S2S-Diff: STRUCTURE-TO-SIGNAL DIFFUSION MODELS FOR TIME SERIES SYNTHESIS

Anonymous authors
Paper under double-blind review

ABSTRACT

Time-series synthesis is fundamental to simulation, data augmentation, and privacy-preserving data sharing. Despite recent advances in generative modeling, most approaches generate sequences directly in high-dimensional data or latent spaces without modeling an explicit structural representation, limiting interpretability and controllability. To address this limitation, we propose *S2S-Diff*, a structure-to-signal cascaded diffusion model that synthesizes time series through sparse structural composition. *S2S-Diff* represents each sequence as a sparse atom-set composition over a learned multi-scale structural dictionary, where the active atoms summarize the dominant structural characteristics of the sequence. Generative modeling is performed in this compact structural space, and sampled structural compositions condition a diffusion model for signal realization. By decoupling high-level structural composition from fine-grained signal realization, *S2S-Diff* reduces the complexity of generative modeling while introducing an explicit structural inductive bias and preserving stochastic flexibility. Experiments across diverse benchmarks show that *S2S-Diff* achieves strong synthesis quality while maintaining structural sparsity and controllability. These findings support structural abstraction as a principled basis for interpretable and controllable time-series synthesis.

1 INTRODUCTION

Time series data are central to modeling evolving systems across domains including energy (Chen et al., 2017), finance (Bao et al., 2017), healthcare (Zammel et al., 2024), and climate science (Zhou et al., 2023; Lim & Zohren, 2021). However, real-world time series (Esteban et al., 2017; Fortuin et al., 2020) are often scarce, incomplete, or subject to strict privacy constraints, limiting their availability for large-scale learning and evaluation. These challenges have driven growing interest in synthetic time series generation (TSG) (Ang et al., 2023; Yuan & Qiao, 2024) as a means to support data augmentation, robustness assessment, and privacy-preserving data sharing.

Early approaches to TSG extended deep generative modeling frameworks (Goodfellow et al., 2014; Lee et al., 2023; Li et al., 2023), including GAN- and VAE-based methods, to the temporal domain. While these models introduced mechanisms for capturing sequential dependencies, they typically operate directly in high-dimensional signal or latent spaces, treating time series as monolithic objects. As a result, they often struggle to reconcile long-range structure with fine-grained dynamics (Kieu et al., 2019; Yu et al., 2016; Lim et al., 2023; Das et al., 2023), and offer limited interpretability or controllability over the generated sequences.

More recently, denoising diffusion probabilistic models (DDPMs) (Ho et al., 2020) have emerged as a dominant paradigm for time series generation. By progressively corrupting data with noise and learning iterative denoising transitions, diffusion models enable stable optimization and high-fidelity synthesis. Early adaptations (Crabbé et al., 2024; Naiman et al., 2024a; Qian et al., 2024; Feng et al., 2024; Tashiro et al., 2021; Shen & Kwok, 2023; Lopez Alcaraz & Strodthoff, 2022; Li et al., 2022; Kollovieh et al., 2023) such as Diffusion-TS (Yuan & Qiao, 2024) extend diffusion modeling directly to temporal sequences, incorporating mechanisms for periodic and long-range dependency modeling. Subsequent work, including TSGDiff (Shen et al., 2026b), augments the denoising process with graph-based temporal structures to better capture inter-step relationships. Prototype-guided variants such as K-ProtoDiff (Duan et al., 2026) further introduce learned temporal

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

prototypes into the denoising dynamics via projection-based alignment, improving local pattern preservation. Beyond single-space diffusion, L2D-Diff (Shen et al., 2026a) proposes a latent-to-data cascaded framework that decomposes generation into two complementary stages: a latent-space diffusion model captures high-level representation distributions, while a data-space diffusion model refines local temporal details under latent guidance. This dual-space design seeks to balance global semantic coherence with local fidelity in multi-modal time series generation.

Despite architectural variations, existing diffusion-based approaches fundamentally generate entire sequences in data or latent space without explicitly modeling temporal structure as compositional units. Structural patterns remain entangled within high-dimensional representations, limiting interpretability and structural controllability. While holistic generation captures global statistics, it does not disentangle reusable temporal components from their temporal arrangement. Thus, the limitation lies not in model capacity, but in the absence of an explicit structural abstraction that treats temporal components as generative primitives.

We argue that *time series are inherently compositional*: global behaviors emerge from the sparse activation of recurring structural patterns across multiple scales. This perspective motivates a reformulation of time-series generation—not as direct sampling of complete sequences, but as learning a distribution over structural components and composing them to synthesize full sequences. The sparsity assumption is central to this formulation. In many real-world time series, sequence-level behavior can often be summarized by a small subset of recurring structural patterns, while the remaining patterns are irrelevant or redundant. We therefore use sparse structural composition as a compact atom-set abstraction, where the active atoms indicate which structural patterns are selected to characterize the sequence. This representation separates high-level structural abstraction from point-wise signal variation, reducing the complexity of generative modeling while improving interpretability and controllability compared with dense latent embeddings.

In this paper, we therefore propose *S2S-Diff*, a structure-to-signal diffusion framework that reframes time-series synthesis as sparse structural composition over a learned multi-scale dictionary. *S2S-Diff* learns reusable structural patterns across temporal scales and represents each sequence as a sparse atom-set composition, where only a small subset of structural atoms is activated to summarize the sequence-level dynamics. Generative modeling is performed in this structural activation space rather than directly in data or latent space, and sampled structural compositions subsequently condition a diffusion model for final signal realization. As illustrated in Figure 1, instead of directly denoising from random latent vectors, *S2S-Diff* reformulates the denoising objective into two stages: recovering an intermediate structural atom representation and subsequently generating the time series. This decomposition induces an easy-to-hard optimization trajectory: the first stage models a compact sparse atom-set representation that captures high-level structural characteristics, while the second stage realizes fine-grained stochastic signal details conditioned on the sampled atom set. Formally, rather than modeling $p(\mathbf{x})$ in the original data space, *S2S-Diff* models a distribution over structural activations $p(\mathbf{a})$, with time-series realization handled by a conditional diffusion process. Structural patterns govern global temporal organization, while diffusion captures local stochastic variation. By treating structural components as explicit stochastic variables, *S2S-Diff* makes the generative representation inherently structured, enabling principled control and analysis of the synthesis process. Unlike prior approaches where structural cues serve only as auxiliary conditioning, *S2S-Diff* embeds structural abstraction directly into the generative space.

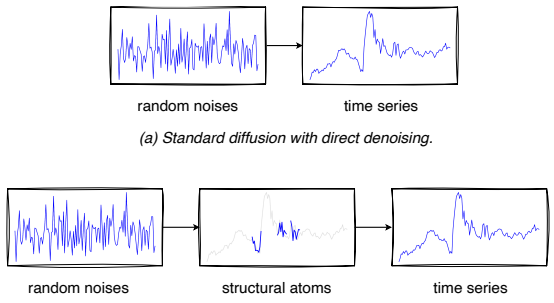


Figure 1: Direct denoising vs. proposed structure-to-signal generation process.

In summary, our contributions are threefold:

- We introduce a structural abstraction perspective on time series synthesis, reformulating synthesis as sparse compositional modeling over learned structure atoms.

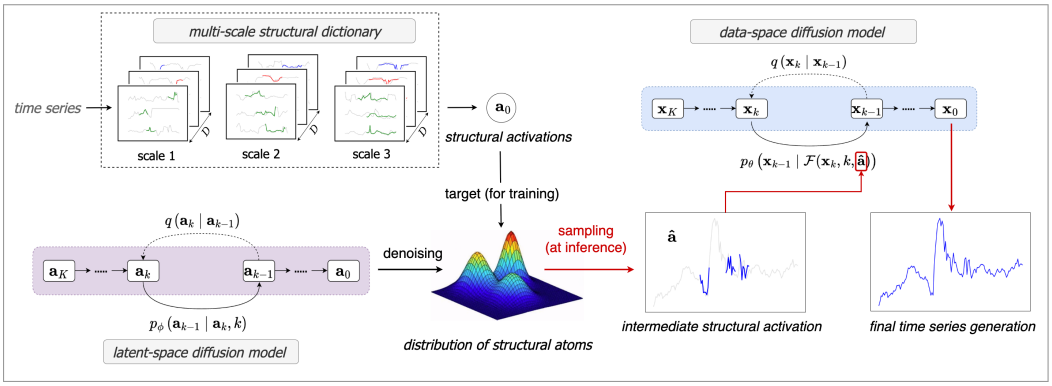


Figure 2: Overview of the proposed structure-to-signal diffusion model *S2S-Diff*.

- We develop *S2S-Diff*, a structure-to-signal diffusion framework that performs modeling in structural activation space and conditions the time-series realization.
- We empirically demonstrate across diverse benchmarks that decoupling structural composition from time-series realization improves distributional fidelity and local pattern preservation, while enabling enhanced interpretability and controllability.

2 Related Work

Existing diffusion-based approaches to time-series synthesis can be roughly grouped according to how they treat temporal structure (Lin et al., 2024).

Holistic diffusion models. Data-space methods such as Diffusion-TS (Yuan & Qiao, 2024) directly model the time-series distribution through denoising in signal space. Latent or cascaded variants, including L2D-Diff (Shen et al., 2026a), first learn compressed representations in the latent space and subsequently refine signals in data space. Despite architectural differences, these approaches fundamentally generate entire sequences in data or latent space, where structural patterns remain implicitly encoded within high-dimensional representations rather than explicitly modeled.

Structure-conditioned diffusion models. Recent works such as TSGDiff (Shen et al., 2026b) and K-ProtoDiff (Duan et al., 2026) incorporate structural priors—e.g., graphs or learned prototypes—into the denoising process. In these frameworks, structural components act as conditioning signals that guide reverse dynamics. However, they are not treated as sample-specific stochastic variables drawn from an explicit structural distribution, and thus do not constitute independent generative factors.

In contrast, the proposed *S2S-Diff* introduces structural activations as intermediate latent variables governed by their own generative distribution $p_\phi(\mathbf{a})$. Time-series synthesis is performed conditionally via $p_\theta(\mathbf{x}|\mathbf{a})$, yielding a hierarchical factorization that explicitly separates structural composition from signal realization. This formulation elevates structural variables from auxiliary guidance to primary stochastic components of generation. Unlike dense latent variables, where structural factors are often entangled, sparse activations explicitly encode the selected structural atoms as an atom-set representation, yielding an intermediate space that is both generative and interpretable.

3 Methodology

Problem Formulation. Let $\mathcal{T} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}\}$ denote a dataset of multivariate time series samples, where each sequence $\mathbf{x} \in \mathbb{R}^{D \times T}$ has dimensionality D and length T. The objective of unconditional time series generation is to learn a model distribution $q(\mathbf{x})$ that approximates the true data distribution $p(\mathbf{x})$, enabling realistic synthesis of new samples. Conventional generative models learn $p(\mathbf{x})$ directly in data or latent space. In contrast, we posit that time series are inherently compositional: global temporal behaviors arise from the sparse activation of reusable structural

patterns across multiple temporal scales. This motivates a structural factorization of the generative process. Further discussion is provided in Appendix B.

Overview. *S2S-Diff* is a structure-to-signal cascaded diffusion model that explicitly models structural composition as a stochastic latent variable. Instead of modeling $p(\mathbf{x})$ directly, *S2S-Diff* defines:

$$p(\mathbf{x}) = \int p_\theta(\mathbf{x} \mid \mathbf{a})\, p_\phi(\mathbf{a})\, d\mathbf{a}. \tag{1}$$

Here, $p_\phi(\mathbf{a})$ models the distribution over structural activations, and $p_\theta(\mathbf{x} \mid \mathbf{a})$ models the conditional distribution of a time series given a structural configuration. In this formulation, structural variables are treated as primary stochastic factors: they determine the compositional organization of a sequence before the realization of its detailed temporal values. The conditional diffusion model parameterizes $p_\theta(\mathbf{x} \mid \mathbf{a})$, capturing residual variability in time series that is not explained by structural composition. By decoupling structural composition from time-series generation, *S2S-Diff* provides an explicit structural abstraction layer while retaining stochastic flexibility in generation. This formulation enables structured manipulation of compositions and promotes sparsity in structural usage within a unified generative framework. An overview is shown in Figure 2.

3.1 Structural Activation Space

To explicitly parameterize structural composition, *S2S-Diff* introduces a learnable multi-scale structural dictionary:

$$\mathcal{D} = \{\mathbf{W}_{s,m} \mid s \in \mathcal{S},\ m = 1, \ldots, M\}, \tag{2}$$

where each structural atom $\mathbf{W}_{s,m} \in \mathbb{R}^{D \times L_s}$ represents a reusable temporal pattern at scale s. M denotes the identical number of atoms for each scale $s \in \mathcal{S}$, D is the dimensionality of the time series, and L_s is the temporal length associated with scale s. Specifically, L_s is determined by the corresponding atom length ratio with respect to the input sequence length; multiple ratios are used to define multiple temporal scales, as detailed in Section 5. Each atom represents a reusable temporal pattern capturing characteristic dynamics at a specific resolution. Multi-scale parameterization enables the dictionary to encode both short-range local segments and longer-range structural segments, forming a compositional basis for representing complex time-series behaviors.

Structural Matching and Activation Construction. Given an input time series $\mathbf{x} \in \mathbb{R}^{D \times T}$, we extract overlapping subsequences of length L_s using sliding windows. Each structural atom $\mathbf{W}_{s,m} \in \mathbb{R}^{D \times L_s}$ serves as a multi-dimensional prototype at scale s. To measure alignment between a subsequence and an atom, we compute the mean absolute deviation:

$$\text{dist}_{s,m,d,t} = \frac{1}{L_s} \sum_{l=1}^{L_s} \left| x_{d,t+l} - W_{s,m,d,l} \right|. \tag{3}$$

The ℓ_1 distance is adopted for its robustness to local amplitude perturbations and outliers, which are common in real-world time-series signals. The normalization by L_s ensures comparability across scales, preventing longer atoms from dominating matching scores.

Then, distances are converted into bounded similarity scores via a radial basis transformation:

$$p_{s,m,d,t} = \exp\left(-(\epsilon_s \,\text{dist}_{s,m,d,t})^2\right). \tag{4}$$

The scale-dependent factor ϵ_s controls the sensitivity of matching. In practice, ϵ_s is set inversely proportional to the empirical standard deviation of distances at scale s, ensuring comparable dynamic ranges across scales. This transformation maps distances into $[0, 1]$, allowing probabilistic interpretation as alignment strength.

To summarize the structural presence of each atom, we aggregate matching responses across temporal positions. First, we compute the best-alignment distance:

$$\text{dist}^{\min}_{s,m,d} = \min_t \text{dist}_{s,m,d,t}, \tag{5}$$

which captures the strongest local correspondence between the atom and the sequence.

Second, we extract the dominant matching response:

$$\hat{p}_{s,m,d} = \max_t p_{s,m,d,t}. \tag{6}$$

Unlike averaging over time, the max-aggregation preserves sparsity and emphasizes the most representative structural occurrence. This design yields a compact structural summary that reflects both alignment quality and peak matching confidence. Note that $\min_t(\cdot)$ and $\max_t(\cdot)$ only take extrema over sliding window time steps within each individual sample, with no cross-minibatch or global dataset aggregation.

The final structural activation is defined as

$$a = \mathrm{AttnFuse}\left(\left\{\mathrm{concat}\left(dist_s^{min}, \hat{p}_s\right)\right\}_{s\in\mathcal{S}}\right) \in \mathbb{R}^{2M\times D}. \tag{7}$$

Each time series is thus mapped into a unified structural activation representation that captures both the *alignment quality* through the minimal deviation and the *alignment strength* through the maximal similarity response, while adaptively integrating information across multiple temporal scales via learnable attention-weighted fusion. This activation space forms the abstraction layer of *S2S-Diff*, providing a compact and interpretable representation of multi-scale structural composition. Although both are derived from the same matching process, they are complementary rather than redundant: $\mathrm{dist}_{\min}$ preserves the absolute deviation in the original distance space, while $\hat{p}$ captures the corresponding normalized matching strength after the nonlinear radial basis transformation.

3.2 Learning the Structural Dictionary

The multi-scale structural dictionary $\mathcal{D} = \{\mathbf{W}_{s,m} \mid s \in \mathcal{S}, m = 1, \ldots, M\}$ is optimized jointly in a stage-wise fashion with the hierarchical generative model in a fully unsupervised manner. Rather than treating structural atoms as fixed templates, we optimize them by gradients so that they form a reusable and statistically meaningful structural basis for the data distribution. After training the structural dictionary, its parameters are fixed and remain unchanged during subsequent diffusion training. To prevent degenerate solutions and promote informative structural primitives, we introduce a composite regularization objective:

$$\mathcal{L}_{\text{struct}} = \lambda_1 \mathcal{L}_{\text{Align}} + \lambda_2 \mathcal{L}_{\ell_1} + \lambda_3 \mathcal{L}_{\text{Div}}, \tag{8}$$

where λ_1 controls the weight of alignment regularization $\mathcal{L}_{Align}$, λ_2 weights the atom sparsity term $\mathcal{L}_{\ell_1}$, and λ_3 balances the inter-atom diversity loss $\mathcal{L}_{Div}$. We set $\lambda_1 = 0.5$, $\lambda_2 = 0.001$, and $\lambda_3 = 0.1$. More implementation details are provided in Section 5.

Data Alignment Regularization. To ensure that structural atoms correspond to representative temporal segments in the data, we introduce an alignment objective based on minimal matching distance:

$$\mathcal{L}_{\text{Align}} = \frac{1}{|\mathcal{S}|} \sum_{s\in\mathcal{S}} \frac{1}{MD} \sum_{m=1}^{M} \sum_{d=1}^{D} d_{s,m,d}^{\min}, \tag{9}$$

where $d_{s,m,d}^{\min}$ denotes the minimal sliding-window distance between atom (s, m) and the data along dimension d.

This term encourages each structural atom to closely match at least one subsequence in the dataset, preventing inactive or uninformative atoms from emerging.

Sparsity Regularization. To encourage compact and localized structural atoms, we impose an ℓ_1 penalty over all dictionary parameters (Wen et al., 2025):

$$\mathcal{L}_{\ell_1} = \frac{1}{|\mathcal{S}|} \sum_{s\in\mathcal{S}} \frac{1}{DML_s} \sum_{d=1}^{D} \sum_{m=1}^{M} \sum_{l=1}^{L_s} |W_{s,m,d,l}|. \tag{10}$$

The normalization by DML_s ensures that the regularization strength remains invariant across different scales and atom sizes, while averaging over $|\mathcal{S}|$ prevents the objective from being dominated by larger scale sets. This sparsity constraint promotes structurally parsimonious atoms that capture salient temporal variations without diffused or trivial patterns. Note we use ℓ_1 regularization on structural atoms $\mathbf{W}_{s,m}$ instead of $\mathbf{a}$, because sparsifying atoms yields concise local patterns, and window-wise max aggregation inherently sparsifies $\hat{\boldsymbol{p}}$, making extra penalty unnecessary.

Atom Diversity Regularization. To avoid redundancy among atoms, we penalize highly correlated structural patterns at each scale. For scale s and variable dimension d, we compute pairwise cosine similarities between flattened atoms (Wen et al., 2025):

$$\mathcal{L}_{\text{Div}} = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \frac{1}{D} \sum_{d=1}^{D} \frac{2}{M(M-1)} \sum_{i<j} \Big\{ \exp\left(\gamma \cdot \max\left(\cos(\mathbf{w}_{s,d,i}, \mathbf{w}_{s,d,j}) - \tau,\ 0\right)\right) - 1 \Big\}, \quad (11)$$

where $\mathbf{w}_{s,d,i} \in \mathbb{R}^{L_s}$ denotes the flattened vector of atom i at scale s and dimension d, τ is a similarity threshold, and γ controls the penalty sharpness. The specific settings of τ and γ are presented in Section 5. Intuitively, The hinge operator $\max(\cos(\cdot) - \tau, 0)$ introduces a soft similarity threshold, penalizing only excessively correlated atoms while allowing moderate structural overlap. This avoids over-constraining the dictionary and preserves modeling flexibility.

Structural Prior Interpretation. The alignment, sparsity, and diversity regularizations jointly induce an implicit structural prior over the dictionary space. Alignment constrains atoms to remain grounded in empirical temporal patterns, sparsity encourages parsimonious usage across sequences, and diversity discourages redundant or degenerate solutions. Under this regularized objective, the learned dictionary converges to a set of reusable multi-scale primitives that capture recurring structural regularities in the data. Rather than serving as a deterministic reconstruction basis, the dictionary defines a structured latent space in which structural compositions can be modeled probabilistically. This perspective naturally motivates diffusion modeling in activation space.

3.3 Modeling Structural Composition $p_\phi(\mathbf{a})$

We model structural activations $\mathbf{a} \in \mathbb{R}^{2M \times D}$ as continuous latent variables governed by an unconditional diffusion process. Concretely, setting $\mathbf{z}_0 = \mathbf{a}_0$, we learn a denoising network $\epsilon_\phi(\mathbf{a}_k, k)$ to approximate the reverse-time dynamics in activation space. The network is instantiated as a Transformer operating along the atom dimension, where each atom activation is treated as a token. Self-attention enables the model to capture inter-atom dependencies and higher-order structural co-occurrences. Through iterative denoising, the model progressively refines noisy activations into coherent structural compositions, thereby defining a generative prior over multi-scale structural activations. The model is optimized using the standard diffusion objective:

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{\mathbf{z}_0, k, \boldsymbol{\epsilon}} \left[\|\boldsymbol{\epsilon} - \epsilon_\psi(\mathbf{z}_k, k)\|_2^2 \right]. \quad (12)$$

3.4 Conditional Time-Series Realization $p_\theta(\mathbf{x}|\mathbf{a})$

Given structural activations $\mathbf{a}$, we generate time-series signals via conditional diffusion. Following Section A, we set $\mathbf{z}_0 = \mathbf{x}_0$ and parameterize the denoising network as $\epsilon_\theta(\mathbf{x}_k, k, \mathbf{a})$, where $\mathbf{a}$ acts as a global latent condition. To integrate structural guidance, we embed $\mathbf{a}$ via feature concatenation:

$$\mathbf{h}(\mathbf{x}_k, k, \mathbf{a}) = W_{in} \left(\text{concat}\left(\mathbf{x}_k, k, \mathbf{a}\right)\right), \quad (13)$$

where $\mathbf{h}(\cdot)$ denotes the fused hidden feature, $\text{concat}(\cdot)$ denotes the feature concatenation operation, and W_{in} is a learnable linear projection matrix.

This mechanism allows global structural composition to modulate intermediate representations without collapsing stochastic variability. Consequently, the conditional diffusion process generates diverse signals that remain structurally consistent with the sampled activation. The conditional model is trained using $\mathcal{L}_{\text{diff-cond}}$ in Equation (20).

The reverse process is implemented using a latent diffusion network with an input layer, multi-layer TransformerEncoder, and an output layer:

$$\begin{aligned} \mathbf{h}_0 &= W_{in} \left(\text{concat}\left(\mathbf{x}_k, k, \mathbf{a}\right)\right), \\ \mathbf{h}_l &= \text{TransformerEncoder}_l\left(\mathbf{h}_{l-1}\right), \quad l = 1, \ldots, L, \\ \hat{\boldsymbol{\epsilon}} &= W_{\text{out}} \mathbf{h}_L, \end{aligned} \quad (14)$$

where W_{out} is a learnable linear projection matrix, $\text{TransformerEncoder}_l$ denotes the l-th layer of the TransformerEncoder, and $\hat{\boldsymbol{\epsilon}}$ represents the predicted noise at diffusion step k.

3.5 Unified Generative Hierarchy

As formulated in Equation 1, *S2S-Diff* introduces sparse structural activations $\mathbf{a}$ as explicit stochastic variables, yielding a hierarchical factorization with a structural prior $p_\phi(\mathbf{a})$ and a conditional signal model $p_\theta(\mathbf{x} \mid \mathbf{a})$. Here, $\mathbf{a}$ serves as a compact atom-set abstraction that specifies the selected structural patterns, while $p_\theta(\mathbf{x} \mid \mathbf{a})$ realizes detailed time-series signals conditioned on this abstraction. This formulation distinguishes *S2S-Diff* from structure-conditioned diffusion models, where structural cues typically act only as auxiliary guidance. In contrast, *S2S-Diff* treats structure as a primary generative factor governed by its own distribution. The resulting hierarchy improves interpretability, enables manipulation of structural variables, and preserves the expressive flexibility of diffusion-based synthesis. The detailed algorithmic pipeline is provided in Appendix C.

4 Experiments

We conduct experiments on nine public real-world time series datasets, including *ETTh*, *Stocks*, *Wind*, *EEG*, *Adiac*, *MiddlePhalanxOutlineCorrect(M.P.)*, *ArabicDigits(A.D.)*, *CharacterTrajectories(C.T.)*, and *Japanese_Vowels(J.V.)*. These datasets cover energy, finance, meteorology, biomedical and image trajectory domains with diverse time spans, sampling frequencies and feature dimensions. Detailed statistics and descriptions are provided in Appendix D.

Additionally, we adopt five widely used metrics for quantitative evaluation of synthetic time series, including Context-FID Score, Correlational Score, Discriminative Score, Predictive Score, and Topo-FID Score. Detailed definitions and descriptions of these metrics are provided in Appendix E.

We select seven representative and high-performance time series generation models for comprehensive comparison, which are divided into three categories: diffusion-based methods including L2D-Diff (Shen et al., 2026a), TSGDiff (Shen et al., 2026b), Diffusion-TS (Yuan & Qiao, 2024); GAN-based methods including TimeGAN (Yoon et al., 2019) and COT-GAN (Xu et al., 2020); as well as VAE-based methods including TimeVAE (Desai et al., 2021) and KoVAE (Naiman et al., 2024b). K-ProtoDiff (Duan et al., 2026) is not included as one of our baseline methods due to the lack of publicly available complete implementation.

4.1 Results for Synthetic Time Series Generation

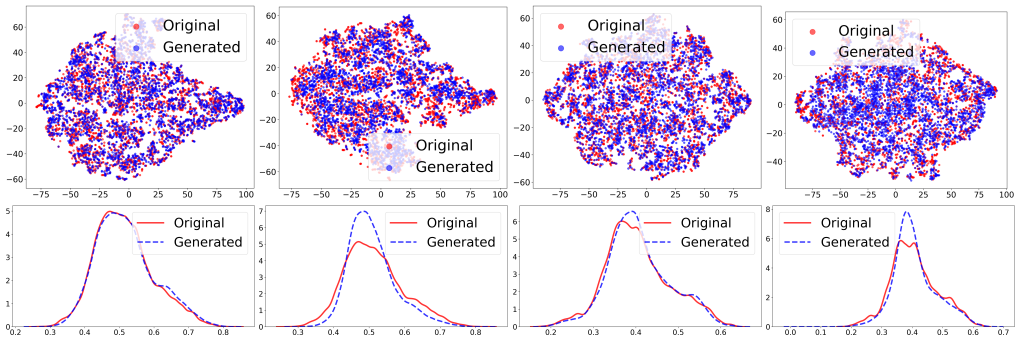
In this section, we follow the experimental settings of all baselines for quantitative comparison and further investigate long-sequence generation in Appendix G.

Evaluation Metrics Performance. Table 1 summarizes quantitative results. We use a standard sequence length of 48 for `ETTh`, `Stocks`, `Wind` and `EEG`, consistent with common practice in time series generation. For `Adiac`, `M.P.`, `A.D.`, `C.T.` and `J.V.`, experiments are conducted with their original dataset-specific fixed lengths. Experimental results demonstrate that *S2S-Diff* consistently outperforms all baseline methods across nearly all evaluation metrics. Specifically, our method achieves a substantial 50% improvement in Context-FID Score on all nine datasets. Notably, *S2S-Diff* maintains prominent performance advantages even high-dimensional datasets such as `EEG` and `A.D.`. This validates that our framework effectively addresses the core challenges in complex time series generation. Complete quantitative results of multivariate time series generation with full evaluation metrics are presented in Appendix F.

t-SNE and Kernel Density Estimation Visualization. Two visualization strategies are adopted to intuitively compare the synthesis performance. First, t-SNE is utilized to project real and synthetic data into a two-dimensional embedding space (van der Maaten & Hinton, 2008). Second, kernel density estimation is applied to visualize data distribution patterns. As illustrated in the first row of Figure 3, the feature distribution of *S2S-Diff* synthetic samples achieves far better overlap with real data than L2D-Diff. The second row of Figure 3 further verifies that the distribution of sequences generated by *S2S-Diff* is highly consistent with the real-world data distribution.

Metric	Methods	ETTh	Stocks	Wind	EEG	Adiac	M.P.	A.D.	C.T.	J.V.
Context-FID ↘ (lower the better)	*S2S-Diff*	**0.071±0.4%**	**0.059±0.5%**	**0.097±0.2%**	**0.005±0.0%**	**0.005±0.6%**	**0.003±0.7%**	**0.108±0.5%**	**0.228±0.3%**	**0.113±0.5%**
	L2D-Diff	0.321±1.1%	0.260±0.5%	0.195±1.4%	0.020±0.6%	0.058±1.0%	0.057±0.7%	3.591±1.2%	0.385±0.3%	0.119±0.6%
	TSGDiff	0.224±0.6%	0.357±0.7%	0.256±1.3%	0.020±0.6%	0.072±0.5%	0.056±0.9%	3.216±0.8%	0.836±0.4%	1.495±0.6%
	Diffusion-TS	0.225±0.4%	0.530±2.5%	0.491±2.4%	0.030±0.6%	0.041±1.0%	0.050±0.1%	1.366±2.1%	1.244±0.9%	1.165±0.6%
	TimeGAN	1.392±0.5%	0.377±1.7%	5.087±2.7%	0.023±0.7%	0.101±0.9%	0.043±2.3%	17.207±1.0%	1.525±1.3%	1.656±0.8%
	Cot-GAN	3.486±0.7%	0.596±0.8%	5.409±1.8%	3.267±1.2%	0.294±2.1%	0.356±1.2%	6.514±0.8%	30.307±0.7%	15.136±0.7%
	TimeVAE	3.452±0.6%	0.830±0.9%	5.174±0.5%	8.874±1.8%	0.074±0.7%	0.041±2.3%	14.309±1.1%	5.960±0.8%	5.989±0.9%
	KoVAE	3.055±0.3%	0.584±1.3%	2.235±0.2%	0.010±0.1%	0.044±0.5%	0.019±0.9%	3.850±1.0%	3.505±1.2%	1.479±0.8%
Correlational ↘ (lower the better)	*S2S-Diff*	**0.022±0.1%**	**0.006±0.1%**	**0.022±0.2%**	**0.088±0.1%**	**0.001±0.2%**	**0.003±0.1%**	**0.037±0.1%**	**0.014±0.1%**	**0.031±0.1%**
	L2D-Diff	0.055±0.4%	0.018±0.5%	0.037±0.4%	0.284±0.6%	0.014±0.3%	0.014±0.1%	0.095±0.3%	0.018±0.5%	0.036±0.4%
	TSGDiff	0.024±0.4%	0.026±2.5%	0.022±2.4%	0.201±0.6%	0.025±1.0%	0.021±0.1%	0.094±0.0%	0.019±0.6%	0.227±0.0%
	Diffusion-TS	0.024±0.4%	0.027±0.1%	0.033±0.7%	0.466±0.2%	0.002±0.5%	0.055±0.3%	0.188±0.5%	0.101±0.2%	0.125±0.3%
	TimeGAN	0.126±0.5%	0.061±0.7%	0.454±0.3%	0.697±0.2%	0.002±0.6%	0.005±0.1%	0.429±0.4%	0.063±0.5%	0.180±0.5%
	Cot-GAN	0.032±0.3%	0.052±1.3%	0.043±0.2%	0.214±0.1%	0.072±0.5%	0.026±0.9%	0.166±0.3%	0.203±0.7%	0.146±0.4%
	TimeVAE	0.025±0.2%	0.053±0.1%	0.064±0.5%	0.209±1.2%	0.001±1.2%	0.029±0.7%	0.294±0.6%	0.131±0.5%	0.558±0.4%
	KoVAE	0.109±0.3%	0.062±1.0%	0.076±0.2%	0.276±0.3%	0.001±0.7%	0.021±0.9%	0.093±0.6%	0.029±0.4%	0.099±0.5%
Topo-FID ↗ (higher the better)	*S2S-Diff*	**0.987±0.0%**	**0.887±0.0%**	**0.869±0.0%**	**0.899±0.0%**	**1.000±0.0%**	**0.999±0.0%**	**0.966±0.0%**	**0.985±0.0%**	0.869±0.0%
	L2D-Diff	0.982±0.0%	0.875±0.0%	0.839±0.0%	0.868±0.0%	1.000±0.0%	0.999±0.0%	0.925±0.0%	0.977±0.0%	0.859±0.0%
	TSGDiff	0.986±0.0%	0.826±0.0%	0.869±0.0%	0.852±0.0%	1.000±0.0%	0.999±0.0%	0.941±0.0%	0.975±0.0%	**0.906±0.0%**
	Diffusion-TS	0.983±0.0%	0.820±0.0%	0.869±0.0%	0.852±0.0%	1.000±0.0%	0.999±0.0%	0.952±0.0%	0.979±0.0%	0.869±0.0%
	TimeGAN	0.949±0.0%	0.812±0.0%	0.824±0.0%	0.828±0.0%	1.000±0.0%	0.999±0.0%	0.964±0.0%	0.978±0.0%	0.863±0.0%
	Cot-GAN	0.968±0.0%	0.823±0.0%	0.825±0.0%	0.841±0.0%	0.999±0.0%	0.978±0.0%	0.939±0.0%	0.972±0.0%	0.845±0.0%
	TimeVAE	0.950±0.0%	0.818±0.0%	0.829±0.0%	0.769±0.0%	1.000±0.0%	0.999±0.0%	0.947±0.0%	0.977±0.0%	0.865±0.0%
	KoVAE	0.972±0.0%	0.832±0.0%	0.843±0.0%	0.864±0.0%	1.000±0.0%	0.999±0.0%	0.951±0.0%	0.984±0.0%	0.843±0.0%

Table 1: Results of multivariate time-series generation (bold indicates best performance, while underline is the second best).



(a) Ours on ETTh (b) L2D-Diff on ETTh (c) Ours on Wind (d) L2D-Diff on Wind

Figure 3: Visual comparison of real and synthesized distributions.

4.2 Ablation Study

We conduct ablation experiments on key model components and structural activation; additional hyperparameter sensitivity results are provided in Appendix H.

Effects of Key Components. To verify the contribution of each core component in the *S2S-Diff* model, we conduct component ablation experiments on the ETTh dataset. We respectively remove $\mathcal{L}_{\ell_1}$, $\mathcal{L}_{\text{Div}}$, and $\mathcal{L}_{\text{Align}}$, and then perform comparative experiments with the complete *S2S-Diff* model. As shown in Table 2, removing any component degrades performance, while the full model achieves the best results, demonstrating the complementary contributions of the three regularization terms.

Table 2: Ablation study on key components of *S2S-Diff* for time series generation on ETTh Dataset (Bold indicates the best).

Method	Context-FID↓	Correlational↓	Discriminative↓	Predictive↓	Topo-FID↑
S2S-Diff	**0.071±0.4%**	**0.022±0.1%**	**0.095±0.3%**	**0.025±0.1%**	**0.987±0.0%**
w/o $\mathcal{L}_{\ell_1}$	0.250±0.4%	0.030±0.2%	0.125±0.3%	0.044±0.1%	0.985±0.0%
w/o $\mathcal{L}_{\text{Div}}$	0.138±0.3%	0.024±0.4%	0.144±0.3%	0.028±0.1%	0.987±0.0%
w/o $\mathcal{L}_{\text{Align}}$	0.452±0.5%	0.048±0.3%	0.214±0.5%	0.029±0.1%	0.986±0.0%

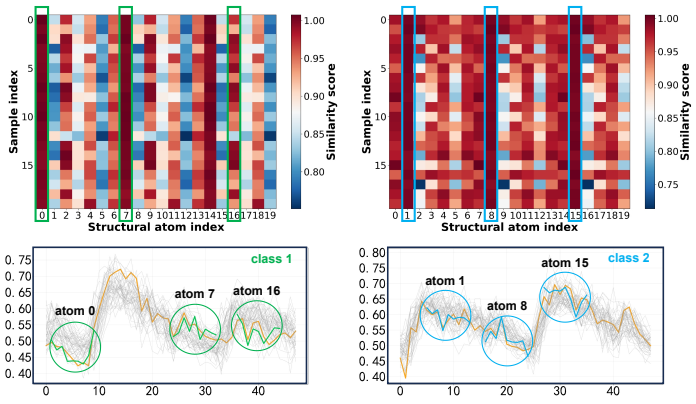


Figure 4: Visualizations of the interpretability of structural atoms in time series synthesis.

Effects of Structural Activation. To verify the effect of structural activation in the *S2S-Diff* model, we conduct component ablation experiments on the `ETTh` dataset. We respectively remove $\hat{p}$, $\text{dist}^{\min}$, and $\mathbf{a}$, and perform comparative experiments with the full *S2S-Diff* model. As shown in Table 3, removing $\hat{p}$, $\text{dist}^{\min}$, or the entire $\mathbf{a}$ consistently degrades performance, particularly on Context-FID, Correlational, and Discriminative scores. These results demonstrate that the complete structural activation effectively improves distributional fidelity and structural coherence.

Table 3: Ablation study on structural activation of *S2S-Diff* on `ETTh` Dataset (Bold indicates the best).

Method	Context-FID↓	Correlational↓	Discriminative↓	Predictive↓	Topo-FID↑
S2S-Diff	**0.071**$\pm$**0.4%**	**0.022**$\pm$**0.1%**	**0.095**$\pm$**0.3%**	**0.025**$\pm$**0.1%**	**0.987**$\pm$**0.0%**
w/o $\hat{p}$	0.278$\pm$0.5%	0.025$\pm$0.3%	0.150$\pm$0.3%	0.029$\pm$0.1%	0.987$\pm$0.0%
w/o $\text{dist}^{\min}$	0.106$\pm$0.3%	0.023$\pm$0.2%	0.102$\pm$0.3%	0.030$\pm$0.1%	0.985$\pm$0.0%
w/o $\mathbf{a}$	0.325$\pm$0.6%	0.040$\pm$0.3%	0.158$\pm$0.5%	0.028$\pm$0.1%	0.986$\pm$0.0%

4.3 Visualization Analysis

As shown in Figure 4, we visualize two groups of time series samples paired with their atom similarity heatmaps. Samples from the same class share consistent high responses over the same set of structural atoms, resulting in closely matched heatmap distributions. By comparison, activation distributions separate clearly between different classes. The most responsive atoms for each class exactly match the highlighted high-scoring regions in their heatmaps. These observations prove our learned structural atoms capture class-specific temporal features and confirm the interpretable activation mechanism for conditional time series generation.

5 Conclusion

In this work, we presented *S2S-Diff*, a structure-to-signal cascaded diffusion model that explicitly factorizes time-series synthesis into structural composition and signal realization. By introducing a learnable multi-scale structural dictionary and modeling a distribution over sparse structural activations, *S2S-Diff* treats structural abstraction as an explicit stochastic variable in the generative process. This factorization separates high-level structure from fine-grained signal variation, improving interpretability and controllability while preserving the expressive power of diffusion-based modeling. Experiments across diverse real-world benchmarks demonstrate consistent gains in fidelity, structural coherence, and controllable generation. These results highlight sparse structural composition as a promising foundation for interpretable and controllable time-series synthesis.

AI USE STATEMENT

No generative-AI large language models were used to create synthetic datasets, build theoretical frameworks, derive mathematical statements, construct proofs, develop hypotheses, design methodologies or experimental pipelines, implement the *S2S-Diff* model, perform qualitative analysis, or interpret experimental results. These core research tasks were solely completed by the human authors. Generative-AI tools were only employed for auxiliary manuscript tasks: brainstorming structure, polishing prose, summarizing background literature, formatting references, and drafting early paragraph versions. All AI-generated text and suggestions were thoroughly reviewed, revised, and fact-checked by the authors. We manually verified AI-assisted writing for logical soundness, technical accuracy and plagiarism against existing time-series generative-model literature. No code or experimental artifacts were directly produced by LLMs. The authors bear full responsibility for all claims, findings, and content in this submission.

ETHICS STATEMENT

This work introduces *S2S-Diff*, a diffusion-based time-series synthesis framework trained on public benchmark datasets (`ETTh`, `Stocks`, `Wind`, `EEG`, `Adiac`, `M.P.`, `A.D.`, `C.T.`, `J.V.`). All datasets are open-access; no new human-subject data was collected, and IRB approval was not required. Our time-series generation method targets legitimate use cases including data augmentation, robustness evaluation, and privacy-preserving data sharing. Like other synthetic data methods, *S2S-Diff* carries ethical risks if misused, e.g., generating misleading financial signals, fabricating biomedical records, or producing deceptive synthetic data, which we discourage. Although the model learns compositional temporal patterns from source benchmarks, synthetic samples may retain original statistical artifacts, noise, and outlier characteristics. Practitioners deploying *S2S-Diff* in real-world scenarios should validate sample fidelity, distribution alignment, and data quality beforehand. We declare no conflicts of interest, commercial sponsorship, or financial biases. All experimental configurations are documented for reproducibility, and this submission complies with the ICLR Code of Ethics.

REPRODUCIBILITY STATEMENT

The following is a detailed description of the experimental parameter settings: In terms of data preprocessing, the sliding window size is 48, the sliding step is 1, and the data normalization range is [-1, 1]; In terms of structural atom module, the number of structural atoms is 20, the multi-scale structural atom length ratio is [0.2, 0.4, 0.6, 0.8], the weight of $\mathcal{L}_{\ell_1}$ is 0.001, the weight of $\mathcal{L}_{\text{Div}}$ is 0.1, γ is 10, τ is 0.5, and the weight of $\mathcal{L}_{\text{Align}}$ is 0.5, the number of training epochs is 100, and the batch size is 128; In terms of the Diffusion model, the hidden layer dimension is 256, the number of Transformer layers is 6, the number of diffusion steps is 1000, the range of diffusion noise coefficients is 0.0001 and 0.2, the number of training epochs is 500, and the batch size is 128. All experiments are repeated five times, implemented in PyTorch (Paszke et al., 2019), and conducted on an NVIDIA RTX 4090 GPU with 24GB memory. The source code will be made public once acceptance.

REFERENCES

Yihao Ang, Qiang Huang, Yifan Bao, Anthony KH Tung, and Zhiyong Huang. Tsgbench: Time series generation benchmark. *Proc. VLDB Endow.*, 2023.

W. Bao, J. Yue, and Y. Rao. A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PLoS ONE*, 12(7):e0180944, 2017.

Y Chen, P Xu, Y Chu, W Li, Y Wu, L Ni, Y Bao, and K Wang. Short-term electrical load forecasting using the support vector regression (svr) model to calculate the demand response baseline for office buildings. *Applied Energy*, 195:659–670, 2017.

Jonathan Crabbé, Nicolas Huynh, Jan Pawel Stanczuk, and Mihaela van der Schaar. Time series diffusion in the frequency domain. In *ICML*, 2024.

Ayan Das, Yongxin Yang, Timothy Hospedales, Tao Xiang, and Yi-Zhe Song. Chirodiff: Modelling chirographic data with diffusion models. *arXiv preprint arXiv:2304.03785*, 2023.

A. Desai, C. Freeman, Z. Wang, and I. Beaver. Timevae: A variational auto-encoder for multivariate time series generation. *arXiv preprint arXiv:2111.08095*, 2021.

Yuhang Duan, Lin Lin, and Xiaoshuai Wu. K-protodiff: Key prototypes-guided diffusion for time series generation. In *AAAI*, 2026.

C. Esteban, S. L. Hyland, and G. Ratsch. Real-valued (medical) time series generation with recurrent conditional gans. *arXiv preprint arXiv:1706.02633*, 2017.

Shibo Feng, Chunyan Miao, Zhong Zhang, and Peilin Zhao. Latent diffusion transformer for probabilistic time series forecasting. In *AAAI*, 2024.

V. Fortuin, D. Baranchuk, G. Ratsch, and S. Mandt. Gp-vae: Deep probabilistic time series imputation. In *AISTATS*, 2020.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Paul Jeha, Michael Bohlke-Schneider, Pedro Mercado, Shubham Kapoor, Rajbir Singh Nirwan, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Psa-gan: Progressive self-attention gans for synthetic time series. In *ICLR*, 2022.

Tung Kieu, Bin Yang, Chenjuan Guo, and Christian S Jensen. Outlier detection for time series with recurrent autoencoder ensembles. In *IJCAI*, 2019.

Marcel Kollovieh, Abdul Fatir Ansari, Michael Bohlke-Schneider, Jasper Zschiegner, Hao Wang, and Yuyang Wang. Predict, refine, synthesize: Self-guiding diffusion models for probabilistic time series forecasting. In *NeurIPS*, 2023.

Daesoo Lee, Sara Malacarne, and Erlend Aune. Vector quantized time series generation with a bidirectional prior model. In *AISTATS*, 2023.

Hongming Li, Shujian Yu, and Jose Principe. Causal recurrent variational autoencoder for medical time series generation. In *AAAI*, 2023.

Yan Li, Xinjiang Lu, Yaqing Wang, and Dejing Dou. Generative time series forecasting with diffusion, denoise, and disentanglement. *NeurIPS*, 35:23009–23022, 2022.

Shujian Liao, Hao Ni, Marc Sabate-Vidales, Lukasz Szpruch, Magnus Wiese, and Baoren Xiao. Conditional sig-wasserstein gans for time series generation. *arXiv preprint arXiv:2006.05421*, 2020.

Bryan Lim and Stefan Zohren. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2194):20200209, 2021.

Haksoo Lim, Minjung Kim, Sewon Park, and Noseong Park. Regular time-series generation using sgm. *arXiv preprint arXiv:2301.08518*, 2023.

Lequan Lin, Zhengkun Li, Ruikun Li, Xuliang Li, and Junbin Gao. Diffusion models for time-series applications: a survey. *FITEE*, 25(1):19–41, 2024.

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.

Ilan Naiman, Nimrod Berman, Itai Pemper, Idan Arbiv, Gal Fadlon, and Omri Azencot. Utilizing image transforms and diffusion models for generative modeling of short and long time series. In *NeurIPS*, 2024a.

Ilan Naiman, N. Benjamin Erichson, Pu Ren, Michael W. Mahoney, and Omri Azencot. Generative modeling of regular and irregular time series data via koopman vaes. In *ICLR*, 2024b.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: an imperative style, high-performance deep learning library. In *NeurIPS*, 2019.

Jian Qian, Bingyu Xie, Biao Wan, Minhao Li, Miao Sun, and Patrick Yin Chiang. Timeldm: Latent diffusion model for unconditional time series generation. *arXiv preprint arXiv:2407.04211*, 2024.

L. Shen and J. T. Kwok. Non-autoregressive conditional diffusion models for time series prediction. In *ICML*, 2023.

Lifeng Shen, Kai Syun Hou, Weiyu Chen, and James T. Kwok. Latent-to-data cascaded diffusion models for unconditional time series generation. In *ICLR*, 2026a.

Lifeng Shen, Xuyang Li, and Lele Long. Tsgdiff: Rethinking synthetic time series generation from a pure graph perspective. In *AAAI*, 2026b.

Y. Tashiro, J. Song, Y. Song, and S. Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. In *NeurIPS*, 2021.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

Yunshi Wen, Tengfei Ma, Ronny Luss, Debarun Bhattacharjya, Achille Fokoue, and Agung Julius. Shedding light on time series classification using interpretability gated networks. In *ICLR*, 2025.

Tianlin Xu, Li K. Wenliang, Michael Munn, and Beatrice Acciaio. Cot-gan: Generating sequential data via causal optimal transport. In *NeurIPS*, 2020.

Jinsung Yoon, Daniel Jarrett, and Mihaela Van der Schaar. Time-series generative adversarial networks. In *NeurIPS*, 2019.

Hsiang-Fu Yu, Nikhil Rao, and Inderjit S Dhillon. Temporal regularized matrix factorization for high-dimensional time series prediction. In *NeurIPS*, 2016.

Xinyu Yuan and Yan Qiao. Diffusion-ts: Interpretable diffusion for general time series generation. In *ICLR*, 2024.

Zina Zammel, Nesrine Khabou, Lotfi Souifi, and Ismael Bouassida Rodriguez. Time series prediction models in healthcare: Systematic literature review. In *ICAART*, 2024.

Linqi Zhou, Michael Poli, Winnie Xu, Stefano Massaroli, and Stefano Ermon. Deep latent state space models for time-series generation. In *ICML*, 2023.

A PRELIMINARIES: DENOISING DIFFUSION PROBABILISTIC MODELS

Denoising diffusion probabilistic models (DDPMs) (Ho et al., 2020) define a generative process by learning to reverse a fixed forward noising Markov chain. Given data $\mathbf{z}_0$, the forward process gradually perturbs it with Gaussian noise:

$$q(\mathbf{z}_k|\mathbf{z}_{k-1}) = \mathcal{N}\big(\sqrt{\alpha_k}\mathbf{z}_{k-1}, \beta_k\mathbf{I}\big), \tag{15}$$

where $\beta_k \in (0,1)$ defines a variance schedule, $\alpha_k = 1-\beta_k$, and $\bar{\alpha}_k = \prod_{i=1}^{k}\alpha_i$. The marginal admits a closed-form:

$$\mathbf{z}_k = \sqrt{\bar{\alpha}_k}\mathbf{z}_0 + \sqrt{1-\bar{\alpha}_k}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0,\mathbf{I}). \tag{16}$$

The reverse process is parameterized as

$$p_\psi(\mathbf{z}_{k-1}|\mathbf{z}_k) = \mathcal{N}\big(\boldsymbol{\mu}_\psi(\mathbf{z}_k,k), \sigma_k^2\mathbf{I}\big), \tag{17}$$

where the mean is expressed via noise prediction:

$$\boldsymbol{\mu}_\psi = \frac{1}{\sqrt{\alpha_k}}\left(\mathbf{z}_k - \frac{\beta_k}{\sqrt{1-\bar{\alpha}_k}}\epsilon_\psi(\mathbf{z}_k,k)\right). \tag{18}$$

The model is trained by minimizing the simplified objective

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{\mathbf{z}_0,k,\boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon} - \epsilon_\psi(\mathbf{z}_k,k)\|_2^2\right], \tag{19}$$

which corresponds to a variational upper bound of the negative log-likelihood.

Conditional diffusion extends this formulation by parameterizing $\epsilon_\psi(\mathbf{z}_k,k,\mathbf{c})$ with an additional conditioning variable $\mathbf{c}$. Then, the corresponding training objective is given by

$$\mathcal{L}_{\text{diff-cond}} = \mathbb{E}_{\mathbf{z}_0,k,\boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon} - \epsilon_\psi(\mathbf{z}_k,k,\mathbf{c})\|_2^2\right]. \tag{20}$$

B FURTHER DISCUSSION ON STRUCTURAL COMPOSITION

Time series often exhibit recurring local behaviors, such as oscillations, peaks, transitions, and cross-variable interactions, which may appear at different temporal resolutions. This motivates representing a time series as a composition of reusable structural atoms rather than generating the entire sequence directly. In *S2S-Diff*, each atom is a short multivariate subsequence jointly describing the temporal evolution of all variables within a local window. Different atom lengths capture structures at different temporal scales, allowing the sequence to be represented as a sparse multi-scale structural composition.

Importantly, structural atoms are not introduced merely as auxiliary features for denoising. As formulated in Equation (1), their activations are explicitly modeled by the structural diffusion distribution $p_\phi(a)$, which is subsequently used to condition signal realization through $p_\theta(x \mid a)$. Thus, *S2S-Diff* models structural composition as an explicit stochastic generative space. This differs from conventional decomposition-based approaches, where decomposition primarily serves representation or denoising; here, reusable structural patterns are explicitly modeled and generated before the realization of fine-grained signals.

C SUPPLEMENTARY ALGORITHMS

The core workflow of *S2S-Diff* comprises two main stages: training and sampling, with their respective implementations detailed in Algorithms 1 and 2.

Algorithm 1 Training of *S2S-Diff* Model

Input: Time series X; Window W; Epochs N_1, N_2, N_3; Diffusion steps K
Output: Trained models $\theta_1, \theta_2, \theta_3$
Preprocess X: Normalize to $[-1, 1]$, slice into subsequences $\mathcal{X}$
Initialize and train structural dictionary θ_1 with Adam optimizer
for epoch $= 1$ to N_1 **do**
  for each batch in $\mathcal{X}$ **do**
    Forward pass and compute total loss $\mathcal{L}_{struct}$
    Update θ_1 by backpropagation
  end for
end for
Extract structural activations $\mathbf{a}$ with trained θ_1
Train latent-space diffusion model θ_2 with MSE loss
for epoch $= 1$ to N_2 **do**
  for each batch in $\mathbf{a}$ **do**
    Sample k, add noise: $a_k = \sqrt{\alpha_k} a_0 + \sqrt{1 - \alpha_k}\epsilon$
    Predict noise and update θ_2 with MSE loss
  end for
end for
Train time series diffusion model θ_3
for epoch $= 1$ to N_3 **do**
  for each batch in $(\mathcal{X}, \mathbf{a})$ **do**
    Sample k, add noise: $x_k = \sqrt{\alpha_k} x_0 + \sqrt{1 - \alpha_k}\epsilon$
    Predict noise and update θ_3 with MSE loss
  end for
end for
Return $\theta_1, \theta_2, \theta_3$

Algorithm 2 Sampling of *S2S-Diff* Model

Input: Trained models $\theta_1, \theta_2, \theta_3$; Diffusion steps K; Samples N
Output: Synthetic time series $\hat{\mathcal{X}}$
Set models to evaluation mode
Initialize noise for Structural activations $a_K \sim \mathcal{N}(0, I)$
for $k = K$ down to 1 **do**
  Denoise step:
    $a_{k-1} = \frac{1}{\sqrt{\alpha_k}} \left(a_k - \frac{1-\alpha_k}{\sqrt{1-\bar{\alpha}_k}} \hat{\epsilon} \right) + \sigma_k \epsilon_k$
end for
Initialize noise for time series $x_K \sim \mathcal{N}(0, I)$
for $k = K$ down to 1 **do**
  Denoise step:
    $x_{k-1} = \frac{1}{\sqrt{\alpha_k}} \left(x_k - \frac{1-\alpha_k}{\sqrt{1-\bar{\alpha}_k}} \hat{\epsilon} \right) + \sigma_k \epsilon_k$
end for
Return synthetic time series $\hat{\mathcal{X}}$

D DATASETS DETAILS

Table 4 shows the statistics of the datasets and all datasets are available online via the link. Detailed descriptions of each dataset are presented as follows: (1) `ETTh`: hourly energy and environmental records from 2016 to 2018, containing over 15,000 time steps; (2) `Stocks`: daily records of multiple stocks with long-term stock prices and trading volume information; (3) `Wind`: hourly wind farm power generation data across different seasons with a time span exceeding one year; (4) `EEG`: high-frequency electroencephalogram signals sampled at 128 Hz, consisting of more than 15,000 samples; (5)`Adiac`: image outlines of different shapes from 70 different classes, containing 176 time steps; (6)`MiddlePhalanxOutlineCorrect (M.P.)`: MiddlePhalanxOutlineCorrect: outlines of the middle bone of the finger from 60 different subjects, consisting of 80 time steps; (7) `ArabicDigits (A.D.)`: handwriting trajectory time series of Arabic digits with over 8,000 samples; (8) `CharacterTrajectories (C.T.)`: three-dimensional motion trajectories of handwritten English characters with multivariate temporal features; (9) `Japanese_Vowels (J.V.)`: speech recordings of two Japanese vowels spoken by 64 male speakers, containing 29 time steps. These datasets differ greatly in time span, sampling frequency and feature dimension, enabling comprehensive evaluation of the adaptability and effectiveness of the proposed model under diverse scenarios.

Table 4: Dataset Details.

Dataset	# of Samples	dim	Link
ETTh(1)	17420	7	https://github.com/zhouhaoyi/ETDataset
Stocks	3685	6	https://finance.yahoo.com/quote/GOOG
Wind	48673	7	https://github.com/PaddlePaddle/.../WindPower
EEG	14980	14	https://archive.ics.uci.edu/.../eeg+eye+state
Adiac	781	377	https://www.cs.ucr.edu/.../time_series_data_2018/
M.P.	892	27	https://www.cs.ucr.edu/.../time_series_data_2018/
A.D.	8800	93	http://www.timeseriesclassification.com/...
C.T.	2858	3	http://www.timeseriesclassification.com/...
J.V.	640	12	http://www.timeseriesclassification.com/...

E EVALUATION METRICS DETAILS

For quantitative assessment of synthetic time series, three primary dimensions are considered: (1) distribution similarity between real and synthetic data; (2) temporal and feature dependency consistency; (3) practicality for downstream forecasting tasks. Five mainstream metrics are employed in our experiments: 1) Discriminative Score (Yoon et al., 2019): a classification-based metric that quantifies distribution similarity by distinguishing real data from synthetic samples as a supervised task; 2) Predictive Score (Yoon et al., 2019): following the Train-on-Synthetic, Test-on-Real (TSTR) protocol, a sequential prediction model is trained on synthetic data and tested on real data to evaluate the practical value of generated sequences; 3) Context-Frechet Inception Distance (Context-FID) Score (Jeha et al., 2022): measures the overall quality of synthetic time series by calculating the discrepancy among local context-aware temporal representations; 4) Correlational Score (Liao et al., 2020): evaluates temporal dependency via the absolute error of correlation matrices between real and synthetic data; 5) Topological Structure Fidelity (Topo-FID) Score (Shen et al., 2026b): assesses structural fidelity from the graph topology perspective and quantifies the consistency of graph-based representations between real and generated time series.

F COMPLETE RESULTS FOR SYNTHETIC TIME SERIES GENERATION

As shown in Table 5, *S2S-Diff* achieves a clear and consistent superiority over all baseline methods across all datasets on all evaluation metrics, including discriminative score and predictive score.

Metric	Methods	ETTh	Stocks	Wind	EEG	Adiac	M.P.	A.D.	C.T.	J.V.
Context-FID ↘ (lower the better)	*S2S-Diff*	**0.071±0.4%**	**0.059±0.5%**	**0.097±0.2%**	**0.005±0.0%**	**0.005±0.6%**	**0.003±0.7%**	**0.108±0.5%**	**0.228±0.3%**	**0.113±0.5%**
	L2D-Diff	0.321±1.1%	0.260±0.5%	0.195±1.4%	0.020±0.6%	0.058±1.0%	0.057±0.7%	3.591±1.2%	0.385±0.3%	0.119±0.6%
	TSGDiff	0.224±0.6%	0.357±0.7%	0.256±1.3%	0.020±0.6%	0.072±0.5%	0.056±0.9%	3.216±0.8%	0.836±0.4%	1.495±0.6%
	Diffusion-TS	0.225±0.4%	0.530±2.5%	0.491±2.4%	0.030±0.6%	0.041±1.0%	0.050±0.1%	1.366±2.1%	1.244±0.9%	1.165±0.6%
	TimeGAN	1.392±0.5%	0.377±1.7%	5.087±2.7%	0.023±0.7%	0.101±0.9%	0.043±2.3%	17.207±1.0%	1.525±1.3%	1.656±0.8%
	Cot-GAN	3.486±0.7%	0.596±0.8%	5.409±1.8%	3.267±1.2%	0.294±2.1%	0.356±1.2%	6.514±0.8%	30.307±0.7%	15.136±0.7%
	TimeVAE	3.452±0.6%	0.830±0.9%	5.174±0.5%	8.874±1.8%	0.074±0.7%	0.041±2.3%	14.309±1.1%	5.960±0.8%	5.989±0.9%
	KoVAE	3.055±0.3%	0.584±1.3%	2.235±0.2%	0.010±0.1%	0.044±0.5%	0.019±0.9%	3.850±1.0%	3.505±1.2%	1.479±0.8%
Correlational ↘ (lower the better)	*S2S-Diff*	**0.022±0.1%**	**0.006±0.1%**	**0.022±0.2%**	**0.088±0.1%**	**0.001±0.2%**	**0.003±0.1%**	**0.037±0.1%**	**0.014±0.1%**	**0.031±0.1%**
	L2D-Diff	0.055±0.4%	0.018±0.5%	0.037±0.4%	0.284±0.6%	0.014±0.3%	0.014±0.1%	0.095±0.3%	0.018±0.5%	0.036±0.4%
	TSGDiff	0.024±0.4%	0.026±2.5%	0.022±2.4%	0.201±0.6%	0.025±1.0%	0.021±0.1%	0.094±0.0%	0.019±0.6%	0.227±0.0%
	Diffusion-TS	0.024±0.4%	0.027±0.1%	0.033±0.7%	0.466±0.2%	0.002±0.5%	0.055±0.3%	0.188±0.5%	0.101±0.2%	0.125±0.3%
	TimeGAN	0.126±0.5%	0.061±0.7%	0.454±0.3%	0.697±0.2%	0.002±0.6%	0.005±0.1%	0.429±0.4%	0.063±0.5%	0.180±0.5%
	Cot-GAN	0.032±0.3%	0.052±1.3%	0.043±0.2%	0.214±0.1%	0.072±0.5%	0.026±0.9%	0.166±0.3%	0.203±0.7%	0.146±0.4%
	TimeVAE	0.025±0.2%	0.053±0.1%	0.064±0.5%	0.209±1.2%	0.001±1.2%	0.029±0.7%	0.294±0.6%	0.131±0.5%	0.558±0.4%
	KoVAE	0.109±0.3%	0.062±1.0%	0.076±0.2%	0.276±0.3%	0.001±0.7%	0.021±0.9%	0.093±0.6%	0.029±0.4%	0.099±0.5%
Discriminative ↘ (lower the better)	*S2S-Diff*	0.095±0.3%	0.075±0.3%	**0.072±0.3%**	**0.297±0.5%**	**0.151±0.5%**	**0.120±0.6%**	**0.175±0.3%**	**0.210±0.2%**	**0.093±0.3%**
	L2D-Diff	0.116±0.4%	0.157±2.5%	0.190±2.4%	0.366±0.6%	0.264±1.0%	0.478±0.1%	0.257±0.5%	0.213±0.5%	0.105±0.7%
	TSGDiff	**0.056±0.6%**	**0.019±1.3%**	0.073±0.3%	0.301±1.5%	0.190±0.9%	0.164±0.1%	0.289±0.7%	0.191±0.5%	0.238±0.6%
	Diffusion-TS	0.111±0.1%	0.075±0.5%	0.161±1.1%	0.492±1.3%	0.372±0.9%	0.125±1.8%	0.499±0.3%	0.395±0.4%	0.367±0.4%
	TimeGAN	0.353±0.7%	0.193±2.5%	0.495±0.5%	0.391±1.7%	0.500±1.5%	0.491±1.8%	0.499±0.5%	0.491±0.6%	0.410±0.4%
	Cot-GAN	0.114±1.5%	0.102±1.6%	0.168±0.7%	0.335±1.1%	0.500±2.5%	0.500±0.8%	0.179±0.4%	0.500±0.3%	0.500±0.4%
	TimeVAE	0.346±1.6%	0.260±3.3%	0.338±0.9%	0.496±2.7%	0.318±2.1%	0.312±1.9%	0.446±0.7%	0.348±0.4%	0.289±0.6%
	KoVAE	0.349±0.3%	0.305±1.3%	0.201±0.2%	0.448±0.1%	0.420±0.5%	0.225±0.9%	0.483±1.0%	0.462±0.5%	0.301±0.7%
Predictive ↘ (lower the better)	*S2S-Diff*	0.025±0.1%	**0.005±0.1%**	**0.007±0.0%**	**0.001±0.1%**	**0.013±0.0%**	**0.006±0.0%**	**0.002±0.0%**	**0.001±0.0%**	**0.001±0.0%**
	L2D-Diff	0.031±0.2%	0.016±0.3%	0.023±0.1%	0.011±0.2%	0.029±0.1%	0.019±0.1%	0.012±0.1%	0.012±0.1%	0.013±0.0%
	TSGDiff	**0.020±0.1%**	0.005±0.1%	0.007±0.0%	0.001±0.1%	0.023±0.0%	0.008±0.0%	0.002±0.1%	0.001±0.1%	0.001±0.1%
	Diffusion-TS	0.026±0.0%	0.005±0.2%	0.009±0.0%	0.002±0.2%	0.013±0.2%	0.009±0.0%	0.002±0.2%	0.002±0.1%	0.003±0.1%
	TimeGAN	0.030±0.2%	0.006±0.1%	0.009±0.1%	0.002±0.3%	0.018±0.1%	0.032±0.1%	0.006±0.2%	0.005±0.1%	0.003±0.2%
	Cot-GAN	0.028±0.2%	0.013±0.1%	0.026±0.2%	0.035±0.3%	0.044±0.3%	0.028±0.2%	0.051±0.2%	0.054±0.1%	0.086±0.3%
	TimeVAE	0.038±0.2%	0.016±0.2%	0.022±0.0%	0.039±0.2%	0.013±0.1%	0.006±0.3%	0.002±0.2%	0.002±0.0%	0.001±0.3%
	KoVAE	0.027±0.3%	0.006±0.3%	0.023±0.2%	0.002±0.1%	0.019±0.3%	0.007±0.2%	0.002±0.3%	0.006±0.2%	0.004±0.1%
Topo-FID ↗ (higher the better)	*S2S-Diff*	**0.987±0.0%**	**0.887±0.0%**	**0.869±0.0%**	**0.899±0.0%**	**1.000±0.0%**	**0.999±0.0%**	**0.966±0.0%**	**0.985±0.0%**	0.869±0.0%
	L2D-Diff	0.982±0.0%	0.875±0.0%	0.839±0.0%	0.868±0.0%	1.000±0.0%	0.999±0.0%	0.925±0.0%	0.977±0.0%	0.859±0.0%
	TSGDiff	0.986±0.0%	0.826±0.0%	0.869±0.0%	0.852±0.0%	1.000±0.0%	0.999±0.0%	0.941±0.0%	0.975±0.0%	**0.906±0.0%**
	Diffusion-TS	0.983±0.0%	0.820±0.0%	0.869±0.0%	0.852±0.0%	1.000±0.0%	0.999±0.0%	0.952±0.0%	0.979±0.0%	0.869±0.0%
	TimeGAN	0.949±0.0%	0.812±0.0%	0.824±0.0%	0.828±0.0%	1.000±0.0%	0.999±0.0%	0.964±0.0%	0.978±0.0%	0.863±0.0%
	Cot-GAN	0.968±0.0%	0.823±0.0%	0.825±0.0%	0.841±0.0%	0.999±0.0%	0.978±0.0%	0.939±0.0%	0.972±0.0%	0.845±0.0%
	TimeVAE	0.950±0.0%	0.818±0.0%	0.829±0.0%	0.769±0.0%	1.000±0.0%	0.999±0.0%	0.947±0.0%	0.977±0.0%	0.865±0.0%
	KoVAE	0.972±0.0%	0.832±0.0%	0.843±0.0%	0.864±0.0%	1.000±0.0%	0.999±0.0%	0.951±0.0%	0.984±0.0%	0.843±0.0%

Table 5: Complete results of multivariate time-series generation (bold indicates best performance, while underline is the second best).

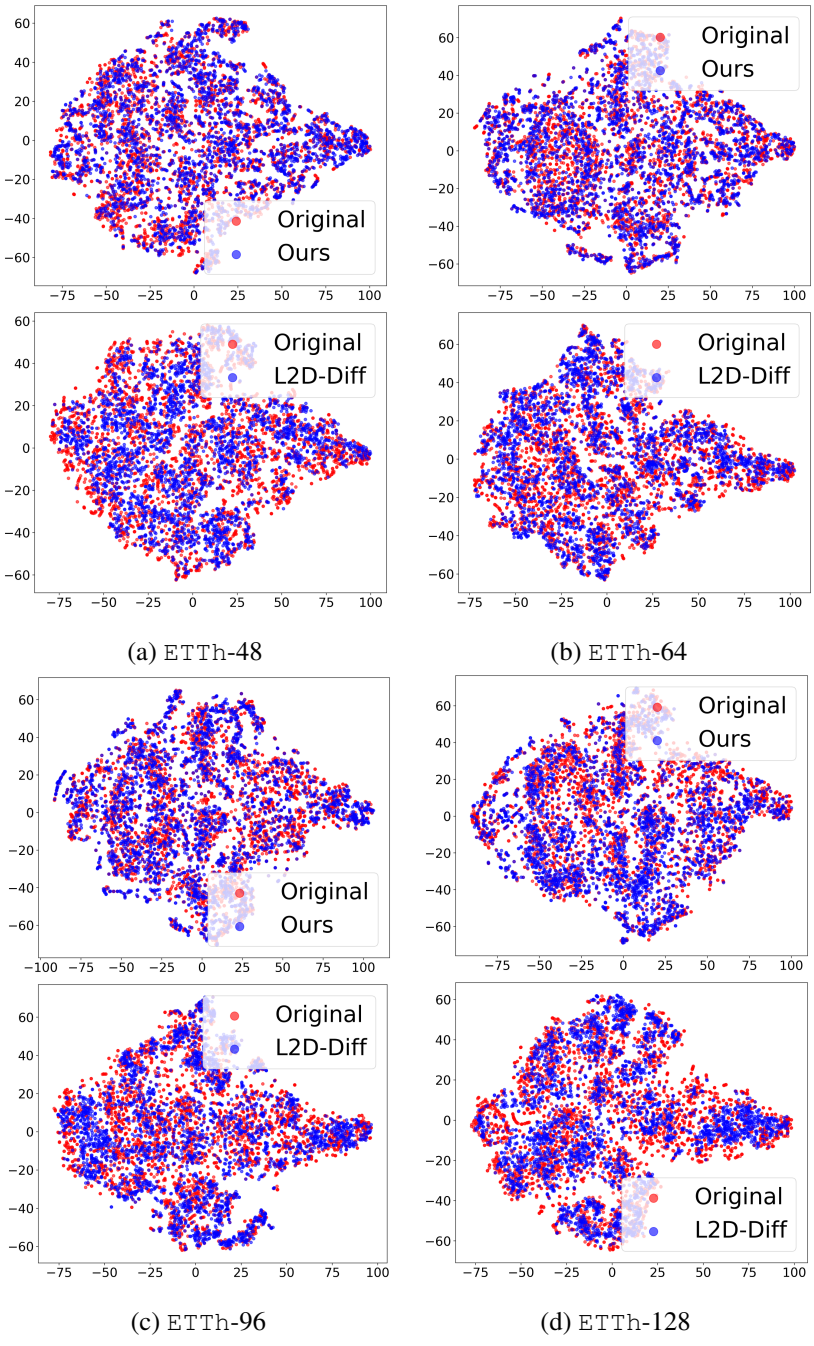
G Supplementary Experimental Results

Results for Synthetic Long-term Time Series Generation. Table 6 presents the generation results of *S2S-Diff*, Diffusion-TS, TimeGAN, Cot-GAN, and TimeVAE for time series of lengths 64, 96, and 128. The results demonstrate that *S2S-Diff* consistently outperforms all baselines across nearly all metrics, producing synthetic samples of superior quality. Notably, *S2S-Diff* excels in the Context-FID score, maintaining excellent performance across all sequence lengths. Unlike other baselines, the performance of *S2S-Diff* remains remarkably stable as sequence length increases. This highlights its superior long-term robustness, making it well-suited for practical applications that involve synthesizing complex, long time series.

Visualization Results. To further illustrate the high quality of the time series generated by *S2S-Diff*, we employed t-SNE, PCA and kernel density estimation analyses to visualize how well the distribution of generated data aligns with that of real data. Figures 5, 6 and 7 show the results of generating time series with lengths of 48, 64, 96, and 128 respectively. Both *S2S-Diff* and Diffusion-TS exhibit substantial overlap with the original data in these visualizations. However, as the sequence length increases, *S2S-Diff* demonstrates exceptional stability, consistently generating data that closely aligns with the original distribution—an ability that Diffusion-TS fails to maintain.

H Hyperparameter Sensitivity Analysis

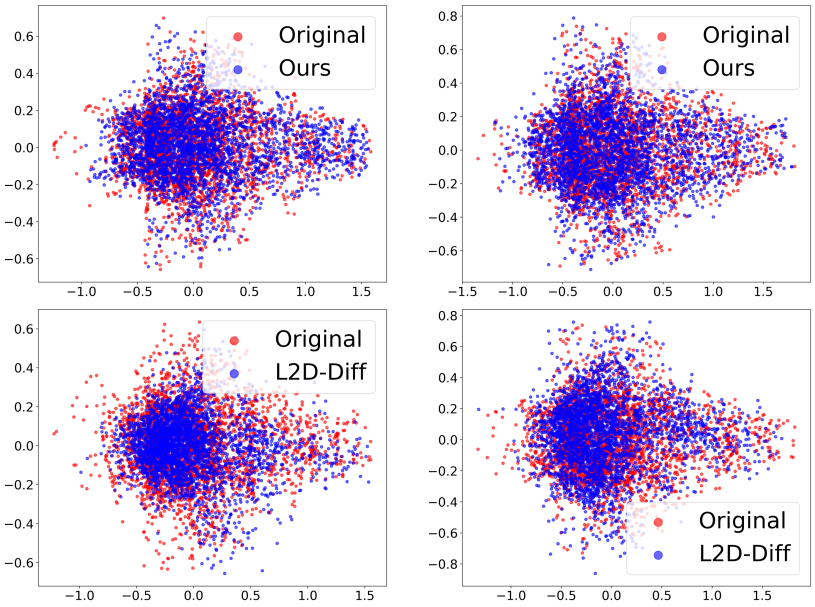
To further investigate how critical hyperparameters affect generation performance, we conduct comprehensive sensitivity experiments over the number of structural atoms, loss coefficients λ_1, λ_2, λ_3, and multi-scale structural atom lengths. Detailed quantitative results are reported in Tables 7–11.



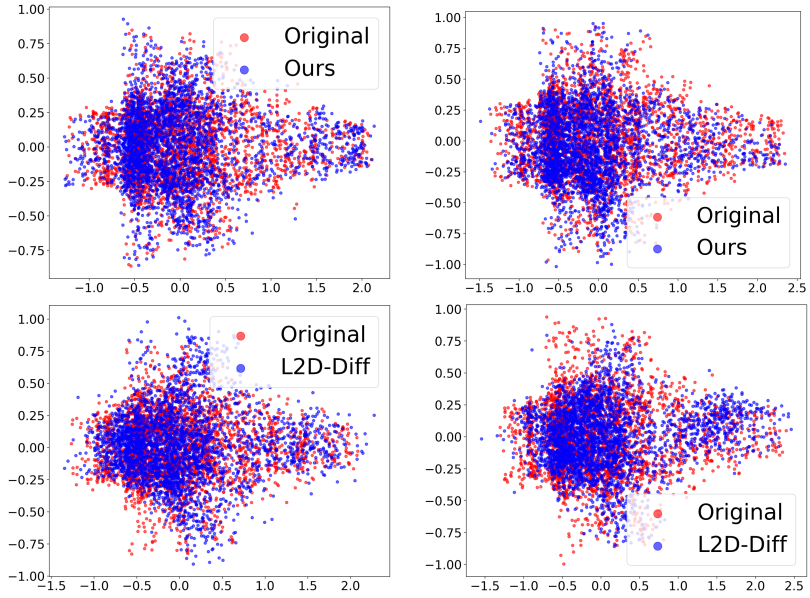
(a) ETTh-48 (b) ETTh-64

(c) ETTh-96 (d) ETTh-128

Figure 5: t-SNE plots of the time series of length 48, 64, 96 and 128 synthesized by Ours and L2D-Diff. Red dots represent real data instances, and blue dots represent generated data samples in all plots.



(a) ETTh-48 (b) ETTh-64

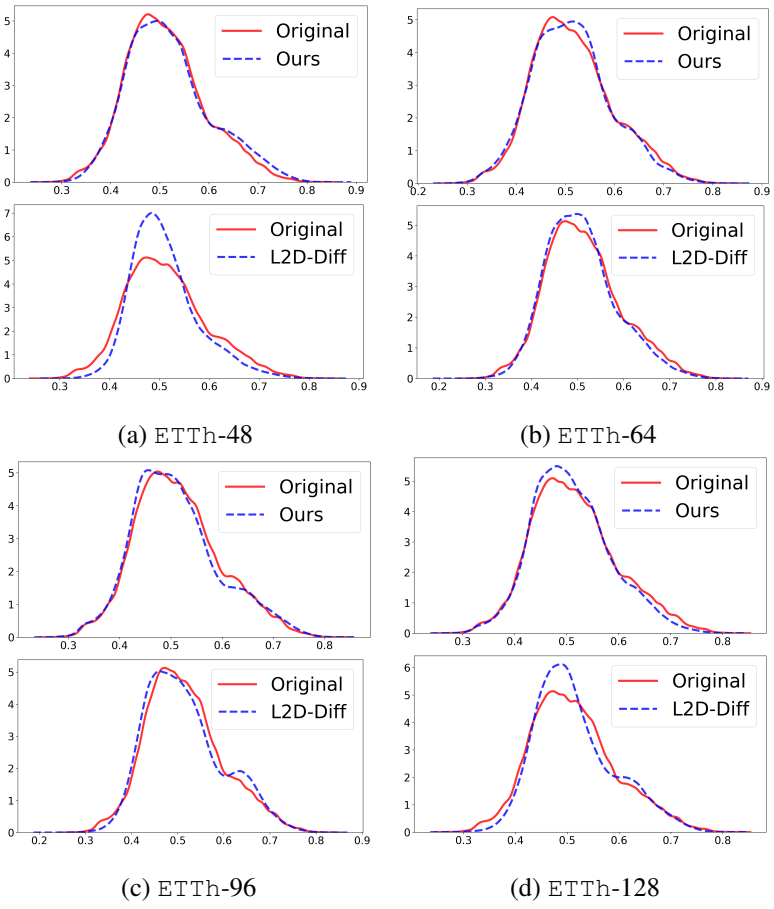


(c) ETTh-96 (d) ETTh-128

Figure 6: PCA plots of the time series of length 48, 64, 96 and 128 synthesized by Ours and L2D-Diff. Red dots represent real data instances, and blue dots represent generated data samples in all plots.

Metric	Length	*S2S-Diff*	L2D-Diff	TSGDiff	Diffsuion-TS	TimeGAN	Cot-GAN	TimeVAE	KoVAE
Context-FID ↘ (lower the better)	64	**0.061**±**0.2%**	0.252±0.4%	0.306±0.5%	0.487±0.9%	6.181±0.6%	3.675±0.7%	3.038±1.2%	3.907±1.3%
	96	**0.183**±**0.3%**	0.255±0.5%	0.367±0.3%	0.412±0.9%	11.631±1.1%	3.823±0.8%	3.476±2.1%	4.657±1.5%
	128	**0.205**±**0.3%**	0.516±0.5%	0.428±0.6%	1.031±0.5%	7.877±0.8%	7.716±0.9%	3.787±1.5%	6.291±0.7%
Correlational ↘ (lower the better)	64	**0.017**±**0.1%**	0.031±0.2%	0.024±0.1%	0.030±0.2%	0.319±0.3%	0.137±0.3%	0.027±0.4%	0.063±0.5%
	96	**0.020**±**0.1%**	0.021±0.2%	0.022±0.1%	0.027±0.3%	0.567±0.5%	0.117±0.2%	0.022±0.2%	0.071±0.6%
	128	**0.020**±**0.2%**	0.027±0.4%	0.021±0.2%	0.047±0.4%	0.489±0.2%	0.144±0.3%	0.023±0.3%	0.070±0.4%
Discriminative ↘ (lower the better)	64	**0.105**±**0.3%**	0.141±0.5%	0.110±0.3%	0.181±0.5%	0.448±1.6%	0.140±1.5%	0.374±1.3%	0.315±0.7%
	96	0.098±0.2%	0.173±0.3%	**0.086**±**0.2%**	0.226±0.7%	0.497±0.7%	0.172±1.0%	0.402±0.5%	0.360±0.4%
	128	**0.103**±**0.1%**	0.144±0.3%	0.104±0.2%	0.349±0.4%	0.492±0.2%	0.118±0.3%	0.407±0.3%	0.375±0.5%
Predictive ↘ (lower the better)	64	**0.020**±**0.0%**	0.044±0.2%	0.020±0.1%	0.025±0.2%	0.034±0.2%	0.088±0.3%	0.145±0.3%	0.026±0.2%
	96	**0.021**±**0.0%**	0.032±0.2%	0.021±0.1%	0.027±0.1%	0.026±0.2%	0.090±0.3%	0.146±0.3%	0.027±0.1%
	128	**0.020**±**0.1%**	0.031±0.1%	0.020±0.2%	0.025±0.1%	0.029±0.2%	0.086±0.2%	0.190±0.3%	0.025±0.1%
Topo-FID ↗ (higher the better)	64	**0.989**±**0.0%**	0.986±0.0%	0.983±0.0%	0.983±0.0%	0.961±0.0%	0.923±0.0%	0.981±0.0%	0.978±0.0%
	96	**0.996**±**0.0%**	0.996±0.0%	0.996±0.0%	0.995±0.0%	0.980±0.0%	0.975±0.0%	0.987±0.0%	0.992±0.0%
	128	0.942±0.0%	0.939±0.0%	**0.943**±**0.0%**	0.934±0.0%	0.918±0.0%	0.863±0.0%	0.925±0.0%	0.930±0.0%

Table 6: Detailed results of long-term time-series generation on ETTh dataset (bold indicates best performance, while underline is the second best).



(a) ETTh-48 (b) ETTh-64

(c) ETTh-96 (d) ETTh-128

Figure 7: kernel density estimation plots of the time series of length 48, 64, 96 and 128 synthesized by Ours and L2D-Diff. Red dots represent real data instances, and blue dots represent generated data samples in all plots.

Atoms	Context-FID ↘	Correlational ↘	Discriminative ↘	Predictive ↘	Topo-FID ↗
5	$0.135\pm_{0.5\%}$	$0.028\pm_{0.2\%}$	$0.098\pm_{0.5\%}$	$0.029\pm_{0.1\%}$	$0.987\pm_{0.0\%}$
10	$0.150\pm_{0.4\%}$	$0.035\pm_{0.2\%}$	$0.116\pm_{0.3\%}$	$0.028\pm_{0.1\%}$	$0.986\pm_{0.0\%}$
15	$0.176\pm_{0.3\%}$	$0.026\pm_{0.3\%}$	$0.118\pm_{0.5\%}$	$0.028\pm_{0.1\%}$	$0.987\pm_{0.0\%}$
20	$\mathbf{0.071\pm_{0.4\%}}$	$\mathbf{0.022\pm_{0.1\%}}$	$\mathbf{0.095\pm_{0.3\%}}$	$\mathbf{0.025\pm_{0.1\%}}$	$\mathbf{0.987\pm_{0.0\%}}$
30	$0.271\pm_{0.4\%}$	$0.035\pm_{0.2\%}$	$0.183\pm_{0.3\%}$	$0.028\pm_{0.1\%}$	$0.983\pm_{0.0\%}$

Table 7: Sensitivity analysis results for the number of structural atoms of *S2S-Diff* on ETTh dataset (bold denotes best performance).

λ_1	Context-FID ↘	Correlational ↘	Discriminative ↘	Predictive ↘	Topo-FID ↗
0.0005	$0.153\pm_{0.4\%}$	$0.037\pm_{0.3\%}$	$0.097\pm_{0.2\%}$	$0.028\pm_{0.1\%}$	$0.985\pm_{0.0\%}$
0.001	$\mathbf{0.071\pm_{0.4\%}}$	$\mathbf{0.022\pm_{0.1\%}}$	$\mathbf{0.095\pm_{0.3\%}}$	$\mathbf{0.025\pm_{0.1\%}}$	$\mathbf{0.987\pm_{0.0\%}}$
0.005	$0.104\pm_{0.3\%}$	$0.025\pm_{0.2\%}$	$0.095\pm_{0.2\%}$	$0.029\pm_{0.2\%}$	$0.987\pm_{0.0\%}$
0.01	$0.157\pm_{0.5\%}$	$0.027\pm_{0.2\%}$	$0.098\pm_{0.3\%}$	$0.028\pm_{0.1\%}$	$0.987\pm_{0.0\%}$
0.05	$0.114\pm_{0.3\%}$	$0.026\pm_{0.3\%}$	$0.223\pm_{0.4\%}$	$0.029\pm_{0.2\%}$	$0.987\pm_{0.0\%}$
0.1	$0.112\pm_{0.4\%}$	$0.028\pm_{0.2\%}$	$0.138\pm_{0.3\%}$	$0.028\pm_{0.1\%}$	$0.986\pm_{0.0\%}$

Table 8: Sensitivity analysis results for sparsity loss weight λ_1 of *S2S-Diff* on ETTh dataset (bold denotes best performance).

λ_2	Context-FID ↘	Correlational ↘	Discriminative ↘	Predictive ↘	Topo-FID ↗
0.05	$0.114\pm_{0.3\%}$	$0.011\pm_{0.2\%}$	$0.110\pm_{0.3\%}$	$0.028\pm_{0.1\%}$	$0.987\pm_{0.0\%}$
0.1	$\mathbf{0.071\pm_{0.4\%}}$	$\mathbf{0.022\pm_{0.1\%}}$	$\mathbf{0.095\pm_{0.3\%}}$	$\mathbf{0.025\pm_{0.1\%}}$	$\mathbf{0.987\pm_{0.0\%}}$
0.2	$0.150\pm_{0.4\%}$	$0.026\pm_{0.2\%}$	$0.105\pm_{0.2\%}$	$0.028\pm_{0.1\%}$	$0.987\pm_{0.0\%}$
0.3	$0.125\pm_{0.3\%}$	$0.029\pm_{0.3\%}$	$0.140\pm_{0.4\%}$	$0.030\pm_{0.2\%}$	$0.987\pm_{0.0\%}$
0.4	$0.097\pm_{0.3\%}$	$0.028\pm_{0.2\%}$	$0.112\pm_{0.3\%}$	$0.030\pm_{0.1\%}$	$0.987\pm_{0.0\%}$
0.5	$0.166\pm_{0.5\%}$	$0.025\pm_{0.2\%}$	$0.121\pm_{0.3\%}$	$0.029\pm_{0.2\%}$	$0.987\pm_{0.0\%}$

Table 9: Sensitivity analysis results for diversity loss weight λ_2 of *S2S-Diff* on ETTh dataset (bold denotes best performance).

λ_3	Context-FID ↘	Correlational ↘	Discriminative ↘	Predictive ↘	Topo-FID ↗
0.2	$0.257\pm_{0.5\%}$	$0.023\pm_{0.2\%}$	$0.133\pm_{0.3\%}$	$0.029\pm_{0.2\%}$	$0.987\pm_{0.0\%}$
0.4	$0.118\pm_{0.3\%}$	$0.022\pm_{0.1\%}$	$0.157\pm_{0.4\%}$	$0.028\pm_{0.1\%}$	$0.987\pm_{0.0\%}$
0.5	$\mathbf{0.071\pm_{0.4\%}}$	$\mathbf{0.022\pm_{0.1\%}}$	$\mathbf{0.095\pm_{0.3\%}}$	$\mathbf{0.025\pm_{0.1\%}}$	$\mathbf{0.987\pm_{0.0\%}}$
0.6	$0.152\pm_{0.4\%}$	$0.031\pm_{0.3\%}$	$0.136\pm_{0.3\%}$	$0.030\pm_{0.2\%}$	$0.987\pm_{0.0\%}$
0.8	$0.192\pm_{0.5\%}$	$0.021\pm_{0.2\%}$	$0.210\pm_{0.4\%}$	$0.028\pm_{0.1\%}$	$0.985\pm_{0.0\%}$
1.0	$0.328\pm_{0.6\%}$	$0.034\pm_{0.3\%}$	$0.145\pm_{0.3\%}$	$0.028\pm_{0.1\%}$	$0.986\pm_{0.0\%}$

Table 10: Sensitivity analysis results for alignment loss weight λ_3 of *S2S-Diff* on ETTh dataset (bold denotes best performance).

Scales	Context-FID ↘	Correlational ↘	Discriminative ↘	Predictive ↘	Topo-FID ↗
[0.2, 0.4, 0.6, 0.8]	**0.071**$\pm$**0.4%**	**0.022**$\pm$**0.1%**	**0.095**$\pm$**0.3%**	**0.025**$\pm$**0.1%**	**0.987**$\pm$**0.0%**
[0.1, 0.25, 0.4, 0.55]	0.194$\pm$0.6%	0.025$\pm$0.3%	0.161$\pm$0.4%	0.029$\pm$0.2%	0.987$\pm$0.0%
[0.3, 0.5, 0.7, 0.9]	0.126$\pm$0.5%	0.026$\pm$0.3%	0.121$\pm$0.2%	0.028$\pm$0.1%	0.986$\pm$0.0%
[0.15, 0.3, 0.6, 0.9]	0.197$\pm$0.6%	0.220$\pm$0.2%	0.239$\pm$0.4%	0.029$\pm$0.1%	0.986$\pm$0.0%
[0.2, 0.5, 0.65, 0.8]	0.103$\pm$0.3%	0.024$\pm$0.3%	0.106$\pm$0.2%	0.028$\pm$0.2%	0.987$\pm$0.0%
[0.1, 0.4, 0.7, 0.9]	0.123$\pm$0.5%	0.024$\pm$0.4%	0.126$\pm$0.3%	0.028$\pm$0.2%	0.987$\pm$0.0%

Table 11: Sensitivity analysis results for multi-scale structural atom lengths on ETTh dataset (bold denotes best performance).