CaT-Diff: Cascaded Text-enhanced Diffusion Model for Time-Series Imputation

Changjian Xu[1], Yong Wang[1*], Ruizheng Huang[1], Zhicheng Zhang[1], Wen Yin[1], Kexin Li[1*]

[1]Laboratory of Intelligent Collaborative Computing, University of Electronic Science and Technology of China
{202422900115, huangrz, zhangzc, yinwen1999}@std.uestc.edu.cn, {cla, likx}@uestc.edu.cn

Abstract

Most state-of-the-art time series imputation methods can leverage textual information to improve imputation quality, but they often struggle because they fail to effectively filter noisy information from large language model (LLM) derived textual information. Some existing solutions only filter over the entire token set, which can introduce erroneous conditional constraints, extreme token frequency effects and increased computational complexity. To address this, we propose CaT-Diff, a novel cascaded text-enhanced diffusion model for probabilistic imputation of multivariate time series under Missing Not At Random (MNAR) scenarios. To suppress irrelevant semantics and focus on context most predictive of missing values, CaT-Diff introduces an innovative Hierarchical Semantic Filter (HSF) that collaborates with a Mixture-of-Experts (MoE) Network. The MoE projects heterogeneous text embeddings into the time series latent space, and the HSF cascade-filters text embeddings from the segment level to the token level, thereby avoiding the pitfalls of direct token-level filtering and reducing overhead. We also incorporate a lightweight Missing Mechanism Estimator, jointly optimized with the denoising network to explicitly capture MNAR missingness patterns. Extensive tests on nine domains show that CaT-Diff outperforms state-of-the-art baselines. Our work presents a new approach for selectively fusing LLM-derived textual information.

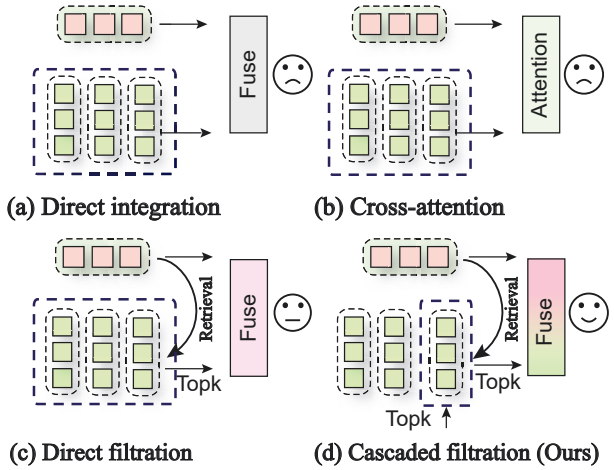

Figure 1: (a) Directly fusing temporal embeddings (in pink) with textual embeddings (in green) containing irrelevant information. (b) Using cross-attention to model the relationships between the two modalities. (c) Directly perform token-level filtering on textual information. (d) Cascaded filtering from segment-level to token-level.

Introduction

Time series data are ubiquitous in domains such as medical monitoring, financial trading, and transportation (Qiu et al. 2024, 2025). However, missing values inevitably arise due to factors including incomplete data entry, equipment failures, or human error (Silva et al. 2012; Yi et al. 2016). Research indicates that missing values themselves can encode latent patterns or critical information, and therefore should not be ignored in time series analysis (Che et al. 2018). Moreover, the quality of missing data imputation critically impacts downstream tasks (Shadbahr et al. 2022); low-quality methods may introduce bias and undermine the validity of subsequent analyses (Zhang et al. 2022).

Deep neural-network-based imputation methods can be divided into deterministic imputation (Liang et al. 2024; Nie et al. 2024) and probabilistic imputation (Liu et al. 2024c; Yang et al. 2024). Deterministic approaches, however, cannot quantify the uncertainty in their predictions, which represents a significant limitation when handling time series data characterized by volatility and noise. Recent studies have introduced diffusion models into the time series imputation task: by virtue of their probabilistic modeling capabilities, these models can flexibly approximate the latent distribution of multivariate time series, achieving significant results (Chen et al. 2023; Zhou et al. 2024).

In many real-world applications, missing values in time series data are closely linked to external factors and contextual information. Relying exclusively on the numerical sequence for imputation neglects these valuable exogenous signals, creating a performance bottleneck and making it difficult to improve the quality of the imputation results. Recently, the introduction of large language model (LLM) derived textual information into time series modeling has led to significant advances (Chen et al. 2024; Jia et al. 2024). Ex-

*Corresponding author

periments with Time-MMD demonstrate that fusing LLM-derived textual information into time series analysis can substantially enhance predictive accuracy (Liu et al. 2024b). Nonetheless, extracting and integrating effective textual information continues to present the following challenges.

LLM-derived textual information inevitably contains irrelevant content that is weakly correlated with time series imputation. If this noise is not filtered out, it can mislead the model's correlation learning and lead to suboptimal performance (Fig. 1(a, b)) (Chang et al. 2024; Jiang et al. 2025). Although a cross-attention mechanism can partially alleviate such interference, processing these unnecessary features still consumes a large portion of the computational budget that could otherwise be devoted to capturing critical temporal patterns (Fig. 1(b)) (Xu et al. 2024). To remove noisy components from LLM-derived textual features, some prior work has attempted direct token selection of the textual information (Fig. 1(c)) (Liu et al. 2024a). However, in realistic settings, a single time series is often associated with multiple text segments organized hierarchically, and direct token retrieval may inadvertently select highly relevant tokens from unrelated sentences or time intervals, imposing erroneous conditional constraints. Moreover, tokens with anomalously high frequency can dominate such filtering ("hijacking" the process), and performing similarity computations over the entire token set still incurs substantial additional overhead.

Existing imputation methods typically assume data are Missing Completely At Random (MCAR) or Missing At Random (MAR), treating the missingness mechanism as negligible and focusing solely on the data distribution (Wang et al. 2024). However, in real-world settings, missingness is often Missing Not At Random (MNAR) and directly related to the unobserved values. Ignoring this mechanism can bias model estimates (Rubin 1976). Therefore, in MNAR scenarios, the data distribution and the missingness mechanism should be modeled jointly, which is a common and recommended practice in statistics (Little 2024).

To address these challenges, we propose CaT-Diff (Cascaded Text-enhanced Diffusion Model), a novel probabilistic imputation method based on Denoising Diffusion Implicit Model (DDIM). The central component of CaT-Diff is the Hierarchical Semantic Filter (HSF), which first uses a Temporal-Semantic Gate to filter out the segments that are most relevant to the time series, then uses Cross-Key Ranking Attention to automatically extract the tokens most helpful for time series imputation from the selected segments, and assigns higher weights to these key pieces of information to suppress the influence of irrelevant information, thereby improving imputation quality (Fig. 1(d)). Compared to direct token selection, the HSF smooths the impact of extremely frequent tokens and, by avoiding filtering over the entire token set, achieves greater stability with reduced noise drift, while also greatly reducing computational overhead. In addition, our Cross-Key Ranking Attention method significantly reduces computational complexity while improving performance compared to Cross-Attention. To simultaneously model data distribution and missing mechanisms in MNAR scenarios, we propose introducing a lightweight Missing Mechanism Estimator into the model and jointly optimizing it with the denoising network in an end-to-end manner. We also explore the use of a computationally efficient Mixture-of-Experts (MoE) (Shazeer et al. 2017) to fine-tune and project token embeddings into the time series feature space, thereby aligning these heterogeneous modalities. Our contributions include:

- We propose CaT-Diff for accurate imputation of time series missing values under MNAR scenarios.
- HSF cascade-filters text embeddings to retain key information and reduce overhead.
- Extensive tests on nine domains show that CaT-Diff outperforms state-of-the-art baselines, cutting MSE by 12.8% and CRPS by 17.2% on average.

Preliminaries

Problem Setup and Notation. We consider a dataset $\mathcal{D} = \{X_i, M_i, S_i\}_{i=1}^{|\mathcal{D}|}$, where $|\mathcal{D}|$ is the total number of samples. $X_i = \{x_{1:K,1:L}\} \in \mathbb{R}^{K\times L}$ is a time series of length L with K variables, it contains missing values and we apply zero padding to each sequence; $M_i = \{m_{1:K,1:L}\} \in \{0,1\}^{K\times L}$ is the corresponding observation mask, where $m_{k,l} = 0$ indicates $x_{k,l}$ is missing and $m_{k,l} = 1$ indicates $x_{k,l}$ is observed. $S_i = \{s_{1:J}\}$ is an ordered set of segments and some segments s_j may be empty. To preserve dimensional consistency for all S_i, we fill these positions with "NaN" and indicate them via a corresponding segment-mask matrix M_i^s. The time series data X_i and the textual data S_i are temporally aligned: $x_{1:K,1:L}$ and $s_{1:J}$ describe information from the same time window. For notational simplicity, the subscript for the variable X will now denote the diffusion timestep rather than the sample index in the following.

We follow the standard DDIM procedure to train our model (Song, Meng, and Ermon 2020). Let X_t for $t = 1, \ldots, T$ be a sequence of latent variables in the same sample space as X_0, which is denoted as $\mathcal{X}$. To simulate real-world scenarios, we adopt a self-supervised learning strategy on the originally incomplete dataset: we further mask a subset of the observed entries using an MNAR mechanism to serve as training targets, denoted by $M^{co} \leq M$. Then during training we define the sets of observed and missing entries as

$$X_0^{ob} = X_0 \odot M^{co}, \quad X_t^{ta} = X_t \odot (M - M^{co}), \tag{1}$$

during sampling we define

$$X_0^{ob} = X_0 \odot M, \quad X_t^{ta} = X_t \odot (1 - M), \tag{2}$$

where $\odot$ denotes element-wise multiplication.

Given a sample X_0 which contains missing values and its corresponding textual information S, we generate imputation targets $X_0^{ta} \in \mathcal{X}^{ta}$ by exploiting the useful textual information and conditional observations $X_0^{ob} \in \mathcal{X}^{ob}$, where $\mathcal{X}^{ob}$ and $\mathcal{X}^{ta}$ are a part of the sample space $\mathcal{X}$ and vary per sample. Let us consider learning a model distribution $p_\theta(X_0^{ta}|X_0^{ob}, S)$ that approximates a data distribution $q(X_0^{ta}|X_0^{ob}, S)$.

MNAR Mechanism. We introduce an advanced data-driven MNAR mechanism that employs a logistic mapping over a standardized latent space to dynamically assign element-wise missingness probabilities, calibrated to any target sparsity via an efficient bisection strategy.

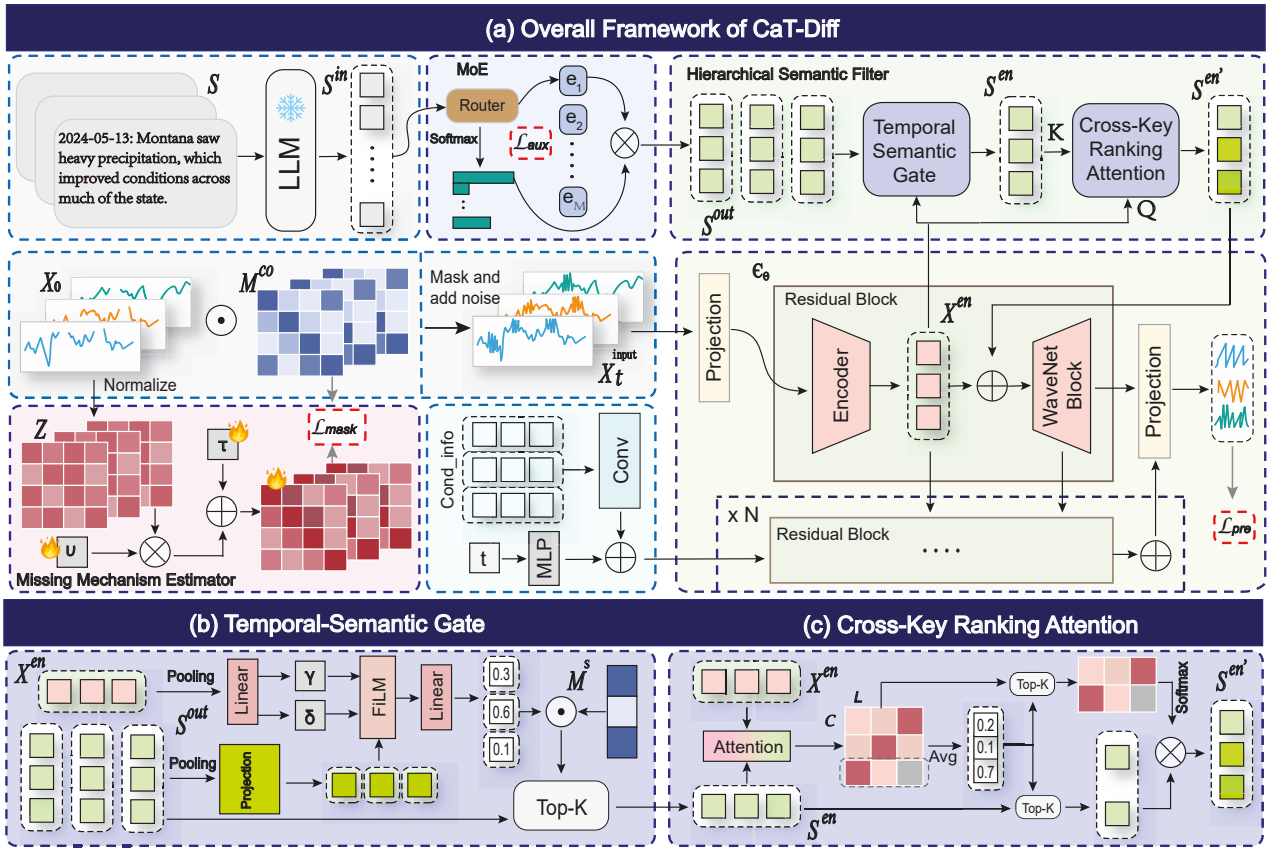

Figure 2: The architecture of the CaT-Diff model. (a) Conditional diffusion pipeline fusing partially observed series and LLM-encoded text via an MoE network and HSF within a denoising model. (b) Temporal-Semantic Gate for selecting top related segments. (c) Cross-Key Ranking Attention for top efficient token attention.

Method

The Model Overview

The illustration of CaT-Diff is shown in Fig. 2(a). We first process the time series according to Eq. (1) to obtain X_t^{ta} and X_0^{ob}. We then concatenate them along a new dimension to form $X_t^{input} \in \mathbb{R}^{2\times K\times L}$, thereby achieving the effect of adding noise to the missing parts of the time series. Next, X_t^{input} is projected and passed through the encoder of first Residual Block to produce the time series embedding $X^{en} \in \mathbb{R}^{L\times D}$ (for notational simplicity, subscripts are omitted), where D is the number of feature dimensions.

On the text side, we first flatten all segments in order and pass the resulting token list through a frozen, pre-trained LLM to obtain token embeddings $S^{in} \in \mathbb{R}^{\mathbf{N}\times d_{in}}$, where $\mathbf{N}$ is the total number of tokens across all segments. An MoE network then projects these embeddings into the time series feature space, and we reshape its output into segment-wise embeddings $S^{out} \in \mathbb{R}^{J\times N\times D}$ with N tokens per segment. Next, HSF uses the time series embedding X^{en} to filter out any segments or tokens in S^{out} that are irrelevant or detrimental, and aligns the text and time series sequence dimensions to produce $S^{en'} \in \mathbb{R}^{L\times D}$. Finally, $S^{en'}$ is fused with the time series embedding X^{en} within several Residual Blocks of the denoising model. Meanwhile, the diffusion step t and conditional information—such as timestamps, target dimensionality, and the conditional mask M^{co}—are encoded via an MLP and a convolutional layer, respectively, and then fused with X^{en} and $S^{en'}$ in the subsequent Residual Blocks. The fused representation is then passed through a WaveNet block (Van Den Oord et al. 2016), processed through multiple stacked layers with aggregated skip connections, and then projected to yield the output at the current time step.

Conditional Diffusion Process

Forward process. The forward process progressively adds Gaussian noise to the original data X_0. A key property of this process is that the noisy sample X_t at any time step t can be sampled directly from X_0. Formally, for each time step $t \in \{1, \ldots, T\}$, the forward process is defined by

$$X_t = \sqrt{\alpha_t} X_0 + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}, \tag{3}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$ is standard Gaussian noise, $\alpha_t = \prod_{i=1}^{t} \bar{\alpha}_i$, $\bar{\alpha}_t = 1 - \beta_t$, $\beta_t > 0$ is hyperparameter defining the noise schedule.

Reverse process. The reverse process is the core of model

learning, whose goal is to recover the original data X_0^{ta} by gradually denoising from Gaussian noise X_T^{ta}. DDIM is an important improvement to DDPM (Ho, Jain, and Abbeel 2020), which constructs a non-Markovian posterior by introducing an arbitrary set of noise scales ξ_t, and greatly improves the sampling speed, while maintaining high quality of the generated results. The conditional distribution of its reverse process depends on X_0^{ta}, which can be derived as a conditional generation process by Bayes' theorem. Let $\epsilon_\theta(X_t^{ta}, t, X_0^{ob}, S)$ be the trainable denoising function, we estimate the clean sample by $\hat{X}_0^{ta} = \frac{X_t^{ta} - \sqrt{1-\alpha_t}\epsilon_\theta}{\sqrt{\alpha_t}}$. For any specified $\xi_t \geq 0$, define:

$$\begin{aligned} p_\theta(X_{t-1}^{ta} | X_t^{ta}, X_0^{ob}, S) &= \mathcal{N}(X_{t-1}^{ta}; \mu_\theta, \xi_t^2 \mathbf{I}), \\ \mu_\theta &= \sqrt{\alpha_{t-1}}\hat{X}_0^{ta} + \sqrt{1-\alpha_{t-1}-\xi_t^2}\epsilon_\theta. \end{aligned} \tag{4}$$

Therefore, the reverse sampling step becomes:

$$X_{t-1}^{ta} = \sqrt{\alpha_{t-1}}\hat{X}_0^{ta} + \sqrt{1-\alpha_{t-1}-\xi_t^2}\epsilon_\theta + \xi_t \boldsymbol{\epsilon}, \tag{5}$$

where $\xi_t^2 = \eta^2 \frac{1-\alpha_{t-1}}{1-\alpha_t}\left(1 - \frac{\alpha_t}{\alpha_{t-1}}\right)$, the hyperparameter η controls the level of stochasticity at each step and implicit sampling is obtained when $\eta = 0$. Given the denoising function ϵ_θ, the temporal data X_0^{ob} and the textual data S, the reverse sampling can be performed according to the above formula to generate X_0^{ta}.

We train the reverse process by minimizing the loss:

$$\mathcal{L}_{\text{pre}} = \mathbb{E}_{\substack{X_0 \sim q(X_0) \\ \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}) \\ t}} \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(X_t^{ta}, t, X_0^{ob}, S) \right\|_2^2. \tag{6}$$

Time Series Encoder

To better capture the complex dependencies within time series data, we introduce a dedicated 2D attention module (Tashiro et al. 2021) for processing time series data in each Residual Block. This module consists of a Temporal Transformer layer and a Feature Transformer layer, which are responsible for learning dependencies between different time points and interactions between different feature dimensions, respectively.

Mixture of Experts Network

The MoE network aims to learn correspondences between multivariate time series patterns and textual tokens. It projects token embeddings into the time series feature space and aligns the dimensions of these heterogeneous modalities. This ensures semantic consistency when the two modalities are integrated into a unified learning framework. The network introduces multiple parallel expert models, enabling the system to dynamically select most relevant expert based on the input. Such selective computation helps expand the model's ability to capture and store information while activating only a few experts in each forward pass, thereby reducing overhead.

A single MoE network consists of a multi-expert router $G(\cdot) : \mathbb{R}^{d_{in}} \to \mathbb{R}^{\mathcal{M}}$ and a set of expert $E = \{e_j\}_{j=1}^{\mathcal{M}}, \quad e_j : \mathbb{R}^{d_{in}} \to \mathbb{R}^D$, where $\mathcal{M}$ denotes the number of experts, d_{in} and D denote the embedding dimensions of input and output tokens, respectively; each expert is a feedforward network with the same structure; the multi-expert router is responsible for assigning each token to one or more experts as input.

For the input sequence tensor S^{in}, we compute the probability matrix:

$$P_{i,j} = \frac{\exp\left((S^{\text{in}} W_r + b_r)_{ij}\right)}{\sum_{\kappa=1}^{\mathcal{M}} \exp\left((S^{\text{in}} W_r + b_r)_{i\kappa}\right)}, \tag{7}$$

where $W_r \in \mathbb{R}^{d_{in} \times \mathcal{M}}$ denotes the trainable parameter matrix and b_r denotes a bias vector. For each token i, the expert index with the highest probability is selected: $k_i^e = \text{TopK}(p_{i,1}, \ldots, p_{i,\mathcal{M}}; \kappa^e)$, where κ^e is a hyperparameter representing the number of experts to be selected each time and k_i^e is the set of indices for these selected experts. Separately feed all tokens routed to the j-th expert into the corresponding feedforward network, assign the output results back to the corresponding positions, and assign corresponding weights based on $P_{i,j}$.

To encourage more balanced token allocation decisions, we adopt auxiliary load-balancing loss to regularize router, ensuring a more uniform load distribution among experts:

$$\mathcal{L}_{\text{aux}} = \mathcal{M} \sum_{j=1}^{\mathcal{M}} \left(\frac{n_j}{\sum_{k=1}^{\mathcal{M}} n_k} \right) \left(\frac{1}{\mathbf{N}} \sum_{i=1}^{\mathbf{N}} P_{i,j} \right), \tag{8}$$

where n_j is the number of tokens routed to the expert j.

Hierarchical Semantic Filter

Temporal-Semantic Gate. We propose a Temporal-Semantic Gate based on FiLM (Perez et al. 2018) to adaptively select the κ^{en} most relevant segments from the candidate segment set. The process first computes a scalar score o_j^{out} for each candidate segment as illustrated in Fig. 2(b):

$$o_j^{\text{out}} = \Phi\left(\tanh\left(\boldsymbol{\gamma} \odot \left(W_e \cdot \text{AvgPool}\left(S_j^{\text{out}}\right)\right) + \boldsymbol{\delta}\right)\right), \tag{9}$$

where $\text{AvgPool}(S_j^{\text{out}})$ denotes average pooling over the valid token embeddings in j-th segment, which are identified by an attention mask. The FiLM parameters, a scaling coefficient $\boldsymbol{\gamma} \in \mathbb{R}^{\mathcal{H}}$ and an offset term $\boldsymbol{\delta} \in \mathbb{R}^{\mathcal{H}}$, are generated by feeding the temporally average-pooled features of X^{en} into a two-layer multilayer perceptron. $W_e \in \mathbb{R}^{\mathcal{H} \times D}$ is a learnable projection matrix and $\Phi(\cdot)$ denotes a linear layer which maps each modulated representation to a scalar score. These scores form the final score vector $o^{out} = (o_1^{out}, \ldots, o_J^{out}) \in \mathbb{R}^J$. We mask invalid segments using a segment mask M^s and select the indices of the top κ^{en} segments:

$$k^s = \text{TopK}\big(o^{\text{out}} \odot M^s,\ \kappa^{en}\big). \tag{10}$$

Finally, the corresponding segment embeddings are extracted from S^{out} based on the index set k^s and concatenated to form the final representation $S^{\text{en}} \in \mathbb{R}^{C \times D}$, where $C = \kappa^{en} N$.

Cross-Key Ranking Attention. To mitigate interference from noisy tokens and reduce the time complexity of conventional cross-attention, we propose Cross-Key Ranking

Attention. As illustrated in Fig. 2(c): For the temporal feature X^{en} and text feature S^{en}, we first calculate the standard scaled dot product attention to obtain the similarity matrix $A \in \mathbb{R}^{L\times C}$ between the i-th time series and the j-th text token, where the temporal feature is used as the query and the text feature is used as the key. To measure the average contribution of each key to the entire query sequence, we take the average over the query dimension to obtain the global importance score for each available key: $o^t = \frac{1}{L}\sum_{i=1}^{L} A_{(i,:)} \in \mathbb{R}^C$, where the attention mask marks the valid tokens.

We select the top κ^t tokens with the highest importance scores as the output for each key, yielding $K^{\kappa^t} \in \mathbb{R}^{\kappa^t\times D}$. Based on the importance scores, we index into the similarity matrix A to extract, for each query, its weights over the κ^t keys, and then re-normalize along the key dimension to obtain $A^{\kappa^t} \in \mathbb{R}^{L\times\kappa^t}$. Finally, the selected keys are weighted and summed to obtain the output:

$$S^{\text{en}'} = A^{\kappa^t} K^{\kappa^t} \in \mathbb{R}^{L\times D}. \tag{11}$$

In this stage, compared to Cross-attention, we reduce the computational complexity from $\mathcal{O}(BLC)$ to $\mathcal{O}(BL\kappa^t)(\kappa^t \ll C)$, significantly reducing computational and memory overhead while maintaining performance.

Modal Fusion

We fuse textual information within several Residual Blocks of the denoising model to control the generation of time series imputations. Specifically, in a shared latent space, we perform element-wise addition of the dimension-aligned time series modality embedding features X^{en} and the text modality embedding features $S^{\text{en}'}$, effectively blending shared information while preserving their interconnections in the latent space.

Missing Mechanism Estimator

To model the binary missing pattern, we introduce a learnable Missing Mechanism Estimator that predicts the observation mask. Specifically, we first perform element-wise normalization on the time series X_0 to obtain $Z = \{z_{1:K,1:L}\} \in \mathbb{R}^{K\times L}$. Then, we apply a learnable affine transformation to the normalized tensor Z to obtain the missingness logits:

$$\ell = \upsilon \odot Z + \tau, \tag{12}$$

where $\ell \in \mathbb{R}^{K\times L}$ and the scale parameter υ and bias τ are learnable parameters. We use the true observation mask M^{co} to supervise the Missing Mechanism Estimator, minimizing the binary cross-entropy loss:

$$\begin{aligned}\mathcal{L}_{\text{mask}} = -\frac{1}{KL}\sum_{i=1}^{K}\sum_{j=1}^{L}\big[&M^{co}_{i,j}\log\sigma(\ell_{i,j}) \\ &+ (1-M^{co}_{i,j})\log\big(1-\sigma(\ell_{i,j})\big)\big],\end{aligned} \tag{13}$$

where σ denotes the sigmoid function.

The Overall Training Process

The final loss of the model is composed of three terms:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{pre}} + \lambda_1\mathcal{L}_{\text{aux}} + \lambda_2\mathcal{L}_{\text{mask}}, \tag{14}$$

where λ_1 and λ_2 are hyperparameters that balance the weights of the auxiliary and mask loss terms.

Experiments

Experimental Setup

Dataset. We employ the Time-MMD dataset to conduct all of our experiments. Time-MMD comprises nine primary domains (Agriculture, Climate, Economy, Energy, Environment, Public Health, Security, SocialGood and Traffic), each providing both time series and paired textual information. The numerical data are drawn from official government agencies at three granularities (daily, weekly and monthly), while the text modalities originate from government publications and targeted web searches. The entire dataset is split into training, validation, and test sets in a 7:1:2 ratio.

Baselines. For a comprehensive evaluation, we benchmark against nine existing models, including the multimodal-input methods MM-TSFlib(integrated with Informer) (Liu et al. 2024b; Zhou et al. 2021), TimeCMA (Liu et al. 2024a) and Time-LLM (Jin et al. 2023); the generative single-modal approaches CSDI (Tashiro et al. 2021), TSDE (Senane et al. 2024) and GP-VAE (Fortuin et al. 2020); and the additional single-modal architectures iTransformer (Liu et al. 2023), BRITS (Cao et al. 2018) and SAITS (Du, Côté, and Liu 2023).

Hardware Platform. All experiments were carried out on a dedicated server equipped with an Intel Xeon Gold 6348 CPU and an NVIDIA A800 GPU (80GB VRAM), running Ubuntu 22.04.3 LTS.

Hyperparameter Settings. We employ a batch size of 32 and the Adam optimizer (Adam et al. 2014). In evaluation settings where 30%, 50% and 70% of the labels are masked during testing, we apply self-supervised training with masking rates of 30%, 30% and 20%, respectively. The denoising model consists of 6 residual layers, with 64 channels per layer and 8 attention heads.

Main Results

We evaluate CaT-Diff against a comprehensive set of baselines on the nine multimodal datasets under the MNAR Scenario. Table 1 summarizes the comparative performance of CaT-Diff and nine state-of-the-art baselines in terms of mean squared error (MSE), where we report the mean and the standard error under three masking ratios (30%, 50% and 70%) for each domain. The superiority of CaT-Diff stems from its effective integration of textual information. Our model achieves best performance in 22 out of 27 experimental setups, significantly outperforming all other baselines across diverse domains. On average, CaT-Diff reduces the MSE by 12.8% compared with all baselines. This substantial improvement highlights the effectiveness of our textual feature selection mechanism in capturing relevant contextual information that complements numerical time series data.

Dataset / Model		SAITS	iTransformer	BRITS	GP-VAE	TSDE	CSDI	TimeCMA	MM-TSFlib	TimeLLM	CaT-Diff
Agriculture	30%	1.121(0.034)	0.818(0.028)	2.689(0.073)	0.647(0.015)	0.438(0.016)	0.435(0.008)	0.420(0.011)	0.830(0.021)	0.493(0.011)	**0.396(0.012)**
	50%	1.312(0.046)	1.371(0.048)	2.764(0.078)	0.718(0.014)	0.713(0.018)	0.653(0.014)	0.643(0.019)	1.392(0.050)	0.654(0.018)	**0.419(0.019)**
	70%	1.481(0.045)	2.217(0.037)	2.993(0.077)	0.932(0.028)	0.882(0.021)	0.768(0.016)	0.864(0.023)	2.250(0.078)	0.716(0.023)	**0.648(0.022)**
Climate	30%	1.255(0.034)	1.194(0.043)	1.220(0.039)	1.260(0.102)	1.116(0.036)	1.127(0.016)	1.114(0.027)	1.212(0.031)	1.112(0.028)	**1.104(0.029)**
	50%	1.336(0.022)	1.346(0.027)	1.387(0.041)	1.348(0.097)	1.225(0.040)	1.201(0.028)	1.204(0.026)	1.366(0.042)	1.185(0.027)	**1.109(0.026)**
	70%	1.498(0.055)	1.381(0.038)	**1.279(0.052)**	1.450(0.038)	1.372(0.053)	1.382(0.036)	1.406(0.033)	1.402(0.047)	1.327(0.042)	1.355(0.031)
Economy	30%	0.823(0.025)	1.258(0.039)	3.919(0.293)	0.159(0.007)	0.198(0.008)	0.348(0.009)	0.262(0.006)	1.277(0.033)	0.543(0.013)	**0.128(0.007)**
	50%	1.201(0.032)	1.594(0.049)	4.719(0.345)	0.262(0.006)	0.275(0.005)	0.579(0.012)	0.584(0.004)	1.618(0.056)	0.775(0.023)	**0.214(0.005)**
	70%	1.251(0.048)	4.531(0.204)	4.919(0.337)	**0.678(0.023)**	0.834(0.012)	0.811(0.016)	0.917(0.028)	4.599(0.073)	0.823(0.026)	0.795(0.018)
Energy	30%	0.168(0.005)	0.303(0.002)	0.863(0.021)	0.488(0.007)	0.228(0.004)	0.151(0.002)	0.131(0.001)	0.461(0.003)	0.151(0.004)	**0.128(0.001)**
	50%	0.189(0.001)	0.454(0.004)	1.009(0.013)	0.521(0.015)	0.362(0.011)	0.172(0.003)	0.137(0.004)	0.308(0.001)	0.173(0.005)	**0.132(0.002)**
	70%	0.213(0.002)	0.745(0.014)	1.278(0.017)	0.624(0.011)	0.419(0.009)	0.191(0.003)	0.166(0.002)	0.756(0.009)	0.192(0.002)	**0.146(0.002)**
Environment	30%	1.227(0.034)	0.980(0.030)	1.331(0.079)	1.106(0.095)	0.794(0.029)	1.221(0.018)	1.219(0.032)	1.128(0.027)	1.374(0.026)	**0.781(0.023)**
	50%	1.316(0.044)	1.226(0.048)	1.448(0.082)	1.208(0.094)	**0.932(0.031)**	1.230(0.022)	1.262(0.028)	1.244(0.009)	1.249(0.029)	1.206(0.025)
	70%	1.375(0.033)	1.638(0.051)	1.606(0.074)	**1.058(0.081)**	1.108(0.033)	1.300(0.027)	1.275(0.038)	1.663(0.062)	1.463(0.047)	1.243(0.034)
Public Health	30%	22.345(0.645)	23.266(0.870)	39.464(0.747)	26.273(0.625)	20.618(0.739)	20.569(0.347)	23.804(0.482)	23.615(0.662)	23.461(0.516)	**16.720(0.432)**
	50%	23.421(0.695)	20.292(0.685)	32.686(0.851)	28.653(0.723)	22.523(0.729)	21.536(0.406)	17.460(0.495)	20.596(0.556)	24.564(0.493)	**15.556(0.451)**
	70%	24.954(0.903)	22.259(0.801)	28.889(0.823)	28.243(0.768)	24.645(0.850)	23.420(0.609)	19.257(0.498)	22.593(0.682)	26.713(0.629)	**18.014(0.567)**
Security	30%	113.924(3.653)	115.933(2.656)	126.189(4.291)	123.692(2.965)	128.372(6.056)	118.778(3.615)	113.456(1.781)	114.305(3.630)	115.674(4.060)	**112.156(3.816)**
	50%	**87.651(1.661)**	90.353(3.046)	101.156(2.806)	97.587(2.413)	103.394(3.235)	92.576(2.606)	90.486(2.104)	97.375(2.516)	90.418(2.450)	90.151(2.114)
	70%	76.211(1.121)	76.524(0.883)	90.417(1.913)	88.645(2.016)	91.060(1.616)	83.011(1.876)	76.456(1.363)	87.630(2.381)	78.165(1.846)	**75.893(1.127)**
SocialGood	30%	1.384(0.027)	1.331(0.047)	2.568(0.086)	2.312(0.087)	1.471(0.041)	1.431(0.031)	1.352(0.032)	1.351(0.030)	1.262(0.031)	**1.125(0.029)**
	50%	1.719(0.055)	1.729(0.065)	2.350(0.052)	2.406(0.071)	1.668(0.063)	1.642(0.026)	1.737(0.049)	1.755(0.028)	1.621(0.034)	**1.613(0.027)**
	70%	1.763(0.052)	2.442(0.058)	2.398(0.068)	2.484(0.063)	1.662(0.054)	1.727(0.023)	1.858(0.049)	2.479(0.041)	1.734(0.023)	**1.525(0.033)**
Traffic	30%	0.201(0.004)	0.324(0.012)	0.719(0.019)	0.346(0.025)	0.281(0.009)	0.204(0.003)	0.250(0.006)	0.329(0.008)	0.186(0.003)	**0.176(0.003)**
	50%	0.236(0.013)	0.490(0.018)	1.179(0.097)	0.537(0.015)	0.343(0.014)	0.264(0.004)	0.293(0.008)	0.497(0.016)	0.218(0.006)	**0.191(0.007)**
	70%	0.310(0.011)	1.536(0.062)	1.807(0.032)	1.113(0.049)	0.369(0.013)	0.469(0.011)	0.377(0.008)	1.559(0.044)	0.281(0.007)	**0.250(0.012)**
Average		13.699	13.983	17.157	15.361	15.085	13.970	13.274	14.651	13.945	**12.710**
1st Count		1	0	1	2	1	0	0	0	0	**22**

Table 1: Comparative Results of Multimodal Time Series Imputation under the MNAR Scenario Across Nine Domains in the Time-MMD Dataset. Best results are shown in **bold**, while second-best results are underlined.

Among the various domains, CaT-Diff achieves notable improvements on the Economy and Energy datasets, reducing the MSE by 6.85% and 6% on average compared with the next-best baselines, respectively. These two datasets generally exhibit relatively smooth fluctuations, but with occasional sharp increases or decreases. CaT-Diff can accurately capture and synchronize the textual information with these abrupt variations, thereby yielding such substantial improvements. On the Security dataset, higher masking ratios are associated with better model performance. This is likely because the series contain pronounced abrupt shifts and our MNAR masking mechanism preferentially masks extreme spikes, so higher missing rates bias the evaluation samples toward smoother, easier-to-predict segments. Furthermore, in challenging domains with high-frequency fluctuations, such as Climate and Environment, CaT-Diff still achieves average MSE improvements of 7.5% and 14.5% over all baselines, attesting to the method's robustness.

While other multimodal models like TimeLLM yield competitive results and validate our motivation to utilize textual information for multivariate time series imputation, they are consistently surpassed by CaT-Diff. This is because they overlook a critical challenge: merely incorporating textual information without an effective selection and integration mechanism can introduce noise rather than valuable signals. TimeCMA achieves the second-best performance thanks to its cross-modality alignment; however, because it only performs token-level filtering, it still underperforms CaT-Diff. The effectiveness of CaT-Diff arises not just from the inclusion of text but from the sophisticated filtering and fusion mechanism enabled by the HSF and MoE network. These components ensure that only the most relevant semantic context is used to guide the imputation process, while the explicit modeling of the missing mechanism further refines performance in realistic MNAR scenarios. The model maintains its leading performance across varying levels of data sparsity, confirming its robustness and generalizability.

Table 2 presents the comparison between CaT-Diff and three probabilistic models using the continuous ranked probability score (CRPS), reporting the mean and the standard error averaged over three masking ratios (30%, 50% and 70%), with 100 samples used to approximate the predictive distribution. Overall, CaT-Diff achieves the lowest average CRPS, corresponding to a 17.2% reduction on average.

Ablation Study

Ablation Studies of Dataset and Model Design. The ablation study was conducted on the Energy and Public Health datasets, and results were averaged over three missing ratios (30%, 50% and 70%) using the MSE, as reported in Table 3. We evaluated six variants of CaT-Diff to isolate the impact of each component: (1) w/o Hierarchical Semantic Filter: fuse time series features with mean-pooled MoE outputs; (2) w/o Temporal-Semantic Gate: feed all segment tokens directly

Dataset / Model	GP-VAE	TSDE	CSDI	CaT-Diff
Agriculture	0.651(0.016)	0.532(0.016)	0.479(0.015)	**0.357(0.011)**
Climate	0.793(0.023)	0.763(0.014)	0.732(0.020)	**0.608(0.021)**
Economy	**0.357(0.007)**	0.416(0.008)	0.534(0.011)	0.360(0.008)
Energy	0.434(0.008)	0.331(0.013)	0.192(0.007)	**0.184(0.004)**
Environment	0.634(0.014)	0.593(0.012)	0.642(0.015)	**0.581(0.008)**
Public Health	2.892(0.087)	2.464(0.031)	2.315(0.137)	**1.857(0.048)**
Security	4.537(0.163)	4.865(0.162)	4.413(0.152)	**4.360(0.147)**
SocialGood	1.018(0.034)	0.621(0.015)	0.625(0.013)	**0.585(0.010)**
Traffic	0.593(0.005)	0.334(0.008)	0.325(0.007)	**0.243(0.004)**
Average	1.323	1.213	1.140	**1.015**

Table 2: Comparison of CaT-Diff with three probabilistic baselines. For each dataset, we report the average results across the three masking ratio settings. Best results are shown in **bold**.

Method / Dataset	Energy	Public Health
CaT-Diff	**0.135(0.002)**	**16.763(0.483)**
w/o Hierarchical Semantic Filter	0.161(0.003)	20.234(0.543)
w/o Temporal-Semantic Gate	0.147(0.003)	18.487(0.482)
w/o Cross-Key Ranking Attention	0.149(0.002)	18.608(0.588)
w/ Cross-Attention	0.138(0.001)	17.110(0.491)
w/o Mixture-of-Experts	0.156(0.004)	19.396(0.528)
w/o Missing Mechanism Estimator	0.141(0.002)	18.359(0.514)
w/o Text	0.170(0.004)	21.680(0.618)
w/ Shuffle-Text	0.182(0.003)	23.153(0.653)
w/ Meaningless-Text	0.183(0.004)	22.997(0.721)

Table 3: Results of ablation studies on dataset and model design. Best results are highlighted in **bold**.

into Cross-Key Ranking Attention; (3) w/o Cross-Key Ranking Attention: apply simple token-level average pooling after selecting top segments; (4) w/ Cross-Attention: replace Cross-Key Ranking Attention with standard cross-attention; (5) w/o Mixture-of-Experts: substitute the MoE module with a vanilla MLP; (6) w/o Missing Mechanism Estimator: remove the missing values prediction module;

We evaluated three dataset variants to isolate the impact of textual information: (1) w/o Text: rely solely on time series inputs; (2) w/ Shuffle-Text: randomly shuffle text descriptions across samples; (3) w/ Meaningless-Text: replace original descriptions with semantically void descriptions.

Results indicate that all components are vital: removing HSF (or its sub-modules) noticeably degrades accuracy, Cross-Key Ranking Attention outperforms Cross-Attention, and ablating either the MoE or the Missing Mechanism Estimator also impairs performance. Performance drops significantly without text. Notably, using misaligned or meaningless text hurts performance even more, proving that irrelevant textual information is harmful and validating our filtering approach. These results confirm that the CaT-Diff's strength lies in its ability to effectively integrate relevant semantic information, not just in its use of a second modality.

Overhead Analysis. As reported in Table 4, we report per-forward compute cost (measured in GFLOPs) and the amount of activated parameters (measured in KBs). Our Cross-Key Ranking Attention requires significantly less computation and memory than standard Cross-Attention, demonstrating its superior design. Similarly, using an MoE network is more efficient than a standard dense network, as it only activates necessary components. Finally, the Temporal-Semantic Gate, while adding minimal parameters, effectively reduces the overall workload by filtering data for subsequent modules. These components work together to make CaT-Diff both powerful and lightweight.

Method	Overhead	
	Activated Params	Computation
CaT-Diff	2970	0.96
w/ Cross-Attention	4410	1.38
w/o Mixture-of-Experts	3460	1.16
w/o Temporal-Semantic Gate	2870	0.99

Table 4: The overhead comparison of four methods.

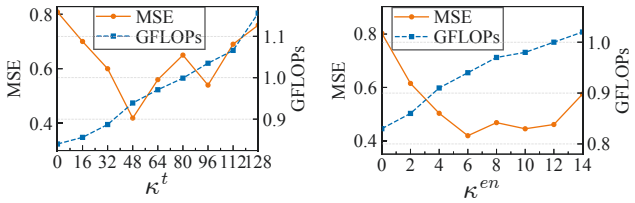

Figure 3: MSE and GFLOPs as functions of κ^t and κ^{en} on the Agriculture dataset with 50% masking.

Hyperparameter Analysis. We evaluate the effects of both κ^t and κ^{en} via sensitivity analyses on the Agriculture dataset with 50% masking. As shown in Fig. 3, when κ^t increases from 16 to 48, the MSE drops sharply—indicating that incorporating more relevant tokens enhances imputation accuracy—while GFLOPs grow approximately linearly with κ^t. Beyond $\kappa^t = 48$, however, the MSE begins to rise as too many tokens introduce noise, revealing a clear accuracy–efficiency trade-off. A parallel pattern emerges for κ^{en}: increasing κ^{en} from 0 to 6 reduces MSE substantially by selecting the most informative segments, but further increases past $\kappa^{en} = 6$ lead to diminishing returns and slight accuracy degradation despite continued GFLOPs growth.

Conclusion

We propose CaT-Diff, a conditional diffusion model that integrates filtered LLM-derived text with observed multivariate time series to accurately impute MNAR data. By introducing an MoE projection layer alongside a novel Hierarchical Semantic Filter and a lightweight Missing Mechanism Estimator, we present a joint modeling approach that aligns heterogeneous modalities, explicitly captures MNAR patterns, and extends the theoretical foundations of conditional diffusion for multimodal imputation. The two-stage HSF provides an extensible, module-agnostic strategy for filtering and weighting long-form context. Future research could integrate our proposed HSF with other models for use in additional downstream tasks.

Acknowledgments

This work was supported by the Sichuan Science and Technology Program (granted No. 2024ZDZX0011).

References

Adam, K. D. B. J.; et al. 2014. A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 1412(6).

Cao, W.; Wang, D.; Li, J.; Zhou, H.; Li, L.; and Li, Y. 2018. Brits: Bidirectional recurrent imputation for time series. *Advances in neural information processing systems*, 31.

Chang, C.; Chan, C.-T.; Wang, W.-Y.; Peng, W.-C.; and Chen, T.-F. 2024. Timedrl: Disentangled representation learning for multivariate time-series. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, 625–638. IEEE.

Che, Z.; Purushotham, S.; Cho, K.; Sontag, D.; and Liu, Y. 2018. Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1): 6085.

Chen, C.; Oliveira, G.; Noghabi, H. S.; and Sylvain, T. 2024. LLM-TS Integrator: Integrating LLM for Enhanced Time Series Modeling. *arXiv preprint arXiv:2410.16489*.

Chen, Y.; Deng, W.; Fang, S.; Li, F.; Yang, N. T.; Zhang, Y.; Rasul, K.; Zhe, S.; Schneider, A.; and Nevmyvaka, Y. 2023. Provably convergent schrödinger bridge with applications to probabilistic time series imputation. In *International Conference on Machine Learning*, 4485–4513. PMLR.

Du, W.; Côté, D.; and Liu, Y. 2023. Saits: Self-attention-based imputation for time series. *Expert Systems with Applications*, 219: 119619.

Fortuin, V.; Baranchuk, D.; Rätsch, G.; and Mandt, S. 2020. GP-VAE: Deep probabilistic multivariate time series imputation. In *Proc. AISTATS*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Jia, F.; Wang, K.; Zheng, Y.; Cao, D.; and Liu, Y. 2024. Gpt4mts: Prompt-based large language model for multimodal time-series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 23343–23351.

Jiang, Y.; Ning, K.; Pan, Z.; Shen, X.; Ni, J.; Yu, W.; Schneider, A.; Chen, H.; Nevmyvaka, Y.; and Song, D. 2025. Multi-modal time series analysis: A tutorial and survey. *arXiv preprint arXiv:2503.13709*.

Jin, M.; Wang, S.; Ma, L.; Chu, Z.; Zhang, J. Y.; Shi, X.; Chen, P.-Y.; Liang, Y.; Li, Y.-F.; Pan, S.; et al. 2023. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*.

Liang, G.; Tiwari, P.; Nowaczyk, S.; and Byttner, S. 2024. Higher-order spatio-temporal physics-incorporated graph neural network for multivariate time series imputation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 1356–1366.

Little, R. J. 2024. Missing Data Analysis. *Annual Review of Clinical Psychology*, 20: 149–173.

Liu, C.; Xu, Q.; Miao, H.; Yang, S.; Zhang, L.; Long, C.; Li, Z.; and Zhao, R. 2024a. Timecma: Towards llm-empowered time series forecasting via cross-modality alignment. *arXiv e-prints*, arXiv–2406.

Liu, H.; Xu, S.; Zhao, Z.; Kong, L.; Prabhakar Kamarthi, H.; Sasanur, A.; Sharma, M.; Cui, J.; Wen, Q.; Zhang, C.; et al. 2024b. Time-mmd: Multi-domain multimodal dataset for time series analysis. *Advances in Neural Information Processing Systems*, 37: 77888–77933.

Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; and Long, M. 2023. itransformer: Inverted transformers are effective for time series forecasting. *arXiv preprint arXiv:2310.06625*.

Liu, Y.; Zhang, H.; Li, C.; Huang, X.; Wang, J.; and Long, M. 2024c. Timer: Generative pre-trained transformers are large time series models. *arXiv preprint arXiv:2402.02368*.

Nie, T.; Qin, G.; Ma, W.; Mei, Y.; and Sun, J. 2024. ImputeFormer: Low rankness-induced transformers for generalizable spatiotemporal imputation. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, 2260–2271.

Perez, E.; Strub, F.; De Vries, H.; Dumoulin, V.; and Courville, A. 2018. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Qiu, X.; Hu, J.; Zhou, L.; Wu, X.; Du, J.; Zhang, B.; Guo, C.; Zhou, A.; Jensen, C. S.; Sheng, Z.; and Yang, B. 2024. TFB: Towards Comprehensive and Fair Benchmarking of Time Series Forecasting Methods. In *Proc. VLDB Endow.*, 2363–2377.

Qiu, X.; Wu, X.; Lin, Y.; Guo, C.; Hu, J.; and Yang, B. 2025. DUET: Dual Clustering Enhanced Multivariate Time Series Forecasting. In *SIGKDD*, 1185–1196.

Rubin, D. B. 1976. Inference and missing data. *Biometrika*, 63(3): 581–592.

Senane, Z.; Cao, L.; Buchner, V. L.; Tashiro, Y.; You, L.; Herman, P. A.; Nordahl, M.; Tu, R.; and Von Ehrenheim, V. 2024. Self-supervised learning of time series representation via diffusion process and imputation-interpolation-forecasting mask. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2560–2571.

Shadbahr, T.; Roberts, M.; Stanczuk, J.; Gilbey, J.; Teare, P.; Dittmer, S.; Thorpe, M.; Torne, R. V.; Sala, E.; Lio, P.; et al. 2022. Classification of datasets with imputed missing values: Does imputation quality matter? *arXiv preprint arXiv:2206.08478*.

Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; and Dean, J. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Silva, I.; Moody, G.; Scott, D. J.; Celi, L. A.; and Mark, R. G. 2012. Predicting in-hospital mortality of icu patients: The physionet/computing in cardiology challenge 2012. In *2012 computing in cardiology*, 245–248. IEEE.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Tashiro, Y.; Song, J.; Song, Y.; and Ermon, S. 2021. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in neural information processing systems*, 34: 24804–24816.

Van Den Oord, A.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; Kavukcuoglu, K.; et al. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 12: 1.

Wang, J.; Du, W.; Yang, Y.; Qian, L.; Cao, W.; Zhang, K.; Wang, W.; Liang, Y.; and Wen, Q. 2024. Deep learning for multivariate time series imputation: A survey. *arXiv preprint arXiv:2402.04059*.

Xu, Z.; Bian, Y.; Zhong, J.; Wen, X.; and Xu, Q. 2024. Beyond trend and periodicity: Guiding time series forecasting with textual cues. *arXiv e-prints*, arXiv–2405.

Yang, X.; Sun, Y.; Chen, X.; et al. 2024. Frequency-aware generative models for multivariate time series imputation. *Advances in Neural Information Processing Systems*, 37: 52595–52623.

Yi, X.; Zheng, Y.; Zhang, J.; and Li, T. 2016. ST-MVL: Filling missing values in geo-sensory time series data. In *Proceedings of the 25th international joint conference on artificial intelligence*.

Zhang, Z.; Xiao, X.; Zhou, W.; Zhu, D.; and Amos, C. I. 2022. False positive findings during genome-wide association studies with imputation: influence of allele frequency and imputation accuracy. *Human molecular genetics*, 31(1): 146–155.

Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; and Zhang, W. 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 11106–11115.

Zhou, J.; Li, J.; Zheng, G.; Wang, X.; and Zhou, C. 2024. Mtsci: A conditional diffusion model for multivariate time series consistent imputation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 3474–3483.