A COMPREHENSIVE SURVEY ON GENERATIVE DIFFUSION MODELS FOR STRUCTURED DATA

Heejoon Koo

University College London heejoon.koo.17@alumni.ucl.ac.uk

To Eun Kim

Carnegie Mellon University toeunkim@cmu.edu

July 11, 2023

ABSTRACT

In recent years, generative diffusion models have achieved a rapid paradigm shift in deep generative models by showing groundbreaking performance across various applications. Meanwhile, structured data, encompassing tabular and time series data, has been received comparatively limited attention from the deep learning research community, despite its omnipresence and extensive applications. Thus, there is still a lack of literature and its reviews on structured data modelling via diffusion models, compared to other data modalities such as visual and textual data. To address this gap, we present a comprehensive review of recently proposed diffusion models in the field of structured data. First, this survey provides a concise overview of the score-based diffusion model theory, subsequently proceeding to the technical descriptions of the majority of pioneering works that used structured data in both data-driven general tasks and domain-specific applications. Thereafter, we analyse and discuss the limitations and challenges shown in existing works and suggest potential research directions. We hope this review serves as a catalyst for the research community, promoting developments in generative diffusion models for structured data.

Keywords Survey · Diffusion Models · Generative Models · Structured Data · Tabular Data · Time Series Data

1 Introduction

Structured data is characterised as its standardised format and ubiquitous across various domains. This type of data can be divided into two categories. The first is tabular data, where information is arranged into rows and columns, representing individual records and their respective attributes. The second category is time series data, which is sequential observations obtained at successive time intervals. The structured data has been extensively applied to many tasks, such as financial modelling [1], fraud detection [2], click-through rate (CTR) prediction [3], clinical event prediction [4], counterfactual estimation [5], and so forth. Enhancing predictive performance and robustness in these applications can provide significant benefits for both end users and organisations offering such solutions. Thus, structured data modelling has been a long-standing research topic in both academia and industry.

Over the past decade, deep learning has revolutionised numerous fields, including computer vision and language modelling [6, 7, 8, 9, 10, 11, 12, 13, 14, 15]. This substantial advancement can be largely attributed to data-driven deep learning technologies, which have received considerable focus from the research community [16]. Nevertheless, traditional machine learning techniques are still widely used for structured data and the volume of literature on structured data via deep learning is relatively insufficient. This is primarily due to the challenges that deep learning methodologies face when applied to structured data. First, datasets related to structured data are generally smaller than those for visual or textual data, thus limiting the full exploitation of deep learning's expressiveness. As a result, traditional machine learning methods are still being broadly utilised [17, 18]. Moreover, dataset complications, such as mixed type of data (both continuous and categorical types), the absence of correlation amongst columns and rows, and the necessity for domain knowledge-guided feature engineering, have made the application of deep learning to structured data a complex task [19]. Kadra *et al.* [19] thus referred to structured data as the final *unconquered castle* in deep learning research community. However, structured data modelling via deep generative models including variational auto-encoders (VAE)

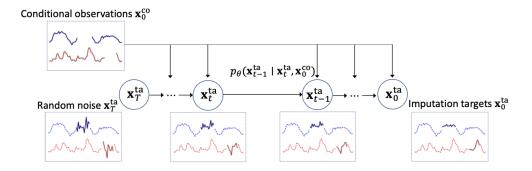


Figure 1: A Process of Time Series Imputation via Diffusion Models. The reverse process p_{θ} progressively removes random noise to generate plausible time series data, conditioned on observed values $\mathbf{x}_{0}^{\text{co}}$. The dashed lines represent observed values, while the solid lines are targets for imputation, denoted as $\mathbf{x}_{0}^{\text{ta}}$, where t corresponds to a specific diffusion step out of the total step T. This illustration is derived from CSDI [39] to provide an intuitive understanding of time series modelling through diffusion models.

[20] and generative adversarial networks (GAN) [21] has been continually explored for applications such as data synthesis against privacy concerns [22, 23], scenario-based simulations [24], and imputations [25]. Additionally, such generative modelling has also been explored to improve predictive performance on structured data.

Score-based diffusion models [26, 27, 28] recently have become very prominent in various domains and applications, due to their superior capabilities compared to previous deep generative model families. Diffusion models were initially introduced by Sohl-Dickstein *et al.* [29], inspired by non-equilibrium statistical physics. Subsequently, they are further developed by [26, 27], verifying their potential in comparison with other state-of-the-art generative models in image synthesis tasks. Hereafter, they have shown exceptional performance across a variety of challenging tasks in different domains, *e.g.*, inverse problems [7, 15], text driven image synthesis and editing [10, 11], language modelling [12, 14], 3D molecule generation [30, 31]. However, albeit its growing body of research, the attention on structured data modelling through diffusion models still remains insufficient. With the aims of promoting future research, we provide a holistic overview of the generative diffusion models for structured data in this paper.

The remaining structure of this paper is outlined as follows. As preliminaries, a brief introduction on backgrounds of generative diffusion models is described in Section 2. Then, this survey dichotomises the existing works into two main categories: data-driven general tasks (Section 3) and domain-specific applications (Section 4). Each section provides an overview and describes cutting-edge works along with their key technical novelties. In Section 5, the limitations and challenges shown in existing works are discussed with potential research directions. Finally, we conclude the survey in Section 6.

What Sets Our Survey Apart from Others There are several existing surveys on diffusion models, either covering algorithmic developments, various data modalities and applications [32, 33], or focusing on specific data modalities such as vision [34], language [35], and graphs [36], or concentrating on medical imaging [37] in a domain specific manner. Lin *et al.* [38] is covering structured data, but their work is limited to time series applications. Recognising the gap in the literature review concerning diffusion models for the structured data, this work aims to address this deficiency by presenting the *first* comprehensive survey dealing with structured data, including both tabular and time series data, and their related applications.

2 Backgrounds on Score-based Diffusion Models

Score-based diffusion models are a class of probabilistic generative models that learn to reversal of the data destruction processes that gradually injects noise, to yield high-quality and realistic synthesised data samples. In other words, the training procedure involves two steps: the forward diffusion process and the subsequent backward denoising process. Despite their diverse applications across various data modalities, the design of the forward and backward processes categorises the current research into three main frameworks: denoising diffusion probabilistic models (DDPMs) [27, 29], score-based models (SGMs) [26], and stochastic differential equations (SDEs) [28]. Therefore, this section provides a concise review on three subcategories of the diffusion models with mathematical formulae. We present only the essential derivations, thus we recommend referring to the original papers for comprehensive equations.

2.1 Denoising Diffusion Probabilistic Models (DDPMs)

DDPMs design the forward and reverse processes via dual Markov chains [27, 29]. The forward process involves the diffusion of data with pre-determined noise, such as Gaussian noise, whilst the reverse process employs deep neural networks to sequentially eliminate noise and recovers the original data.

Forward Pass Suppose that there is a clean data point $\mathbf{x_0}$ drawn from a data distribution $q(\mathbf{x_0})$. Then, the forward diffusion process progressively perturbs the clean original data distribution by adding Gaussian noise, ultimately converging towards the standard Gaussian distribution \mathbf{z}_T . During the diffusion step T, noised latent variables $\mathbf{x_1}$, $\mathbf{x_2}$, ..., \mathbf{x}_T are yielded. In other words, it generates \mathbf{x}_T using sequential transition kernel $q(\mathbf{x}_t|\mathbf{x}_{t-1})$, which can be formulated as below:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) := \mathcal{N}(\mathbf{x}_t; \sqrt{1 - \beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{I}), \ \forall t \in \{1, \cdots, T\}$$
(1)

thus, the forward process is defined by using a series of transition kernels:

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) := \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$$
(2)

where $\beta_T \in (0,1)$ is a variance schedule that controls step sizes, and \mathbf{I} is the identity matrix with the same dimension as input data $\mathbf{x_0}$. Also, $\mathcal{N}(\mathbf{x}; \mu, \sigma)$ is the Gaussian distribution of \mathbf{x} with the mean μ and covariance σ . Let $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^t \alpha_s$ and ϵ as a Gaussian noise, it becomes feasible to yield a noisy sample from an arbitrary step from the distribution, conditioned on the original input $\mathbf{x_0}$ [40] as follows:

$$q(\mathbf{x}_t|\mathbf{x}_0) := \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, \sqrt{1 - \bar{\alpha}_t}\mathbf{I}), \tag{3}$$

where
$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon$$
. (4)

Reverse Process With the forward process, the reverse process removes the noise at each step in reverse time direction until the destroyed original data is reconstructed. We start with $p_{\theta}(\mathbf{x}_T)$ to generate $p_{\theta}(\mathbf{x}_0)$ that obeys the true data distribution $q(\mathbf{x}_0)$. Again, $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. A series of reverse Gaussian transition kernels p_{θ} , where θ denotes the learnable parameters, is parameterised in the form of a deep neural network as:

$$p_{\theta}(\mathbf{x}_{0:T}) := p(\mathbf{x}_T) \prod_{t=1}^{T} p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t), \tag{5}$$

$$p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \mu_{\theta}(\mathbf{x}_t, t), \sigma_{\theta}(\mathbf{x}_t, t)\mathbf{I}). \tag{6}$$

Additionally, the model parameterises both the mean $\mu_{\theta}(\mathbf{x}_t, t)$ and variance $\sigma_{\theta}(\mathbf{x}_t, t)$. The model learns to approximate the true data distribution during the reverse process, which is achieved through optimising variational upper bound on negative log-likelihood (NLL):

$$\mathbb{E}\left[-\log p_{\theta}\left(\mathbf{x}_{0}\right)\right] \leq \mathbb{E}_{q}\left[-\log \frac{p_{\theta}\left(\mathbf{x}_{0:T}\right)}{q\left(\mathbf{x}_{1:T} \mid \mathbf{x}_{0}\right)}\right] = \mathbb{E}_{q}\left[-\log p\left(\mathbf{x}_{T}\right) - \sum_{t \geq 1} \log \frac{p_{\theta}\left(\mathbf{x}_{t-1} \mid \mathbf{x}_{t}\right)}{q\left(\mathbf{x}_{t} \mid \mathbf{x}_{t-1}\right)}\right] := -L_{\text{VLB}}.$$
 (7)

It can be rewritten using Kullback-Leibler divergence (KL divergence) as:

$$\mathbb{E}_{q} \left[\underbrace{D_{\mathrm{KL}}(q(\mathbf{x}_{T}|\mathbf{x}_{0})p(\mathbf{x}_{T}))}_{L_{T}} + \sum_{t>1} \underbrace{D_{\mathrm{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_{t},\mathbf{x}_{0}) \mid\mid p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_{t}))}_{L_{t-1}} \underbrace{-\log p_{\theta}(\mathbf{x}_{0}|\mathbf{x}_{1})}_{L_{0}} \right]. \tag{8}$$

The equation 8 is decomposed into three parts: L_T , L_{t-1} , and L_0 . First, L_T minimises the KL divergence between $q(\mathbf{x}_T|\mathbf{x}_0)$ and a standard Gaussian distribution $p(\mathbf{x}_T)$. Next, in L_{t-1} , the $p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)$ can be computed against

posteriors from the forward process. The last term L_0 denotes a negative log-likelihood. Especially, we condition L_{t-1} on \mathbf{x}_0 to make it tractable:

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \mathbf{I})$$
(9)

where
$$\tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1 - \bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t}\mathbf{x}_t \text{ and } \tilde{\beta}_t := \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t}\beta_t.$$
 (10)

Considering the equation 6, 9 and 10, the term L_{t-1} in equation 8 can be reformulated as:

$$L_{t-1} = \mathbb{E}_q \left[\frac{1}{2\sigma_t^2} \| \tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0) - \mu_\theta(\mathbf{x}_t, t) \|^2 \right] + C$$
(11)

where C does not depend on θ , thus considered as a constant. Especially, Ho *et al.* [27] highlight that instead of parameterising the mean $\mu_{\theta}(\mathbf{x}_t, t)$, predicting noise vector ϵ at each time step t in the forward process by parameterising $\epsilon_{\theta}(\mathbf{x}_t, t)$ enhances training efficiency and improves sample quality. They rewrite the equation 11 as below:

$$\mathbb{E}_{t \sim \mathcal{U}(1,T), \mathbf{x}_0 \sim q(\mathbf{x}_0), \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} \left[\lambda(t) \left\| \epsilon - \epsilon_{\theta}(\mathbf{x}_t, t) \right\|^2 \right]$$
(12)

where $\mathcal{U}(1,T)$ denotes a uniform distribution, $\lambda(t)$ is a weighting function to adjust the scales of noise evenly, and ϵ_{θ} is a deep parameterised model that predicts the Gaussian noise $\epsilon \sim \mathcal{N}(\mathbf{0},\mathbf{I})$. Once trained, it samples \mathbf{x}_0 that resembles the original data.

Whilst the DDPM can only be applied to continuous data, such as image and audio, its application can be extended beyond these types. Hoogeboom *et al.* [41] have introduced a multinomial diffusion, specifically designed for categorical data. This model corrupts the data using a discrete Markovian process, then restores the original data through a reversal process. It is particularly crucial in certain data modalities and applications in which the usage of categorical data types is involved: tables, text, segmentation maps, and such [35, 41, 42]. We provide an explanation of multinomial diffusion in the Appendix A.

2.2 Score-based Generative Models (SGMs)

Here, we employ a score function of a probabilistic density function, denoted as $p(\mathbf{x})$. The score function, $\nabla_{\mathbf{x}} \log p(\mathbf{x})$, is defined as the gradient of the logarithm of the probabilistic density with respect to the input \mathbf{x} . In order to estimate the score function, we train a deep neural network $s_{\theta}(\mathbf{x})$, where θ represents the learnable parameters and \mathbf{x} is the input data, to approximate the score of the original data distribution $p(\mathbf{x})$. It is mathematically formulated as following:

$$\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} ||s_{\theta}(\mathbf{x}) - \nabla_{\mathbf{x}} \log p(\mathbf{x})||_{2}^{2}.$$
(13)

Nevertheless, it is computationally infeasible to obtain $\nabla_{\mathbf{x}} \log p(\mathbf{x})$ in the context of high-dimensional data and deep neural networks. Thus, there are various works to address the issue: score matching [43], denoising score matching [26, 44], and sliced score matching [45].

In particular, noise-conditioned score network (NCSN) (a.k.a score matching with langevin dynamics, SMLD) [26] emphasise issues regarding manifold hypothesis. In circumstances where real-world data concentrate on the low-dimensional manifolds that are embedded within a high-dimensional space, the estimated score functions are inevitably imprecise in regions of low density. Thus, they propose to inject the random Gaussian noise to the original data with a sequence of intensifying scale, making the data distribution more amenable to SGMs, and estimate the score corresponding to each noise level. Mathematically, we have a sequence of Gaussian noise scales $0 < \sigma_1 < \sigma_2 < \cdots < \sigma_t < \cdots < \sigma_T$, thus $p_{\sigma_1}(\mathbf{x}) \approx p(\mathbf{x}_0)$, $p_{\sigma_T}(\mathbf{x}) \approx \mathcal{N}(\mathbf{x}; 0, \mathbf{I})$ and $p_{\sigma_t}(\mathbf{x}_t|\mathbf{x}) \approx \mathcal{N}(\mathbf{x}_t; \mathbf{x}, \sigma_t^2 \mathbf{I})$. Moreover, a single noise-conditioned score networks $s_{\theta}(\mathbf{x}, \sigma_t)$ aims to approximate the gradient logarithm density function $\nabla_{\mathbf{x}} \log p_{\sigma_t}(\mathbf{x})$. For a specific \mathbf{x}_t , the derivation of $\nabla_{\mathbf{x}} \log p_{\sigma_t}(\mathbf{x})$ is:

$$\nabla_{\mathbf{x}} \log p_{\sigma_t}(\mathbf{x}_t | \mathbf{x}) = -\frac{\mathbf{x}_t - \mathbf{x}}{\sigma_t}.$$
 (14)

In particular, the directions of the gradients are towards regions where the density of samples is high. Then, the combination of the equation 13 and 14 with a weighting function λ (σ_t) derives a new equation as:

$$\frac{1}{T} \sum_{t=1}^{T} \lambda\left(\sigma_{t}\right) \mathbb{E}_{p(\mathbf{x})} \mathbb{E}_{\mathbf{x}_{t} \sim p_{\sigma_{t}}(\mathbf{x}_{t}|\mathbf{x})} \left\| s_{\theta}\left(\mathbf{x}_{t}, \sigma_{t}\right) + \frac{\mathbf{x}_{t} - \mathbf{x}}{\sigma_{t}} \right\|_{2}^{2}.$$
(15)

With the annealed Langevin dynamics, new samples are generated by a progressive denoising process from the prior Gaussian distribution. The annealed Langevin dynamics exploits a Markov chain Monte Carlo (MCMC) to draw a sample from the distribution $p(\mathbf{x})$ using the score function $\nabla_{\mathbf{x}} \log p(\mathbf{x})$. It recursively samples \mathbf{x}_i as follows:

$$\mathbf{x}_{i} = \mathbf{x}_{i-1} + \frac{\gamma}{2} \nabla_{\mathbf{x}} \log p(\mathbf{x}) + \sqrt{\gamma} \omega_{i}, \forall i \in \{1, \cdots, N\}$$
 (16)

where $\omega_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, γ determines both magnitude and direction of the score update. After executing N iterations, the \mathbf{x} becomes a sample derived from the original distribution $p(\mathbf{x})$. Notably, in NCSN [26], the magnitude of ω_i undergoes gradual decrement, thereby subtly introducing uncertainty and preventing the model from mode failure.

2.3 Stochastic Differential Equations (SDEs)

Since the objective forms of both SGMs [26] and DDPMs [27] are similar, Song *et al.* [28] have integrated and further generalised these into a single framework where the number of noise scales is extended to infinity via stochastic differential equations (SDEs). The corresponding continuous diffusion process can be described using Itô SDE as:

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}, t \in [0, T].$$
(17)

where f is a drift coefficient of x(t) and g is a diffusion coefficient that is interwined with standard Wiener process w. Also, dt is a infinitesimal negative time step.

Similar to DDPMs and SGMs, $\mathbf{x_0}$ and \mathbf{x}_T denote data samples from the clean distribution p_0 and standard Gaussian distribution p_T , respectively. Accordingly, it synthesises new samples from the known prior distribution p_T , by solving the reverse-time SDE:

$$d\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - g^2(t)\nabla_{\mathbf{x}}\log p_t(\mathbf{x})\right]dt + g(t)d\bar{\mathbf{w}}$$
(18)

where $\bar{\mathbf{w}}$ is the reverse standard Brownian motion. The solution to reverse-time SDE is to approximate a time-dependent deep neural network $s_{\theta}(\mathbf{x},t)$ to a score function $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ via score matching objective function. Instead of directly approximating the score function which is computationally intractable, it estimates the transition probability $\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t | \mathbf{x}_0)$ that follows the Gaussian distribution during the forward diffusion process as:

$$\mathbb{E}_{t,\mathbf{x}_{0},\mathbf{x}_{t}}\left[\lambda(t)\left|\left|s_{\theta}\left(\mathbf{x}_{t},t\right)-\nabla_{\mathbf{x}_{t}}\log p_{0t}\left(\mathbf{x}_{t}|\mathbf{x}_{0}\right)\right|\right|_{2}^{2}\right].$$
(19)

Here, $p_{0t}(\mathbf{x}_t|\mathbf{x}_0)$ denotes the transition kernel of \mathbf{x}_t given \mathbf{x}_0 and $\lambda(t)$ is the weighting function. Upon completion of the training process, we can generate samples employing various techniques such as the Euler-Maruyama (EM), Prediction-Correction (PC), or Probability Flow ODE method.

EM solves the equation 18 by using a simple discretisation technique, where $d\mathbf{x}$ is substituted with Δt and $d\bar{\mathbf{w}}$ is replaced by the Gaussian noise $z \sim \mathcal{N}(\mathbf{0}, \Delta t\mathbf{I})$. In PC method, it operates in a sequential manner, alternating between predictor and corrector. The predictor can employ any numerical solver for the reverse-time SDE following a fixed discretisation strategy, such as the EM method. Subsequently, the corrector can be any score-based Markov chain Monte Carlo (MCMC) method, like annealed Langevin dynamics. Thus, the equation 16 can be solved using Langevin dynamics. In Probability Flow ODE method, the equation 17 can be reformulated into an ODE given by:

$$d\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - \frac{1}{2} g^2(t) \nabla_{\mathbf{x}} \log p_t(\mathbf{x}) \right] dt.$$
 (20)

This equation maintains the identical marginal probability density $p_t(t)$, as that of the SDE. Thus, sampling via solving the above reverse-time ODE is equivalent to solving the time reversal SDE.

3 Data-driven General Tasks

In this and subsequent sections, we delve into an in-depth review of diffusion-based methodologies for structured data, which are divided into two main categories: data-driven general tasks and domain-specific applications. For data-driven tasks, we focus on two data modalities: tabular and time series data. When it comes to tabular data modelling, we focus on generation and imputation, whilst for time series data, we delve into generation, imputation, and forecasting. Moving on to the domain-specific applications, we divide them into three categories: electronic health records (EHR), bioelectrical signal processing, and recommendation systems (RecSys). For EHR and bioelectrical signal processing, we explore both generation and specific tasks, with forecasting for EHR and enhancement for bioelectrical signal processing. Table 1 presents a quick summary on these categories along with the corresponding works. For further details, such as information on the generative modelling framework, specific datasets used in the experiments and links to accessible code, please refer to the Appendix B.

| Categories | Data Type | Task | Papers |
|------------------------------|---------------------------|-------------|----------------------------|
| Data-driven General Task | Tabular Data | Generation | TabDDPM [42] |
| | | | SOS [46] |
| | | | STaSy [47] |
| | | | CoDi [48] |
| | | Imputation | TabCSDI [49] |
| | Time Series Data | Generation | TSGM [50] |
| | | Imputation | CSDI [39] |
| | | | SSSD ^{S4} [51] |
| | | | DSPD/CSPD [52] |
| | | Forecasting | TimeGrad [53] |
| | | | ScoreGrad [54] |
| | | | SSSD ^{S4} [51] |
| | | | D ³ VAE [55] |
| | | | DSPD/CSPD [52] |
| Domain-specific Applications | Electronic Health Records | Generation | EHRDiff [56] |
| | | | TabDDPM [57] |
| | | | MedDiff [58] |
| | | | EHR-DPM [59] |
| | | Forecasting | TDSTF [60] |
| | Bioelectrical Signals | Generation | SSSD-ECG [61] |
| | | Enhancement | DeScoD-ECG [62] |
| | | | DS-DDPM [63] |
| | Recommendation Systems | | CODIGEM [64] |
| | | | DiffuRec [65] |
| | | | DiffRec (Du et al.) [66] |
| | | | DiffRec (Wang et al.) [67] |
| | | | CDDRec [68] |

Table 1: A hierarchical table on generative diffusion models for structured data

3.1 Tabular Data

Tabular data, an organised data format in a rectangular grid of rows and columns, is one of the most universal data types in real-world applications. Nevertheless, handling tabular datasets directly using deep learning has been obstructed by various challenges, such as potential privacy issues and missing information due to data storage or human errors [47, 49]. Thus, deep generative models (including VAE [20] and GAN [21]) have been investigated for both tabular data synthesis [22, 69] or imputation [25, 70]. To further improve the performance, researchers have started to use diffusion models, a new paradigm in generative models, for the generation and imputation of tabular data.

3.1.1 Generation

TabDDPM [42] is the first pioneering work in the field of tabular data synthesis. To tackle mixed-type characteristics of tabular data, it employs the Gaussian diffusion to model continuous features and multinomial diffusion to model categorical features [41]. To pre-process the mixed-type of data, they convert the continuous features using min-max scaler whilst they convert categorical features through one-hot encoding and process each feature using a separate

diffusion process. During post-processing, they apply reverse scaling when generating continuous variables. For categorical features, they use softmax function, followed by a rounding operator. These pre- and post-processing techniques are widely used by the most of the follow-up works.

TabDDPM [42] uses class-conditional model for classification datasets and inserts target values as an additional feature for regression datasets. A simple MLP architecture is optimised through a combination of mean squared error (MSE) and KL divergence, with each term tailored to continuous and categorical features. The equation is:

$$L_t^{TabDDPM} = L_t^{mse} + \frac{\sum_{i \le C} L_t^i}{C}$$
 (21)

where C is the number of categorical features, L_t^{mse} is inherently equation 12, and L_t^i minimises the KL divergence for each multinomial diffusion [41]. It outperforms other strong baselines in tabular data synthesis and even verified its efficacy on privacy criteria against SMOTE [71].

Next, to address the long-standing problem of class imbalance in tabular data, Kim *et al.* [46] propose the first work on Score-based Over Sampling (SOS). It utilises style transfer to transform samples from the majority classes to the minority classes. In detail, the samples from the majority classes are corrupted by a forward SDE and recovered using the reverse SDE that originally learns to sample minority classes. Class-conditional fine-tuning is optionally applied to further improve the over-sampling performance.

STaSy [47] is another seminal work on tabular data synthesis. Kim *et al.* [47] highlight that directly applying SDE [28] to the tabular data makes it challenging to learn the joint probability of multiple columns, particularly when there is little to no correlation amongst them. To mitigate the issue, they design architectures comprised of MLP residual blocks, which are dataset-dependent. They further integrate self-paced learning with denoising score matching objective to improve the performance. Fake tabular samples are generated by solving reverse SDEs using probability flow ODE method. It is also shown that sampling quality improves after fine-tuning in general.

CoDi [48] addresses the training challenges that arise from mixed-data types by adopting a dual diffusion model approach; one model is for modelling continuous features, while the other is for discrete (categorical) features. These models are comprised of UNet-based architecture where convolutional layers are replaced with linear layers [72] and trained in a co-evolutionary fashion, being conditioned reciprocally. In other words, they receive each other's output as additional input. To elaborate, there is a pair of data, $(\mathbf{x}_0^C, \mathbf{x}_0^D)$, and the perturbed data \mathbf{x}_t^C after t steps is a condition to \mathbf{x}_t^D in the discrete diffusion model, and vice versa. Then, the diffusion objective function for model handling continuous features is updated as follows:

$$L_{\text{Diff}_{C}}(\theta_{C}) = \mathbb{E}_{t, \mathbf{x}_{0}^{C}, \epsilon} \left[\left\| \epsilon - \epsilon_{\theta_{C}}(\mathbf{x}_{t}^{C}, t \mid \mathbf{x}_{t}^{D}) \right\|^{2} \right].$$
 (22)

For brevity, we only provide equations regarding the continuous features here. To further improve, triplet loss [73] is applied to each diffusion model independently. By learning a combination of two losses, it surpasses the other strong baselines in the real-world benchmark datasets.

3.1.2 Imputation

TabCSDI [49] is a novel approach on tabular data imputation built upon CSDI [39], which is for time-series data imputation and forecasting. To generate both continuous and categorical features of tabular data, it explores three conventional techniques: one-hot encoding, analog bits encoding [74], and feature tokenisation [75]. Especially, when utilising feature tokenisation, both continuous and categorical features undergo transformation via an embedding layer. Zheng *et al.* [49] emphasise the importance of feature tokenization as it mitigates the issue of column imbalance, which is commonly observed in the other two techniques. This approach leads to superior performance.

3.2 Time Series Data

Time series data consist of a sequence of observations recorded over regular intervals of time. It plays a crucial role in myriad fields, including finance, healthcare and climate. Its significance is particularly highlighted in decision-making processes, making time series forecasting methodologies evolve significantly from traditional statistical methods [76], to Recurrent Neural Networks (RNNs) [77], and Transformer-based models [78]. However, the performance of predictive tasks using these models can degrade due to the presence of missing data, often attributable to device failures or human error [79]. As such, time series imputation strategies have also been explored [79]. Concurrently, to facilitate scenario-based simulation and address privacy concerns, researchers have also focused on time series synthesis [23, 24].

Aiming to comprehensively address the three core challenges—generation, imputation, and forecasting—researchers have delved into the time series modelling with generative diffusion models in recent years¹.

3.2.1 Generation

TSGM [50] is the first work on time-series synthesis. It generates fake time series data at each time step, based on the past synthesised observations. The framework follows a two-stage process: pre-training and main training, which are performed by encoder, decoder, and conditional score-matching network. During the pre-training stage, both the encoder and decoder are trained to reconstruct the input time series data with MSE loss. Next, in the main training stage, a conditional score matching network, based on UNet [72] with linear layers, is trained with the pre-trained encoder and decoder to synthesise the time series data. PC sampler [28] is harnessed to solve the time-reverse SDE and to yield the synthesised time series data. Its remarkable performance is demonstrated by surpassing prior baselines built upon the frameworks of VAE and GAN on five real-world datasets.

3.2.2 Imputation

Although CSDI [39] is mainly introduced for time-series data imputation, it is also applicable to forecasting. Tashiro *et al.* propose novel training and sampling strategies, which are inspired by masked language modelling [80]. During the training process, a subset of observed values is employed as conditional data, with the rest of the observed data for imputation targets. On the other hand, during sampling, all observed data points are utilised as conditional information, whereas all missing values are considered as targets for imputation. Let conditional observations \mathbf{x}_0^{co} and imputation targets \mathbf{x}_0^{ta} , then the noisy target to be sampled can be written as $\mathbf{x}_t^{\text{ta}} = \sqrt{\alpha_t}\mathbf{x}_0^{\text{ta}} + (1 - \alpha_t)\epsilon$. Accordingly, the imputation network ϵ_θ is trained by minimising the following objective function:

$$\mathbb{E}_{\mathbf{x}_0 \sim p(\mathbf{x}_0), \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), t} \left[\left| \left| \left(\epsilon - \epsilon_{\theta}(\mathbf{x}_t^{\text{ta}}, t \mid \mathbf{x}_0^{\text{co}}) \right) \right| \right|_2^2 \right]. \tag{23}$$

To effectively harness both temporal and feature-based dependencies of time series, they design a temporal and feature Transformer layer [80], instead of using a convolution layer. It outperforms other strong VAE-based baselines on both healthcare and environmental data.

SSSD^{S4} [51] is a pioneering approach that integrates structured state-space models (SSM) [81] into the framework of diffusion models. Specifically, SSM is proven to be effective in handling long-term dependencies on time-series data. It is formulated as:

$$x'(t) = Ax(t) + Bu(t) \text{ and } y(t) = Cx(t) + Du(t)$$
(24)

where u(t) and y(t) are 1D input and output sequence, respectively, and x(t) is an N-dimensional hidden state. When learning with diffusion model framework, it denoises the segments, either in part or as a whole, making it suitable for both imputation and forecasting applications. First, for the imputation task, a given time series \mathbf{x}^0 and a binary mask \mathbf{M} indicate observed elements, then the concatenated matrix is denoted as $\mathbf{x}_c^0 = concat(\mathbf{x}^0 \odot \mathbf{M}, \mathbf{M})$. They propose the modification of the objective function that incorporates a binary mask, built upon the DDPM framework [27]:

$$\mathbb{E}_{k,\mathbf{x}_{0},\epsilon} \left[\lambda(k) \left\| \epsilon \odot \mathbf{M} - \epsilon_{\theta} \left(\sqrt{\bar{\alpha_{k}}} \mathbf{x}_{0} + \sqrt{1 - \bar{\alpha_{k}}} \epsilon, \mathbf{x}_{c}^{0}, k \right) \odot \mathbf{M} \right\|^{2} \right]$$
(25)

where ϵ_{θ} is the SSM and k is the diffusion step. In forecasting, \mathbf{x}^{0} is replaced with $(1 - \mathbf{M}) \odot \mathbf{x}^{0}$. SSSD^{S4} employs SSSD and modifies the previous works [39, 82] to form a more direct representation of a diffusion process along the time axis. It has shown its competitiveness compared to most of the prior works on handling long sequence time series.

3.2.3 Forecasting

TimeGrad, presented by Rasul $et\ al.\ [53]$, is an autoregressive multivariate probabilistic time series forecasting model on the basis of DDPM framework [27]. The time interval for prediction is perturbed using Gaussian noise, which is then gradually removed, conditioned on historical time series data. The historical data is encoded with a conditional distribution approximation using a hidden state, generated by an RNN-based module. At time point t, given the multivariate time series vector \mathbf{x}_t^0 , the recurrent model RNN encodes the time series using covariates \mathbf{c}_t and the hidden state \mathbf{h}_{t-1} from the previous step t-1 by:

¹To stay within our scope, we exclude time series modelling approaches based on spatio-temporal graph data, despite their relevance in certain applications. Our focus remains on the tabular and time series data modalities.

$$\mathbf{h}_t = \text{RNN}_{\theta}(\mathbf{x}_t^0, \mathbf{c}_t, \mathbf{h}_{t-1}). \tag{26}$$

Then, the conditional distribution is approximated as:

$$\prod_{t=t_0}^T p_{\theta}(\mathbf{x}_t^0 | \mathbf{h}_{t-1}). \tag{27}$$

They follow a similar derivation introduced in Section 2.1, yielding an appropriate objective function using equation 27. After training, the iterative sampling process is conducted until the desired forecast length is attained.

ScoreGrad [54] is built upon SDE framework [28] for time series foreceasting. It comprises a feature extraction module and conditional score-matching module. The feature extraction module, as well as target distribution, is identical to the previous work of TimeGrad [53] in that it encodes historical data and generates hidden states. Next, the conditional score-matching module has multiple residual blocks, each comprising a bidirectional dilated convolution layer, a gated activation unit, and a 1D convolutional neural network for generating output. PC sampler [28] is employed to forecast the future time datapoints by solving the time-reverse SDE.

Li *et al.* [55] propose D³VAE, where a series of diffusion, denoising, and disentanglement is applied to bidirectional VAE (BVAE) [83] with the aim of improving both performance and interpretability in time series forecasting. First, to improve performance, it harnesses a coupled forward diffusion process for data augmentation on both input and target data. It reduces both epistemic and aleatoric uncertainties, where the former is induced by the model and the latter by the data. Meanwhile, the backward process, which encompasses prediction and further refinement of the disturbed prediction, is facilitated by leveraging the BVAE and multi-scale denoising score matching. It further effectively cleans the generated time series using a single-step gradient denoising jump [84].

Next, interpretability involves discerning the independent factors within the data [85]. It can be accomplished by disentangling the latent variables, which signify trends or seasonality. In this regard, D³VAE [55] minimises the Total Correlation (TC) [85] of the BVAE to disentangle latent variables. Therefore, its optimisation process is guided by a combination of four loss terms: 1) KL divergence between the estimated target distribution and target distribution, 2) denoising score matching, 3) minimisation of TC, and 4) MSE loss between prediction and ground truth. It surpasses previous VAE-based works by a significant margin.

Bilovš *et al.* [52] approach time series diffusion modelling differently from the prior works, assuming that the time series data can be represented as a series of values derived from the underlying continuous function $\mathbf{x}(\cdot)$. They propose a novel score-based generative diffusion framework where noise injection and removal processes are performed on the entire continuous function, instead of being applied to individual data points. In other words, the continuous function transitions to the prior stochastic process during the forward process, whilst the reverse process yields the new function samples. This means that this continuous function should be continuous and computationally tractable so as to facilitate both training and sampling. Bilovš *et al.* [52] fulfill these requirements by designing a Gaussian stochastic process with a covariate Σ as $\epsilon(\cdot) \sim \mathcal{GP}(\mathbf{0}, \Sigma)$, instead of the conventional noise vector $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

They first introduce Discrete Stochastic Process Diffusion (DSPD), which is built upon the DDPM formulation [27]. The transition kernels in the forward process and backward process are modified from the original formulations of equations 3 and 6 to equations 28 and 29, respectively, as following:

$$q(\mathbf{X}^k|\mathbf{X}^0) = \mathcal{N}(\sqrt{\bar{\alpha}_k}\mathbf{X}^0, (1-\bar{\alpha}_k)\mathbf{\Sigma}), \tag{28}$$

$$p_{\theta}(\mathbf{X}^{k-1}|\mathbf{X}^k) = \mathcal{N}(\mu_{\theta}(\mathbf{X}^k, k), (1 - \alpha_k)\Sigma)$$
(29)

where $\mathbf{X}^k = (\mathbf{x}^k(t_0), ..., \mathbf{x}^k(t_{M-1}))$ represents the time-indexed sequence of points observed at M distinct timestamps, t_i , within the interval $t \in [0, T]$, at k-th diffusion step. As aforementioned, these points are assumed to be derived from the continuous function $\mathbf{x}(\cdot)$. Next, the objective function is modified from equation 12 to 30,

$$\mathbb{E}_{k,\mathbf{X}_{F}^{0},\epsilon}\left[\lambda(k)\left\|\epsilon - \epsilon_{\theta}\left(\sqrt{\bar{\alpha}_{k}}\mathbf{X}_{F}^{0} + \sqrt{1 - \bar{\alpha}_{k}}\epsilon_{k},\mathbf{X}_{H}^{0},k\right)\right\|^{2}\right]$$
(30)

where X_F^0 and X_H^0 denote the input future and historical data. The DSPD can be applied to both forecasting and imputation tasks. First, the forecasting methodology of DSPD is similar to TimeGrad [53], with respect to objective function, architecture, and the sampling process. However, it advances in two points: 1) it is capable of predicting any

future point within a continuous time interval and 2) it facilitates simultaneous prediction of multiple time points in a single run, not in an autoregressive fashion. Second, DSPD assumes that observed time series are formed by the values of the continuous function $\mathbf{x}(\cdot)$ at specific time points, which allows missing values to be computed by extracting the value of the continuous function at the corresponding time point. Thus, it can be repurposed for imputation by replacing the variance term with a covariance matrix, which is indicated in the equation on neural network of CSDI [39].

Bilovš *et al.* [52] extend DSPD to a continuous variant termed as Continuous Stochastic Process Diffusion (CSPD), developed from the SDE formulation [28]. This continuous variant can also be employed for both imputation and forecasting tasks, particularly useful in imputation tasks. This is because the continuous noise process is more natural when handling irregular time intervals, which is a prevalent issue in imputation tasks.

4 Domain-specific Applications

4.1 Electronic Health Records (EHR)

EHR encompasses a vast amount of patient-centered information, including patient's medical history, diagnoses, medications and such. It has significantly benefited from the advancements in deep learning applications, such as predictive diagnosis [86], medication recommendation [87], and continuous monitoring in intensive care units (ICUs), which significantly diminished the likelihood of complications and mortality rates [88]. However, due to the privacy and ethical concerns surrounding the EHR, release of public data has been limited, which has constrained further research and development [89]. To mitigate these concerns, high-quality and realistic EHR synthesis using GAN [90] has been investigated amongst researchers. With more powerful generative diffusion models, researchers have explored both synthesis and forecasting of the EHR.

4.1.1 Generation

MedDiff [58] is the first literature that applies diffusion models to EHR. He *et al.* [58] propose novel three points: first, it accelerates the generation process via Anderson acceleration [91]. Next, to reflect label information to the generated samples, it utilises classifier-guided sampling process [92]. Lastly, it uses 1D convolutional layer to U-Net architecture to enhance the learning of feature correlations amongst neighbor features. Nonetheless, it is still limited in that it only generates continuous variables, as the authors confine the data to follow Gaussian distribution.

To mitigate the aforementioned issue, Ceritli *et al.* [57] adopt TabDDPM framework [42] to generate 1) continuous data using the Gaussian multinomial diffusion processes and 2) categorical data using the multinomial diffusion processes. The results demonstrate that their work [57] surpasses previous works in data usability criteria but with the exception of privacy evaluation, raising the necessity to improve in such direction.

EHRDiff [56] is built upon the work of SDE [28], especially solving the reverse process via Heun's second order method ODEs to produce more precise and realistic EHR samples in a deterministic manner. Also, it utilises an adaptive parameterisation [93] to address the issue of amplified prediction error due to the variance of scale noise σ . For the architecture, it uses multilayer perceptron (MLP) layers. Similar to TabDDPM [57], it shows outstanding performance on data usability, but moderate performance on privacy criteria.

Kuo *et al.* [59] propose a EHR-DPM. It introduces two auxiliary loss functions: 1) one-step reconstruction loss to alleviate training instability and 2) MSE loss between the clean data and its restored counterpart in a randomly projected latent space. Unlike MedDiff [58] that applies a 1D convolutional layer to individually denoise each feature, EHR-DPM adds linear layers. It involves two steps: 1) a linear layer is added to perform denoising at the latent level, as opposed to the variable level, 2) an additional linear layer is incorporated into each up-sampling and down-sampling 1D CNN, and the final up-sampling output.

4.1.2 Forecasting

TDSTF [60] is an initial work on EHR forecasting via diffusion models. The training and sampling process are the same as other traditional diffusion models for time series forecasting. However, to overcome data sparsity, Chang *et al.* [60] convert the data into triplet form instead of utilising conventional techniques such as aggregation and imputation. Also, its architecture consists of multiple residual layers including Transformer encoder and decoder [80]. It surpasses previous existing models in terms of both predictive performance on critical vital sign and inference speed by a large margin on MIMIC-III dataset [94].

4.2 Bioelectrical Signal Processing

Bioelectrical signals are measured through electrical potential differences across a cell or an organ [95]. They include Electrocardiogram (ECG), Electroencephalogram (EEG), Electromyogram (EMG) and Electrocaulogram (EOG). These signals, expressing electrical activity in the heart, brain, muscles, and eyes, respectively, serve as non-invasive but informative diagnostic tools [96]. Recent advancements in deep learning methodologies have led to notable predictive capability improvements on bioelectrical signals. However, two significant hurdles persist [97]: restricted availability of public bioelectrical signal data due to privacy concerns and inevitable data corruption owing to noise from sources like patient respiration and body movements. These challenges impede further improvements of deep predictive models, thus researchers have explored deep generative models to both synthesise [98] and reduce wander or noise [99] from bioelectrical signal data.

4.2.1 Generation

SSSD-ECG [61] is built upon the work of SSSD^{S4} [51], which integrates diffusion models and structured state space models. It synthesises 12-lead ECG data in a multi-label fashion by conditioning on over 70 ECG statements. It outperforms previous GAN-based works in ECG synthesis in terms of both quantitative evaluation and qualitative analysis from domain experts.

4.2.2 Enhancement

Li *et al.* [62] propose a novel ECG enhancement model, DeScoD-ECG, that is designed for the removal of noise and wander. It operates conditionally on noisy observations and employs an iterative process to restore signals from Gaussian distributions. With a clean input signal $\tilde{\mathbf{x}}$ and a noisy ECG signal \mathbf{x}_0 , the equation 5 and 6 from DDPM [27] is modified as:

$$p_{\theta}\left(\mathbf{x}_{0:T} \mid \tilde{\mathbf{x}}\right) = p\left(\mathbf{x}_{T}\right) \prod_{t=1}^{T} p_{\theta}\left(\mathbf{x}_{t-1} \mid \mathbf{x}_{t}, \tilde{\mathbf{x}}\right) \text{ where } \mathbf{x}_{T} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),$$
(31)

$$p_{\theta}\left(\mathbf{x}_{t-1} \mid \mathbf{x}_{t}, \tilde{\mathbf{x}}\right) = \mathcal{N}\left(\mathbf{x}_{t-1}; \boldsymbol{\mu}_{\theta}\left(\mathbf{x}_{t}, t \mid \tilde{\mathbf{x}}\right), \sigma_{\theta}\left(\mathbf{x}_{t}, t \mid \tilde{\mathbf{x}}\right) \mathbf{I}\right). \tag{32}$$

Also, it utilises a self-ensemble strategy, wherein it averages multiple output results to enhance the performance of signal reconstruction. The architecture has two backbones that extract features from both the noisy observations and the latent variables, then they are combined through bridge modules. Inspired by the DeepFilter [99], two backbones conduct multi-scale feature aggregation and channel-wise concatenation processes. Also, the bridge modules are conditioned on noise level and perform encoding using sinusoidal positional embeddings [80], followed by a convolutional layer with a 1 x 1 kernel size. The absence of any up/downsampling procedures within the architecture ensures its applicability to ECG signals of any length. DeScoD-ECG [62] is validated on real-world datasets, including the QT database [100] and the MIT-BIH Noise Stress Test Database [101]. The results demonstrate the efficacy of stable DeScoD-ECG compared to other baselines, particularly in situations with extreme corruption.

Duan *et al.* [63] explore the reconstruction of EEG signal under subject-specific variability. They propose DS-DDPM based on the assumption that noisy EEG signals (input) can be decomposed into domain specific noise and clean signals. Thus, they propose a novel diffusion and the reversal process, in which the domain specific noise and clean signal are divided and aggregated at every time step. As both noise and signal exist in orthogonal spaces, DS-DDPM learns using an Additive Angular Margin classification (Arc-Margin) loss [102] to improve intra-class cohesion and inter-class discrepancy. Additionally, they implement an input overlap segmentation strategy to minimise temporal differences in overlapping segments. It also adopts a classifier guidance strategy [92] that exploits human subject index to improve generalisation capability. It learns with modified UNet architecture [72] with multi-head attention [80]. The effectiveness of DS-DDPM is validated on the BCI Competition IV motor imagery decoding dataset [103]. It is also tested on downstream classification tasks to corroborate its reconstruction capability.

4.3 Recommendation Systems (RecSys)

RecSys aim at modelling personalised users' preferences based on their previous user-item interactions such as clicks, ratings, or purchases [104]. Due to their profound significant value in industry, research on RecSys has progressed from traditional methods such as collaborative filtering (CF) based techniques [105] to deep learning methodologies [106]. However, they often show limited generalisation performance, on account of weak collaborative signals, inadequate latent representations, or noisy data scenarios [66]. Researchers have explored the generative models of VAE [107] and

GAN [108] to mitigate these challenges, but these models also have their own limitations of restricted capability in capturing personalised user preferences (from VAE-based models) and training instability (from GAN-based models) [67]. To address the issues, researchers have delved into diffusion models, owing to their strong representation capability and training stability.

Walker *et al.* [64] propose CODIGEM, which is the first work exploiting diffusion models on RecSys. It generates strong collaborative signals and robust latent representations on user-item interactions, thus outperforming previous VAE-based works. However, the model falls short in handling sequential scenarios.

DiffuRec [65] aims to address the limitations from modelling the item representations as fixed vectors. To handle latent representation of items and multi-level interests of users, they utilise diffusion models to represent them as distributions. During the diffusion process, a truncated linear schedule is used for noise addition. The introduction of noise functions as an uncertainty factor that steers the learning process towards enhanced robustness. Also, instead of using conventional MSE loss in DDPM [27], it learns with cross-entropy loss. This modification is driven by two reasons: the static nature of item embedding and the use of inner product for relevance calculation in the reverse process. They also exploit Transformer [80] as an approximator to reconstruct target item representation. These modifications enable their model to show remarkable performance on several benchmark datasets.

Du *et al.* [66] develop another novel sequential diffusion model for RecSys. It introduces an additional transition to convert the items with discrete features. It perturbs the original item, instead of the whole item sequences, by injecting noise and it restores the target by MSE loss term. Also, it samples only important diffusion steps instead of the entire steps. Those strategies enable an efficient training and sampling. On the architectural side, it leverages a transformer-based encoder [80] with learnable positional embedding. It significantly surpasses preceding generative and contrastive learning based benchmarks on real world datasets.

Wang *et al.* [67] design two novel diffusion recommendation models for sequential RecSys of L-DiffRec and T-DiffRec, highlighting two major challenges in RecSys: 1) high computational costs for large-scale item prediction and 2) temporal transition for user preference. L-DiffRec addresses the first challenge. It groups items into clusters, then compresses the user-item interaction using multiple VAEs and conducts diffusion processes in the latent space to generate top-K recommendations. It addresses the high computational demands arising from large-scale item prediction.

Second, T-DiffRec mitigates the second issue, temporal shifts in user preferences. It introduces time-aware reweighting strategy to model user interactions based on the assumption that the recent interactions of a user may capture user preference more effectively. This enhances the model's adaptability to dynamic user behaviour. The two variants of DiffRec outperform previous VAE-based baselines by a substantial margin on several public datasets.

CDDRec [68] achieves diffusion models for sequential RecSys in conditional autoregressive and ranking aware manners. Its network comprises of three models: a step-wise diffuser, a sequence encoder and a cross-attentive conditional denoising decoder. First, the step-wise diffuser corrupts the original data using increasing Gaussian noise. The sequence encoder utilises a self-attention mechanism on historical interactions whilst the cross-attentive conditional denoising decoder adopts a direct-condition mechanism wherein the sequence embeddings from the encoder are directly conditioned to the decoder at every denoising step. It enables a conditional sample generation in an autoregressive fashion.

Further, to mitigate the inherent long-tailed and sparse item distribution, CDDRec [68] is optimised through a combination of a cross-divergence loss and multi-view contrastive loss with denoising diffusion loss. The cross-divergence loss term utilises KL divergence to minimise the discrepancy between the forecasted mean and the target item embedding, as compared to the discrepancy between the predicted mean and unrelated target embeddings. It helps avoid a situation where the rank scores uniformly converge across all items for all users, enabling the model to function in a ranking-aware fashion. The multi-view contrastive loss with denoising diffusion loss applies contrastive loss to enhance the robustness against the noisy interference by maximising the agreement between the original view and its counterpart, achieved through random cropping, shuffling and masking randomly. The combination of two loss terms, therefore, guides the CDDRec to be more discriminative and robust.

5 Current Challenges and Future Research Directions

Despite the astonishing advancements in generative diffusion models for the structured data, there are still challenges that this field continues to grapple with. The objective of this section is to describe these challenges and provide potential future research directions.

Customised Designs for Structured Data Generative diffusion models are largely influenced by factors such as architectural design, training strategy, and noise scheduling. However, most of the approaches on structured data

have directly adopted or slightly modified from existing seminal works from other data modalities. Customising these elements may potentially lead to significant improvements in modelling capability and overcoming generative modelling trilemma: sampling quality, diversity, and speed [109].

Causal Learning and Counterfactual Reasoning Causal learning aims to discern and identify the casual relationships or inter-dependencies amongst variables within a given dataset [110]. For instance, it allows us to model potential outcomes or the most indicative factor in financial forecasting or disease progression. On the other hand, counterfactual reasoning aims to predict an individual's outcome under different circumstances [5]. Specifically, it predicts what the result would be like if certain variables were changed from their observed real-world values. Hence, the integration of causal learning or counterfactual reasoning methodologies into the diffusion models can potentially enhance their performance. By harnessing cause-and-effect relationships or counterfactual estimation rather than simply exploiting correlations or conditioning on variables, we can optimise diffusion models to be more reliable and robust.

Bias in Dataset Another noteworthy challenge is the inherent bias present within the publicly available datasets. Specifically, the demographic features in the EHRs and the bioelectrical signals extracted from the subjects are often biased towards specific classes [63, 111]. This inevitable skewness in data consequently impacts the generalisability of the generative models and limits their applicability to related fields. To circumvent these issues, it is vital to utilise more diverse and balanced datasets. Otherwise, novel methods should be developed to neutralise this inherent bias.

Extensions to Multi-modality Learning The fusion of structured data with other data modalities not only enhances the model performance but also expands the possible tasks, *e.g.*, improving financial stock price prediction by jointly learning both text and time series data [112] and text synthesis from table [113]. In this regard, future research direction may focus on the development of novel methodologies adept at integrating multiple data modalities in an efficient and effective manner as well as exploring the potential untapped tasks.

Miscellany In the realm of tabular data modelling, it is possible to devise an enhanced strategy that effectively models both continuous and categorical data types. Furthermore, a universal framework can be developed to encompass generation, imputation, and forecasting for both tabular and time-series modelling. For applications involving EHR and biological signal processing, current works may benefit from an incorporation of domain-specific knowledge or issues, *e.g.*, medical ontology and irregular visit interval [114]. Furthermore, the long inference time compared to GAN-based methods poses a challenge for real-time or on-chip deployment in medical equipment, highlighting the need for improvements to facilitate practical use.

6 Conclusion

The generative diffusion models have verified themselves by exhibiting remarkable performance across diverse applications, and also have shown excellent performance on structured data. However, current research on generative diffusion models specifically tailored for structured data has not received active attention, compared to the other data modalities. To facilitate exploration and advancement in this field, we provide a comprehensive survey on generative diffusion models for structured data. Our survey encompasses a brief introduction to the underlying theory of score-based diffusion models, followed by a concise review of existing literature categorized into data-driven general tasks and domain-specific applications. Additionally, we discuss current challenges and outline potential research directions for the future. We hope that this survey will serve as a valuable guide for those interested in this field, thereby fostering further research and advancement in the area.

References

- [1] Yu Lin, Yan Yan, Jiali Xu, Ying Liao, and Feng Ma. Forecasting stock index price using the ceemdan-lstm model. *The North American Journal of Economics and Finance*, 57:101421, 2021.
- [2] Pooja Tiwari, Simran Mehta, Nishtha Sakhuja, Jitendra Kumar, and Ashutosh Kumar Singh. Credit card fraud detection using machine learning: a study. *arXiv preprint arXiv:2108.10005*, 2021.
- [3] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 5941–5948, 2019.
- [4] Juan-Jose Beunza, Enrique Puertas, Ester García-Ovejero, Gema Villalba, Emilia Condes, Gergana Koleva, Cristian Hurtado, and Manuel F Landecho. Comparison of machine learning algorithms for clinical event prediction (risk of coronary heart disease). *Journal of biomedical informatics*, 97:103257, 2019.
- [5] Licheng Liu, Ye Wang, and Yiqing Xu. A practical guide to counterfactual estimators for causal inference with time-series cross-sectional data. *American Journal of Political Science*, 2022.

- [6] Seunghoi Kim and Daniel C. Alexander. Agen: Adversarial graph convolutional network for 3d point cloud segmentation. In *British Machine Vision Conference*, 2021.
- [7] Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. Solving inverse problems in medical imaging with score-based generative models. In *NeurIPS 2021 Workshop on Deep Learning and Inverse Problems*, 2021.
- [8] Atijit Anuchitanukul and Julia Ive. Surf: Semantic-level unsupervised reward function for machine translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4508–4522, 2022.
- [9] To Eun Kim and Aldo Lipani. A multi-task based neural model to simulate users in goal oriented dialogue systems. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2115–2119, 2022.
- [10] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2426–2435, June 2022.
- [11] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10696–10706, 2022.
- [12] Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in Neural Information Processing Systems*, 35:4328–4343, 2022.
- [13] Jerome Ramos, To Eun Kim, Zhengxiang Shi, Xiao Fu, Fanghua Ye, Yue Feng, and Aldo Lipani. Condita: A state machine like architecture for multi-modal task bots. In *Alexa Prize TaskBot Challenge Proceedings*, 2022.
- [14] Peiyu Yu, Sirui Xie, Xiaojian Ma, Baoxiong Jia, Bo Pang, Ruigi Gao, Yixin Zhu, Song-Chun Zhu, and Ying Nian Wu. Latent diffusion energy-based model for interpretable text modeling. *arXiv preprint arXiv:2206.05895*, 2022.
- [15] Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *The Eleventh International Conference on Learning Representations*, 2023.
- [16] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. nature, 521(7553):436–444, 2015.
- [17] Shereen Elsayed, Daniela Thyssens, Ahmed Rashed, Hadi Samer Jomaa, and Lars Schmidt-Thieme. Do we really need deep learning models for time series forecasting? *arXiv preprint arXiv:2101.02118*, 2021.
- [18] Ravid Shwartz-Ziv and Amitai Armon. Tabular data: Deep learning is not all you need. *Information Fusion*, 81:84–90, 2022.
- [19] Arlind Kadra, Marius Lindauer, Frank Hutter, and Josif Grabocka. Well-tuned simple nets excel on tabular datasets. *Advances in neural information processing systems*, 34:23928–23941, 2021.
- [20] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114, 2013.
- [21] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.
- [22] L Vivek Harsha Vardhan and Stanley Kok. Generating privacy-preserving synthetic tabular data using oblivious variational autoencoders. In *Proceedings of the Workshop on Economics of Privacy and Data Labor at the 37 th International Conference on Machine Learning*, 2020.
- [23] Cristóbal Esteban, Stephanie L Hyland, and Gunnar Rätsch. Real-valued (medical) time series generation with recurrent conditional gans. *arXiv preprint arXiv:1706.02633*, 2017.
- [24] Magnus Wiese, Robert Knobloch, Ralf Korn, and Peter Kretschmer. Quant gans: deep generation of financial time series. *Quantitative Finance*, 20(9):1419–1440, 2020.
- [25] Jinsung Yoon, James Jordon, and Mihaela Schaar. Gain: Missing data imputation using generative adversarial nets. In *International conference on machine learning*, pages 5689–5698. PMLR, 2018.
- [26] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.
- [27] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.
- [28] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

- [29] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.
- [30] Lei Huang, Hengtong Zhang, Tingyang Xu, and Ka-Chun Wong. Mdm: Molecular diffusion model for 3d molecule generation. *arXiv preprint arXiv:2209.05710*, 2022.
- [31] Minkai Xu, Alexander Powers, Ron Dror, Stefano Ermon, and Jure Leskovec. Geometric latent diffusion models for 3d molecule generation. *arXiv preprint arXiv:2305.01140*, 2023.
- [32] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Yingxia Shao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *arXiv* preprint arXiv:2209.00796, 2022.
- [33] Hanqun Cao, Cheng Tan, Zhangyang Gao, Guangyong Chen, Pheng-Ann Heng, and Stan Z Li. A survey on generative diffusion model. *arXiv* preprint arXiv:2209.02646, 2022.
- [34] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.
- [35] Yuansong Zhu and Yu Zhao. Diffusion models in nlp: A survey. arXiv preprint arXiv:2303.07576, 2023.
- [36] Mengchun Zhang, Maryam Qamar, Taegoo Kang, Yuna Jung, Chenshuang Zhang, Sung-Ho Bae, and Chaoning Zhang. A survey on graph diffusion models: Generative ai in science for molecule, protein and material. *arXiv* preprint arXiv:2304.01565, 2023.
- [37] Amirhossein Kazerouni, Ehsan Khodapanah Aghdam, Moein Heidari, Reza Azad, Mohsen Fayyaz, Ilker Hacihaliloglu, and Dorit Merhof. Diffusion models for medical image analysis: A comprehensive survey, 2023.
- [38] Lequan Lin, Zhengkun Li, Ruikun Li, Xuliang Li, and Junbin Gao. Diffusion models for time series applications: A survey. *arXiv preprint arXiv:2305.00624*, 2023.
- [39] Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. Advances in Neural Information Processing Systems, 34:24804–24816, 2021.
- [40] Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. Digress: Discrete denoising diffusion for graph generation. *arXiv preprint arXiv:2209.14734*, 2022.
- [41] Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. *Advances in Neural Information Processing Systems*, 34:12454–12465, 2021.
- [42] Akim Kotelnikov, Dmitry Baranchuk, Ivan Rubachev, and Artem Babenko. Tabddpm: Modelling tabular data with diffusion models. *arXiv preprint arXiv:2209.15421*, 2022.
- [43] Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005.
- [44] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.
- [45] Yang Song, Sahaj Garg, Jiaxin Shi, and Stefano Ermon. Sliced score matching: A scalable approach to density and score estimation. In *Uncertainty in Artificial Intelligence*, pages 574–584. PMLR, 2020.
- [46] Jayoung Kim, Chaejeong Lee, Yehjin Shin, Sewon Park, Minjung Kim, Noseong Park, and Jihoon Cho. Sos: Score-based oversampling for tabular data. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 762–772, 2022.
- [47] Jayoung Kim, Chaejeong Lee, and Noseong Park. Stasy: Score-based tabular data synthesis. *arXiv preprint arXiv:2210.04018*, 2022.
- [48] Chaejeong Lee, Jayoung Kim, and Noseong Park. Codi: Co-evolving contrastive diffusion models for mixed-type tabular synthesis. *arXiv* preprint arXiv:2304.12654, 2023.
- [49] Shuhan Zheng and Nontawat Charoenphakdee. Diffusion models for missing value imputation in tabular data. *arXiv preprint arXiv:2210.17128*, 2022.
- [50] Haksoo Lim, Minjung Kim, Sewon Park, and Noseong Park. Regular time-series generation using sgm. *arXiv* preprint arXiv:2301.08518, 2023.
- [51] Juan Miguel Lopez Alcaraz and Nils Strodthoff. Diffusion-based time series imputation and forecasting with structured state space models. *arXiv* preprint arXiv:2208.09399, 2022.

- [52] Marin Biloš, Kashif Rasul, Anderson Schneider, Yuriy Nevmyvaka, and Stephan Günnemann. Modeling temporal data as continuous functions with process diffusion. *arXiv preprint arXiv:2211.02590*, 2022.
- [53] Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *International Conference on Machine Learning*, pages 8857–8868. PMLR, 2021.
- [54] Tijin Yan, Hongwei Zhang, Tong Zhou, Yufeng Zhan, and Yuanqing Xia. Scoregrad: Multivariate probabilistic time series forecasting with continuous energy-based generative models. *arXiv preprint arXiv:2106.10121*, 2021.
- [55] Yan Li, Xinjiang Lu, Yaqing Wang, and Dejing Dou. Generative time series forecasting with diffusion, denoise, and disentanglement. *Advances in Neural Information Processing Systems*, 35:23009–23022, 2022.
- [56] Hongyi Yuan, Songchi Zhou, and Sheng Yu. Ehrdiff: Exploring realistic ehr synthesis with diffusion models. *arXiv preprint arXiv:2303.05656*, 2023.
- [57] Taha Ceritli, Ghadeer O Ghosheh, Vinod Kumar Chauhan, Tingting Zhu, Andrew P Creagh, and David A Clifton. Synthesizing mixed-type electronic health records using diffusion models. *arXiv preprint arXiv:2302.14679*, 2023.
- [58] Huan He, Shifan Zhao, Yuanzhe Xi, and Joyce C Ho. Meddiff: Generating electronic health records using accelerated denoising diffusion model. *arXiv preprint arXiv:2302.04355*, 2023.
- [59] Nicholas I Kuo, Louisa Jorm, Sebastiano Barbieri, et al. Synthetic health-related longitudinal data with mixed-type variables generated using diffusion models. *arXiv preprint arXiv:2303.12281*, 2023.
- [60] Ping Chang, Huayu Li, Stuart F Quan, Janet Roveda, and Ao Li. Tdstf: Transformer-based diffusion probabilistic model for sparse time series forecasting. *arXiv* preprint arXiv:2301.06625, 2023.
- [61] Juan Miguel Lopez Alcaraz and Nils Strodthoff. Diffusion-based conditional ecg generation with structured state space models. *arXiv preprint arXiv:2301.08227*, 2023.
- [62] Huayu Li, Gregory Ditzler, Janet Roveda, and Ao Li. Descod-ecg: Deep score-based diffusion model for ecg baseline wander and noise removal. *IEEE Journal of Biomedical and Health Informatics*, 2023.
- [63] Yiqun Duan, Jinzhao Zhou, Zhen Wang, Yu-Cheng Chang, Yu-Kai Wang, and Chin-Teng Lin. Domain-specific denoising diffusion probabilistic models for brain dynamics. *arXiv preprint arXiv:2305.04200*, 2023.
- [64] Joojo Walker, Ting Zhong, Fengli Zhang, Qiang Gao, and Fan Zhou. Recommendation via collaborative diffusion generative model. In *Knowledge Science*, *Engineering and Management: 15th International Conference*, *KSEM 2022*, *Singapore*, *August 6–8*, *2022*, *Proceedings*, *Part III*, pages 593–605. Springer, 2022.
- [65] Zihao Li, Aixin Sun, and Chenliang Li. Diffurec: A diffusion model for sequential recommendation. arXiv preprint arXiv:2304.00686, 2023.
- [66] Hanwen Du, Huanhuan Yuan, Zhen Huang, Pengpeng Zhao, and Xiaofang Zhou. Sequential recommendation with diffusion models. *arXiv preprint arXiv:2304.04541*, 2023.
- [67] Wenjie Wang, Yiyan Xu, Fuli Feng, Xinyu Lin, Xiangnan He, and Tat-Seng Chua. Diffusion recommender model. arXiv preprint arXiv:2304.04971, 2023.
- [68] Yu Wang, Zhiwei Liu, Liangwei Yang, and Philip S Yu. Conditional denoising diffusion for sequential recommendation. *arXiv preprint arXiv:2304.11433*, 2023.
- [69] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Modeling tabular data using conditional gan. *Advances in Neural Information Processing Systems*, 32, 2019.
- [70] Ramiro D Camino, Christian A Hammerschmidt, and Radu State. Improving missing data imputation with deep generative models. *arXiv preprint arXiv:1902.10666*, 2019.
- [71] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.
- [72] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.
- [73] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.
- [74] Ting Chen, Ruixiang Zhang, and Geoffrey Hinton. Analog bits: Generating discrete data using diffusion models with self-conditioning. *arXiv* preprint arXiv:2208.04202, 2022.

- [75] Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. Advances in Neural Information Processing Systems, 34:18932–18943, 2021.
- [76] Siu Lau Ho and Min Xie. The use of arima models for reliability forecasting and analysis. *Computers & industrial engineering*, 35(1-2):213–216, 1998.
- [77] Hansika Hewamalage, Christoph Bergmeir, and Kasun Bandara. Recurrent neural networks for time series forecasting: Current status and future directions. *International Journal of Forecasting*, 37(1):388–427, 2021.
- [78] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 11106–11115, 2021.
- [79] Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. Brits: Bidirectional recurrent imputation for time series. *Advances in neural information processing systems*, 31, 2018.
- [80] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [81] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021.
- [82] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761*, 2020.
- [83] Arash Vahdat and Jan Kautz. Nvae: A deep hierarchical variational autoencoder. *Advances in neural information processing systems*, 33:19667–19679, 2020.
- [84] Saeed Saremi and Aapo Hyvarinen. Neural empirical bayes. arXiv preprint arXiv:1903.02334, 2019.
- [85] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International Conference on Machine Learning*, pages 2649–2658. PMLR, 2018.
- [86] Yikuan Li, Shishir Rao, José Roberto Ayala Solares, Abdelaali Hassaine, Rema Ramakrishnan, Dexter Canoy, Yajie Zhu, Kazem Rahimi, and Gholamreza Salimi-Khorshidi. Behrt: transformer for electronic health records. *Scientific reports*, 10(1):1–12, 2020.
- [87] Suman Bhoi, Mong Li Lee, Wynne Hsu, Hao Sen Andrew Fang, and Ngiap Chuan Tan. Personalizing medication recommendation with a graph-based approach. *ACM Transactions on Information Systems (TOIS)*, 40(3):1–23, 2021.
- [88] Akira-Sebastian Poncette, Claudia Spies, Lina Mosch, Monique Schieler, Steffen Weber-Carstens, Henning Krampe, Felix Balzer, et al. Clinical requirements of future patient monitoring in the intensive care unit: qualitative study. *JMIR medical informatics*, 7(2):e13064, 2019.
- [89] Nithesh Naik, BM Hameed, Dasharathraj K Shetty, Dishant Swain, Milap Shah, Rahul Paul, Kaivalya Aggarwal, Sufyan Ibrahim, Vathsala Patil, Komal Smriti, et al. Legal and ethical consideration in artificial intelligence in healthcare: who takes responsibility? *Frontiers in surgery*, page 266, 2022.
- [90] Mrinal Kanti Baowaly, Chia-Ching Lin, Chao-Lin Liu, and Kuan-Ta Chen. Synthesizing electronic health records using improved generative adversarial networks. *Journal of the American Medical Informatics Association*, 26(3):228–241, 2019.
- [91] Donald G Anderson. Iterative procedures for nonlinear integral equations. *Journal of the ACM (JACM)*, 12(4):547–560, 1965.
- [92] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.
- [93] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022.
- [94] Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.
- [95] Anita Pal, Ajeet Kumar Gautam, and Yogendra Narain Singh. Evaluation of bioelectric signals for human recognition. *Procedia Computer Science*, 48:746–752, 2015.
- [96] Peter J Zimetbaum and Mark E Josephson. Use of the electrocardiogram in acute myocardial infarction. *New England Journal of Medicine*, 348(10):933–940, 2003.
- [97] Shubhojeet Chatterjee, Rini Smita Thakur, Ram Narayan Yadav, Lalita Gupta, and Deepak Kumar Raghuvanshi. Review of noise removal techniques in ecg signals. *IET Signal Processing*, 14(9):569–590, 2020.

- [98] Anne Marie Delaney, Eoin Brophy, and Tomas E Ward. Synthesis of realistic ecg using generative adversarial networks. *arXiv preprint arXiv:1909.09150*, 2019.
- [99] Francisco P Romero, David C Piñol, and Carlos R Vázquez-Seisdedos. Deepfilter: An ecg baseline wander removal filter using deep learning techniques. *Biomedical Signal Processing and Control*, 70:102992, 2021.
- [100] Pablo Laguna, Roger G Mark, A Goldberg, and George B Moody. A database for evaluation of algorithms for measurement of qt and other waveform intervals in the ecg. In *Computers in cardiology 1997*, pages 673–676. IEEE, 1997.
- [101] George B Moody, W Muldrow, and Roger G Mark. A noise stress test for arrhythmia detectors. *Computers in cardiology*, 11(3):381–384, 1984.
- [102] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019.
- [103] Michael Tangermann, Klaus-Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai J Miller, Gernot Mueller-Putz, et al. Review of the bci competition iv. *Frontiers in neuroscience*, page 55, 2012.
- [104] Qian Zhang, Jie Lu, and Yaochu Jin. Artificial intelligence in recommender systems. *Complex & Intelligent Systems*, 7:439–457, 2021.
- [105] Xiangnan He, Hanwang Zhang, Min-Yen Kan, and Tat-Seng Chua. Fast matrix factorization for online recommendation with implicit feedback. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 549–558, 2016.
- [106] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1441–1450, 2019.
- [107] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 world wide web conference*, pages 689–698, 2018.
- [108] Ruiyang Ren, Zhaoyang Liu, Yaliang Li, Wayne Xin Zhao, Hui Wang, Bolin Ding, and Ji-Rong Wen. Sequential recommendation with self-attentive multi-adversarial network. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pages 89–98, 2020.
- [109] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion GANs. In *International Conference on Learning Representations*, 2022.
- [110] Raha Moraffah, Paras Sheth, Mansooreh Karami, Anchit Bhattacharya, Qianru Wang, Anique Tahir, Adrienne Raglin, and Huan Liu. Causal inference for time series analysis: Problems, methods and evaluation. *Knowledge and Information Systems*, 63:3041–3085, 2021.
- [111] Chuizheng Meng, Loc Trinh, Nan Xu, James Enouen, and Yan Liu. Interpretability and fairness evaluation of deep learning models on mimic-iv dataset. *Scientific Reports*, 12(1):7166, 2022.
- [112] Pratyush Muthukumar and Jie Zhong. A stochastic time series model for predicting financial trends using nlp. *arXiv preprint arXiv:2102.01290*, 2021.
- [113] Tianyu Liu, Kexiang Wang, Lei Sha, Baobao Chang, and Zhifang Sui. Table-to-text generation by structure-aware seq2seq learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.
- [114] Ke Niu, You Lu, Xueping Peng, and Jingni Zeng. Fusion of sequential visits and medical ontology for mortality prediction. *Journal of Biomedical Informatics*, 127:104012, 2022.

Appendices

A Multinomial Diffusion for Categorical Data

In this section, we introduce multinomial diffusion [41], which is designed to process categorical data. For K categorical data, each variable is encoded as a one-hot vector, denoted as $\mathbf{x}_t \in \{0,1\}^K$. Using categorical distribution, the multinomial diffusion process is formulated utilising the uniform diffusion noise schedule β_t and categorical distribution C:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{C}(\mathbf{x}_t; (1-\beta_t)\mathbf{x}_{t-1} + \beta_t/K). \tag{33}$$

Then, we can compute the probability of any \mathbf{x}_t given \mathbf{x}_0 with $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{\tau=0}^t \alpha_\tau$:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{C}(\mathbf{x}_t; \bar{\alpha}_t \mathbf{x}_0 + (1 - \bar{\alpha}_t)/K). \tag{34}$$

According to Hoogeboom *et al.* [41], the distribution for the preceding time step t-1 can be expressed from the value \mathbf{x}_t at the next step and the ground truth value \mathbf{x}_0 as following:

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{C}(\mathbf{x}_{t-1}; \tilde{\theta}/A)$$
(35)

where $\tilde{\theta} = [\alpha_t \mathbf{x}_t + (1 - \alpha_t)/K] \odot [\bar{\alpha}_{t-1} \mathbf{x}_0 + (1 - \bar{\alpha}_{t-1})/K]$ and A is a normalising constant that guarantees the cumulative total of all probabilities equals one. It is noteworthy that $q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)$ simplifies to $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$, following the Markov property of the forward process.

Lastly, the reverse process $p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is also learned via a deep neural network, denoted as $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \hat{\mathbf{x}}_0(\mathbf{x}_t, t))$ where $\hat{\mathbf{x}}_0$ is the predicted probabilities by the deep neural network. The model is trained by minimising KL divergence (the L_{t-1} term in Equation 8) between the true distribution and predicted one as:

$$D_{\mathrm{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \mid\mid p_{\theta}(\mathbf{x}_{t-1}|\mathbf{x}_t)) = D_{\mathrm{KL}}(\mathcal{C}(\tilde{\theta}(\mathbf{x}_t, \mathbf{x}_0)) \mid\mid \mathcal{C}(\tilde{\theta}(\mathbf{x}_t, \hat{\mathbf{x}}_0))). \tag{36}$$

B A Detailed Table on Generative Diffusion Models on Structured Data

Table 2: A detailed table on generative diffusion models for structured data, including their frameworks, datasets used for experiments and accessible code links

| Paper | Year | Task | Framework | Dataset | Code |
|----------------------------|------|---|-----------|--|------|
| TabDDPM [42] | 2022 | Tabular Generation | DDPM | Abalone, Adult ROC, Buddy, California Housing, Cardio, Churn Modelling, Diabetes, Facebook Comments Volume, Gesture Phase, Higgs Small, House 16H, Insurance, King, MiniBooNE, Wilt | |
| SOS [46] | 2022 | Tabular Generation | SDE | Buddy, Default, Satimage, Shoppers, Surgical, WeatherAUS | Link |
| STaSy [47] | 2023 | Tabular Generation | SDE | Bean, Beijing, Credit, Crowdsource, Contraceptive, Default, HTRU, Magic, News, Obesity, Phishing, Robot, Shoppers, Shuttle, Spambase | |
| CoDi [48] | 2023 | Tabular Generation | DDPM | Absent, Bank, CMC, Customer, Drug, Faults, Heart, Insurance, Obesity, Seismic, Stroke | |
| TabCSDI [49] | 2022 | Tabular Imputation | DDPM | Breast, Census, Concrete, COVID-19, Diabetes, Libras, Wine | |
| TSGM [50] | 2023 | Time Series Generation | SDE | Air, AI4I, Energy, Occupancy, Stocks | N/A |
| CSDI [39] | 2021 | Time Series Imputation Time Series Forecasting | DDPM | Air Quality, PhysioNet 2012 Challenge | |
| SSSD ^{S4} [51] | 2022 | Time Series Imputation Time Series Forecasting | DDPM | ETTm1, MuJoCo, PTB-XL, Solar | |
| DSPD/CSPD [52] | 2022 | Time Series Imputation Time Series Forecasting | SDE | CIR, Lorenz, OU, Predator-prey, Sine, Sink | |
| TimeGrad [53] | 2021 | Time Series Forecasting | DDPM | Electricity, Exchange, Solar, Taxi, Traffic, Wiki | Link |
| ScoreGrad [54] | 2021 | Time Series Forecasting | SDE | Electricity, Exchange, Solar, Taxi, Traffic, Wiki | Link |
| D ³ VAE [55] | 2022 | Time Series Forecasting | DDPM | Electricity, ETTm1, ETTh1, Traffic, Weather, Wind | |
| EHRDiff [56] | 2023 | EHR Generation | SDE | MIMIC-III | |
| TabDDPM [57] | 2023 | EHR Generation | DDPM | MIMIC-III | |
| MedDiff [58] | 2023 | EHR Generation | SDE | MIMIC-III, Patient Treatment Classification | |
| EHR-DPM [59] | 2023 | EHR Generation | DDPM | MIMIC-III, EuResist | N/A |
| TDSTF [60] | 2023 | EHR Forecasting | DDPM | MIMIC-III | Link |
| SSSD-ECG [61] | 2022 | Biosignal Generation | DDPM | PTB-XL | Link |
| DeScoD-ECG [62] | 2022 | Biosignal Enhancement | DDPM | MIT-BIH Noise Stress Test Database, QT Database | Link |
| DS-DDPM [63] | 2022 | Biosignal Enhancement | DDPM | BCI-Competition-IV dataset | Link |
| CODIGEM [64] | 2022 | RecSys | DDPM | Amazon Electronics, MovieLens-1m, MovieLens-20m | Link |
| DiffuRec [65] | 2023 | RecSys | DDPM | Beauty, MovieLens-1M, Steam, Toys | N/A |
| DiffRec (Du et al.) [66] | 2023 | RecSys | DDPM | Beauty, MovieLens-1M, Toys | N/A |
| DiffRec (Wang et al.) [67] | 2023 | RecSys | DDPM | Amazon-book, MovieLens-1M, Yelp | Link |
| CDDRec [68] | 2023 | RecSys | DDPM | Beauty, Home, Office, Tools | N/A |