MORITS: MORPHOLOGY-AWARE MODELING FOR IRREGULAR TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Irregular multivariate time series (IMTS) are prevalent in healthcare, biomechanics, climate science, and astronomy. Observations occur at irregular intervals within variables and at misaligned timestamps across variables. Existing methods typically treat timestamps and missingness masks only as point-level auxiliary information, without explicitly modeling patch-level observation morphology jointly formed by temporal distributions and missingness patterns. This makes it difficult for local semantic extraction and cross-patch interactions to adapt to different observation structures. To address this issue, we propose **MorITS**, a morphology-aware framework for IMTS forecasting. The Observation Morphology Modeling module uses localized Gaussian basis functions to jointly model all timestamps and missingness masks within each patch, constructing morphology representations independent of observed values. The Morphology-Guided Expert Mixture module adaptively extracts patch semantics through morphology-conditioned expert routing and semantic aggregation. The Semantic–Morphology Patch Interaction module combines semantic similarity with inter-patch morphology relations to guide cross-patch interaction. MorITS achieves state-of-the-art forecasting performance on five real-world datasets.

1 INTRODUCTION

Irregular Multivariate Time Series Forecasting (IMTSF) has important applications in healthcare, biomechanics, climate science, and astronomy (Yao et al., 2018; Vidulin et al., 2010; Menne et al., 2016; Vio et al., 2013). In these domains, data collection is often driven by clinical demand, sensor availability, or natural events rather than a fixed sampling schedule. Observations are therefore unevenly spaced within each variable and recorded at misaligned timestamps across variables, as shown in Figure 1(a). Such irregular sampling creates substantial differences in observation density, temporal distribution, and missingness patterns across local segments, making effective local semantic extraction and inter-segment dependency modeling essential for accurate forecasting.

Patch-based IMTSF methods typically use timestamps and missingness masks to encode observation times and missingness, supporting the learning of semantic representations for local patches (Qiu et al., 2026). In practice, within each patch, the temporal positions of observations and missingness patterns jointly shape its observation distribution. As shown in Figure 1(b), patches obtained under the same partitioning scheme can exhibit sparse, dense, dispersed, or concentrated observation patterns. In this paper, we refer to this value-independent patch-level structure, formed jointly by temporal distributions and missingness patterns, as observation morphology. Observation density, temporal distribution, and clustering patterns within a patch jointly form its morphology, which affects how its semantic representation is learned. As shown in Figure 1(c), observation points with identical values and timestamps can convey different meanings during semantic extraction in their respective patches because their surrounding observations are distributed differently. Patches with the same pattern of value changes can express different semantics when their morphologies differ.

However, existing methods typically treat timestamps and missingness masks only as point-level auxiliary information and do not explicitly model observation morphology, making it difficult to adapt patch semantic representations to different observation structures. This limitation has two aspects. (1) As shown in Figure 2(a), existing methods lack a morphology-conditioning mechanism when extracting patch semantics, making it difficult to adaptively adjust feature extraction to struc-

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.

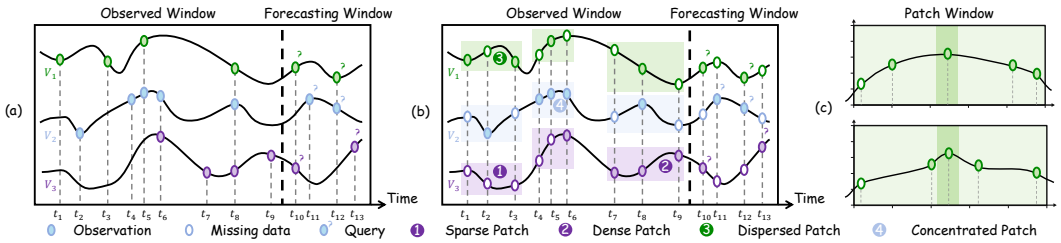
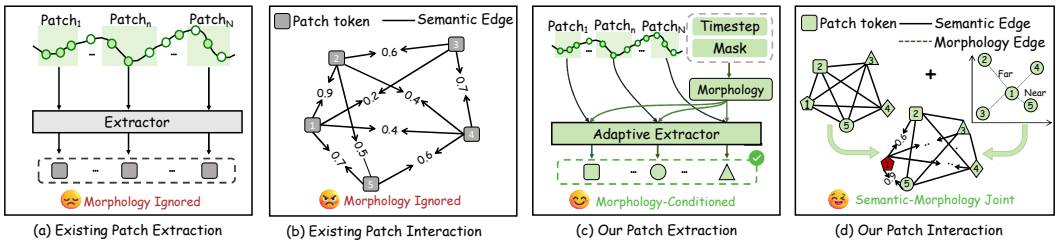


Figure 1: Observation morphology in IMTS. (a) Intra-variable irregularity and inter-variable asynchrony. (b) Patches under the same partitioning exhibit sparse, dense, dispersed, or concentrated patterns. (c) Different observation distributions give points sharing values and timestamps different meanings, and patches sharing value-change patterns different semantics.



(a) Existing Patch Extraction (b) Existing Patch Interaction (c) Our Patch Extraction (d) Our Patch Interaction

Figure 2: Comparison of existing methods and MorITS. (a) Existing methods lack morphology conditioning during patch semantic extraction. (b) Existing methods rely mainly on semantic similarity without explicitly modeling inter-patch morphology relations. (c) MorITS uses morphology conditioning for adaptive patch semantic extraction. (d) MorITS combines semantic similarity and morphology relations to guide cross-patch interaction.

tural differences across patches. (2) As shown in Figure 2(b), cross-patch interactions in existing methods rely mainly on semantic similarity, without explicitly modeling morphology relations between patches, making it difficult to use structural differences to regulate information propagation.

To address these limitations, we propose **MorITS**, a morphology-aware framework for IMTSF. Observation Morphology Modeling (OMM) uses localized Gaussian basis functions to jointly model all timestamps and missingness masks within each patch, explicitly constructing patch-level observation morphology representations. Building on these representations, the Morphology-Guided Expert Mixture (MGEM) adaptively extracts patch semantics through morphology-conditioned expert routing and semantic aggregation, as shown in Figure 2(c). Semantic–Morphology Patch Interaction (SMPI) uses semantic similarity and inter-patch morphology relations to guide cross-patch interaction, as shown in Figure 2(d).

Our main contributions are summarized as follows:

- We propose **MorITS**, a morphology-aware framework for IMTSF, and are the first to introduce the concept of patch-level observation morphology.
- We design coordinated OMM, MGEM, and SMPI modules to perform patch-level observation morphology modeling, morphology-conditioned local semantic extraction, and cross-patch interaction integrating semantic similarity and morphology relations, respectively.
- We conduct extensive experiments on multiple real-world IMTSF datasets, demonstrating that MorITS achieves state-of-the-art performance.

2 Related Work

2.1 Irregular Time-Series Modeling

Methods for irregular time series handle nonuniform observations through discrete-time updates, continuous-time dynamics, and time-aware aggregation (Shukla & Marlin, 2020). Discrete-time recurrent models use temporal decay, missingness masks, or time gates in state updates (Che et al.,

2018; Baytas et al., 2017; Cao et al., 2018; Neil et al., 2016). Neural ODE and Latent ODE parameterize continuous hidden dynamics, while GRU-ODE-Bayes, ODE-LSTM, and CRU combine continuous evolution with observation updates (Chen et al., 2018; Rubanova et al., 2019; De Brouwer et al., 2019; Lechner & Hasani, 2020; Schirmer et al., 2022). The Neural Flows framework avoids numerical ODE solvers (Biloš et al., 2021), and Neural CDE describes observed paths with controlled differential equations (Kidger et al., 2020). IP-Net and mTAN aggregate irregular observations through interpolation and time-aware attention, respectively (Shukla & Marlin, 2019; 2021). Set-function, pretraining, and Transformer approaches include SeFT, PrimeNet, Warpformer, and ContiFormer (Horn et al., 2020; Chowdhury et al., 2023; Zhang et al., 2023; Chen et al., 2023). Much of this work focuses on state evolution and observation aggregation within individual series.

Recent studies further organize asynchronous observations through structured representations. Raindrop propagates information over learned sensor dependencies (Zhang et al., 2021), while GraFITi formulates forecasting as edge-weight estimation on a sparse bipartite graph (Yalavarthi et al., 2024). HyperIMTS represents observations and their dependencies with a hypergraph (Li et al., 2025), and GNeuralFlow models systemic interactions through graph neural flows (Mercatali et al., 2024). ASTGI constructs an adaptive causal graph among observation points (Liu et al., 2026a), whereas KAFNet retains canonical pre-alignment (CPA) and compresses aligned sequences through temporal kernel aggregation (Zhou et al., 2026). ReIMTS recursively divides each sample into shorter time periods while preserving the original timestamps, then fuses representations across scales (Li et al., 2026). Patch-based approaches provide another way to organize irregular observations into local temporal units for dependency modeling.

2.2 Patch-based Irregular Multivariate Time Series Forecasting

In regularly sampled forecasting, patch representations shorten sequences through local aggregation and provide a common modeling unit (Nie et al., 2022; Zhang & Yan, 2023; Huang et al., 2024). For IMTSF, T-PatchGNN divides each univariate series into patches with fixed temporal spans but varying observation counts, and models variable relationships with time-adaptive graphs (Zhang et al., 2024). TimeCHEAT models channel relationships over locally organized irregular observations (Liu et al., 2025). Hi-Patch captures multiscale temporal dependencies through hierarchical patching (Luo et al., 2025). APN learns adaptive soft temporal windows and uses time-aware weighted aggregation to alleviate uneven information density from fixed partitioning (Liu et al., 2026b). TFMixer aggregates local patches with learnable queries and models global periodicity with frequency-domain representations (Qiu et al., 2026).

Overall, the patch-based methods above use irregular sampling information mainly for patch construction, scale selection, or local aggregation (Zhang et al., 2024; Luo et al., 2025; Liu et al., 2026b; Qiu et al., 2026), and rarely construct a separate representation of observation morphology from the temporal distribution and missingness pattern within each patch. MorITS models this morphology to guide patch semantic extraction and cross-patch interaction.

3 Preliminary

3.1 Problem Definition

An IMTS $\mathcal{O}$ comprises N univariate sequences, $\mathcal{O} = \{\mathbf{o}^n_{1:L_n}\}_{n=1}^N$. Each $\mathbf{o}^n_{1:L_n}$ contains L_n observations (t^n_i, x^n_i), where t^n_i and x^n_i respectively denote the timestamp and observed value of variable n's i-th observation. Observation intervals are irregular within variables, and timestamps are generally unaligned across variables. Given historical observations $\mathcal{O}$ and query times $\mathcal{Q} = \{[q^n_j]_{j=1}^{Q_n}\}_{n=1}^N$, Q_n is the query count for variable n and q^n_j its j-th query time. We learn $\mathcal{F}(\cdot)$ to predict $\widehat{\mathcal{V}} = \{[\widehat{v}^n_j]_{j=1}^{Q_n}\}_{n=1}^N$, where $\widehat{v}^n_j$ is the forecast for variable n at q^n_j. Thus, $\widehat{\mathcal{V}} = \mathcal{F}(\mathcal{O}, \mathcal{Q})$.

3.2 Canonical Pre-Alignment Representation and Patching for IMTS

Canonical Pre-Alignment. MorITS uses CPA to align asynchronous observations as $(\mathcal{T}, \mathcal{X}, \mathcal{M})$. The shared timeline $\mathcal{T} = [t_l]_{l=1}^L = \bigcup_{n=1}^N [t^n_i]_{i=1}^{L_n} \in \mathbb{R}^L$ contains the unique historical timestamps in chronological order, with L timestamps and t_l denoting its l-th entry. The value matrix $\mathcal{X} =$

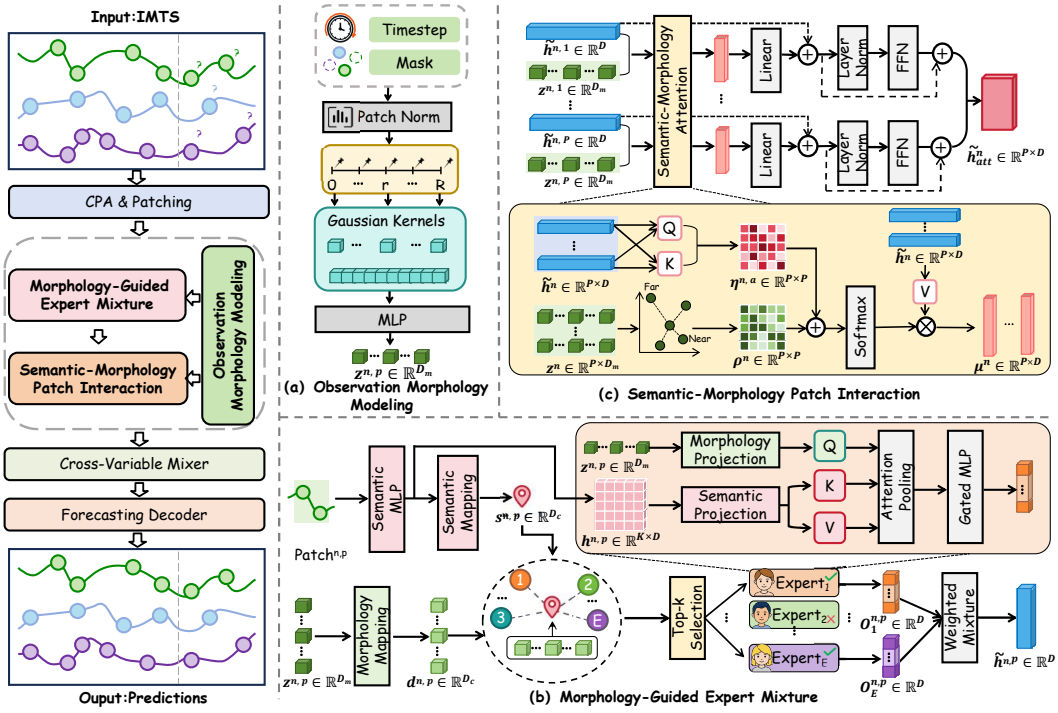


Figure 3: Overview of MorITS. After CPA and patching, (a) OMM models value-independent observation morphology from timestamps and masks, (b) MGEM uses morphology-conditioned routing and aggregation to extract patch semantics, and (c) SMPI combines semantic similarity and morphology relations for cross-patch interaction. The cross-variable mixer and decoder produce forecasts.

$[[\widetilde{x}_l^n]_{n=1}^N]_{l=1}^L \in \mathbb{R}^{L\times N}$ satisfies $\widetilde{x}_l^n = x_i^n$ when variable n is observed at $t_l = t_i^n$, and $\widetilde{x}_l^n = 0$ otherwise. The mask $\mathcal{M} = [[m_l^n]_{n=1}^N]_{l=1}^L \in \{0,1\}^{L\times N}$ uses $m_l^n = 1$ for valid observations of variable n at t_l, and $m_l^n = 0$ otherwise (Zhang et al., 2024; Che et al., 2018).

Patching. Along the shared timeline, MorITS partitions the aligned sequence into $P = \lceil L/K \rceil$ non-overlapping patches, each containing up to K consecutive timestamps with their values and masks. The final patch is padded to length K if needed (Appendix B.1). Variables share patch boundaries; irregular intervals allow different physical patch durations.

4 Methodology

4.1 Framework Overview

Figure 3 illustrates the overall architecture of MorITS. After CPA and patching, MorITS uses patch-level observation morphology to guide patch semantic extraction and cross-patch interaction. OMM constructs value-independent morphology representations from timestamps and observation masks. Different observation morphologies may require different ways of aggregating observations. MGEM therefore uses multiple experts to learn complementary responses and extracts patch semantics through morphology-conditioned routing and aggregation. SMPI incorporates inter-patch morphology relations into the semantic-similarity scores used to compute interaction weights, so information propagation accounts for both semantic content and observation structure. Finally, the Cross-Variable Mixer integrates information across variables, and the forecasting decoder combines historical representations with future query times to generate predictions.

4.2 Observation Morphology Modeling

To explicitly model patch-level observation morphology jointly formed by temporal distributions and missingness patterns, OMM takes timestamps and observation masks as input. For patch p,

t_k^p denotes the timestamp at its k-th position, and $m_k^{n,p} \in \{0,1\}$ indicates whether variable n is observed at that position. Because patches may span different physical durations, we convert timestamps to positions relative to each patch as $\tau_k^p = (t_k^p - t_1^p)/\max(t_K^p - t_1^p, \varepsilon)$, so that observation locations are described on a common relative time scale. Here, ε is a small positive constant that prevents division by zero. To jointly characterize observation density, temporal distribution, and local clustering within a patch, we use localized Gaussian basis functions to compute weighted statistics of the temporal positions of valid observations. Specifically, we uniformly divide the relative time axis into R intervals and use the resulting $R+1$ boundaries as Gaussian-basis centers. The centers are $c_r = r/R$ for $r = 0, \ldots, R$, with bandwidth $\sigma_R = 1/R$. The response of variable n in patch p to the r-th Gaussian basis is

$$g_r^{n,p} = \frac{1}{K} \sum_{k=1}^{K} m_k^{n,p} \exp\left[-\frac{(\tau_k^p - c_r)^2}{2\sigma_R^2} \right], \qquad r = 0, \ldots, R. \tag{1}$$

The mask $m_k^{n,p}$ excludes missing positions, and the Gaussian weight reflects the temporal distance between valid observations and center c_r; the fixed patch length K serves as the normalization denominator, preserving the relative observation density within the patch. The responses at all centers form the Gaussian observation field $\mathbf{g}^{n,p} = [g_0^{n,p}, g_1^{n,p}, \ldots, g_R^{n,p}] \in \mathbb{R}^{R+1}$. Finally, a multilayer perceptron maps the Gaussian observation field $\mathbf{g}^{n,p}$ to a morphology representation $\mathbf{z}^{n,p} \in \mathbb{R}^{D_m}$ for subsequent patch semantic extraction and cross-patch interaction.

4.3 Morphology-Guided Expert Mixture

Patch morphology combines observation density, temporal distribution, and clustering patterns. MGEM uses independently parameterized experts to disentangle them and learn complementary responses, extracting patch semantics through morphology-conditioned routing and aggregation.

Morphology-Conditioned Expert Routing. For variable n in patch p, we encode timestamp t_k^p with a continuous time embedding $\boldsymbol{\phi}(t_k^p)$ (Zhang et al., 2024). Concatenating it with the observed value, a semantic MLP produces the observation-level semantic vector $\mathbf{h}_k^{n,p} = \mathrm{MLP}_{\mathrm{sem}}([x_k^{n,p}; \boldsymbol{\phi}(t_k^p)]) \in \mathbb{R}^D$. The vectors within the patch are denoted by $\mathbf{h}^{n,p}$; the observation mask excludes missing positions, and masked mean pooling yields the semantic summary $\overline{\mathbf{h}}^{n,p}$. In the shared D_c-dimensional expert-matching space, the semantic summary and morphology representation are mapped to the semantic routing vector $\mathbf{s}^{n,p} = \Gamma_s(\overline{\mathbf{h}}^{n,p})$ and positive morphology metric weights $\mathbf{d}^{n,p} = \Gamma_m(\mathbf{z}^{n,p})$, respectively.

Each expert e has a learnable prototype $\mathbf{p}_e \in \mathbb{R}^{D_c}$ as a reference in the routing space. Because the importance of semantic dimensions for matching may vary with observation morphology, $\mathbf{d}^{n,p}$ weights the coordinate-wise differences between $\mathbf{s}^{n,p}$ and $\mathbf{p}_e$. The resulting score depends on both observation content and structure:

$$\ell_e^{n,p} = -\frac{(\mathbf{s}^{n,p} - \mathbf{p}_e)^\top \mathrm{Diag}(\mathbf{d}^{n,p})(\mathbf{s}^{n,p} - \mathbf{p}_e)}{\beta}, \qquad e = 1, \ldots, E, \tag{2}$$

where $\mathrm{Diag}(\mathbf{d}^{n,p})$ places the morphology weights along the diagonal, E is the number of experts, and β is a learned positive temperature. Following sparse mixture-of-experts routing (Shazeer et al., 2017; Fedus et al., 2022), the K_{top} highest-scoring experts form the selected set $\mathcal{E}^{n,p}$; their Softmax-normalized scores define the routing weights $\pi_e^{n,p}$.

Morphology-Guided Semantic Aggregation. Within the selected experts, morphology guides the aggregation of within-patch semantic vectors. For expert e, $\mathbf{z}^{n,p}$ serves as the Query, while $\mathbf{h}^{n,p}$ provides the Keys and Values. Masked attention and a gated MLP produce the expert output:

$$\mathbf{o}_e^{n,p} = \mathrm{MLP}_{\mathrm{gate}}(\mathrm{Attention}_{\mathrm{mask}}(\mathbf{z}^{n,p}, \mathbf{h}^{n,p}, \mathbf{h}^{n,p})) \in \mathbb{R}^D. \tag{3}$$

Attention applies learned Query, Key, and Value projections and masks missing positions. The selected expert outputs are combined using the routing weights:

$$\widetilde{\mathbf{h}}^{n,p} = \mathrm{LN}\left(\sum_{e \in \mathcal{E}^{n,p}} \pi_e^{n,p} \, \mathrm{LN}_e(\mathbf{o}_e^{n,p}) \right) \in \mathbb{R}^D. \tag{4}$$

The resulting $\widetilde{\mathbf{h}}^{n,p}$ represents patch semantics for subsequent cross-patch interaction.

4.4 SEMANTIC–MORPHOLOGY PATCH INTERACTION

Using only semantic similarity to compute cross-patch interaction weights makes it difficult to explicitly distinguish patches with similar semantics but different observation morphologies. SMPI therefore introduces morphology relations into the attention scores so that differences in observation structure affect information propagation. For variable n, let $\widetilde{\mathbf{h}}^n = [\widetilde{\mathbf{h}}^{n,1}, \dots, \widetilde{\mathbf{h}}^{n,P}]^\top$ and $\mathbf{z}^n = [\mathbf{z}^{n,1}, \dots, \mathbf{z}^{n,P}]^\top$ denote the semantic and morphology representations of all patches, respectively. For attention head a, a scaled dot product measures semantic matching (Vaswani et al., 2017), while negative squared Euclidean distance in a learnable morphology projection space measures structural proximity between each pair of patches:

$$\boldsymbol{\eta}^{n,a} = \frac{\left(\widetilde{\mathbf{h}}^n \omega^a_{\text{query}}\right)\left(\widetilde{\mathbf{h}}^n \omega^a_{\text{key}}\right)^\top}{\sqrt{D/A}} \in \mathbb{R}^{P\times P}, \qquad \boldsymbol{\rho}^n = -\frac{\text{Dist}^2(\mathbf{z}^n \omega_{\text{mor}})}{\sqrt{D_\rho}} \in \mathbb{R}^{P\times P}. \tag{5}$$

Here A is the number of attention heads; ω^a_{query} and ω^a_{key} are head a's Query and Key projections; ω_{mor} is the morphology projection shared by all heads; D_ρ is the morphology-relation space dimension; and $\text{Dist}^2(\cdot)$ computes all pairwise squared Euclidean distances between projected morphology representations. The negative squared distance assigns higher morphology relation scores to patch pairs that are closer in the projected space; the shared $\boldsymbol{\rho}^n$ provides the same structural reference to all semantic attention heads, whose semantic projections are learned separately.

Adding morphology scores to semantic scores allows structural proximity to directly modulate attention weights. Morphology relations help determine which patches contribute and by how much; the aggregated Value vectors are projected from patch semantics:

$$\boldsymbol{\mu}^{n,a} = \text{Softmax}_{\text{mask}}(\boldsymbol{\eta}^{n,a} + \boldsymbol{\rho}^n)\left(\widetilde{\mathbf{h}}^n \omega^a_{\text{value}}\right). \tag{6}$$

Here the patch-availability mask excludes invalid positions, and ω^a_{value} is head a's Value projection. To preserve the original patch semantics while incorporating cross-patch information, SMPI concatenates the head outputs along the feature dimension to form $\boldsymbol{\mu}^n$, then applies an output projection and two residual updates to obtain the updated patch representations:

$$\mathbf{h}^n_{\text{att}} = \widetilde{\mathbf{h}}^n + \boldsymbol{\mu}^n \omega_{\text{att}}, \qquad \widetilde{\mathbf{h}}^n_{\text{att}} = \mathbf{h}^n_{\text{att}} + \text{FFN}(\text{LN}(\mathbf{h}^n_{\text{att}})) \in \mathbb{R}^{P\times D}. \tag{7}$$

Here ω_{att} is the output projection; the updated $\widetilde{\mathbf{h}}^n_{\text{att}}$ is passed to the Cross-Variable Mixer.

4.5 CROSS-VARIABLE MIXER AND FORECASTING DECODER

Cross-Variable Mixer. SMPI models cross-patch relations within each variable. Because variables share patch boundaries, the Cross-Variable Mixer can integrate information across variables at the same patch position. Let $\widetilde{\mathbf{H}}_{\text{att}} = [\widetilde{\mathbf{h}}^1_{\text{att}}; \dots; \widetilde{\mathbf{h}}^N_{\text{att}}] \in \mathbb{R}^{N\times P\times D}$ denote the patch representations of all variables. To retain each variable's original representation during fusion, the mixer applies residual linear mixing along the variable dimension:

$$\mathbf{H}_{\text{var}} = \widetilde{\mathbf{H}}_{\text{att}} + \omega_{\text{var}} \widetilde{\mathbf{H}}_{\text{att}} \in \mathbb{R}^{N\times P\times D}. \tag{8}$$

Here $\omega_{\text{var}} \in \mathbb{R}^{N\times N}$ is a linear map along the variable dimension shared across patches and feature positions; an availability mask excludes invalid positions. SMPI and the Cross-Variable Mixer form one interaction layer. MorITS stacks G interaction layers, alternating cross-patch propagation and cross-variable fusion. The final output is denoted by $\mathbf{H}^{(G)}_{\text{var}} \in \mathbb{R}^{N\times P\times D}$.

Forecasting Decoder. To summarize each variable's final patch representations for forecasting, the decoder applies masked mean pooling over valid patches:

$$\mathbf{H}_{\text{hist}} = \text{MaskedMean}^P_{p=1}\left(\mathbf{H}^{(G)}_{\text{var}}\right) \in \mathbb{R}^{N\times D}. \tag{9}$$

The n-th row of $\mathbf{H}_{\text{hist}}$, denoted by $\mathbf{h}^n_{\text{hist}}$, is the historical representation obtained by pooling the valid patch representations of variable n. For future query times $\{q^n_j\}_{j=1}^{Q_n}$, the decoder uses learnable time embeddings with the same affine–periodic form as the historical time embedding. The historical representation and query-time embedding jointly determine the forecast:

$$\widehat{\mathbf{v}}^n = \left[\widehat{v}^n_j\right]_{j=1}^{Q_n} = \left[\mathrm{MLP}_{\text{dec}}\left(\mathbf{h}^n_{\text{hist}}, \boldsymbol{\phi}_q(q^n_j)\right)\right]_{j=1}^{Q_n}. \tag{10}$$

Here $\boldsymbol{\phi}_q(q^n_j)$ is the embedding of query time q^n_j. $\mathrm{MLP}_{\text{dec}}(\cdot,\cdot)$ maps the historical representation and query-time embedding into the same feature space, fuses them, and produces the forecast. The predictions of all variables form $\widehat{\mathcal{V}} = \{\widehat{\mathbf{v}}^n\}_{n=1}^N$.

To optimize forecasting accuracy while discouraging expert routing from concentrating on a few experts, MorITS minimizes a joint training objective that combines mean squared error over the observed forecasting targets across all variables with an auxiliary expert load-balancing loss:

$$\mathcal{L} = \mathcal{L}_{\text{MSE}} + \lambda \mathcal{L}_{\text{bal}}, \qquad \mathcal{L}_{\text{bal}} = E \sum_{e=1}^{E} f_e P_e. \tag{11}$$

Here E is the number of experts, f_e is the fraction of instances assigned to expert e, P_e is its mean routing probability before Top-K_{top} selection, and λ is the load-balancing coefficient.

5 Experiments

5.1 Experimental Settings

Datasets. To ensure a comprehensive and fair comparison, we conduct experiments on five benchmark datasets for IMTSF: Human Activity (Vidulin et al., 2010), PhysioNet 2012 (Silva et al., 2012), USHCN (Menne et al., 2016), MIMIC-III (Johnson et al., 2016), and MIMIC-IV (Johnson et al., 2023). These datasets span healthcare, biomechanics, and climate science. Further dataset details are provided in Appendix B.2.

Baselines. To evaluate the performance of MorITS, we compare it with 13 baseline models. GRU-D (Che et al., 2018), mTAN (Shukla & Marlin, 2021), Raindrop (Zhang et al., 2021), and Warpformer (Zhang et al., 2023) are general-purpose models for irregular time series, while NeuralFlows (Biloš et al., 2021), CRU (Schirmer et al., 2022), GNeuralFlow (Mercatali et al., 2024), t-PatchGNN (Zhang et al., 2024), GraFITi (Yalavarthi et al., 2024), Hi-Patch (Luo et al., 2025), APN (Liu et al., 2026b), ASTGI (Liu et al., 2026a), and TFMixer (Qiu et al., 2026) are designed for IMTSF. Further details of these methods are provided in Appendix B.3.

Implementation Details. All experiments are implemented in PyTorch 2.13.0+cu130 using the PyOmniTS framework and conducted on a server equipped with four NVIDIA GeForce RTX 4090 GPUs. MorITS is trained with the Adam optimizer (Kingma & Ba, 2014). The maximum number of training epochs is set to 200, and training terminates early if the validation loss does not improve for 10 consecutive epochs. All methods use the same data splits and target filtering. To ensure a fair comparison, we primarily adopt the hyperparameter settings reported in the original papers for the baseline models and further search and tune key hyperparameters on the validation set so that all models achieve competitive performance. Each experiment is independently repeated with five random seeds from 2024 to 2028, and we report the mean and standard deviation of mean squared error (MSE) and mean absolute error (MAE) over observed forecasting targets.

5.2 Main Results

Table 1 presents the forecasting results of MorITS and the baseline models, showing that MorITS achieves state-of-the-art forecasting performance on all datasets. Compared with TFMixer, MorITS reduces average MSE and MAE by **10.64%** and **8.19%**, respectively, across the five datasets, demonstrating a consistent and substantial overall advantage. Furthermore, MorITS maintains a performance advantage across healthcare, biomechanics, and climate science under different irregular sampling patterns and observation conditions.

This improvement primarily stems from morphology-aware modeling: MGEM uses morphology-conditioned expert routing and semantic aggregation to adapt patch semantic extraction to each

Table 1: IMTSF results on five datasets (mean ± standard deviation over five runs). Best and second-best values are **bold** and underlined, respectively; lower is better.

Method	Human Activity		USHCN		PhysioNet 2012		MIMIC-III		MIMIC-IV	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
NeuralFlows	$0.1685_{\pm 0.0072}$	$0.3113_{\pm 0.0083}$	$0.4168_{\pm 0.0429}$	$0.3353_{\pm 0.0075}$	$0.4047_{\pm 0.0031}$	$0.4459_{\pm 0.0028}$	$0.6592_{\pm 0.0046}$	$0.5314_{\pm 0.0027}$	$0.4198_{\pm 0.0037}$	$0.4529_{\pm 0.0018}$
CRU	$0.1422_{\pm 0.0117}$	$0.2670_{\pm 0.0178}$	$0.2423_{\pm 0.0127}$	$0.3189_{\pm 0.0118}$	$0.6195_{\pm 0.0026}$	$0.5794_{\pm 0.0010}$	$0.6575_{\pm 0.0043}$	$0.5255_{\pm 0.0025}$	$0.4099_{\pm 0.0058}$	$0.4460_{\pm 0.0036}$
mTAN	$0.0856_{\pm 0.0027}$	$0.2082_{\pm 0.0038}$	$0.5891_{\pm 0.0992}$	$0.5163_{\pm 0.0329}$	$0.3720_{\pm 0.0036}$	$0.4219_{\pm 0.0033}$	$0.9181_{\pm 0.0462}$	$0.6497_{\pm 0.0168}$	$0.5129_{\pm 0.0255}$	$0.5034_{\pm 0.0132}$
GNeuralFlow	$0.3488_{\pm 0.0849}$	$0.4300_{\pm 0.0446}$	$0.2621_{\pm 0.0401}$	$0.3434_{\pm 0.0228}$	$0.8208_{\pm 0.1051}$	$0.6705_{\pm 0.0422}$	$0.9783_{\pm 0.0777}$	$0.6629_{\pm 0.0338}$	$0.5962_{\pm 0.0055}$	$0.5447_{\pm 0.0027}$
GRU-D	$0.1787_{\pm 0.0381}$	$0.3173_{\pm 0.0317}$	$0.2690_{\pm 0.0100}$	$0.2915_{\pm 0.0038}$	$0.3418_{\pm 0.0024}$	$0.3997_{\pm 0.0011}$	$0.5177_{\pm 0.0095}$	$0.4518_{\pm 0.0047}$	$0.5899_{\pm 0.0405}$	$0.5399_{\pm 0.0266}$
Raindrop	$0.0914_{\pm 0.0082}$	$0.2119_{\pm 0.0124}$	$0.2447_{\pm 0.0294}$	$0.3367_{\pm 0.0314}$	$0.3442_{\pm 0.0010}$	$0.4015_{\pm 0.0011}$	$0.5142_{\pm 0.0036}$	$0.4452_{\pm 0.0036}$	$0.4229_{\pm 0.0351}$	$0.4475_{\pm 0.0176}$
Warpformer	$0.0449_{\pm 0.0009}$	$0.1229_{\pm 0.0014}$	$0.1802_{\pm 0.0035}$	$0.2701_{\pm 0.0049}$	$0.3077_{\pm 0.0038}$	$0.3641_{\pm 0.0028}$	$0.4514_{\pm 0.0025}$	$0.3981_{\pm 0.0018}$	$0.2433_{\pm 0.0020}$	$0.2970_{\pm 0.0030}$
t-PatchGNN	$0.0454_{\pm 0.0028}$	$0.1281_{\pm 0.0074}$	$0.2132_{\pm 0.0247}$	$0.2854_{\pm 0.0214}$	$0.3091_{\pm 0.0015}$	$0.3666_{\pm 0.0019}$	$0.4641_{\pm 0.0066}$	$0.4048_{\pm 0.0037}$	$0.2504_{\pm 0.0016}$	$0.2955_{\pm 0.0021}$
GraFITi	$0.0436_{\pm 0.0003}$	$0.1184_{\pm 0.0016}$	$0.1852_{\pm 0.0021}$	$0.2748_{\pm 0.0079}$	$0.3057_{\pm 0.0007}$	$0.3687_{\pm 0.0006}$	$\underline{0.4175_{\pm 0.0072}}$	$\underline{0.3789_{\pm 0.0023}}$	$0.2375_{\pm 0.0009}$	$0.3056_{\pm 0.0020}$
Hi-Patch	$0.0481_{\pm 0.0006}$	$0.1256_{\pm 0.0014}$	$0.2331_{\pm 0.0121}$	$0.2847_{\pm 0.0086}$	$0.3172_{\pm 0.0030}$	$0.3751_{\pm 0.0018}$	$0.4818_{\pm 0.0055}$	$0.4161_{\pm 0.0013}$	$0.2443_{\pm 0.0049}$	$0.2950_{\pm 0.0051}$
APN	$0.0441_{\pm 0.0006}$	$0.1230_{\pm 0.0019}$	$0.1935_{\pm 0.0283}$	$0.2811_{\pm 0.0058}$	$0.3128_{\pm 0.0006}$	$0.3658_{\pm 0.0009}$	$0.4739_{\pm 0.0069}$	$0.4137_{\pm 0.0027}$	$0.2444_{\pm 0.0055}$	$0.3019_{\pm 0.0015}$
ASTGI	$\underline{0.0418_{\pm 0.0004}}$	$0.1195_{\pm 0.0011}$	$0.2184_{\pm 0.0287}$	$0.2828_{\pm 0.0082}$	$\underline{0.3029_{\pm 0.0011}}$	$\underline{0.3638_{\pm 0.0012}}$	$0.4310_{\pm 0.0033}$	$0.3899_{\pm 0.0010}$	$\underline{0.2297_{\pm 0.0025}}$	$\underline{0.2930_{\pm 0.0018}}$
TFMixer	$0.0431_{\pm 0.0004}$	$\underline{0.1169_{\pm 0.0011}}$	$\underline{0.1780_{\pm 0.0190}}$	$\underline{0.2642_{\pm 0.0041}}$	$0.3079_{\pm 0.0035}$	$0.3707_{\pm 0.0032}$	$0.4635_{\pm 0.0026}$	$0.4126_{\pm 0.0023}$	$0.2455_{\pm 0.0021}$	$0.3134_{\pm 0.0024}$
MorITS (Ours)	$\mathbf{0.0396}_{\pm 0.0003}$	$\mathbf{0.1126}_{\pm 0.0006}$	$\mathbf{0.1595}_{\pm 0.0024}$	$\mathbf{0.2590}_{\pm 0.0067}$	$\mathbf{0.2984}_{\pm 0.0005}$	$\mathbf{0.3587}_{\pm 0.0009}$	$\mathbf{0.4085}_{\pm 0.0036}$	$\mathbf{0.3679}_{\pm 0.0015}$	$\mathbf{0.1971}_{\pm 0.0021}$	$\mathbf{0.2468}_{\pm 0.0009}$

Table 2: Ablation studies for MorITS in terms of the lowest MSE and MAE highlighted in bold.

Variant	Human Activity		USHCN		PhysioNet 2012		MIMIC-III		MIMIC-IV	
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
w/o OMM	$0.0415_{\pm 0.0039}$	$0.1177_{\pm 0.0063}$	$0.1656_{\pm 0.0083}$	$0.2686_{\pm 0.0067}$	$0.3102_{\pm 0.0010}$	$0.3722_{\pm 0.0007}$	$0.4283_{\pm 0.0035}$	$0.3846_{\pm 0.0019}$	$0.2098_{\pm 0.0027}$	$0.2576_{\pm 0.0079}$
w/o MGEM	$0.0410_{\pm 0.0017}$	$0.1163_{\pm 0.0019}$	$0.1652_{\pm 0.0055}$	$0.2677_{\pm 0.0079}$	$0.3126_{\pm 0.0003}$	$0.3750_{\pm 0.0007}$	$0.4222_{\pm 0.0053}$	$0.3811_{\pm 0.0020}$	$0.2087_{\pm 0.0018}$	$0.2563_{\pm 0.0046}$
w/o SMPI	$0.0409_{\pm 0.0002}$	$0.1161_{\pm 0.0003}$	$0.1668_{\pm 0.0023}$	$0.2704_{\pm 0.0045}$	$0.3078_{\pm 0.0009}$	$0.3711_{\pm 0.0021}$	$0.4252_{\pm 0.0085}$	$0.3821_{\pm 0.0046}$	$0.2070_{\pm 0.0017}$	$0.2539_{\pm 0.0008}$
w/o MW	$0.0407_{\pm 0.0002}$	$0.1154_{\pm 0.0009}$	$0.1623_{\pm 0.0027}$	$0.2644_{\pm 0.0078}$	$0.3084_{\pm 0.0002}$	$0.3695_{\pm 0.0009}$	$0.4152_{\pm 0.0042}$	$0.3754_{\pm 0.0013}$	$0.2068_{\pm 0.0044}$	$0.2533_{\pm 0.0064}$
w/o MGA	$0.0403_{\pm 0.0062}$	$0.1149_{\pm 0.0094}$	$0.1639_{\pm 0.0082}$	$0.2656_{\pm 0.0072}$	$0.3046_{\pm 0.0030}$	$0.3673_{\pm 0.0022}$	$0.4199_{\pm 0.0050}$	$0.3796_{\pm 0.0010}$	$0.2051_{\pm 0.0015}$	$0.2528_{\pm 0.0061}$
w/o MR	$0.0404_{\pm 0.0003}$	$0.1151_{\pm 0.0009}$	$0.1647_{\pm 0.0030}$	$0.2665_{\pm 0.0086}$	$0.3035_{\pm 0.0002}$	$0.3662_{\pm 0.0009}$	$0.4212_{\pm 0.0032}$	$0.3780_{\pm 0.0016}$	$0.2051_{\pm 0.0034}$	$0.2513_{\pm 0.0041}$
w/o $\mathcal{L}_{\text{bal}}$	$0.0401_{\pm 0.0001}$	$0.1143_{\pm 0.0006}$	$0.1617_{\pm 0.0027}$	$0.2638_{\pm 0.0057}$	$0.3017_{\pm 0.0004}$	$0.3641_{\pm 0.0007}$	$0.4161_{\pm 0.0037}$	$0.3760_{\pm 0.0017}$	$0.2030_{\pm 0.0029}$	$0.2491_{\pm 0.0008}$
MorITS (Ours)	$\mathbf{0.0396}_{\pm 0.0003}$	$\mathbf{0.1126}_{\pm 0.0006}$	$\mathbf{0.1595}_{\pm 0.0024}$	$\mathbf{0.2590}_{\pm 0.0067}$	$\mathbf{0.2984}_{\pm 0.0005}$	$\mathbf{0.3587}_{\pm 0.0009}$	$\mathbf{0.4085}_{\pm 0.0036}$	$\mathbf{0.3679}_{\pm 0.0015}$	$\mathbf{0.1971}_{\pm 0.0021}$	$\mathbf{0.2468}_{\pm 0.0009}$

patch's observation structure; SMPI combines semantic similarity and inter-patch morphology relations to compute interaction weights, allowing structural differences to influence cross-patch information propagation. These results demonstrate the effectiveness of explicitly modeling observation morphology for IMTS forecasting.

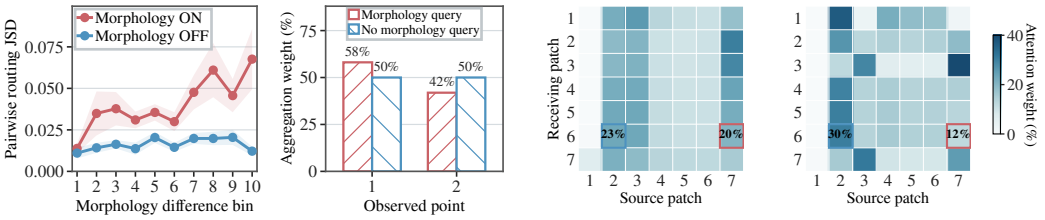
5.3 Ablation Studies

We evaluate seven ablation variants (Table 2). w/o OMM, w/o MGEM, and w/o SMPI remove the respective modules. w/o MW (Metric Weight) omits morphology weighting in expert matching (Eq. (2)); w/o MGA (Morphology-Guided Attention) omits morphology-query attention (Eq. (3)) and mean-pools valid semantic vectors; w/o MR (Morphology Relation) omits the morphology score (Eq. (5)); and w/o $\mathcal{L}_{\text{bal}}$ omits the balancing loss (Eq. (11)). All other settings are unchanged.

The full model achieves the best MSE and MAE on all five datasets. w/o OMM shows the largest overall degradation, underscoring the structural information that morphology modeling supplies to subsequent modules. w/o MGEM and w/o SMPI raise both errors on every dataset, supporting morphology-conditioned patch semantic extraction and morphology-aware cross-patch interaction. w/o MW and w/o MGA also impair performance, indicating that morphology helps expert matching and within-patch aggregation. w/o MR performs worse despite retaining semantic similarity, showing the value of morphology relations. Removing $\mathcal{L}_{\text{bal}}$ causes a smaller but consistent decline.

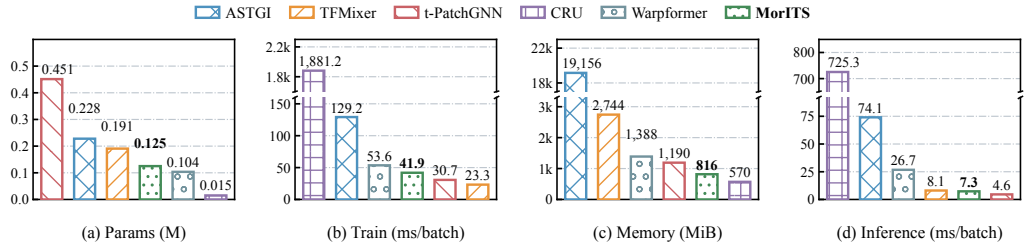
5.4 Mechanism Analysis

On PhysioNet 2012, we examine morphology's effects on expert routing, within-Patch aggregation, and cross-Patch interaction (Figure 4). In panel (a), we group semantically similar Patch pairs into ten bins by morphology difference. Across the bins, their expert-assignment difference generally grows with morphology modulation but changes less without it, showing that routing distinguishes Patches with different observation structures. In panel (b), we fix the Patch and expert and compare aggregation with and without the morphology query. The two observations' weights shift from 50%/50% to 58%/42%, showing that morphology adjusts their contributions to Patch semantics. In panels (c,d), with semantic representations fixed, we compare semantic-only attention with attention incorporating morphology relations. The latter reduces the weight on a morphologically different Patch from 20% to 12% and raises that on a similar Patch from 23% to 30%, showing that observation structure helps regulate cross-Patch interaction.



(a) Expert routing (b) Within-patch aggregation (c) Semantic only (d) Semantic + morphology

Figure 4: Morphology effects on routing, aggregation, and interaction in MorITS.



(a) Params (M) (b) Train (ms/batch) (c) Memory (MiB) (d) Inference (ms/batch)

Figure 5: Efficiency comparison of MorITS and baselines on USHCN.

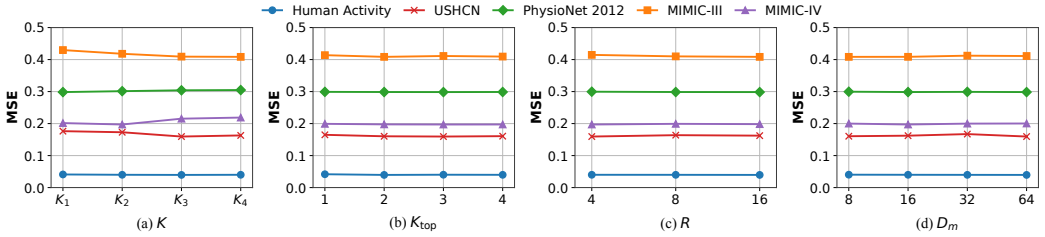


Figure 6: One-at-a-time test-MSE sensitivity of MorITS to K, K_{top}, R, and D_m.

5.5 Efficiency Analysis

On USHCN, we compare parameter count, GPU memory, and per-batch training and inference times using one RTX 4090 and a batch size of 32 (Figure 5). MorITS performs competitively across all four metrics without leading in any single one. CRU is much slower in training and inference, TFMixer uses more memory, t-PatchGNN has more parameters, and ASTGI incurs higher memory and runtime costs. MorITS thus strikes a balance across the four metrics.

5.6 Parameter Sensitivity

Figure 6 plots test MSE against aligned observation positions per patch K, selected experts K_{top}, localized Gaussian-field resolution R, and morphology dimension D_m. Figure 6(a) examines K. For each dataset, K_1–K_4 denote four values in ascending order: $(100, 200, 300, 400)$ for Human Activity, $(5, 10, 15, 20)$ for USHCN, $(3, 6, 9, 12)$ for PhysioNet 2012, $(6, 12, 18, 24)$ for MIMIC-III, and $(24, 48, 72, 96)$ for MIMIC-IV. Overall, test MSE varies moderately across the tested settings, indicating stable forecasting performance. At $K_{\text{top}} = 1$, MSE is highest on all five datasets. This suggests that complementary responses from multiple experts help accommodate different combinations of morphology factors.

6 Conclusion

This paper presents MorITS, a morphology-aware framework for IMTSF that explicitly models patch-level observation morphology formed by temporal distributions and missingness patterns. Building on this representation, the framework uses morphology conditioning to extract patch semantics and combines semantic similarity with morphology relations to guide cross-patch interaction. MorITS achieves the best MSE and MAE on all five real-world datasets.

REFERENCES

Inci M Baytas, Cao Xiao, Xi Zhang, Fei Wang, Anil K Jain, and Jiayu Zhou. Patient subtyping via time-aware lstm networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 65–74, 2017.

Marin Biloš, Johanna Sommer, Syama Sundar Rangapuram, Tim Januschowski, and Stephan Günnemann. Neural flows: Efficient alternative to neural odes. *Advances in neural information processing systems*, 34:21325–21337, 2021.

Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. Brits: Bidirectional recurrent imputation for time series. *Advances in neural information processing systems*, 31, 2018.

Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1):6085, 2018.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

Yuqi Chen, Kan Ren, Yansen Wang, Yuchen Fang, Weiwei Sun, and Dongsheng Li. Contiformer: Continuous-time transformer for irregular time series modeling. *Advances in Neural Information Processing Systems*, 36:47143–47175, 2023.

Ranak Roy Chowdhury, Jiacheng Li, Xiyuan Zhang, Dezhi Hong, Rajesh K Gupta, and Jingbo Shang. Primenet: Pre-training for irregular multivariate time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 7184–7192, 2023.

Edward De Brouwer, Jaak Simm, Adam Arany, and Yves Moreau. Gru-ode-bayes: Continuous modeling of sporadically-observed time series. *Advances in neural information processing systems*, 32, 2019.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Max Horn, Michael Moor, Christian Bock, Bastian Rieck, and Karsten Borgwardt. Set functions for time series. In *International Conference on Machine Learning*, pp. 4353–4363. PMLR, 2020.

Qihe Huang, Lei Shen, Ruixin Zhang, Jiahuan Cheng, Shouhong Ding, Zhengyang Zhou, and Yang Wang. Hdmixer: Hierarchical dependency with extendable patch for multivariate time series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 12608–12616, 2024.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):160035, 2016.

Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1, 2023.

Patrick Kidger, James Morrill, James Foster, and Terry Lyons. Neural controlled differential equations for irregular time series. *Advances in neural information processing systems*, 33:6696–6707, 2020.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Mathias Lechner and Ramin Hasani. Learning long-term dependencies in irregularly-sampled time series. *arXiv preprint arXiv:2006.04418*, 2020.

Boyuan Li, Yicheng Luo, Zhen Liu, Junhao Zheng, Jianming Lv, and Qianli Ma. Hyperimts: Hypergraph neural network for irregular multivariate time series forecasting. *arXiv preprint arXiv:2505.17431*, 2025.

Boyuan Li, Zhen Liu, Yicheng Luo, and Qianli Ma. Learning recursive multi-scale representations for irregular multivariate time series forecasting. *arXiv preprint arXiv:2602.21498*, 2026.

Jiexi Liu, Meng Cao, and Songcan Chen. Timecheat: A channel harmony strategy for irregularly sampled multivariate time series analysis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 18861–18869, 2025.

Xvyuan Liu, Xiangfei Qiu, Hanyin Cheng, Xingjian Wu, Guo Guo, Bin Yang, and Jilin Hu. Astgi: Adaptive spatio-temporal graph interactions for irregular multivariate time series forecasting. In *International Conference on Learning Representations*, volume 2026, pp. 23940–23958, 2026a.

Xvyuan Liu, Xiangfei Qiu, Xingjian Wu, Zhengyu Li, Chenjuan Guo, Jilin Hu, and Bin Yang. Rethinking irregular time series forecasting: A simple yet effective baseline. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 23873–23881, 2026b.

Yicheng Luo, Bowen Zhang, Zhen Liu, and Qianli Ma. Hi-patch: Hierarchical patch gnn for irregular multivariate time series. In *Forty-second International Conference on Machine Learning*, 2025.

M.J. Menne, C.N. Williams, Jr., and R.S. Vose. Long-term daily and monthly climate records from stations across the contiguous united states (u.s. historical climatology network), 2016. URL `https://www.osti.gov/biblio/1394920`.

Giangiacomo Mercatali, Andre Freitas, and Jie Chen. Graph neural flows for unveiling systemic interactions among irregularly sampled time series. *Advances in Neural Information Processing Systems*, 37:57183–57206, 2024.

Daniel Neil, Michael Pfeiffer, and Shih-Chii Liu. Phased lstm: Accelerating recurrent network training for long or event-based sequences. *Advances in neural information processing systems*, 29, 2016.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

Xiangfei Qiu, Kangjia Yan, Xvyuan Liu, Xingjian Wu, and Jilin Hu. Bridging time and frequency: A joint modeling framework for irregular multivariate time series forecasting. *arXiv preprint arXiv:2602.00582*, 2026.

Yulia Rubanova, Ricky TQ Chen, and David K Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. *Advances in neural information processing systems*, 32, 2019.

Mona Schirmer, Mazin Eltayeb, Stefan Lessmann, and Maja Rudolph. Modeling irregular time series with continuous recurrent units. In *International conference on machine learning*, pp. 19388–19405. PMLR, 2022.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

Satya Narayan Shukla and Benjamin M Marlin. Interpolation-prediction networks for irregularly sampled time series. *arXiv preprint arXiv:1909.07782*, 2019.

Satya Narayan Shukla and Benjamin M Marlin. A survey on principles, models and methods for learning from irregularly sampled time series. *arXiv preprint arXiv:2012.00168*, 2020.

Satya Narayan Shukla and Benjamin M Marlin. Multi-time attention networks for irregularly sampled time series. In ... *International Conference on Learning Representations*, volume 2021, pp. 14897, 2021.

Ikaro Silva, George Moody, Daniel J Scott, Leo A Celi, and Roger G Mark. Predicting in-hospital mortality of icu patients: The physionet/computing in cardiology challenge 2012. In *2012 computing in cardiology*, pp. 245–248. IEEE, 2012.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Vedrana Vidulin, Mitja Lustrek, Bostjan Kaluza, Rok Piltaver, and Jana Krivec. Localization data for person activity, 2010. URL https://doi.org/10.24432/C57G8X.

Roberto Vio, María Diaz-Trigo, and Paola Andreani. Irregular time series in astronomy and the use of the lomb–scargle periodogram. *Astronomy and Computing*, 1:5–16, 2013.

Vijaya Krishna Yalavarthi, Kiran Madhusudhanan, Randolf Scholz, Nourhan Ahmed, Johannes Burchert, Shayan Jawed, Stefan Born, and Lars Schmidt-Thieme. Grafiti: Graphs for forecasting irregularly sampled time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 16255–16263, 2024.

Zhen-Jie Yao, Jie Bi, and Yi-Xin Chen. Applying deep learning to individual and community health monitoring data: A survey. *International Journal of Automation and Computing*, 15(6):643–655, 2018.

Jiawen Zhang, Shun Zheng, Wei Cao, Jiang Bian, and Jia Li. Warpformer: A multi-scale modeling approach for irregular clinical time series. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3273–3285, 2023.

Weijia Zhang, Chenlong Yin, Hao Liu, Xiaofang Zhou, and Hui Xiong. Irregular multivariate time series forecasting: A transformable patching graph neural networks approach. In *ICML*, pp. 60179–60196, 2024.

Xiang Zhang, Marko Zeman, Theodoros Tsiligkaridis, and Marinka Zitnik. Graph-guided network for irregularly sampled multivariate time series. *arXiv preprint arXiv:2110.05357*, 2021.

Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *The eleventh international conference on learning representations*, 2023.

Ziyu Zhou, Yiming Huang, Yanyun Wang, Yuankai Wu, James Kwok, and Yuxuan Liang. Revitalizing canonical pre-alignment for irregular multivariate time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 29115–29123, 2026.

A Research Transparency and Reproducibility

A.1 AI Use Statement

Codex was used to improve the clarity and readability of the manuscript's wording. It was not used to generate data, develop research ideas or methodology, formulate or prove mathematical claims, design experiments, implement methods, process data, or interpret results. The authors reviewed the AI-assisted edits and take responsibility for the final manuscript, results, and artifacts.

A.2 Ethics Statement

All experiments use five previously released benchmark datasets; no new participant data were collected. Human Activity contains human movement measurements, and USHCN contains climate records. PhysioNet 2012 contains intensive-care measurements and is openly accessible under its dataset license. MIMIC-III and MIMIC-IV contain de-identified intensive-care records and are available only through credentialed access, which requires completion of the relevant training and acceptance of their data use agreements. We used all five datasets under their respective access terms. The supplementary material contains the experimental code, and we report only aggregate performance metrics without disclosing patient-level records.

A.3 Reproducibility Statement

Section 4 specifies the model architecture, computations, and training objective. Section 5 describes the optimization setup, hardware, evaluation metrics, and five random seeds. Appendix B.1 specifies final-patch padding, Appendix B.2 documents the datasets and data splits, Appendix B.3 describes baseline implementations. The experimental code is included in the supplementary files.

B Additional Experimental Details

B.1 Final Patch Padding

If the aligned input length L is not divisible by K, the final patch is right-padded with $PK - L$ positions. Values and observation masks at these positions are set to zero, and their timestamps repeat the last timestamp of the input sequence. Thus, every patch has length K.

B.2 Dataset Details

We evaluate MorITS on five widely used irregular multivariate time-series forecasting datasets: Human Activity, PhysioNet 2012, USHCN, MIMIC-III, and MIMIC-IV. They cover biomechanics, healthcare, and climate science. Table 3 reports processed-benchmark statistics from Li et al. (2026); the four datasets other than MIMIC-IV also appear in Liu et al. (2026b). The lookback periods are 36 hours for PhysioNet 2012, MIMIC-III, and MIMIC-IV, 3000 ms for Human Activity, and 3 years for USHCN. Human Activity forecasts the following 300 ms; the other four datasets use the next three timestamps as targets.

Table 3: Statistics of the five processed forecasting datasets.

Dataset	Variables	Samples	Avg. observations	Max. length
Human Activity	12	1,359	362.2	131
PhysioNet 2012	36	11,981	308.6	47
USHCN	5	1,114	313.5	337
MIMIC-III	96	21,250	144.6	96
MIMIC-IV	100	17,874	304.8	971

Human Activity. Human Activity (Vidulin et al., 2010) contains three-dimensional position measurements from subjects performing different activities. The processed benchmark has 12 variables, with timestamps rounded to one-millisecond intervals.

PhysioNet 2012. The PhysioNet Challenge 2012 dataset (Silva et al., 2012) provides clinical time series from the first 48 hours of intensive-care stays. Its processed forecasting version contains 36 clinical variables, with timestamps rounded to one-hour intervals.

USHCN. The United States Historical Climatology Network (USHCN) (Menne et al., 2016) provides historical weather measurements from stations across the United States. We use the five-variable subset prepared by De Brouwer et al. (2019), which covers approximately four years between 1996 and 2000.

MIMIC-III. MIMIC-III (Johnson et al., 2016) contains de-identified intensive-care measurements. The processed benchmark contains 96 clinical variables, with timestamps rounded to 30-minute intervals.

MIMIC-IV. MIMIC-IV (Johnson et al., 2023) contains de-identified intensive-care measurements. The processed benchmark contains 100 clinical variables, with timestamps rounded to one-minute intervals.

Following the benchmark preprocessing pipeline, we first hold out 10% of samples for testing and then assign 10% of the remainder to validation, yielding approximately 81%/9%/10% training/validation/test partitions. USHCN uses a randomized split with seed 1; the other four datasets retain the existing sample order. All methods use the same data partitions.

B.3 Baseline Details

We compare MorITS with 13 representative baselines covering recurrent, continuous-time, flow-based, graph-based, attention-based, and patch-based approaches for irregular time-series modeling.

GRU-D. GRU-D (Che et al., 2018) augments gated recurrent units with trainable decay mechanisms to account for elapsed time and missing observations.

mTAN. mTAN (Shukla & Marlin, 2021) uses multi-time attention to map irregular observations onto a set of reference points for downstream prediction.

Raindrop. Raindrop (Zhang et al., 2021) represents variables as graph nodes and uses graph attention to model dependencies between sensors under irregular sampling.

Warpformer. Warpformer (Zhang et al., 2023) introduces temporal warping into a Transformer architecture to capture multi-scale patterns in irregular and non-stationary sequences.

NeuralFlows. NeuralFlows (Biloš et al., 2021) models continuous latent dynamics with time-dependent normalizing flows as an alternative to ordinary differential equation solvers.

CRU. CRU (Schirmer et al., 2022) uses continuous recurrent state transitions to propagate latent states between irregular observations.

GNeuralFlow. GNeuralFlow (Mercatali et al., 2024) combines neural flows with graph neural networks to jointly model continuous-time dynamics and dependencies among variables.

t-PatchGNN. t-PatchGNN (Zhang et al., 2024) partitions each irregular univariate series into transformable patches and employs graph neural networks to capture inter-variable dependencies.

GraFITi. GraFITi (Yalavarthi et al., 2024) represents historical observations and prediction queries using a bipartite graph and performs forecasting through message passing.

Hi-Patch. Hi-Patch (Luo et al., 2025) constructs hierarchical patch representations to capture temporal information at multiple granularities.

APN. APN (Liu et al., 2026b) uses time-aware patch aggregation to convert irregular observations into regularized patch representations, followed by query-based historical aggregation.

ASTGI. ASTGI (Liu et al., 2026a) adaptively constructs spatio-temporal interaction graphs to model temporal relations between observations and dependencies between variables.

TFMixer. TFMixer (Qiu et al., 2026) jointly models time- and frequency-domain information using a learnable non-uniform discrete Fourier transform and mixing blocks.