electronics

MDPI

Article

Keyword Data Analysis Using Generative Models Based on Statistics and Machine Learning Algorithms

Sunghae Jun

Department of Data Science, Cheongju University, Cheongju 28503, Chungbuk, Republic of Korea; shjun@cju.ac.kr; Tel.: +82-10-7745-5677; Fax: +82-43-229-8432

Abstract: For text big data analysis, we preprocessed text data and constructed a document–keyword matrix. The elements of this matrix represent the frequencies of keywords occurring in a document. The matrix has a zero-inflation problem because many elements are zero values. Also, in the process of preprocessing, the data size of the document–keyword matrix is reduced. However, various machine learning algorithms require a large amount of data, so to solve the problems of data shortage and zero inflation, we propose the use of generative models based on statistics and machine learning. In our experimental tests, we compared the performance of the models using simulation and practical data sets. Thus, we verified the validity and contribution of our research for keyword data analysis.

Keywords: data shortage; generative model; synthetic data; keyword data; zero inflation; machine learning

Citation: Jun, S. Keyword Data Analysis Using Generative Models Based on Statistics and Machine Learning Algorithms. *Electronics* **2024**, *13*, 798. https://doi.org/10.3390/electronics13040798

Academic Editors: Wanfu Gao, Yuzhou Liu and Ping Zhang

Received: 15 January 2024
Revised: 14 February 2024
Accepted: 18 February 2024
Published: 19 February 2024

1. Introduction

Keyword data analysis has been actively performed in various big data fields [1–3]. This is because a significant portion of big data consists of text-based data. To carry out a text data analysis, we preprocess the text data, such as a document, and extract the keywords from the preprocessed text data by text mining techniques [4,5]. In general, we construct a document–keyword matrix, with documents and keywords corresponding to its rows and columns [1,5–7]. Each element of the matrix is the frequency value of a keyword occurring in a document. Figure 1 shows a document–keyword matrix [7].

	communication	computing	cylinder	damage	data	database	deep	depth	detection
1	0	0	0	0	1	0	0	0	0
2	0	0	0	0	39	0	0	0	0
3	0	0	0	0	0	0	0	0	0
4	0	0	0	0	1	0	0	0	5
5	0	0	0	0	1	0	0	0	0
6	0	0	0	0	2	0	0	0	1

	device	diagnosis	digital	display	earth	earthquake	echo	edge	electric	energy
1	0	0	0	0	0	0	0	0	1	0
2	7	0	2	3	0	0	0	0	0	0
3	0	13	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	1	0
6	2	0	0	0	0	0	0	0	0	0

	engine	engineering	environment	estimation	fault	feedback	fire	flow	fluid
1	0	0	1	0	1	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	3	0	0	0	12	0	0	1	0
4	0	0	7	0	0	0	15	0	0
5	0	0	0	0	0	0	1	0	0
6	0	0	0	0	1	0	0	0	0

Figure 1. A document–keyword matrix.

Even if a large amount of text big data is collected, the size of the data set is reduced through preprocessing to build structured data which can be analyzed by statistics and machine learning. Additionally, as can be seen in Figure 1, the preprocessed text data contain

many zero values because a keyword that is included only once among all documents is assigned to one column in the document–keyword data.

In the previous research attempting to overcome the zero-inflation problem in text data analysis, Uhm and Jun (2022) proposed a generative model based on statistics by synthpop to address the zero-inflation problem in patent data analysis [6]. Jun (2023) studied generative adversarial network (GAN) and statistical modeling [1]. He compared the performance of GAN with the statistical model without GAN. Park and Jun (2023) used compound Poisson models to perform zero-inflation patent data analysis [7]. This was not a generative model but an exponential dispersion model. In this paper, to solve the problems of zero inflation as well as data shortage, we propose a keyword data analysis using generative models based on statistics and machine learning. Using these generative models, we generate synthetic data from the original data, and add the synthetic data to the original training and test data sets for keyword data analysis by various machine learning algorithms such as deep learning, linear modeling, Bayesian neural networks, etc.

Our paper consists of the following sections. In Section 2, we outline the background to our research, such as keyword data analysis and generative modeling. We propose the generative models based on statistics and machine learning for keyword data analysis in Section 3. In the next section, we carry out the experiments using simulation and patent document data to show the performance results for the compared models. We explain the conclusions and contributions of our paper in the last section.

2. Research Background

2.1. Keyword Data Analysis

In this paper, we employ keyword data analysis to find the relationship between the keywords extracted from various text documents. Generally, we can extract the keywords from documents using term frequency (tf), inverse document frequency (idf) and domain experts' knowledge and analyze them [4,5]. Figure 2 shows the process of keyword data analysis [4,5].

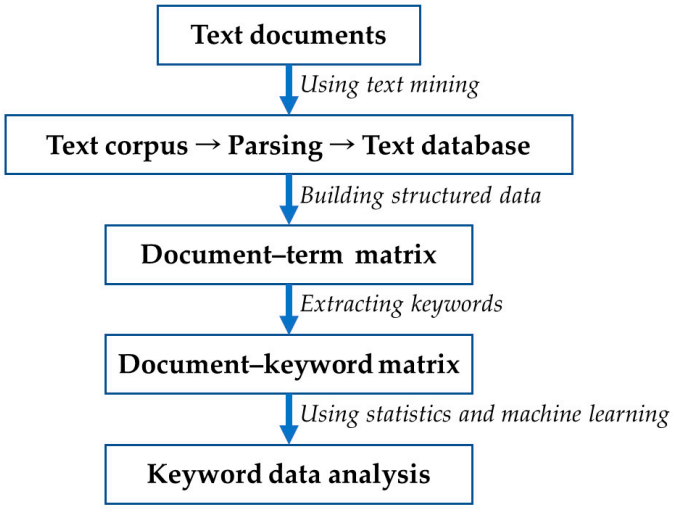

Figure 2. Process of keyword data analysis.

In the process of keyword data analysis, we first collect text documents to be analyzed. Using text mining techniques, we create a text corpus, carry out parsing and construct a text database [5]. Next, we build a document–term matrix to serve as structured data for statistical analysis and machine learning. Finally, we extract keywords from the document–term matrix for keyword data analysis. In general, the data size of the document–keyword matrix constructed from collected text documents is reduced as it goes through preprocessing for text mining. Therefore, we need new methods to solve the data shortage

problem. In this paper, we study the use of generative models to increase the data size of a document–keyword matrix.

2.2. Generative Modeling

A generative model is a probabilistic and machine learning model to generate synthetic data that resemble given original data [8–10]. Figure 3 illustrate the process of generative modeling [8].

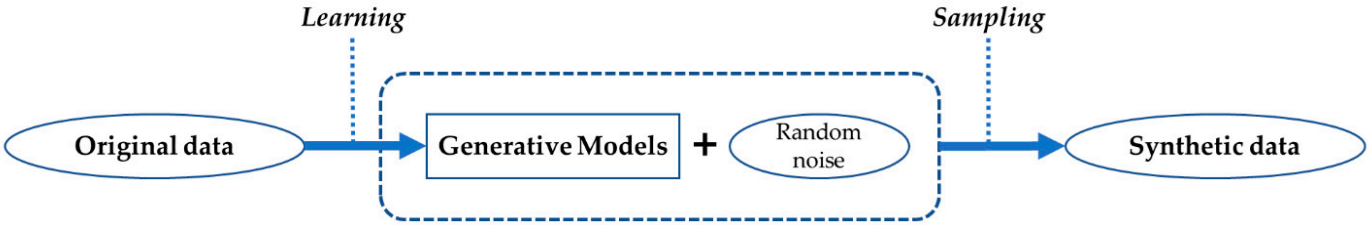

Figure 3. Generative modeling.

By learning from the original data, we can construct generative models. To generate the synthetic data, we sample new data from the generative models with random noise [11]. The generative models use a probability distribution such as normal with mean (μ) and variance (σ^2) [12,13]. That is, the generative models are not deterministic and they generate different data every time. Therefore, we have to estimate the parameters of the probability distribution to explain the original data well. In the generative modeling, we estimate the density $p(x)$ of the observed data (x) [14,15]. Currently, various studies are being actively conducted on generative models based on statistics and machine learning [16–22]. In the field of biology, single-cell genomic data also take the form of large count matrices characterized by a high occurrence of zeros. This presents the same issue as zero inflation in our study. Therefore, various tailored analysis methods have been investigated to address the unique challenges posed by such data [23,24]. The following two studies focus on generative models to solve the zero-inflation problem that arises in the analysis of single-cell genomic data. The first, by Liu et al. (2021), introduced one of the most widely adopted approaches to this problem [24]. The second, by Ji et al. (2023), detailed a method that employs a generative model in the single-cell genomic data analysis [23]. These references serve as a valuable entry point for extensive research on the analysis of single-cell genomic data using generative models. Most studies on generative models focus on image data, but in this study, we deal with generative models for count data because the document–keyword matrix consists of the frequency values of keywords occurring in documents.

2.3. Zero-Inflation Problems in Keyword Data Analysis

For keyword data analysis, we construct the document–keyword matrix according to the process of keyword data analysis illustrated in Figure 2. As we explained in Figure 1, there is a zero-inflation problem in the matrix. Many previous works have been conducted to solve the problem. Most of them were based on statistical methods. The zero-inflated count model is a popular method to analyze keyword data with zero inflation. This is defined as follows [25]:

$$P(X = x) = \begin{cases} \pi + (1 - \pi) f(0) & , x = 0 \\ (1 - \pi) f(x) & , x > 0 \end{cases} \quad (1)$$

where π is the probability of zero. In Equation (1), the zero-inflated model $P(X = x)$ consists of two components, zero and non-zero parts. Also, $f(x)$ is a base model of probability mass function (pmf). In the zero-inflated count model, we use Poisson and negative binomial distributions for the pmf, and we call them zero-inflated Poisson (ZIP) and zero-inflated negative binomial (ZINB) models [26]. To improve the models based on ZIP and ZINB, Bayesian inference has been applied to the zero-inflated count model [27,28].

Neelon and Chung (2017) used the Bayesian latent factor model for the ZIP, and called this LZIP. They compared the LZIP with ZIP, and verified the improved performance of LZIP in the illustrative examples [27]. Moriña et al. (2021) proposed a method to analyze zero-inflated binomial data using Bayesian inference [28]. Seo and Hwang (2022) studied a Bayesian inference for ZINB regression model [29]. Another approach to overcome the zero-inflation problem is methods based on machine learning, such as classification and regression trees or generative models [1,6]. In this paper, we conduct research on generative models based on statistics and machine learning algorithms to solve the zero-inflation problem that occurs during keyword data analysis. Also, in our experiments, we compare the performance between models with synthetic data generation using the generative models based on statistical methods and machine learning algorithms.

3. Proposed Method

Much big data is in text form. Therefore, we have to extract the keywords from text data and analyze them by methods such as constructing document–keyword matrices, N-grams and correlation analysis between keywords, sentiment analysis, topic modeling, etc. [4,5]. The first task to be performed in keyword data analysis is to collect text documents on a given topic. As explained in Figure 2, the collected document data are preprocessed using text mining and natural language processing techniques. The preprocessed text document data set has a frequency matrix structure in which the rows and columns are documents and terms, respectively, as shown in Figure 1. This matrix is called the document–term matrix [5]. Next, we extract the keywords from the document–term matrix and construct the document–keyword matrix, as shown in Table 1.

Table 1. Document–keyword matrix and frequency element.

Data Matrix	**Keyword$_1$**	**Keyword$_2$**	$\cdots$	**Keyword$_p$**
Document$_1$	Frequency$_{11}$	Frequency$_{12}$	$\cdots$	Frequency$_{1p}$
Document$_2$	Frequency$_{21}$	Frequency$_{22}$	$\cdots$	Frequency$_{2p}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
Document$_n$	Frequency$_{n1}$	Frequency$_{n2}$	$\cdots$	Frequency$_{np}$

In Table 1, the $Frequency_{ij}$ is the frequency value of $Keyword_j$ occurring in $Document_i$. Through the data preprocessing, the size of the initially collected document data set gradually decreases, and when the document–keyword matrix is finally constructed, the data set sometimes becomes so small that it is difficult to analyze. In addition, many elements of the matrix are zero values, as shown in Figure 1. Not only the data shortage but also the zero-inflation problem must be solved in keyword data analysis [1,6,7,25,26,28,30]. To solve these problems, we use generative models based on statistics and machine learning. First, we consider the synthpop package of R data language to generate synthetic data [31,32]. This is a generative model based on statistics. In the synthpop modeling, the original input data are represented as follows.

$$X_i = (X_{i1}, X_{i2}, \ldots, X_{ip}), \quad i = 1, 2, \ldots, n \tag{2}$$

In the data of (2), p is the number of input variables. The synthpop model uses classification and regression trees (CART) to generate the synthetic data [16,31]. We generate the synthetic data for the current variable using the previous variables as follows: We start with the second variable and exclude the first variable [31]. We generate X_j by running $(X_1, X_2, \ldots, X_{j-1})$ using CART growth. That is, we sample X_j from the CART model, $P(X_j|X_1, X_2, \ldots, X_{j-1})$ [16]. Figure 4 explains the method of synthetic data generation by the synthpop in our keyword data analysis.

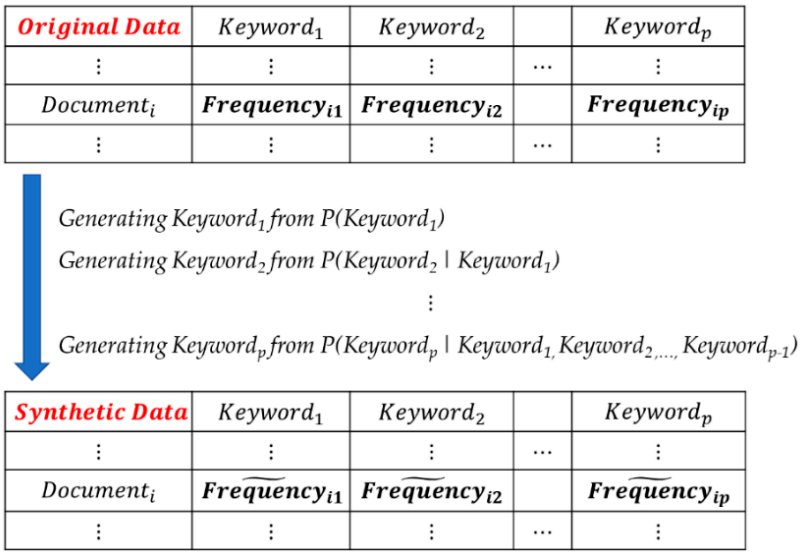

Figure 4. Generating a synthetic document–keyword matrix using synthpop.

In Figure 4, we fit statistical models of the keywords to the original data and generate synthetic data that is completely new data of the keywords. We denote original and synthetic data as $Frequency_j$ and $\widetilde{Frequency}_j$. Using the original data, we find the joint probability distribution of keywords. We represent the probability distribution of $Keyword_j$ as Equation (3).

$$P\left(Keyword_j \middle| Keyword_1,\ Keyword_2, \ldots, Keyword_{j-1}\right) \tag{3}$$

In Equation (3), we generate the $Keyword_j$ from the conditional probability distribution of $Keyword_j$ given $\left(Keyword_1,\ Keyword_2, \ldots, Keyword_{j-1}\right)$. The synthpop begins by estimating the probability distribution of the first keyword, $P(Keyword_1)$. Next, we generate new data $Keyword_1$ (synthetic) that resembles $Keyword_1$ (original) by $P(Keyword_1)$. That is, the synthetic data $\widetilde{Frequency}_1$ represent the original data $Frequency_1$. Using this result, we build the conditional distribution $P\left(Keyword_2 | Keyword_1\right)$ and generate the synthetic data $Frequency_2$ for $Keyword_2$ by the conditional distribution. In this way, we generate the final synthetic data $\widetilde{Frequency}_p$.

Unlike the synthpop, the generative model based on machine learning is performed by deep neural networks [9,33]. In this paper, we use GAN for the generative model for document–keyword data generation. GAN is a machine learning model which generates new synthetic data that resemble the given original data [9,11–13,17]. GAN performs an adversarial training process, involving two neural networks, a generator and discriminator. The formula of GAN is defined as follows [9].

$$V(D,G) = E_{p(x)}(logD(x)) + E_{q(z)}(log(1 - D(G(z)))) \tag{4}$$

where D and G represent the generator and discriminator respectively. x and z are input data and random noise, which follow a normal distribution. In Equation (4), z also is the latent representation of x. $p(x)$ and $q(z)$ are the generative and latent models. The discriminator wants to maximize $V(D,G)$, and on the other side, the generator tries to minimize $V(D,G)$. In this paper, the document–keyword matrix is used as x in Equation (4). Figure 5 illustrates the process of generating a synthetic document–keyword matrix using GAN.

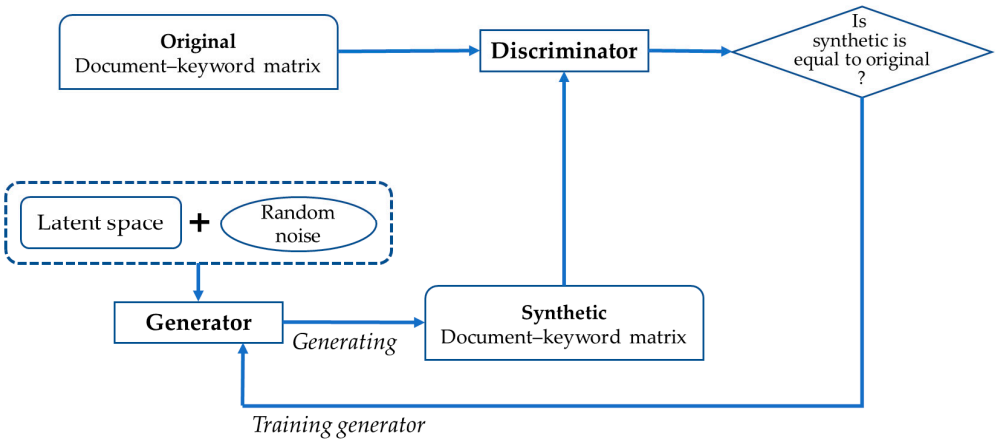

Figure 5. Generating synthetic document–keyword matrix using GAN.

In Figure 5, the generator creates a new synthetic document–keyword matrix from latent space and random noise. Also, the generator tries to make the synthetic data as similar as possible to the original data. The latent space is a learning space representing sample data with low dimension. Sample data that are similar to each other are located close to each other in the latent space. We select the initial data point from the latent space and add random noise to the data. Thus, we generate the synthetic document–keyword matrix from the latent space and random noise. The discriminator predicts whether the input data are real (original) or fake (not original). Sampling the document–keyword data randomly from original and synthetic data sets, and combining the sample data, the discriminator uses this data to learn to accurately classify real and fake. The generator is trained so that the discriminator can judge the synthetic data as original. Ultimately, the generator aims to generate synthetic data to the extent that the discriminator cannot distinguish whether the synthetic data are original or not. When training the entire model, only the weights of generator should be updated and the weights of discriminator should not be updated so that the synthetic data with good performance are generated. Next, we combine the results of synthpop and GAN. Figure 6 shows the synthetic data generation.

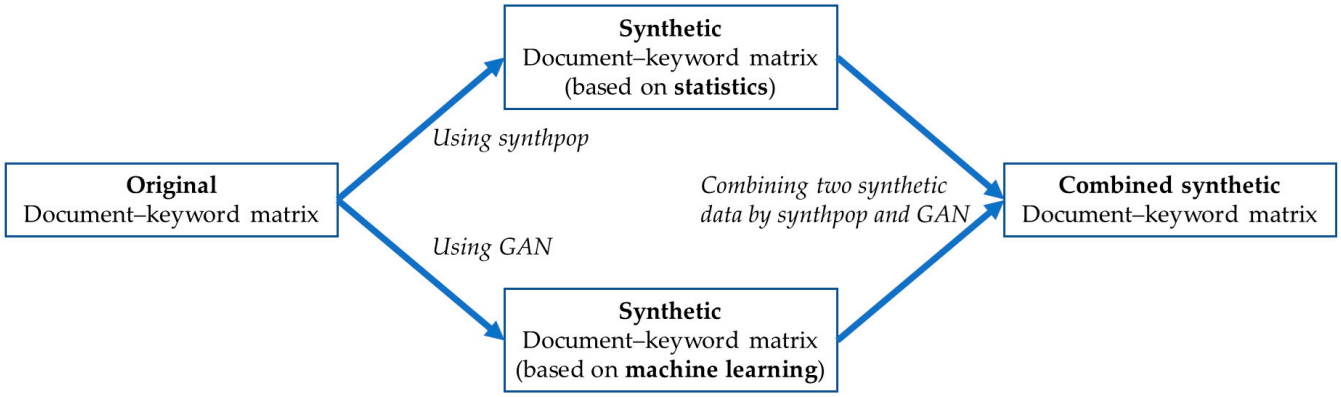

Figure 6. Generating synthetic document–keyword matrix using synthpop and GAN.

Our third generative model combines the two data sets generated by synthpop and GAN. In Figure 6, synthpop and GAN generate synthetic document–keyword data sets based on statistics and machine learning, respectively. Therefore, we use the three generative models to analyze the keyword data. In our keyword data analysis, we build the linear model shown in Equation (5).

$$Keyword_Y = b_0 + b_1 Keyword_1 + b_2 Keyword_2 + \cdots + b_k Keyword_k \quad (5)$$

where $(Keyword_1, Keyword_2, \ldots, Keyword_k)$ are k explanatory keywords and $Keyword_Y$ is a response keyword. Each variable of (5) represents the frequency value of a keyword. To evaluate the performance between generative models, in this paper, we divide the given data into training (70%) and test (30%) data sets. Using the training data, we calculate the Akaike information criterion (AIC) and use this value to compare the explanatory power of the linear model [34]. AIC is a measure used in predictive modeling to verify the goodness of fit of a model, as shown in Equation (6) [35,36].

$$AIC = -2ln(argmaxL(\theta; X)) + 2p \tag{6}$$

where θ and X represent the model parameter and data, respectively. p is the number of model parameters. $argmaxL(\theta; X)$ is maximum likelihood estimator of θ. The better the fitting performance of model, the smaller the AIC value. We use another measure, mean squared error (MSE), to evaluate performance of the compared models. MSE is defined as shown in Equation (7) [34–37].

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{7}$$

where y_i and $\hat{y}_i$ are real and predicted values, respectively, and n is the size of the given data. We calculate MSE value of each model using the test data. The smaller the MSE value of the model, the better its predictive performance. In the next section, we perform a performance comparison between the compared generative models using MSE and AIC.

4. Experimental Results

4.1. Simulation Data Analysis

In keyword data analysis, each element of a document–keyword matrix is the frequency value of keyword occurring in document. This count data follows a Poisson distribution [7,38]. Thus, we generated the random numbers from the multivariate generalized Poisson distribution [39,40]. We used the R data language and R package for generating simulation data and data analysis [32,40]. For our simulation data for keyword data analysis, we considered the mixture of Poisson distributions with two parameters, rate and dispersion [39–41]. In our simulation study, we consider a linear model with six variables (keywords), as shown in Equation (8).

$$Y = b_0 + b_1X_1 + b_2X_2 + \cdots + b_5X_5 \tag{8}$$

where Y is the response keyword and $(X_1, X_2, \ldots, X_5)$ are explanatory keywords. Each keyword in $(Y, X_1, X_2, \ldots, X_5)$ follows a Poisson distribution with different parameters. Also, they are correlated with each other. Figure 7 shows the correlation structure of the generated simulation data.

	Y	X1	X2	X3	X4	X5
Y	1.0000	0.3483	0.2838	0.3673	0.5441	0.7884
X1	0.3483	1.0000	0.2654	0.1771	0.4443	0.3164
X2	0.2838	0.2654	1.0000	0.6299	0.7000	0.2189
X3	0.3673	0.1771	0.6299	1.0000	0.7487	0.4517
X4	0.5441	0.4443	0.7000	0.7487	1.0000	0.5862
X5	0.7884	0.3164	0.2189	0.4517	0.5862	1.0000

Figure 7. Correlation matrix of a simulation data set.

In Figure 7, we can see that among the explanatory variables, some variables have a high correlation with the response variable Y, while others have a relatively low correlation. Also, in Table 2, we illustrate the zero ratio of each variable in the simulation data.

Table 2. Zero ratio of each variable in a simulation data set.

Variable	Y	X_1	X_2	X_3	X_4	X_5
Zero ratio	14.3%	22.7%	57.0%	31.7%	52.6%	39.0%

We can see that the percentage of zeros in X_2 and X_4 is more than 50%. In this way, we created simulation data so that the zero-inflation problem could be included in the original data. Next, we utilized the generative models to create synthetic data and analyze them based on the linear model in (7). In addition, we calculated the AIC and MSE values of the constructed models using original and synthetic data sets. We show the AIC values of the compared models in Table 3.

Table 3. AIC values of compared models: simulation data.

Iteration	Original	Synthpop	GAN	SynthGAN
1	1842.19	1324.18	509.89	1815.29
2	1867.63	1350.04	953.32	2100.60
3	1837.09	1341.35	768.61	2264.31
4	1859.86	1343.14	706.52	2080.11
5	1862.00	1356.58	645.50	2201.26
6	1859.98	1345.17	462.33	2061.93
7	1846.71	1330.72	591.87	2071.94
8	1858.55	1334.23	403.80	1920.64
9	1836.82	1329.28	534.35	1794.90
10	1871.46	1340.30	620.48	1993.39
11	1861.15	1356.74	536.33	2006.04
12	1834.99	1326.46	530.13	2474.01
13	1854.99	1317.95	563.45	2332.94
14	1845.23	1337.51	374.92	1690.55
15	1852.55	1349.32	479.86	2029.23
16	1849.44	1353.55	425.15	1866.02
17	1870.77	1336.73	695.17	2102.80
18	1849.38	1334.34	709.39	2183.69
19	1872.85	1358.24	651.02	1930.75
20	1867.92	1314.11	653.04	1832.50
21	1815.57	1329.93	602.27	1839.60
22	1848.14	1345.97	728.53	2044.62
23	1885.86	1356.95	618.60	2343.37
24	1865.49	1339.58	468.00	2270.44
25	1857.93	1337.17	652.76	2059.51
26	1886.26	1334.46	468.81	2074.06
27	1833.16	1347.06	483.89	1991.80
28	1881.45	1374.20	590.23	2248.10
29	1876.67	1358.93	551.00	2093.37
30	1857.03	1307.41	635.69	2003.78

In Table 3, We built the linear model in Equation (7) for the four data sets and calculated the AIC value for each model. The elements of original data are count data because they were generated from a generalized Poisson distribution. The synthetic data generated by synthpop from the original data are also count data because they are created by a joint distribution of frequency count data. However, the synthetic data set created by GAN is a continuous data type because the generator of GAN creates the new data from the latent space of a normal distribution and random noise. In this paper, it is possible to use continuous data because the constructed linear model of keywords allows such data. Lastly, we combined the synthetic data sets from synthpop and GAN (SynthGAN). Figure 8 is a visualization of the AIC values in Table 3.

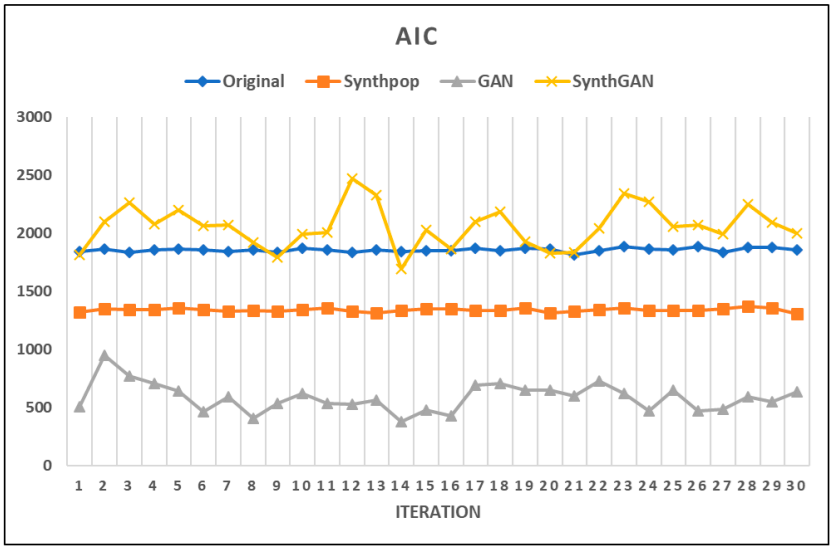

Figure 8. AIC plot of compared models.

The AIC values of the GAN synthetic data are the smallest. The values for the synthpop synthetic data are also smaller than the original data, but the AIC values for synthGAN are mostly larger than the original data. Therefore, from the perspective of model fitting, we were able to confirm the validity of using the synthetic data by synthpop and GAN. Although the AIC values of the GAN synthetic data are smaller than those of synthpop, the dispersion of the AIC values is larger than for the original. Also, in Figure 8, we can see that there is a large variation in the AIC values of the models based on GAN. This is because random noise was used in the synthetic data generation process using the GAN model, thereby increasing the variance in the data. Next, we present the MSE results of models for the comparative data sets in Table 4.

By comparing the MSE values of the models, we can check the prediction performance between the compared models. We confirmed in Table 4 that the difference in MSE values for the compared models was not large. Figure 9 shows the MSE plot of compared models with the original and synthetic data sets.

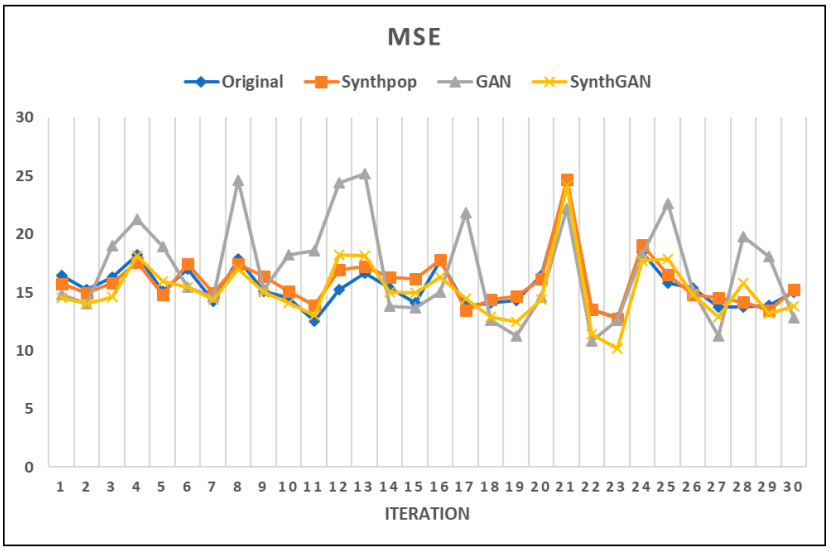

Figure 9. MSE plot of compared models.

Table 4. MSE values of compared models: simulation data.

Iteration	Original	Synthpop	GAN	SynthGAN
1	16.47	15.73	14.82	14.54
2	15.22	14.94	14.06	14.04
3	16.29	15.83	19.01	14.60
4	18.21	17.60	21.25	18.15
5	15.11	14.84	18.90	15.92
6	16.98	17.43	15.46	15.46
7	14.23	14.94	14.59	14.42
8	17.87	17.42	24.64	17.01
9	15.10	16.36	15.22	15.04
10	14.52	15.12	18.19	14.12
11	12.57	13.90	18.58	13.05
12	15.26	16.94	24.43	18.20
13	16.67	17.20	25.19	18.12
14	15.35	16.33	13.80	15.04
15	14.11	16.16	13.70	14.98
16	17.73	17.77	15.01	16.31
17	13.85	13.43	21.88	14.48
18	14.09	14.41	12.60	12.89
19	14.29	14.65	11.26	12.50
20	16.43	16.14	14.61	14.39
21	24.38	24.66	22.22	24.21
22	13.54	13.52	10.86	11.39
23	12.81	12.78	12.63	10.21
24	18.45	19.08	18.31	17.74
25	15.80	16.50	22.61	17.89
26	15.35	14.78	15.19	14.82
27	13.75	14.52	11.23	12.86
28	13.76	14.20	19.77	15.80
29	13.89	13.43	18.07	13.18
30	15.04	15.25	12.84	13.81

In many repeated experiments, we confirmed that the MSE values for the GAN synthetic data were calculated to be larger than those for the remaining three comparison data sets in Figure 9. Therefore, in terms of the predictive performance, the use of synthetic data by synthpop is not significantly different from the original data. Thus, we confirmed the validity of using the synthetic data by synthpop. In addition, we can see that the MSE values of the GAN model are distributed over a wide range. Similar to the results in Figure 8, this is because we use random noise from the latent space when we generate reproduction data using the GAN model. Table 5 shows a summary of the MSE values of the original and synthetic data sets.

Table 5. Summary of MSE values.

Data	Min	Q1	Median	Mean	Q3	Max
Original	12.57	14.10	15.16	15.57	16.46	24.38
Synthpop	12.78	14.55	15.49	15.86	16.83	24.66
GAN	10.86	13.87	15.34	17.03	19.58	25.19
SynthGAN	10.21	13.87	14.71	15.17	16.21	24.21

In Table 5, Q1 and Q3 are the percentiles of 25% and 75%, respectively. The mean value of the synthetic data from GAN is the largest among the compared models. Also, the minimum value in the GAN synthetic data is smaller than that of the original and synthpop data, and the maximum value in the GAN data is larger than those of the other data sets. Therefore, we can see that the performance of the GAN synthetic data was poor compared with the other models and that the dispersion of GAN synthetic data was large. Therefore,

considering the results of simulation data analysis, although the AIC values of the synthetic data generated by GAN are small compared with those of the other data sets, the MSE values are large and their dispersion is also large. We found that synthpop synthetic data is the most appropriate for the linear prediction model for keyword data analysis.

4.2. Practical Data Analysis

To show how the proposed research can be applied to a practical domain, we used the patent document data related to artificial intelligence (AI) technology for disaster and security for text data. We searched the patent documents from the United States Patent and Trademark Office and the Korea Intellectual Property Rights Information Service [42,43]. The collected patent documents were filed until 2022 in patent databases around the world. We obtained a total of 16,875 valid patents through the patent verification process. Figure 10 shows the valid patent data.

	A	B	C	D	E
1	country	title	abstract	claim	date
2	CN	One kind being suitable	It is suitable for GIS electronic mutual	1. one kinc	2015-01-21
3	CN	Three-dimensional liquid	The invention discloses three-dimensic	1. a kind o	2011-06-02
4	CN	A kind of efficient fault c	The invention discloses a kind of effici	1. a kind o	2016-01-08
5	CN	The detection method o	The detection method of liquid combi	1. the dete	2015-12-10
6	CN	The method constructed	The invention discloses a kind of meth	1. a kind o	2016-03-15
7	CN	Pivot degree of associati	The present invention relates to a kin	1. a kind o	2016-03-31
8	CN	A kind of multiple anten	The present invention discloses a kind	1. a kind o	2016-06-21
9	CN	Drilling power simulated	A kind of drilling power simulated tes	1. a kind o	2015-03-13
10	CN	The foreseeable method	The present invention relates to a kin	1. a kind o	2015-03-20
11	CN	A kind of schedulable ca	The invention discloses a kind of sche	1. a kind o	2016-07-15
12	CN	Tunnels and undergroun	It dashes forward discharge disaster p	The discha	2016-07-29
13	CN	A kind of image defoggi	The present invention discloses a kind	1. a kind o	2016-01-07
14	CN	A kind of water cooler fa	A kind of method that the present inv	1. a kind o	2016-09-20
15	CN	A kind of high mountain	A kind of high mountain permafrost c	1. a kind o	2016-11-25

Figure 10. Valid patent documents.

In this experiment, we selected the title and abstract from the patent data in Figure 10 for keyword data analysis. Using text mining techniques, we constructed the document–keyword matrix from the valid patent documents, as shown in Figure 11.

```
abnormal acoustic air alarm amplitude analysis antenna artificial audio automatic band battery beam big
       0        0   0     0         0        1       0          0     0         0    0       0    0   0
cable camera car channel cloud cluster coal communication computing cylinder damage data database deep
    0      0   0       0     0       0    0             0         0        0      0    1        0    0
depth detection device diagnosis digital display earth earthquake echo edge electric energy engine
    0         0      0         0       0       0     0          0    0    0        1      0      0
engineering environment estimation fault feedback fire flow fluid forecast frame fuzzy gas geological
          0           1          0     1        0    0    0     0        0     0     0   1          0
grid health hole human image information intelligence interaction interface land language laser layer
   1      0    1     0     0           0            0           0         0    0        0     0     0
learning life light lightning liquid machine magnetic map measurement memory metal mobile monitoring
       0    1     0         0      0       4        0   0           6      0     0     0           0
natural negative network neural node normal oil optical parallel patient pattern physical picture pipe
      0        0       0      0    0       0   0       0         0       0       0         0       0    0
pipeline pixel plane platform power prediction pressure probability protection protocol pulse pump radar
      11     0     0        6     0          0         0           0          0         0     0    0     0
radio remote risk road robot rock sampling satellite scale scanning sea security seismic sensor signal
    0      0    0    0     0    0        0         0     0        0   0        0       0      8      1
software soil space spatial speed stability statistics steel stream surface switch tank temperature time
       0    0     0       0     0         0          0     0       0       0      3    0           3    0
transmission tree tunnel turbine ultrasonic underground user valve vehicle velocity video virtual visual
            0    0       0       0          0          0    0     0       0        0     0       0      0
voice voltage warning water wave waveform wavelet weather web wheel wind wire wireless
    0       5       0     0    0        0       0       0   0     0    0    0        0
```

Figure 11. Our document–keyword matrix of AI patents.

Figure 11 shows the first rows of 162 keywords in a document–keyword matrix. We carried out a 10% sampling from the matrix with 16,875 rows (documents). To compare

the performance between the original and synthetic data sets, we used a linear model, as shown in Equation (9).

$$Analysis = b_0 + b_1 Data + b_2 Image + b_3 Information + b_4 Signal + b_5 Time \quad (9)$$

We selected *Analysis* for the response keyword, and *Data*, *Image*, *Information*, *Signal* and *Time* for the explanatory keywords. We show the AIC values of the compared models in Table 6.

Table 6. AIC values of compared models: patent data.

Iteration	Original	Synthpop	GAN	SynthGAN
1	5820.07	5498.04	1292.23	7796.35
2	5898.53	5663.81	1712.42	7954.24
3	5708.29	5869.78	1180.56	8230.99
4	6023.27	5768.70	950.62	8105.72
5	6052.42	6080.04	479.14	8068.74
6	5562.51	5294.46	72.43	7525.59
7	5749.34	5543.17	1770.41	8002.39
8	6010.72	6118.29	340.51	8382.23
9	5724.70	5803.23	2149.38	8497.16
10	5476.35	5503.69	2076.72	8048.08
11	5969.25	6187.38	2039.60	8526.81
12	5867.03	5894.44	244.00	8061.97
13	5664.92	5917.38	2346.98	8421.98
14	6023.38	5958.67	1092.81	8024.77
15	6124.64	5896.04	133.84	8029.70
16	5837.25	5256.60	2182.39	7843.76
17	6054.10	5787.18	202.72	7933.51
18	6127.58	6033.42	−40.01	8125.56
19	5375.92	5056.72	−401.05	7419.40
20	5680.26	5939.15	1260.75	8036.63

In Table 6, as for the results of the simulation data, the AIC values of the synthetic data by GAN are the smallest in all data sets, both original and synthetic. In addition, the AIC values of the original and synthpop synthetic data are similar to each other. The AIC values of synthetic data by synthGAN are larger than the original and synthpop values. Therefore, we confirmed that synthpop synthetic data can well replace original data. Next, we calculated the MSE values of the compared models. We show the experimental results in Table 7.

Similar to the experimental results using simulated data, we found that the GAN synthetic data had the largest MSE values in the comparison using practical patent data. Also, the MSE dispersion of the GAN synthetic data is larger than that of the others. In comparison to this, the MSE values of the synthpop synthetic data are similar to the results of the original data. We were able to see that the MSE results for synthGAN were also similar to the results for the original and synthpop data. Therefore, we confirmed that the synthpop generative model is the most efficient way to perform synthetic data generation for keyword data analysis.

Table 7. MSE values of compared models: patent data.

Iteration	Original	Synthpop	GAN	SynthGAN
1	145.43	143.57	21,154.99	141.82
2	140.85	136.58	5080.91	138.63
3	166.23	168.30	8292.04	168.86
4	143.40	145.94	488.06	145.54
5	157.62	158.19	228.39	157.76
6	129.52	129.01	406.02	128.80
7	161.39	161.01	15,904.39	160.53
8	154.98	156.24	1118.97	157.03
9	188.42	189.59	9794.08	188.36
10	171.78	174.88	2378.39	172.65
11	132.72	135.06	15,602.14	133.90
12	144.67	144.91	167.14	142.12
13	131.58	128.93	28,130.54	129.93
14	148.64	148.91	2768.86	147.38
15	165.18	166.39	245.63	165.89
16	139.04	139.93	11,970.70	141.56
17	152.31	153.70	920.31	149.10
18	172.88	174.17	173.62	169.22
19	151.85	153.83	201.37	150.83
20	169.04	171.53	1306.92	169.17

5. Discussion

In this paper, we tried to solve the zero-inflation problem that occurs in the process of keyword data analysis. We carried out two experiments using simulation and practical models. From the experimental results, we found that the performances of the synthpop and GAN models were better than the original model. This means that using the generative models based on statistics and machine learning is better than not using them. Also, the performance of the GAN model was better than that of synthpop. For example, all the AIC values of GAN were smaller than the values of the synthpop model in the simulation data analysis of Figure 8. This was similar to the results for AIC values when comparing models in the practical data analysis of Table 6. The GAN model has the best performance due to its characteristics as a generative model. When we generate the synthetic data using the GAN model, we generally perform random sampling from a normal distribution representing the latent space. Thus, most of the generated data values are distributed around the mean. Because the variance of the synthetic data generated by GAN is relatively small compared with other generative models, the performance of the compared linear model is stable and shows excellent explanatory power.

Currently, generative models are actively used in various machine learning domains. In this study, we used this model to solve the zero-inflation problem that occurs during the analysis of keyword data extracted from text documents. Of course, generative models show excellent performance in the image data field, but we showed the utility and improved performance of generative models in the field of numerical data including zero. We generated synthetic data from the generative models and analyzed them by statistical methods such as linear regression. Finally, we overcame the zero-inflation problem using the generative models for keyword data analysis. We expect that our research will be used more broadly to solve data sparsity problems, including the zero-inflation problem. Finally, we found that using simulation data to evaluate the performance of generative models can lead to over-optimistic conclusions. Therefore, we decided that experiments using more diverse practical data would be necessary to efficiently evaluate the performance of the generative model.

6. Conclusions

The aim of our study is to generate synthetic data and analyze it for prediction. Most studies related to generative models are interested in the generative model itself that

creates synthetic data, but our focus is on constructing a predictive model by analyzing data sampled from the constructed generative models. For this reason, AIC and MSE were used to evaluate the performance of the model in this paper. Thus, we generated and analyzed more synthetic data. In the process of keyword data analysis, we face the problems of a lack of data and zero inflation. These problems become factors that degrade the performance of machine learning models for keyword data analysis. To solve these problems, we proposed the use of generative models. We considered generative models based on statistics and machine learning, such as synthpop, GAN and synthGAN, in this paper. Also, we compared the model performance between the original and synthetic data sets using the measures of AIC and MSE. From the experimental results using simulation data and practical patent documents, we verified the better performance of generative modeling by synthpop compared with the other models. We also found that the AIC values of the GAN synthetic data were the smallest among the compared models. However, its MSE values were not the smallest but the largest. Due to these results, we confirmed the difficulty in analyzing keyword data using the synthetic data by GAN.

In this paper, we conclude that the synthpop generative model is the best method to generate synthetic data for keyword data analysis. Of course, the generative model using GAN is also an excellent model from the AIC perspective, but its performance is poor from the MSE perspective. Therefore, in our future works, we will study new methods to improve the MSE of GAN synthetic data for keyword data analysis. We will add Bayesian learning or various probability distributions to traditional GAN to improve the performance of generative models using GAN in keyword data analysis. Our research contributes a method for creating new synthetic data in text big data analysis. This is necessary because the data size is reduced during the preprocessing of text data. In particular, a sufficient amount of data is required to perform large-scale machine learning such as deep learning. Therefore, synthetic data from generative models will contribute to solving the lack of original data to perform machine learning.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available on request from the corresponding author (status: privacy).

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Jun, S. Zero-Inflated Text Data Analysis using Generative Adversarial Networks and Statistical Modeling. *Computers* **2023**, *12*, 258. [CrossRef]
2. Shin, H.; Lee, H.J.; Cho, S. General-use unsupervised keyword extraction model for keyword analysis. *Expert Syst. Appl.* **2023**, *233*, 120889. [CrossRef]
3. Bzhalava, L.; Kaivo-oja, J.; Hassan, S.S. Digital business foresight: Keyword-based analysis and CorEx topic modeling. *Futures* **2024**, *155*, 103303. [CrossRef]
4. Julia, S.; Robinson, D. *Text Mining with R*; O'Reilly: Sebastopol, CA, USA, 2017.
5. Feinerer, I.; Hornik, K. *Package 'tm' Version 0.7-11, Text Mining Package*; CRAN of R Project, R Foundation for Statistical Computing: Vienna, Austria, 2023.
6. Uhm, D.; Jun, S. Zero-Inflated Patent Data Analysis Using Generating Synthetic Samples. *Future Internet* **2022**, *14*, 211. [CrossRef]
7. Park, S.; Jun, S. Zero-Inflated Patent Data Analysis Using Compound Poisson Models. *Appl. Sci.* **2023**, *13*, 4505. [CrossRef]
8. Foster, D.; Friston, K. *Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play*, 2nd ed.; O'REILLY: Sebastopol, CA, USA, 2023.
9. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 1–9.
10. Bau, D.; Liu, S.; Wang, T.; Zhu, J.Y.; Torralba, A. Rewriting a deep generative model. In Proceedings of the 16th European Conference on Computer Vision–ECCV, Glasgow, UK, 23–28 August 2020; pp. 351–369.
11. Deng, L.; He, C.; Xu, G.; Zhu, H.; Wang, H. PcGAN: A Noise Robust Conditional Generative Adversarial Network for One Shot Learning. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 25249–25258. [CrossRef]
12. Li, C.; Xu, K.; Zhu, J.; Liu, J.; Zhang, B. Triple Generative Adversarial Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 9629–9640. [CrossRef] [PubMed]

13. Rosenfeld, B.; Simeone, O.; Rajendran, B. Spiking Generative Adversarial Networks with a Neural Network Discriminator: Local Training, Bayesian Models, and Continual Meta-Learning. *IEEE Trans. Comput.* **2022**, *71*, 2778–2791. [CrossRef]
14. Ruthotto, L.; Haber, E. An introduction to deep generative modeling. *GAMM-Mitteilungen* **2021**, *44*, e202100008. [CrossRef]
15. Zhou, X.; Hu, Y.; Wu, J.; Liang, W.; Ma, J.; Jin, Q. Distribution Bias Aware Collaborative Generative Adversarial Network for Imbalanced Deep Learning in Industrial IoT. *IEEE Trans. Ind. Inform.* **2023**, *19*, 570–580. [CrossRef]
16. Nowok, B.; Raab, G.M.; Dibben, C. synthpop: Bespoke Creation of Synthetic Data in R. *J. Stat. Softw.* **2016**, *74*, 1–26. [CrossRef]
17. Xu, M.; Baraldi, P.; Lu, X.; Zio, E. Generative Adversarial Networks with AdaBoost Ensemble Learning for Anomaly Detection in High-Speed Train Automatic Doors. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 23408–23421. [CrossRef]
18. Yan, C.; Chang, X.; Li, Z.; Guan, W.; Ge, Z.; Zhu, L.; Zheng, Q. ZeroNAS: Differentiable Generative Adversarial Networks Search for Zero-Shot Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 9733–9740. [CrossRef]
19. Tang, C.; He, Z.; Li, Y.; Lv, J. Zero-Shot Learning via Structure-Aligned Generative Adversarial Network. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 6749–6762. [CrossRef] [PubMed]
20. You, H.; Cheng, Y.; Cheng, T.; Li, C.; Zhou, P. Bayesian Cycle-Consistent Generative Adversarial Networks via Marginalizing Latent Sampling. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 4389–4403. [CrossRef] [PubMed]
21. Li, H.; Prasad, R.G.N.; Sekuboyina, A.; Niu, C.; Bai, S.; Hemmert, W.; Menze, B. Micro-Ct Synthesis and Inner Ear Super Resolution via Generative Adversarial Networks and Bayesian Inference. In Proceedings of the IEEE 18th International Symposium on Biomedical Imaging (ISBI), Nice, France, 13–16 April 2021; pp. 1500–1504.
22. Yan, R.; Yuan, Y.; Wang, Z.; Geng, G.; Jiang, Q. Active Distribution System Synthesis via Unbalanced Graph Generative Adversarial Network. *IEEE Trans. Power Syst.* **2022**, *38*, 4293–4307. [CrossRef]
23. Ji, X.; Tsao, D.; Bai, K.; Tsao, M.; Xing, L.; Zhang, X. scAnnotate: An automated cell-type annotation tool for single-cell RNA-sequencing data. *Bioinform. Adv.* **2023**, *3*, vbad030. [CrossRef] [PubMed]
24. Liu, Q.; Chen, S.; Jiang, R.; Wong, W.H. Simultaneous deep generative modelling and clustering of single-cell genomic data. *Nat. Mach. Intell.* **2021**, *3*, 536–544. [CrossRef]
25. Cameron, A.C.; Trivedi, P.K. *Regression Analysis of Count Data*, 2nd ed.; Cambridge University Press: New York, NY, USA, 2013.
26. Hilbe, J.M. *Modeling Count Data*; Cambridge University Press: New York, NY, USA, 2014.
27. Neelon, B.; Chung, D. The LZIP: A Bayesian Latent Factor Model for Correlated Zero-Inflated Counts. *Biometrics* **2017**, *73*, 185–196. [CrossRef] [PubMed]
28. Moriña, D.; Puig, P.; Navarro, A. Analysis of zero inflated dichotomous variables from a Bayesian perspective: Application to occupational health. *BMC Med. Res. Methodol.* **2021**, *21*, 277. [CrossRef]
29. Seo, G.T.; Hwang, B.S. A Bayesian zero-inflated negative binomial regression model based on Pólya-Gamma latent variables with an application to pharmaceutical data. *Korean J. Appl. Stat.* **2022**, *35*, 311–325.
30. Sidumo, B.; Sonono, E.; Takaidza, I. Count Regression and Machine Learning Techniques for Zero-Inflated Overdispersed Count Data: Application to Ecological Data. *Ann. Data Sci.* **2023**. [CrossRef]
31. Nowok, B.; Raab, G.M.; Snoke, J.; Dibben, C.; Nowok, M.B. *Package 'synthpop' Ver. 1.8–0, Generating Synthetic Versions of Sensitive Microdata for Statistical Disclosure Control*; CRAN of R Project, R Foundation for Statistical Computing: Vienna, Austria, 2022.
32. R Development Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria; Available online: http://www.R-project.org (accessed on 1 October 2023).
33. Neunhoeffer, M. *Package 'RGAN' Version 0.1.1, Generative Adversarial Nets (GAN) in R*; CRAN of R Project, R Foundation for Statistical Computing: Vienna, Austria, 2022.
34. Montgomery, D.C.; Peck, E.A.; Vining, G.G. *Introduction to Linear Regression Analysis*; John Wiley & Sons: Hoboken, NJ, USA, 2012.
35. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; MIT Press: Cambridge, MA, USA, 2012.
36. Theodoridis, S. *Machine Learning A Bayesian and Optimization Perspective*; Elsevier: London, UK, 2015.
37. Bruce, P.; Bruce, A.; Gedeck, P. *Practical Statistics for Data Scientists*; O'Reilly Media: Sebastopol, CA, USA, 2020.
38. Makowski, M.; Piotrowski, E.W. Transactional Interpretation and the Generalized Poisson Distribution. *Entropy* **2022**, *24*, 1416. [CrossRef] [PubMed]
39. Li, H.; Demirtas, H.; Chen, R. RNGforGPD An R Package for Generation of Univariate and Multivariate Generalized Poisson Data. *R J.* **2020**, *12*, 173–188. [CrossRef]
40. Li, H.; Chen, R.; Nguyen, H.; Chung, Y.; Gao, R.; Demirtas, H. *Package 'RNGforGPD' Version 1.1.0, Random Number Generation for Generalized Poisson Distribution*; CRAN of R Project, R Foundation for Statistical Computing: Vienna, Austria, 2022.
41. Li, X.-J.; Tian, G.-L.; Zhang, M.; Ho, G.T.S.; Li, S. Modeling Under-Dispersed Count Data by the Generalized Poisson Distribution via Two New MM Algorithms. *Mathematics* **2023**, *11*, 1478. [CrossRef]
42. USPTO. The United States Patent and Trademark Office. Available online: http://www.uspto.gov (accessed on 1 October 2023).
43. KIPRIS. Korea Intellectual Property Rights Information Service. Available online: www.kipris.or.kr (accessed on 1 October 2023).