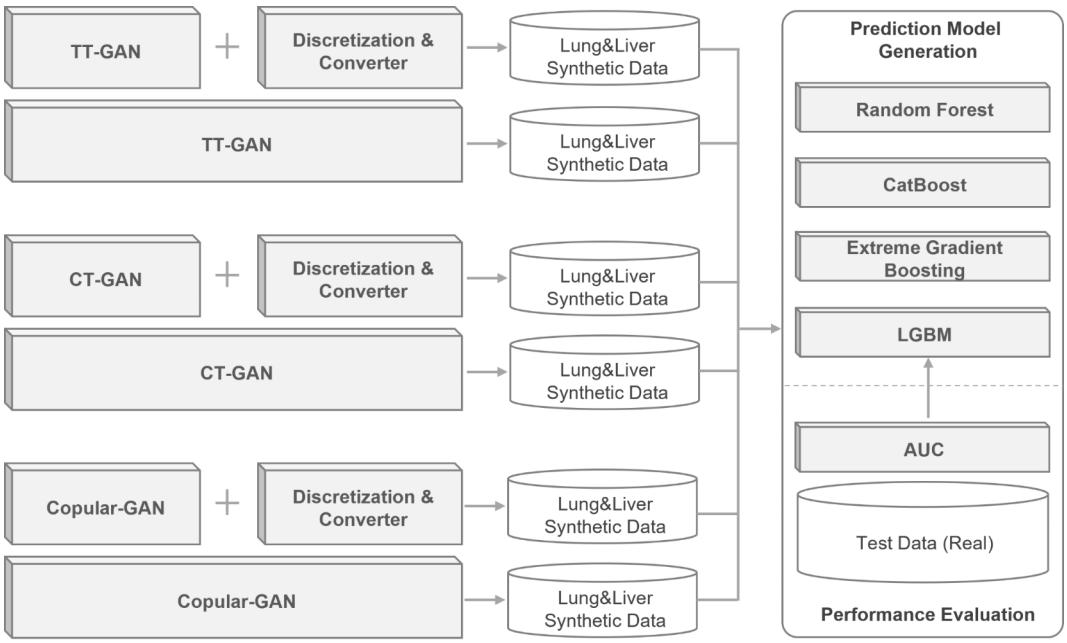

Fig. 2. Experiment setting for TT-GAN.

Results

Experimental setting

To objectively evaluate the performance of the proposed algorithm, we conducted the following experiments as shown in Fig. 2. First, synthetic data were generated using CTGAN, copula GAN, and TT-GAN, and data were generated with and without discretization and converter methodology in each generation. Secondly, we developed a prediction model for liver and lung cancer mortality based on synthetic data. Third, the model generated using the synthetic data was evaluated by measuring the AUC(area under the curve) based on the original TEST data. The model was evaluated using a two-phase approach, whereby the training synthetic data set was employed to train the model, and the test real data set was used to assess its performance[6], and the model test was repeated five times to calculate the mean and standard deviation. The objective evaluation of the performance of the synthetic data-generated model was conducted using the test data for liver(n = 1,363) and lung cancer(n = 460) (Appendix 1).

Dataset

Study population

Lung and liver cancer data from the Korea Central Cancer Registry at the National Cancer Center (https://kccrsurvey.cancer.go.kr/index.do) were used in this study, and the data were reviewed by the institutional review board[16]. Our study used non duplicate lung cancer and liver data after excluding missing variables. Lung cancer data were divided into development (n = 1, 616), validation (n = 228), and test (n = 460) groups, which were further divided into development (n = 4, 767), validation (n = 681), and test (n = 1, 363) groups using stratified random sampling. The basic characteristics of the datasets showed similar distributions (Appendix 1).

Generation and validation of STDs

The CTGAN, copula GAN, and TT-GAN models were trained for comparison (Appendix 2). Subsequently, we employed RF, CatBoost, XGBoost, and LightGBM as regression classifiers which served as converters to predict continuous variables. Subsequently, we implemented the discretization and converter methodology. To assess prediction performance, we conducted evaluations using the RF, CatBoost, XGBoost, and LightGBM models individually for each of the generated GAN models (Appendix 3). We generated 1,616 lung cancer STD and 4,766 liver cancer STD.

As shown in Table 1, for the lung cancer dataset, the following AUC values were obtained for the original dataset: RF: 85.02%, CatBoost: 86.02%, XGBoost: 84.24%, and LightGBM: 84.49%. The model performance was observed through the STD, which was generated by each GAN model without the preprocessing stage. For the STD generated by CTGAN, the AUC for RF was 84.00 ± 0.55, while CatBoost achieved 83.80 ± 0.45, XGBoost attained 81.20 ± 0.48, and LightGBM obtained 82.88 ± 0.72. When the STD was produced by copula GAN, the values were 84.45 ± 0.26 for RF, 81.58 ± 0.77 CatBoost, 79.40 ± 0.71 for XGBoost, and 84.07 ± 0.58 for LightGBM. The STD generated by TT-GAN yielded an AUC of 81.53 ± 0.46 for RF, 82.64 ± 0.44 for CatBoost, 84.45 ± 0.51 for XGBoost, 84.32 ± 0.18 for LightGBM.

The model performance was assessed by examining the STD generated by each GAN model after the preprocessing stage. The STD generated by CTGAN with the RF converter yielded AUC of 82.31 ± 0.50 for RF, 83.68 ± 0.74 for CatBoost, 81.33 ± 0.55 for XGBoost, and 80.59 ± 0.32 for LightGBM. With the CatBoost converter, the AUC was 83.39 ± 0.33 for RF, 83.93 ± 0.67 for CatBoost, 83.07 ± 1.13 for XGBoost, and 82.38 ± 0.31 for LightGBM. When the XGBoost converter was used, the AUC was 82.95 ± 0.55 for RF, 83.58 ± 0.24 for CB,

Data	Generator	Converter	Prediction model			
			RF	CatBoost	XGBoost	LightGBM
Original	–	–	85.02%	86.02%	84.24%	84.49%
Without discretization and converter	CTGAN	–	84.00 ± 0.55	83.80 ± 0.45	81.20 ± 0.48	82.88 ± 0.72
	Copula GAN	–	84.45 ± 0.26	81.58 ± 0.77	79.40 ± 0.71	84.07 ± 0.58
	TT-GAN	–	81.53 ± 0.46	82.64 ± 0.44	84.45 ± 0.51	84.32 ± 0.18
Discretization and converter	CTGAN	RF	82.31 ± 0.50	83.68 ± 0.74	81.33 ± 0.55	80.59 ± 0.32
		CatBoost	83.39 ± 0.33	83.93 ± 0.67	83.07 ± 1.13	82.38 ± 0.31
		XGBoost	82.95 ± 0.55	83.58 ± 0.24	82.09 ± 1.02	82.67 ± 0.54
		LightGBM	83.32 ± 0.67	82.99 ± 0.28	80.76 ± 0.58	82.97 ± 1.71
	Copula GAN	RF	81.18 ± 0.83	80.50 ± 0.80	78.30 ± 0.92	77.78 ± 1.61
		CatBoost	81.90 ± 0.21	82.17 ± 0.50	79.81 ± 1.01	81.65 ± 1.16
		XGBoost	81.50 ± 0.76	82.30 ± 0.77	80.35 ± 0.60	82.50 ± 0.87
		LightGBM	81.18 ± 0.78	80.04 ± 0.92	81.36 ± 0.44	81.27 ± 1.07
	TT-GAN	RF	83.53 ± 0.22	83.92 ± 0.44	83.19 ± 0.77	82.58 ± 0.19
		CatBoost	84.69 ± 0.55	85.86 ± 0.30	85.94 ± 0.51	84.55 ± 0.56
		XGBoost	84.84 ± 0.47	85.91 ± 0.14	85.44 ± 0.19	85.34 ± 0.60
		LightGBM	84.69 ± 0.37	85.69 ± 0.09	82.97 ± 0.36	85.42 ± 0.71

Table 1. Performance evaluation of prediction models using lung cancer SSD test dataset.

Data	Generator	Converter	Prediction model			
			RF	CatBoost	XGBoost	LightGBM
Original	–	–	85.96%	86.69%	85.14%	85.91%
Without Discretization and converter	CTGAN	–	83.31 ± 0.17	83.81 ± 0.23	81.20 ± 0.50	82.69 ± 0.19
	Copula GAN	–	82.46 ± 0.07	83.61 ± 0.24	80.93 ± 0.62	82.53 ± 0.42
	TT-GAN	–	80.29 ± 0.14	81.98 ± 0.35	80.43 ± 0.31	80.33 ± 0.37
Discretization and converter	CTGAN	RF	81.77 ± 0.21	82.78 ± 0.52	79.60 ± 0.62	80.94 ± 0.34
		CatBoost	82.65 ± 0.24	81.00 ± 0.33	77.60 ± 0.27	80.34 ± 0.60
		XGBoost	82.96 ± 0.21	82.44 ± 0.50	80.43 ± 0.50	81.81 ± 0.48
		LightGBM	82.47 ± 0.43	81.47 ± 0.29	78.76 ± 0.41	80.34 ± 0.19
	Copula GAN	RF	78.95 ± 0.28	71.70 ± 0.70	65.62 ± 2.14	74.54 ± 1.42
		CatBoost	80.95 ± 0.38	79.41 ± 0.69	75.10 ± 1.09	78.69 ± 1.41
		XGBoost	78.96 ± 0.45	79.75 ± 1.13	74.95 ± 1.24	74.30 ± 1.42
		LightGBM	77.67 ± 1.01	70.46 ± 1.15	68.46 ± 1.43	71.32 ± 0.56
	TT-GAN	RF	83.24 ± 0.26	83.83 ± 0.13	82.99 ± 0.26	82.76 ± 0.19
		CatBoost	83.32 ± 0.24	83.96 ± 0.19	83.10 ± 0.31	82.37 ± 0.18
		XGBoost	83.32 ± 0.18	84.06 ± 0.15	83.29 ± 0.15	84.04 ± 0.20
		LightGBM	82.32 ± 0.37	84.13 ± 0.12	83.28 ± 0.46	83.16 ± 0.48

Table 2. Performance evaluation of prediction models using liver cancer SSD test dataset.

82.09 ± 1.02 for XGBoost, and 82.67 ± 0.54 for LightGBM. As for the application of the LightGBM converter, the AUC was 83.32 ± 0.67 for RF, 82.99 ± 0.28 for CatBoost, 80.76 ± 0.58 for XGBoost, and 82.97 ± 1.71 for LightGBM.

The STD was produced by copula GAN using RF converter, the AUC was 81.18 ± 0.83 for RF, 80.50 ± 0.80 for CatBoost, 78.30 ± 0.92 for XGBoost, and 77.78 ± 1.61 for LightGBM. When the CatBoost converter was utilized, the AUC was 81.90 ± 0.21 for RF, 82.17 ± 0.50 for CB, 79.81 ± 1.01 for XGBoost, and 81.65 ± 1.16 for LightGBM. Further, with the XGBoost converter, the AUC was 81.50 ± 0.76 for RF, 82.30 ± 0.77 for CatBoost, 80.35 ± 0.60 for XGBoost, and 82.50 ± 0.87 for LightGBM. The application of the LightGBM converter yielded AUC values of 81.18 ± 0.78 for RF, 80.04 ± 0.92 for CatBoost, 81.36 ± 0.44 for XGBoost, and 81.27 ± 1.07 for LightGBM.

Further, the TT-GAN-derived lung STD when used with the RF converter yielded AUC values of 83.53 ± 0.22 for RF, 83.92 ± 0.44 for CatBoost, 83.19 ± 0.77 for XGBoost, and 82.58 ± 0.19 for LightGBM. When the CatBoost converter was used, the AUC was 84.69 ± 0.55 for RF, 85.86 ± 0.30 for CatBoost, 85.94 ± 0.51 for XGBoost, 84.55 ± 0.56 for LightGBM. The utilization of the XGBoost converter, 84.84 ± 0.47 for RF, 85.91 ± 0.14 for CatBoost, and 85.44 ± 0.19 for XGBoost, 85.34 ± 0.60 for LightGBM. When the LightGBM converter was used, the AUC was 84.69 ± 0.37 for RF, 85.69 ± 0.09 for CatBoost, 82.97 ± 0.36 for XGBoost, and 85.42 ± 0.71 for LightGBM.

In Table 2, the performances of the RF, CatBoost, XGBoost, and LightGBM prediction models for the liver cancer dataset were evaluated using the AUC metric for the test sets. The original dataset showed AUC values of 85.96% for RF, 86.69% for CatBoost, 85.14% for XGBoost, and 85.91% for LightGBM. Without the preprocessing stage, the STD from CTGAN, exhibited AUC values of 83.31 ± 0.17 for RF, 83.81 ± 0.23 for CatBoost, 81.20 ± 0.50 for XGBoost, and 82.69 ± 0.19 for LightGBM. The STD from copula GAN exhibited AUC values of 82.46 ± 0.07 for RF, 83.61 ± 0.24 for CatBoost, 80.93 ± 0.62 for XGBoost, and 82.53 ± 0.42 for LightGBM. The STD from TT-GAN exhibited AUC values of 80.29 ± 0.14 for RF, 81.98 ± 0.35 for CatBoost, 80.43 ± 0.31 for XGBoost, and 80.33 ± 0.37 for LightGBM.

When evaluating the impact of pre-processing, the STD generated from CTGAN, in conjunction with the RF converter, yielded AUC values of 81.77 ± 0.21 for RF, 82.78 ± 0.52 for CatBoost, 79.60 ± 0.62 for XGBoost, and 80.94 ± 0.34 for LightGBM. The implementation of the CatBoost converter results in an AUC of 82.65 ± 0.24 for RF, 81.00 ± 0.33 for CatBoost, 77.60 ± 0.27 for XGBoost, and 80.34 ± 0.60 for LightGBM. Employing the XGBoost converter yielded AUC values of 82.96 ± 0.21 for RF, 82.44 ± 0.50 for CatBoost, 80.43 ± 0.50 for XGBoost, and 81.81 ± 0.48 for LightGBM. Finally, using the LightGBM converter, AUC values of 82.47 ± 0.43 for RF, 81.47 ± 0.29 for CatBoost, 78.76 ± 0.41 for XGBoost, and 80.34 ± 0.19 for LightGBM were obtained as it shown in Table 2.

The STD generated by copula GAN exhibited AUC values of 78.95 ± 0.28 for RF, 71.70 ± 0.70 for CatBoost, 65.62 ± 2.14 for XGBoost, and 74.54 ± 1.42 for LightGBM when utilized by the RF converter. The CatBoost converter yielded AUC values of 80.95 ± 0.38 for RF, 79.41 ± 0.69 for CatBoost, 75.10 ± 1.09 for XGBoost, and 78.69 ± 1.41 for LightGBM. The XGBoost converter yielded AUC values of 78.96 ± 0.45 for RF, 79.75 ± 1.13 for CatBoost, 74.95 ± 1.24 for XGBoost, and 74.30 ± 1.42 for LightGBM. Whereas the LightGBM converter yielded AUC values of 77.67 ± 1.01 for RF, 70.46 ± 1.15 for CatBoost, 68.46 ± 1.43 for XGBoost, and 71.32 ± 0.56 for LightGBM .

The STD obtained using the TT-GAN yielded various AUC. When employing the RF converter, the AUC values were 83.24 ± 0.26 for RF, 83.83 ± 0.13 for CatBoost, 82.99 ± 0.26 for XGBoost, and 82.76 ± 0.19 for LightGBM. The application of the CatBoost converter yielded AUC values of 83.32 ± 0.24 for RF, 83.96 ± 0.19 for CatBoost, 83.10 ± 0.31 for XGBoost, and 82.37 ± 0.18 for LightGBM. Implementing the XGBoost converter yielded AUC values of 83.32 ± 0.18 for RF, 84.06 ± 0.15 for CatBoost, 83.29 ± 0.15 for XGBoost, and 84.04 ± 0.20

LightGBM. Finally, the AUC with the LightGBM converter was 82.32 ± 0.37 for RF, 84.13 ± 0.12 for CatBoost, 83.28 ± 0.46 for XGBoost, and 83.16 ± 0.48 for LightGBM in Table 2.

The TT-GAN preserved the attributes of the original data and the relationships between variables, thereby maintaining connections between continuous and categorical values during the generation of the STD. It exhibited good efficacy in safeguarding real-world patterns and commendable performance in terms of model efficiency.

Discussion

Synthetic data are commonly perceived as irreversibly generated in traditional practice[17,18]. However, techniques that estimate explicit distributions during synthetic data generation can potentially reconstruct the original data, posing risks when dealing with sensitive information, such as healthcare data. To address this concern, synthetic data generation should rely on implicit density models, which focus on directly learning to generate realistic samples without explicitly modeling or calculating the underlying probability densities. Unlike explicit models (e.g., Variational Autoencoders), implicit density models do not require a predefined formula for the data distribution and instead compare generated data with real data[19]. These models can be broadly categorized into two families: (1) Markov Chain-Based Models, which use iterative sampling with multiple steps to refine a random starting point into a realistic sample (e.g., Generative Stochastic Networks)[20]; and (2) Single-Step Generative Models, which generate samples in a single step, offering faster and more scalable solutions (e.g., GANs, kernel-based moment matching methods)[21,22]. However, certain studies often overlook the distinct differences between explicit and implicit density methods. Consequently, the performance of algorithms is compared and evaluated without considering the differences between explicit and non-explicit density methods[23–25]. This experimental design can be considered irrational. The evaluation of algorithms based on the implicit density of sensitive data is considered an appropriate objective approach. By employing implicit density models, the generation process avoids explicit distributions, mitigating the risks of reconstructing original data and ensuring compliance with data privacy requirements, particularly in sensitive fields like healthcare. In cases where sensitive information is not included, synthetic data based on explicit density may have a higher quality and performance.

In contrast to the data typically observed in other fields, HTDs encompass a diverse array of clinically collected information, with the values represented in each column being of particular significance. In particular, continuous variables, particularly those gathered in a clinical setting, are obtained through the use of disparate devices, actions, and experiments, and consequently, these variables showed the heterogeneous distribution. Also, HTDs often exhibit a high degree of interdependence between variables, as they typically contain multiple clinical characteristics within a single record. Accurately capturing and reflecting these complex relationships among variables is a critical challenge in data synthesis. Generative models based on implicit density methods, such as GANs, are suited for this task because they generate STD without explicitly learning or representing the underlying probability distribution. This unique feature not only ensures the preservation of complex variable interdependencies but also mitigates the risk of unintentional disclosure of sensitive information. However, even the most widely used models, such as CTGAN and Copula GAN, continue to face significant challenges in generating realistic HTD. These methods struggle to model complex dependencies between features, especially when the relationships are nonlinear or hierarchical. It relies on relatively simple fully connected layers, which may fail to capture the intricate patterns often present in HTD. Also it has limited ability to generalize to datasets with highly nonstandard distributions or sparse data points. It struggles with scalability and adaptability when dealing with datasets that have diverse and highly interdependent variables. Also, it may generate synthetic data that lacks variability, leading to over smoothing in feature distributions[26].

Recently, advancements in deep learning, particularly those centered on Transformer architectures, have demonstrated promising applications in handling tabular datasets[27–29]. A notable development involves the implementation of a transformer-based GAN for the generation of synthetic data in the text and sequence areas[30,31]. Even for STDs, transformer-based GANs such as TabMT[32], have shown promising results in handling of missing values in tabular data sets through masking techniques. These models address challenges like filling missing data and handling diverse field types effectively. However, they fall short in modeling continuous variables with complex distributions, such as non-normal or multimodal continuous distributions that are commonly found in healthcare tabular datasets. One major challenge that must be overcome before its application in healthcare is the processing of continuous variables. Typically, it is ideal if all the continuous variables in the synthetic data follow a normalized Gaussian distribution during the learning process. However, cases wherein the actual data follow a Gaussian distribution are rare. We developed TT-GAN using the transformer with discretization and converter to address and improve this challenge. First, that all continuous variables were discretized before training a model to generate synthetic data. Consequently, a model was built to predict the continuous variables of these discretized variables. Subsequently, the model was used to predict the continuous variables of these discretized variables after generating synthetic data.

Based on our methodology, TT-GAN was found to be remarkably simple, user-friendly, and powerful. In our experimental results, the Transformer model applying the discretization and converter method exhibited outstanding performance. As observed in our experimental results, synthetic data generated by the Transformer model without the application of discretization and converter methodology exhibited significantly worse performance. Thus, although Transformer-based synthetic data generation models exhibit significant potential in the healthcare domain characterized by high inter-column interdependence, their capabilities cannot be fully realized without effective handling of continuous variables. However, the application of discretization and transformers to all healthcare datasets may not be necessary. In cases involving minimal continuous variables, or wherein such variables have a minor impact on the dependent variables of predictive models, disregarding them may not result in significant differences in performance.

One of the most challenging aspects of applying and enabling AI in healthcare is ensuring the security and accessibility of the data involved. In light of regulations, particularly in cases where data is highly protected, a closed analytics environment with no external access is required. In such environments, the opportunity for research into the latest methodologies is severely limited, which can give rise to a number of challenges. The most realistic solution to these issues is the generation of synthetic data using non-explicit methodologies. Synthetic data can mimic the statistical properties of real clinical data while safeguarding patient privacy[33,34]. Beyond privacy concerns, the scarcity of training data in real-world healthcare environments exacerbates the problem. Collecting and processing such data is often time-consuming and costly, posing significant burdens for researchers. In this regard, synthetic data offers a viable alternative, enabling the augmentation of limited datasets to improve model performance efficiently. Many studies demonstrated that synthetic data increased the volume of scarce training data and improved the performance of the model[35–38]. In this way, TT-GAN has the potential to effectively address the ongoing challenge of data scarcity in healthcare, while ensuring robust privacy protection. In practice, several platforms have been developed to generate and share synthetic health data, ensuring privacy while enabling AI research. For example, in the United States, Synthea[39], developed by the MITRE Corporation, provides open-source synthetic patient data that mimics real-world EHRs to support disease modeling and public health research. Similarly, in the United Kingdom, the Simulacrum[40], created by Health Data Insight CIC, synthesizes cancer research data and makes it safely accessible for researchers. In Korea, AI-Hub[41], managed by the National Information Society Agency (NIA), generates a variety of synthetic medical datasets, including medical text and images, to support AI model development. Together, these platforms, led by their respective institutions, address key challenges in accessing and utilizing sensitive healthcare data while accelerating advancements in AI-driven healthcare solutions.

Consequently, synthetic data is expected to make a significant contribution to the development and activation of healthcare AI. However, it should be noted that each synthesis algorithm has its own advantages and disadvantages, and sufficient consideration of the direction pursued by each synthetic data algorithm and a clear definition of the target data should always be given. Nonetheless, our study has certain limitation. While explicit methodologies are generally considered to outperform non-explicit methodologies for non-sensitive data, prospective experimental validation using a comprehensive set of algorithms and benchmark datasets is required to objectively verify this claim.

Conclusion

This study proposed TT-GAN as a specialized GAN algorithm for healthcare within the practical constraints of clinical settings. The TT-GAN operated on a devised three-stage framework: discretization, generation, and conversion stages. The discretization and converter methodology were the primary process applied to transform continuous variables into categorical data, thereby facilitating the subsequent vectorization process for the transformer of the generator. The entire dataset was cast in a categorical format, thereby enabling the Transformer to capture the unique attributes associated with each value. Subsequently, the original continuous data of the generated dataset were reconverted into continuous data by applying a prediction model. The integration of the Transformer encoder into the GAN framework ensured that the relational characteristics between the columns were preserved during the generation process. In particular, the TT-GAN exhibited better performance than the representative algorithms of CTGAN, and copulaGAN.

Finally, the TT-GAN effectively produced mixed variable types, including multinomial, discrete, and continuous, which closely resemble the characteristics of the original HTD. In particular, the discretization and converter methodology could be interpreted as a demonstration of the potential of the existing LLM model to be used effectively with a wide variety of data.

Data availability

Anyone can use the original data after registering as a member on the Korea Central Cancer Registry (KCCR) portal[16] and passing through the data application and review. Users need to fill out an application form, including a research proposal describing how they will use the data and that the data access request will be accessed by the KCCR and the National Statistics Office. All synthetic data can be shared for research purposes by contacting the authors. Please note that this service is only available to Koreans; it is a domestic service.All code for data generation and validation associated with the current submission is available in a GitHub repository[27].

Received: 30 October 2024; Accepted: 4 March 2025
Published online: 25 March 2025

References

1. Borisov, V. et al. Deep neural networks and tabular data: a survey. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, 1–21. https://doi.org/10.1109/TNNLS.2022.3229161 (2022).
2. de Kok, J. W. T. M. et al. A guide to sharing open healthcare data under the General Data Protection Regulation. *Sci. Data* **10**:404. https://doi.org/10.1038/s41597-023-02256-2 (2023).
3. Hernandez, M., Epelde, G., Alberdi, A., Cilla, R. & Rankin, D. Synthetic data generation for tabular health records: a systematic review. *Neurocomputing* **493**, 28–45. https://doi.org/10.1016/j.neucom.2022.04.053 (2022).
4. Giuffrè, M. & Shung, D. L. Harnessing the power of synthetic data in healthcare: innovation, application, and privacy. *Npj Digit. Med.* **6**, 186. https://doi.org/10.1038/s41746-023-00927-3 (2023).
5. Rankin, D. et al. Reliability of supervised machine learning using synthetic data in health care: model to preserve privacy for data sharing. *JMIR Med. Inf.* **8**, e18910. https://doi.org/10.2196/18910 (2020).
6. Xu, L., Skoularidou, M., Cuesta-Infante, A. & Veeramachaneni, K. Modeling tabular data using conditional GAN. *Adv. Neural Inf. Process. Syst.* **32**, 117 (2019).

7. Quiroz, J. C. et al. Development and validation of a machine learning approach for automated severity assessment of COVID-19 based on clinical and imaging data: retrospective study. *JMIR Med. Inf.* **9**, e24572. https://doi.org/10.2196/24572 (2021).
8. Syed, A. R. P. et al. Implementation of ensemble machine learning algorithms on exome datasets for predicting early diagnosis of cancers. *BMC Bioinform.* **23**, 496. https://doi.org/10.1186/s12859-022-05050-w (2022).
9. Kang, H. Y. J. et al. Synthetic tabular data based on generative adversarial networks in health care: generation and validation using the divide-and-conquer strategy. *JMIR Med. Inf.* **24**, e47859. https://doi.org/10.2196/47859 (2023).
10. Khan, A. & Swaleha, Z. Expansion of regularized k means discretization machine learning approach in prognosis of dementia progression. In *11th Int Conf Comp Commun Netw Technol (ICCCNT)* (2020).
11. Garcia, S., Luengo, J., Sáez, J. A., Lopez, V. & Herrera, F. A survey of discretization techniques: taxonomy and empirical analysis in supervised learning. *IEEE Trans. Knowl. Data Eng.* **25**, 734–750 (2012).
12. Ho, T. K. Random decision forests. In *Proc. 3rd Int. Conf. Doc. Anal. Recog* (1995).
13. Dorogush, A. V., Vasily, E. & Andrey, G. CatBoost: gradient boosting with categorical features support. *ArXiv Preprint* arXiv:1810.11363 (2018).
14. Chen, T. & Carlos, G. XGBoost: a scalable tree boosting system. In *Proc 22nd ACM SIGKDD int Conf Knowl Discov Data Min* (2016).
15. Guolin, K. et al. LightGBM: a highly efficient gradient boosting decision tree. *Adv. Neural Inf. Process. Syst.* ;**30**, 178 (2017).
16. Home page. Korea Central Cancer Registry (2024, accessed 8 Mar 2024). https://kccrsurvey.cancer.go.kr/index.do.
17. Ansari, A. F., Scarlett, J. & Soh, H. A characteristic function approach to deep implicit. generative modeling. In *Proc IEEE/CVF Conf Comp Vis Pattern Recog* (2020).
18. Subakan, C. & Oluwasanmi Ko, Paris, S. Learning the base distribution in implicit generative models. *ArXiv Preprint* arXiv:1803.04357 (2018).
19. Goodfellow, I. et al. Generative adversarial networks. *Commun. ACM* **63**(11), 139–144 (2020).
20. Bengio, Y., Laufer, E., Alain, G. & Yosinski, J. Deep generative stochastic networks trainable by backprop. In *International Conference on Machine Learning* 226–234 (PMLR, 2014).
21. Ren, Y., Zhu, J., Li, J. & Luo, Y. Conditional generative moment-matching networks. *Adv. Neural. Inf. Process. Syst.* **2016**, 29 (2016).
22. Li, Y., Swersky, K. & Zemel, R. Generative moment matching networks. In *International Conference on Machine Learning* 1718–1727 (PMLR, 2015.
23. Zhang, Y., Zaidi, N. A., Zhou, J. & Li, G. GANBLR: a tabular data generation model. *IEEE Int. Conf. Data Min. (ICDM)* **2021**, 181. https://doi.org/10.1109/ICDM51629.2021.00103 (2021).
24. Zhang, Y., Zaidi, N., Zhou, J. & Li, G. GANBLR++ Incorporating capacity to generate numeric attributes and leveraging unrestricted Bayesian networks. In *Proc 2022 SIAM Int Conf Data Mining (SDM), Society for Industrial and Applied Mathematics* (2022).
25. Han, P. et al. *C3-TGAN-controllable Tabular Data Synthesis With Explicit Correlations and Property Constraints* (Authorea Preprints, 2023).
26. Miletic, M. & Sariyar, M. Challenges of using synthetic data generation methods for tabular microdata. *Appl. Sci.* **14**(14), 5975. https://doi.org/10.3390/app14145975 (2024).
27. Huang, X., Khetan, A., Cvitkovic, M. & Karnin, Z. Tabtransformer: tabular data modeling using contextual embeddings. *ArXiv Preprint* arXiv:2012.06678 (2020).
28. Gorishniy, Y., Rubachev, I., Khrulkov, V. & Babenko, A. Revisiting deep learning models for tabular data. *Adv. Neural Inf. Process. Syst.* **34**, 18932–18943 (2021).
29. Solatorio, A. V., Dupriez, O. & REaLTabFormer generating realistic relational and tabular data using Transformers. *ArXiv Preprint* arXiv:2302.02041 (2023).
30. Diao, S., Shen, X., Shum, K., Song, Y. & Zhang, T. TILGAN: Transformer-based implicit latent GAN for diverse and coherent text generation. *Find. Ass Comput. Linguist ACL-IJCNLP* **2021**, 4844–4858 (2021).
31. Li, X., Metsis, V., Wang, H. & Ngu, A. H. H. Tts-gan: a transformer-based time-series generative adversarial network. *Int. Conf. Artif. Intell. Med.* **2022**, 133–143 (2022).
32. Gulati, M. & Roysdon, P. TabMT: generating tabular data with masked Transformers. *Adv. Neural. Inf. Process. Syst.* **2024**, 36. (2024).
33. Venugopal, R. et al. Privacy preserving generative adversarial networks to model electronic health records. *Neural Netw.* **153**, 339–348 (2022).
34. Ramzan, F., Sartori, C., Consoli, S. & Reforgiato Recupero, D. Generative adversarial networks for synthetic data generation in finance: evaluating statistical similarities and quality assessment. *AI* **5**(2), 667–685 (2024).
35. Scroggins, J. K., Topaz, M., Song, J. & Zolnoori, M. Does synthetic data augmentation improve the performances of machine learning classifiers for identifying health problems in patient–nurse verbal communications in home healthcare settings? *J. Nurs. Scholar.* (2024).
36. Waheed, A. et al. Covidgan: data augmentation using auxiliary classifier Gan for improved covid-19 detection. *Ieee Access.* **8**, 91916–91923 (2020).
37. Wan, C. & Jones, D. T. Protein function prediction is improved by creating synthetic feature samples with generative adversarial networks. *Nat. Mach. Intell.* **2**(9), 540–550 (2020).
38. Kim, H. et al. Synthetic data improve survival status prediction models in early-onset colorectal cancer. *JCO Clin. Cancer Inf.* **8**, e2300201. https://doi.org/10.1200/CCI.23.00201 (2024).
39. Walonoski, J. et al. Synthea: an approach, method, and software mechanism for generating synthetic patients and the synthetic electronic health care record. *J. Am. Med. Inform. Assoc.* **25**(3), 230–238 (2018).
40. Health Data Insight CiC. The Simulacrum: a synthetic dataset derived from anonymous cancer data provided by the National Disease Registration Service, NHS England (2024, accessed 8 Mar 2024). https://simulacrum.healthdatainsight.org.uk.
41. AI-Hub. Synthetic data for AI research (2024, accessed 8 Mar 2024). https://aihub.or.kr.
42. Kwang, S. R. Sally/ttgan. GitHub. https://github.com/KwangSun-Ryu/Sally.git (2024).

Acknowledgements

This study was supported by a grant (no: 2310440-3) offered by the National Cancer Center of Korea, Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (no: NRF-2022R1F1A107504).

Author contributions

Conceptualization was managed by HYJK, MSK, and KSR; methodology, HYJK, MSK, and KSR; validation, HYJK, MSK, and KSR; investigation, HYJK; data curation, HYJK and KSR; writing of the original draft preparation, HYJK, and KSR. All the authors assisted in drafting and editing the manuscript.

Competing interests
The authors declare no competing interests.

Additional information
Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-025-93077-3.

Correspondence and requests for materials should be addressed to K.S.R.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.