ADDP: COORDINATED HISTORICAL PRUNING VIA A SHARED CONTINUOUS ROUTING SIGNAL FOR IRREGULAR MULTIVARIATE TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Effectively utilizing historical observations is fundamental to irregular multivariate time series forecasting. However, real-world time series often contain redundant segments that distract model attention and degrade forecasting performance. Existing pruning methods for low-information data typically rely on independently designed decision rules for different components, making it difficult for different pruning strategies to coordinate or conflict on the same sample. We propose Adaptive Data-Dependent Pruning (ADDP), a framework that uses a shared signal to guide multiple pruning decisions during training. Specifically, ADDP summarizes each sample's historical variation in a shared signal that guides subsequent pruning decisions. Lightweight learnable calibrators then use this signal to determine the redundancy of historical segments, the amount of information shared between aggressive and conservative pruning views, and the weights assigned to these views. Based on these decisions, ADDP constructs two complementary views that retain different amounts of historical information and jointly trains a shared forecasting model. Experiments on PhysioNet, Activity, USHCN, and MIMIC-III show that ADDP consistently outperforms competing methods, achieving the lowest mean MSE on each of the four datasets. On MIMIC-III, ADDP reduces MSE by 5.05% relative to Hi-Patch, the strongest evaluated baseline.

1 INTRODUCTION

Irregular Multivariate Time Series (IMTS) forecasting is widely used in healthcare monitoring, human-activity analysis, and climate forecasting, where effective utilization of historical information is essential for predicting future states (Silva et al., 2012; Johnson et al., 2016; Vidulin et al., 2010; Menne et al., 2009). However, historical sequences can contain repetitive or low-information segments, motivating selective utilization of historical context (Qiu et al., 2025). Meanwhile, different samples may exhibit substantial differences in observation counts, sampling intervals, missingness levels, and temporal variation patterns (Che et al., 2018; Rubanova et al., 2019; Shukla & Marlin, 2021; Zhang et al., 2024). Therefore, a single historical pruning strategy may not be suitable for all inputs.

To address this issue, existing studies have explored patch-level masking, random dropping, and adaptive or transformable patching strategies, which selectively retain, perturb, or reorganize historical segments to improve representation and forecasting (Nie et al., 2023; Qiu et al., 2025; Zhang et al., 2024; Liu et al., 2026). Despite the promising progress, we identify two underexplored issues in historical pruning for irregular time series.

① **Fixed historical selection preferences are difficult to adapt to heterogeneous irregular inputs.** Existing random dropping or fixed scoring strategies typically apply the same information-selection rule to different samples. However, irregular histories may differ substantially in both observation density and temporal variation patterns. For example, a densely observed interval may mainly contain repeated or slowly varying values, whereas a sparsely observed interval may contain a pronounced state transition. Therefore, relying only on a unified observation-density criterion, fixed scoring weights, or a fixed routing rule makes it difficult to determine which historical patches should be

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.

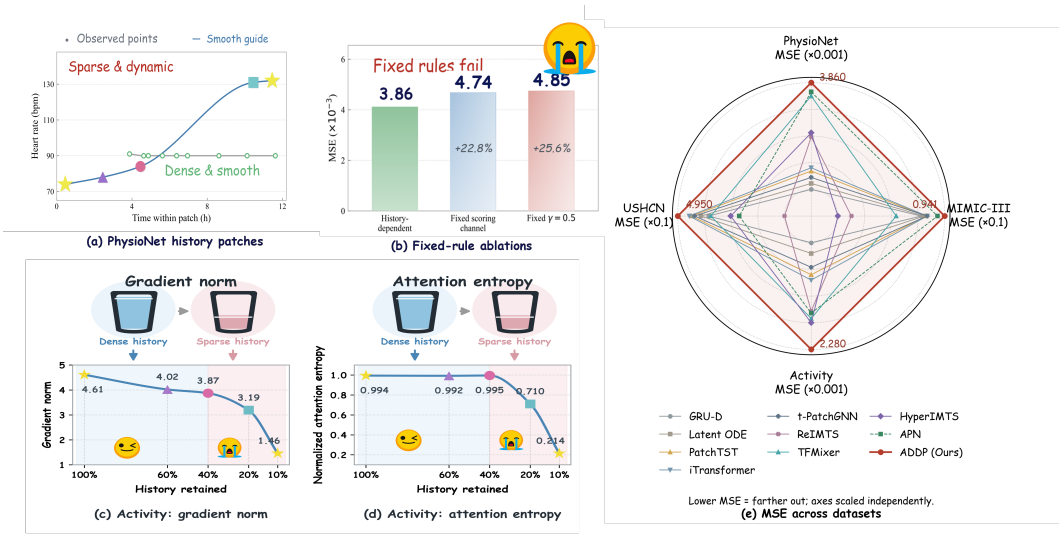


Figure 1: Motivation and performance overview of ADDP. (a) Illustrative PhysioNet-style history patches: dense observations can remain nearly unchanged, whereas sparse observations can capture a clear rise. Lines connect observed points for visualization. (b) On PhysioNet, fixed scoring and fixed routing increase MSE from 3.86×10^{-3} to 4.74×10^{-3} and 4.85×10^{-3}, respectively. (c) On Activity, retaining less history reduces the attention-parameter gradient norm measured at initialization. (d) For a model trained with the full reference history, attention entropy stays near 0.99 through 40% retention, then decreases at 20% and 10% retention. Lower entropy indicates more concentrated attention. The water levels in (c) and (d) illustrate the amount of retained history; both diagnostics use iTransformer. (e) Mean-MSE comparison across four datasets. Each axis is scaled independently, with lower MSE farther from the center.

preferentially retained for different samples. Figure 1(a) schematically illustrates this contrast: a densely observed patch changes little, whereas a sparse patch captures a clear rise.

Figure 1(b) compares two fixed-rule variants on PhysioNet. ADDP achieves an MSE of 0.003860. A single fixed scoring channel increases MSE to 0.00474, and a fixed routing signal $\gamma = 0.5$ increases MSE to 0.00485. These results suggest that adapting patch-selection preferences to the current history is more appropriate than applying the same selection rule to all samples.

② Historical information retention and model optimization lack a unified adaptation signal. Deciding *what to retain* is only one aspect of historical pruning; *how much to retain* can also alter the model's learning behavior. Retaining more history provides more complete context, whereas more aggressive pruning introduces stronger input perturbation. As shown in Figure 1(c), in our diagnostic experiments with iTransformer on Activity, reducing the retained observation ratio from 100% to 10% decreases the attention-parameter gradient norm at initialization from 4.61 to 1.46. Meanwhile, Figure 1(d) shows that, for a model trained with the full reference history, severe information removal makes the attention distribution substantially more concentrated, with normalized attention entropy decreasing from 0.994 to 0.214. These results show that historical pruning not only changes the information available to the model, but also affects its optimization signals and attention allocation. In other words, different pruning strengths create training views with different levels of difficulty. Therefore, the model should not only determine what to retain and how much to retain, but also adjust the importance of different views during training. If patch scoring, retention size, and training weights are controlled by independent rules, these components may respond inconsistently to the same historical sample. This motivates us to use a unified history-dependent signal to jointly coordinate historical selection, information retention, and training weighting.

To address these limitations, we propose **A**daptive **D**ata-**D**ependent **P**runing (ADDP), a training framework that uses one historical signal to coordinate which patches to retain and how to learn from them. ADDP first computes a continuous roughness index γ from changes between consecutive observations, normalized by their dispersion and count. This index is computed from the observed history and guides three decisions: patch scoring, shared-core retention, and the relative weight of

two training views. For patch scoring, γ balances two scoring channels that consider local variation, deviation from the historical mean, observation density, and recency. The highest-ranked patches form a shared core whose size also depends on γ; adding different amounts of other history produces a conservative view and an aggressive view. The shared core keeps important context available to both views, while the additional patches vary how much information the model receives. Both views use the same forecasting model, and a smooth worst-view objective gives more weight to the view with the larger prediction error. Its temperature is also controlled by γ, so the historical signal determines how strongly training emphasizes the harder view. Lightweight constrained calibrators adjust these responses while preserving the scoring roles and the direction of the retention and temperature mappings. Figure 1(e) summarizes forecasting MSE across the four datasets, with lower error farther from the center on each independently scaled axis.

Our contributions are threefold:

- **History-dependent adaptive pruning.** We propose a history-dependent adaptive pruning method that uses a unified historical variation signal to jointly adjust patch selection, shared information retention, and training weights, enabling historical pruning to adapt to heterogeneous irregular inputs.
- **Shared-core dual-view training.** We design a shared-core dual-view training strategy. The two training views jointly retain high-scoring historical patches and incorporate different amounts of additional historical information to form conservative and aggressive pruning views. During training, the model assigns a larger weight to the view with higher prediction error, allowing it to learn under different levels of historical retention.
- **Evaluation across heterogeneous irregular forecasting benchmarks.** We evaluate ADDP on PhysioNet, Activity, USHCN, and MIMIC-III, covering healthcare, human-activity monitoring, and climate forecasting (Silva et al., 2012; Johnson et al., 2016; Vidulin et al., 2010; Menne et al., 2009). ADDP achieves the lowest average MSE among all evaluated methods. Routing ablations, retention-ratio sensitivity analysis, and observation-removal diagnostics further examine the key design choices.

2 Related Work

Irregular historical representation. Existing methods for irregular multivariate time series mainly focus on learning effective representations from asynchronous observations, missing values, and uneven sampling patterns. GRU-D incorporates missingness patterns and temporal decay mechanisms into recurrent networks, enabling hidden states to be updated according to observation intervals (Che et al., 2018). Latent ODE models irregular observations as continuous latent dynamics and describes hidden state evolution between observations (Rubanova et al., 2019). mTAN introduces multi-time attention to aggregate observations at arbitrary timestamps under irregular sampling conditions (Shukla & Marlin, 2021). t-PatchGNN further explores structured representation learning by combining temporal patching with adaptive graph construction to jointly model temporal patterns and variable dependencies (Zhang et al., 2024). ReIMTS learns recursive multi-scale representations to capture irregular dynamics at different temporal resolutions (Li et al., 2026). TFMixer jointly models temporal and frequency information to extract complementary historical representations (Qiu et al., 2026). HyperIMTS employs hypergraph structures to model higher-order interactions among variables and observations, providing expressive representations for irregular multivariate dependencies (Li et al., 2025). APN introduces adaptive patching and time-aware aggregation to dynamically organize irregular observations according to input characteristics (Liu et al., 2026). Although these methods substantially improve irregular history representation, they mainly focus on how to encode and aggregate observed information, while how to selectively retain or remove historical information during training remains underexplored.

Historical information selection and pruning. Recent studies have explored masking, dropping, or selecting parts of the input sequence to reduce redundancy and improve representation learning. PatchTST introduces patch-level masking in self-supervised learning, where masked patches encourage robust temporal representation learning (Nie et al., 2023). DropPatch further investigates temporal patch dropping by removing redundant patches to improve masked time-series modeling (Qiu et al., 2025). VarDrop explores adaptive token pruning from the variable dimension by identifying redundant variables and removing unnecessary variable tokens (Kang et al., 2025). Pathformer

employs multi-scale routing to select different patch resolutions and adaptively determine interactions among patches (Chen et al., 2024). iTransformer reinterprets multivariate time series by applying attention across variables, enhancing variable-wise dependency modeling (Liu et al., 2024). However, existing methods typically rely on random dropping, predefined selection rules, or dimension-specific selection strategies, which makes it difficult for models to adaptively select important historical information according to each input and coordinate different pruning decisions. ADDP addresses this limitation by using a shared history-dependent routing signal to coordinate historical selection, information retention, and training optimization.

3 METHODOLOGY

We first define the forecasting task, then describe how ADDP uses the observed history to coordinate patch scoring, view construction, and training. Figure 2 shows the complete pipeline.

3.1 PROBLEM SETUP

For sample i, let the irregular historical input be

$$\mathcal{X}_i = \{(t_{ij}, \mathbf{x}_{ij}, \mathbf{m}_{ij})\}_{j=1}^{T_i},$$

where t_{ij} denotes an observation timestamp, $\mathbf{x}_{ij} \in \mathbb{R}^V$ contains the input variables, and $\mathbf{m}_{ij} \in \{0,1\}^V$ indicates which variables are observed at that timestamp. All historical timestamps precede the forecasting cutoff, and the number and timing of observations may vary across samples and variables.

Let

$$\mathbf{t}_i^+ = (t_{i1}^+, \ldots, t_{iH_i}^+)$$

denote the future query times. The corresponding prediction targets and observation mask are

$$\mathbf{Y}_i \in \mathbb{R}^{H_i \times V_Y}, \qquad \mathbf{M}_i^Y \in \{0,1\}^{H_i \times V_Y},$$

where V_Y is the number of prediction channels. We distinguish the input dimension V from the target dimension V_Y so that auxiliary input variables need not be treated as forecasting targets. Given the irregular history and future query times, the forecasting model produces

$$\widehat{\mathbf{Y}}_i = f_\theta(\mathcal{X}_i, \mathbf{t}_i^+).$$

3.2 TEMPORAL PATCHES AND FORECASTING LOSS

The historical window is partitioned into M temporally ordered patches. For the temporal interval I_{ip} corresponding to patch p, define

$$\mathcal{P}_{ip} = \{(t_{ij}, \mathbf{x}_{ij}, \mathbf{m}_{ij}) : \ t_{ij} \in I_{ip}\}.$$

Each patch therefore contains the values, timestamps, and observation masks associated with a local portion of the historical window. Because the data are irregular, the number of observed entries can differ across patches, samples, and variables. These patches are the units selected by ADDP.

Forecasting error is evaluated only on observed future targets. Define

$$\mathcal{O}_i^Y = \{(u,v) : M_{iuv}^Y = 1\}, \qquad N_i^Y = |\mathcal{O}_i^Y|.$$

For any prediction $\widehat{\mathbf{Y}}_i$, the per-sample observed-target mean squared error is

$$\mathcal{L}_{\text{MSE}}\left(\widehat{\mathbf{Y}}_i, \mathbf{Y}_i\right) = \frac{1}{N_i^Y} \sum_{(u,v)\in\mathcal{O}_i^Y} \left(\widehat{Y}_{iuv} - Y_{iuv}\right)^2. \tag{1}$$

Samples with $N_i^Y = 0$ do not contribute to the supervised objective. We apply the same observed-target loss to each training view before combining the losses. Evaluation across samples and channels is specified in Appendix F.4.

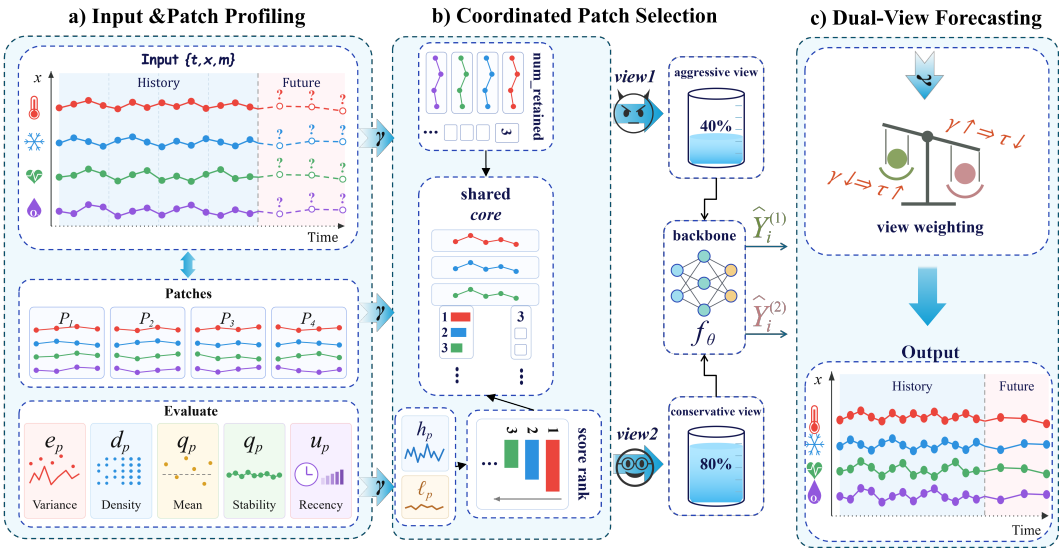


Figure 2: Overview of ADDP. A history-derived continuous routing index γ provides a shared conditioning signal for three lightweight constrained calibrators: learnable patch-scoring weights, monotone shared-core retention, and monotone temperature modulation of the smooth worst-view objective. Conservative and aggressive views retain a common high-ranked core and are processed by the same forecasting backbone during training. At inference, the predictor processes the complete history in a single forward pass.

3.3 Structure Overview

ADDP constructs two training views of the same history and passes both through one forecasting model. First, a continuous roughness index γ summarizes changes in the observed values. Next, this index controls the balance between two patch-scoring channels and the size of a shared set of high-ranked patches. We add different amounts of other history to obtain conservative and aggressive views. Finally, a smooth worst-view loss gives greater weight to the harder view, with γ controlling the strength of this weighting. Small constrained calibrators adjust how each decision responds to the shared index. Algorithm 1 summarizes the training and inference steps.

3.4 Shared Forecasting Backbone

ADDP controls training inputs and losses around a forecasting model f_θ. We use a t-PatchGNN-style predictor with temporal aggregation, Transformer layers (Vaswani et al., 2017), adaptive graph convolution, and a forecasting decoder (Zhang et al., 2024). The two views share all predictor parameters. Their intended difference is the historical information they retain. At inference, f_θ receives the original history and future query times without the additional pruning mask. Appendix B lists the dataset-specific configurations.

3.5 History-Derived Continuous Routing Index

We need a common signal that describes the current history before making any pruning decision. ADDP uses normalized changes between consecutive observations to obtain this signal. For clarity, the sample index i is omitted in this subsection and in the patch-scoring formulation. For variable v, let $x_{1v}, \ldots, x_{n_v,v}$ denote its observed historical values in temporal order. For $n_v > 1$, define

$$r_v = \frac{\sum_{j=2}^{n_v} |x_{jv} - x_{j-1,v}|}{n_v \,\mathrm{std}(x_{1v}, \ldots, x_{n_v,v}) + \epsilon_r}, \qquad \epsilon_r = 0.01. \tag{2}$$

The numerator accumulates changes between consecutive observed values, while the denominator normalizes these changes by historical dispersion and observation count. We use the sample-standard-deviation convention.

Let

$$\mathcal{V}_{\text{obs}} = \{v : n_v > 1\}$$

denote the variables with sufficient observations. The sample-level routing index is

$$\gamma = \text{clip}\left(\frac{1}{c|\mathcal{V}_{\text{obs}}|}\sum_{v\in\mathcal{V}_{\text{obs}}} r_v, 0, 1\right), \qquad c = 1.5. \tag{3}$$

When $\mathcal{V}_{\text{obs}}$ is empty, we set $\gamma = 0$. The resulting $\gamma \in [0, 1]$ is a history-derived routing signal that summarizes normalized sequential roughness. The index measures variation along the observed sequence; timestamps enter the temporal representation of the forecasting backbone. All routing statistics are computed solely from the historical input and outside the gradient path.

We reuse γ for patch scoring, shared-core retention, and loss weighting. Each calibrator maps the shared signal through its own response parameters. The shared index provides a common sample-level coordinate for pruning intensity: histories with different normalized variation occupy different positions along the same adaptation axis. The three calibrators retain decision-specific response functions, so patch ranking, core size, and view weighting can react at different rates while remaining coordinated with the same historical state.

3.6 Learnably Calibrated Patch Scoring

Patch scoring decides which parts of the history to retain first. We combine measures of local variation, observation availability, and departure from the sample's usual values. For patch p, let e_p denote its variance-based energy statistic, d_p its observation density, and a_p its deviation from the historical sample mean. We additionally define

$$q_p = (e_p + 10^{-3})^{-1} \tag{4}$$

and the index-conditioned recency term

$$u_p = \exp[-0.1(1-\gamma)(M-p)]. \tag{5}$$

Instead of fixing the internal weights of the two scoring channels, ADDP learns nonnegative convex coefficients:

$$[\alpha_e, \alpha_d] = \text{softmax}(\mathbf{a}_h), \tag{6}$$
$$[\beta_a, \beta_q, \beta_d] = \text{softmax}(\mathbf{a}_\ell), \tag{7}$$
$$\lambda_u = \text{softplus}(a_u). \tag{8}$$

The two channel scores and final patch score are

$$h_p = \alpha_e e_p + \alpha_d d_p, \tag{9}$$
$$\ell_p = \beta_a a_p + \beta_q q_p + \beta_d d_p, \tag{10}$$
$$s_p = \gamma h_p + (1-\gamma)\ell_p + \lambda_u u_p. \tag{11}$$

For larger γ, the score relies more on the variation and density channel h_p. For smaller γ, it relies more on deviation, inverse variance, and density through ℓ_p. The recency term favors later patches. Convex coefficients keep the contributions within each channel nonnegative and summing to one. The inverse-variance term favors low-variance patches.

The statistics are computed on normalized historical values. The energy statistic uses the sample variance of the patch values, and the deviation statistic is the absolute difference between the patch mean and the historical sample mean. Empty patches receive zero score before ranking.

3.7 Monotone Shared-Core Calibration

The two views should differ in how much history they retain while sharing high-ranked patches. We first set the shared-core ratio with a bounded mapping of γ. Its decreasing direction encodes the design choice that a rougher history should not require a larger shared core:

$$b_c = \mathrm{softplus}(\widetilde{b}_c), \tag{12}$$

$$\rho_i = \rho_{\min} + (\rho_{\max} - \rho_{\min})\sigma(a_c - b_c\gamma_i), \tag{13}$$

where $0 < \rho_{\min} < \rho_{\max} < 1$ are fixed bounds. The number of shared patches is

$$k_{\mathrm{core},i} = \max\{1, \lfloor M\rho_i \rfloor\}. \tag{14}$$

Because $b_c > 0$, ρ_i is monotone non-increasing in γ_i. The intercept sets the typical core size, and the positive slope controls how strongly it changes with roughness. Patch ranking and core selection use the resulting scores and integer counts. The core-ratio mapping is parameterized by trainable calibration parameters.

After sorting patches by s_p, the top $k_{\mathrm{core},i}$ patches form the shared core $\mathcal{C}_i$. The conservative and aggressive retained sets, $\mathcal{R}_{i1}$ and $\mathcal{R}_{i2}$, both contain this core. Additional non-core patches are sampled without replacement toward nominal retention targets

$$r_1 = 0.8, \qquad r_2 = 0.4.$$

The full shared core is always retained. If its size already exceeds a branch's target count, that branch retains the core without adding further patches.

For view $\widetilde{\mathcal{X}}_i^{(b)}$, values in a pruned patch are replaced by $(1 - \gamma_i)$ times the sample-specific per-variable historical mean; variables without historical observations use their training-set means. The corresponding routing/topology value is set to $1 - \gamma_i$. Before Transformer processing, the routing mask is converted into a Boolean key-padding mask, and patch-validity information is propagated to the graph stage. Both views are processed by the same predictor f_θ.

3.8 Monotonically Calibrated Smooth Worst-View Objective

For training view $b \in \{1, 2\}$, let $\widehat{\mathbf{Y}}_i^{(b)}$ denote the prediction obtained from $\widetilde{\mathcal{X}}_i^{(b)}$. Its observed-target forecasting loss is

$$L_{ib} = \mathcal{L}_{\mathrm{MSE}}\left(\widehat{\mathbf{Y}}_i^{(b)}, \mathbf{Y}_i\right). \tag{15}$$

Both branches use the same future query times, targets, and target observation mask.

An average of the two losses treats both views equally even when one is harder to predict from. We instead use a smooth approximation to the larger loss, related to tilted empirical risk minimization (Li et al., 2021). Its temperature controls how much more weight the harder view receives. We derive the temperature from the same historical signal:

$$b_\tau = \mathrm{softplus}(\widetilde{b}_\tau), \tag{16}$$

$$\tau_i = \tau_{\min} + (\tau_{\max} - \tau_{\min})\sigma(a_\tau - b_\tau\gamma_i), \tag{17}$$

with fixed positive bounds $0 < \tau_{\min} < \tau_{\max}$. This gives a positive temperature that decreases with γ_i for finite parameters. A larger routing index therefore increases the emphasis on the harder view. Appendix B lists the dataset-specific backbone settings.

Let

$$L_i^{\max} = \max_b L_{ib}.$$

The sample loss is

$$L_i = L_i^{\max} + \tau_i \log\left[\frac{1}{2}\sum_{b=1}^{2}\exp\left(\frac{L_{ib} - L_i^{\max}}{\tau_i}\right)\right]. \tag{18}$$

The induced branch weights are

$$w_{ib} = \frac{\exp(L_{ib}/\tau_i)}{\sum_{b'=1}^{2}\exp(L_{ib'}/\tau_i)}. \tag{19}$$

The higher-loss view therefore receives greater weight, and smaller temperatures increase the contrast between branch contributions. Thus, the same historical signal guides patch selection, common

context, and view weighting. The temperature controls the relative emphasis on the two constructed views and is updated through the forecasting loss. Appendix C gives the corresponding bounds and limiting behavior.

Inference. Given the historical values, timestamps, and observation masks $\mathcal{X}_i$ and future query times $\mathbf{t}_i^+$, the trained shared predictor returns

$$\widehat{\mathbf{Y}}_i = f_\theta(\mathcal{X}_i, \mathbf{t}_i^+). \tag{20}$$

The original observation mask specifies the available history. Inference uses one forecasting pass with all observed historical entries.

4 Experiments

Datasets. We evaluate ADDP on four datasets covering clinical monitoring, human activity, and climate observations. PhysioNet and MIMIC-III provide intensive-care measurements (Silva et al., 2012; Johnson et al., 2016); Activity contains recordings from wearable localization tags (Vidulin et al., 2010); and USHCN contains station-based climate records (Menne et al., 2009). Dataset representations, observation statistics, and forecasting tasks are summarized in Table 8 of Appendix E.

Settings. We report MSE on observed future targets; lower values are better. ADDP uses a shared temporal/Transformer/graph predictor and Adam (Kingma & Ba, 2015), with a maximum budget of 50 epochs. PhysioNet and USHCN use early-stopping patience 10. Activity uses the full 50 epochs. MIMIC-III uses early stopping with patience 10 after epoch 25. Checkpoints are selected by validation MSE. Dataset-specific settings appear in Appendix B.

Baselines. Table 1 compares ADDP with representative recurrent, continuous-time, Transformer, patch-based, and irregular-time-series forecasters, including GRU-D (Che et al., 2018), Latent ODE (Rubanova et al., 2019), PatchTST (Nie et al., 2023), iTransformer (Liu et al., 2024), t-PatchGNN (Zhang et al., 2024), ReIMTS (Li et al., 2026), TFMixer (Qiu et al., 2026), HyperIMTS (Li et al., 2025), APN (Liu et al., 2026), GraFITi (Yalavarthi et al., 2024), ISTS-PLM (Zhang et al., 2025), and Hi-Patch (Luo et al., 2025). Results for all 26 baselines appear in Appendix D.

Reporting. Relative error reduction is $(\mathrm{MSE}_{\mathrm{baseline}} - \mathrm{MSE}_{\mathrm{ADDP}})/\mathrm{MSE}_{\mathrm{baseline}}$. Appendix F.4 defines the evaluation metrics.

4.1 Main Results

Table 1 presents forecasting results across four datasets. ADDP achieves the lowest MSE: 0.003860 on PhysioNet, 0.094191 on MIMIC-III, 0.002287 on Activity, and 0.495418 on USHCN. The corresponding reductions relative to the strongest baselines are 4.93%, 5.05%, 9.60%, and 2.25%, respectively.

APN gives the lowest baseline MSE on PhysioNet, Hi-Patch on MIMIC-III, HyperIMTS on Activity, and iTransformer on USHCN. The following analyses examine the contributions of patch scoring, history-dependent routing, and dual-view training.

4.2 Model Analyses

Figure 3 organizes four analyses around the decisions made by ADDP. The first three panels report MSE increases relative to their respective full configurations; panel (d) compares the same forecasting backbones with and without ADDP.

Adaptive selection and view weighting. On PhysioNet, random dual views with mean loss yield MSE 0.004107; adaptive ranking, fixed-temperature weighting, and full ADDP reduce it to 0.003950, 0.003916, and 0.003860, respectively (Figure 3(a)). Random dual views are 6.40% higher.

History-dependent routing. Figure 3(b) compares the history-derived signal with fixed, linear-decay, and random signals, which increase PhysioNet MSE by 25.65%, 27.46%, and 40.16%, respectively.

Coordinating the three decisions. The shared-γ configuration obtains Activity MSE 0.002280, compared with 0.002331 for independent signals and 0.002348, 0.002356, and 0.002371 for scoring-, temperature-, and core-only adaptation (Figure 3(c)).

Table 1: Forecasting performance against representative baselines across four datasets. Red and blue mark the lowest and second-lowest MSE, respectively. Gap is the relative MSE reduction over the strongest baseline on each dataset. The complete comparison appears in Appendix D.

Algorithm	**PhysioNet** (24h → 24h)	**MIMIC-III** (24h → 24h)	**Activity** (4s → 5steps)	**USHCN** (24months → 1month)
	MSE (10^{-3}) ↓	MSE (10^{-2}) ↓	MSE (10^{-3}) ↓	MSE (10^{-1}) ↓
GRU-D	6.18 ± 0.42	11.8 ± 1.50	3.28 ± 0.35	5.267 ± 0.189
Latent ODE	6.05 ± 0.38	11.5 ± 1.40	3.18 ± 0.31	5.189 ± 0.165
PatchTST	5.78 ± 0.29	11.3 ± 1.30	2.98 ± 0.25	5.091 ± 0.138
iTransformer	5.71 ± 0.27	11.2 ± 1.30	2.93 ± 0.23	5.068 ± 0.129
t-PatchGNN	5.92 ± 0.31	11.2 ± 1.20	3.05 ± 0.28	5.123 ± 0.142
ReIMTS	5.04 ± 0.41	18.95 ± 2.24	2.64 ± 0.30	6.051 ± 0.319
TFMixer	4.15 ± 0.36	14.36 ± 1.81	2.57 ± 0.28	5.291 ± 0.238
HyperIMTS	4.95 ± 0.44	20.33 ± 2.56	2.53 ± 0.26	5.496 ± 0.285
APN	4.06 ± 0.10	10.14 ± 0.96	2.62 ± 0.93	5.5834 ± 0.015
GraFITi	5.11 ± 0.19	10.49 ± 0.13	3.03 ± 0.14	5.07 ± 0.03
ISTS-PLM	4.92 ± 0.05	10.08 ± 0.23	2.58 ± 0.03	5.23 ± 0.07
Hi-Patch	4.86 ± 0.03	9.92 ± 0.14	2.57 ± 0.02	5.14 ± 0.05
ADDP	3.860 ± 0.050	9.419 ± 0.020	2.287 ± 0.009	4.954 ± 0.040
Gap (%)	↓ 4.93	↓ 5.05	↓ 9.60	↓ 2.25

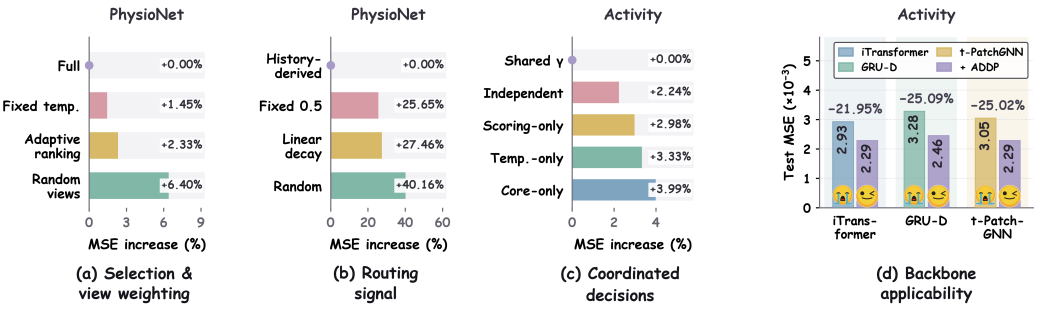


Figure 3: ADDP component analyses. (a) Adaptive selection and view weighting reduce PhysioNet MSE. (b) History-derived routing outperforms fixed, scheduled, and random signals. (c) A shared γ improves coordinated decisions. (d) ADDP improves three forecasting backbones. Panels (a)–(c) report relative MSE increases; (d) reports absolute MSE.

Backbone applicability. Figure 3(d) shows that ADDP reduces Activity MSE from 0.002930 to 0.002287 with iTransformer, 0.003280 to 0.00245714 with GRU-D, and 0.003050 to 0.002287 with t-PatchGNN.

Appendices E.2–E.4 and B.2 provide the exact values; Appendices E.6, F.1, and F.2 report further scoring, retention, and routing-scale analyses.

5 Conclusion

We introduced ADDP, which uses a shared history-dependent signal to coordinate patch selection, information retention, and training-view weighting for irregular multivariate time-series forecasting. Constrained calibrators adapt these decisions, while inference uses the original history in a single pass. ADDP achieves the lowest MSE among the evaluated methods on four datasets, and its ablations support the coordinated design. By aligning independently designed pruning decisions, ADDP enables historical pruning to adapt to heterogeneous irregular inputs.

REFERENCES

Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific Reports*, 8:6085, 2018. doi: 10.1038/s41598-018-24271-9.

Peng Chen, Yingying Zhang, Yunyao Cheng, Yang Shu, Yihang Wang, Qingsong Wen, Bin Yang, and Chenjuan Guo. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/2be6705de7412adf107900add727a795-Abstract-Conference.html.

Alistair E. W. 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*, 3:160035, 2016. doi: 10.1038/sdata.2016.35.

Junhyeok Kang, Yooju Shin, and Jae-Gil Lee. VarDrop: Enhancing training efficiency by reducing variate redundancy in periodic time series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(17):17742–17749, 2025. doi: 10.1609/aaai.v39i17.33951.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6980.

Boyuan Li, Yicheng Luo, Zhen Liu, Junhao Zheng, Jianming Lv, and Qianli Ma. HyperIMTS: Hypergraph neural network for irregular multivariate time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 35502–35518, 2025. URL https://proceedings.mlr.press/v267/li25bl.html.

Boyuan Li, Zhen Liu, Yicheng Luo, and Qianli Ma. Learning recursive multi-scale representations for irregular multivariate time series forecasting. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/410dacbcfda0388f69b8983ed771c44e-Abstract-Conference.html.

Tian Li, Ahmad Beirami, Maziar Sanjabi, and Virginia Smith. Tilted empirical risk minimization. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2007.01162.

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, 2026.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.06625.

Yicheng Luo, Bowen Zhang, Zhen Liu, and Qianli Ma. Hi-Patch: Hierarchical patch GNN for irregular multivariate time series. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 41494–41519, 2025. URL https://proceedings.mlr.press/v267/luo25r.html.

Matthew J. Menne, Claude N. Williams, Jr., and Russell S. Vose. The U.S. Historical Climatology Network monthly temperature data, version 2. *Bulletin of the American Meteorological Society*, 90 (7):993–1007, 2009. doi: 10.1175/2008BAMS2613.1.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2211.14730.

Tianyu Qiu, Yi Xie, Hao Niu, Yun Xiong, and Xiaofeng Gao. Enhancing masked time-series modeling via dropping patches. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(19):20077–20085, 2025. doi: 10.1609/aaai.v39i19.34211. URL https://doi.org/10.1609/aaai.v39i19.34211.

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. URL https://arxiv.org/abs/2602.00582.

Yulia Rubanova, Ricky T. Q. Chen, and David K. Duvenaud. Latent ODEs for irregularly-sampled time series. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://arxiv.org/abs/1907.03907.

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 *Computing in Cardiology*, volume 39, pp. 245–248, 2012. URL https://physionet.org/content/challenge-2012/1.0.0/.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://arxiv.org/abs/1706.03762.

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

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. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38 (15):16255–16263, 2024. doi: 10.1609/aaai.v38i15.29560.

Weijia Zhang, Chenlong Yin, Hao Liu, Xiaofang Zhou, and Hui Xiong. Irregular multivariate time series forecasting: A transformable patching graph neural networks approach. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 60179–60196, 2024. URL https://proceedings.mlr.press/v235/zhang24bw.html.

Weijia Zhang, Chenlong Yin, Hao Liu, and Hui Xiong. Unleashing the power of pre-trained language models for irregularly sampled time series. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, 2025. doi: 10.1145/3711896.3737171.

A Notation and Pipeline Pseudocode

Table 2 summarizes the notation. Algorithm 1 describes the training and inference pipeline.

Table 2: Notation used in the method and pseudocode.

Symbol	Meaning
$\mathcal{X}_i$	Historical values, timestamps, and observation masks
$\mathbf{t}_i^+, H_i$	Future query times and their number
$\mathbf{Y}_i, \mathbf{M}_i^Y$	Future targets and target observation mask
V, V_Y	Input and target dimensions
$M, \mathcal{P}_{ip}$	Number of temporal patches and patch p of sample i
$\mathcal{O}_i^Y, N_i^Y$	Observed target set and its size
n_v	Historical observation count for variable v
$\mathcal{V}_{\text{obs}}$	Variables with at least two historical observations
r_v, γ_i	Variable roughness and sample routing index
c, ϵ_r	Index scale and denominator stabilizer
e_p, d_p, a_p	Variance-based energy, density, and deviation
q_p, u_p	Inverse-variance proxy and recency
$\alpha_e, \alpha_d, \beta_a, \beta_q, \beta_d, \lambda_u$	Learnable scoring-calibration coefficients
h_p, ℓ_p, s_p	Two scoring channels and final patch score
$\rho_i, k_{\text{core},i}, \mathcal{C}_i$	Calibrated shared-core ratio, size, and patch set
b_c, b_τ	Positive monotonicity slopes for core and temperature calibrators
$\rho_{\min}, \rho_{\max}, \tau_{\min}, \tau_{\max}$	Fixed bounds of the monotone calibration maps
$r_b, \mathcal{R}_b$	Nominal retention target and retained patch set
$\widetilde{\mathcal{X}}_i^{(b)}$	Perturbed history for branch b
f_θ	Shared forecasting model
L_{ib}, L_i	Branch loss and aggregated sample loss
τ_i, w_{ib}	Temperature and induced branch weight

Algorithm 1 ADDP training and inference pipeline

Require: Histories $\mathcal{X}_i$, future query times $\mathbf{t}_i^+$, targets $\mathbf{Y}_i$, and masks $\mathbf{M}_i^Y$
Require: Shared predictor f_θ, learnable calibrator parameters ψ, and backbone configuration
Ensure: Updated forecasting and calibrator parameters (θ, ψ)

for each training batch **do**
  for each sample i with observed targets **do**
    Compute r_v and γ_i using Equations 2–3
    Compute patch statistics e_p, d_p, a_p, q_p, u_p
    Compute learnable scoring weights using Equation 8
    Compute s_p using Equation 11
    Compute ρ_i and select $\mathcal{C}_i$ using Equations 13–14
    for $b = 1, 2$ **do**
      Initialize $\mathcal{R}_{ib} \leftarrow \mathcal{C}_i$
      Sample additional non-core patches toward nominal target r_b
      Construct $\widetilde{\mathcal{X}}_i^{(b)}$ by replacement and masking
      Predict $\widehat{\mathbf{Y}}_i^{(b)} \leftarrow f_\theta(\widetilde{\mathcal{X}}_i^{(b)}, \mathbf{t}_i^+)$
      Compute L_{ib} using Equation 15
    end for
    Compute calibrated τ_i and L_i using Equations 17–18
  end for
  Update the predictor and learnable calibrators using the training objective
end for
At inference, predict $f_\theta(\mathcal{X}_i, \mathbf{t}_i^+)$ from the original history

B Dataset-Specific Implementation Details

B.1 ADDP Configuration

The four configurations use nominal retention targets $(r_1, r_2) = (0.8, 0.4)$. All three calibrator groups are included in the optimizer for each dataset. The routing statistic and scoring settings are specified below.

Table 3 lists the forecasting-backbone configurations.

Table 3: ADDP backbone configurations. Activity uses the full training budget; MIMIC-III enables early stopping from epoch 25 with patience 10.

Setting	**PhysioNet**	**Activity**	**USHCN**	**MIMIC-III**
Hidden dimension	64	32	32	64
Model layers	1	1	1	2
Attention heads	2	2	2	4
Transformer layers	1	1	1	2
Graph hops	2	1	1	2
Batch size	32	32	64	8
Maximum epochs	50	50	50	50
Early-stopping patience	10	None	10	10 (from epoch 25)

B.2 Forecasting Backbone Comparison

We evaluate ADDP with iTransformer and GRU-D on PhysioNet and Activity. With iTransformer, ADDP obtains MSE 0.003900 and 0.002287 on the two datasets, respectively; with GRU-D, it obtains 0.00427676 and 0.00245714. The iTransformer backbone reduces MSE relative to GRU-D by 8.81% on PhysioNet and 6.92% on Activity. Table 5 reports the five-seed MSE and MAE for each backbone. Each result is the mean and standard deviation over five seeds.

Table 4 pairs each Activity backbone with its corresponding ADDP result.

Table 4: Activity backbone applicability. Relative reductions compare each ADDP result with the corresponding unmodified backbone.

Backbone	Baseline MSE	ADDP MSE	Reduction
iTransformer	0.002930	0.002287 ± 0.000009	21.95%
GRU-D	0.003280	0.00245714 ± 0.000003	25.09%
t-PatchGNN	0.003050	0.002287 ± 0.000009	25.02%

Table 5: ADDP with iTransformer and GRU-D forecasting backbones. Results are reported as mean $\pm$ standard deviation over five seeds.

Dataset	Backbone	Test MSE ↓	Test MAE ↓
PhysioNet	iTransformer	0.003900 ± 0.000007	0.03400212 ± 0.00004
PhysioNet	GRU-D	0.00427676 ± 0.000005	0.03799358 ± 0.00003
Activity	iTransformer	0.002287 ± 0.000009	0.02920952 ± 0.00001
Activity	GRU-D	0.00245714 ± 0.000003	0.03102945 ± 0.00002

B.3 PhysioNet Selection Strategies

Table 6 compares four history-selection strategies on PhysioNet. ADDP obtains the lowest MSE, reducing it by 8.31%, 3.02%, and 6.99% relative to full history, random pruning, and fixed pruning, respectively.

Table 6: PhysioNet history-selection strategies. Results are reported as mean ± standard deviation over five seeds.

Strategy	Test MSE ↓	Test MAE ↓
Full history	0.00421 ± 0.000001	0.03351 ± 0.00003
Random pruning	0.00398 ± 0.000002	0.03578 ± 0.00002
ADDP adaptive pruning	0.003860 ± 0.000050	–
Fixed pruning	0.00415 ± 0.000002	0.03564 ± 0.00003

C Properties of the Training Objective

Equation 18 is equivalent to

$$L_i = \tau_i \log \left[\frac{e^{L_{i1}/\tau_i} + e^{L_{i2}/\tau_i}}{2} \right]. \tag{21}$$

For $\tau_i > 0$, Jensen's inequality gives

$$\frac{L_{i1} + L_{i2}}{2} \le L_i \le \max(L_{i1}, L_{i2}). \tag{22}$$

The upper bound follows because each exponential is at most $\exp(\max_b L_{ib}/\tau_i)$. In addition,

$$\max_b L_{ib} - \tau_i \log 2 \le L_i, \tag{23}$$

so the objective approaches the maximum branch loss as the temperature approaches zero.

The ratio between branch weights is

$$\frac{w_{i1}}{w_{i2}} = \exp\left(\frac{L_{i1} - L_{i2}}{\tau_i} \right). \tag{24}$$

Consequently, the relative loss difference $(L_{i1} - L_{i2})/\tau_i$ determines the strength of worst-view weighting. When this quantity is close to zero, the two branches receive nearly equal weight.

D Complete Baseline Comparison

Table 7 reports the complete comparison with all 26 baselines. The evaluation protocol and scaling follow Table 1.

E Additional Experimental Details

E.1 Dataset Tasks

Activity uses four-second windows with a 500 ms stride, producing 10,739 instances. The source contains 25 recordings from five people wearing four three-dimensional localization tags (Vidulin et al., 2010). The processed representation contains 12 coordinate channels.

For PhysioNet, the historical interval is $[0\text{h}, 24\text{h}]$ and the forecast interval is $(24\text{h}, 48\text{h}]$. The 41-dimensional input representation includes the processed input features; the target mask identifies the available forecasting measurements. The source distinguishes time-varying measurements from general descriptors (Silva et al., 2012).

For MIMIC-III, the historical interval is $[0\text{h}, 24\text{h})$ and the forecast interval is $[24\text{h}, 48\text{h})$. The processed representation contains 96 variables.

The USHCN experiment uses five processed channels: mean, maximum, and minimum temperature, precipitation, and snowfall. The task predicts the following month.

E.2 Adaptive Selection and View Weighting

Table 9 gives the PhysioNet values underlying Figure 3(a).

Table 7: Complete forecasting comparison across four datasets. Red and blue mark the lowest and second-lowest MSE, respectively.

Algorithm	**PhysioNet** (24h → 24h)	**MIMIC-III** (24h → 24h)	**Activity** (4s → 5steps)	**USHCN** (24months → 1month)
	MSE (10^{-3}) ↓	MSE (10^{-2}) ↓	MSE (10^{-3}) ↓	MSE (10^{-1}) ↓
GRU-D	6.18 ± 0.42	11.8 ± 1.50	3.28 ± 0.35	5.267 ± 0.189
Latent ODE	6.05 ± 0.38	11.5 ± 1.40	3.18 ± 0.31	5.189 ± 0.165
PatchTST	5.78 ± 0.29	11.3 ± 1.30	2.98 ± 0.25	5.091 ± 0.138
iTransformer	5.71 ± 0.27	11.2 ± 1.30	2.93 ± 0.23	5.068 ± 0.129
t-PatchGNN	5.92 ± 0.31	11.2 ± 1.20	3.05 ± 0.28	5.123 ± 0.142
ReIMTS	5.04 ± 0.41	18.95 ± 2.24	2.64 ± 0.30	6.051 ± 0.319
TFMixer	4.15 ± 0.36	14.36 ± 1.81	2.57 ± 0.28	5.291 ± 0.238
HyperIMTS	4.95 ± 0.44	20.33 ± 2.56	2.53 ± 0.26	5.496 ± 0.285
APN	4.06 ± 0.10	10.14 ± 0.96	2.62 ± 0.93	5.5834 ± 0.015
DLinear	41.86 ± 0.05	14.93 ± 0.09	4.03 ± 0.01	6.21 ± 0.00
TimesNet	16.48 ± 0.11	14.71 ± 0.12	3.12 ± 0.01	5.58 ± 0.05
Crossformer	6.66 ± 0.11	12.36 ± 0.10	4.29 ± 0.20	5.25 ± 0.04
Graph WaveNet	6.04 ± 0.28	12.76 ± 0.33	2.89 ± 0.03	5.29 ± 0.04
MTGNN	6.26 ± 0.18	13.04 ± 0.14	3.03 ± 0.03	5.39 ± 0.05
StemGNN	6.86 ± 0.28	13.78 ± 0.33	8.81 ± 0.37	5.75 ± 0.09
CrossGNN	7.22 ± 0.36	14.69 ± 0.12	3.03 ± 0.10	5.66 ± 0.04
FourierGNN	6.84 ± 0.35	14.21 ± 0.18	2.99 ± 0.02	5.82 ± 0.06
SeFT	9.22 ± 0.18	10.4 ± 0.22	12.2 ± 0.17	5.8 ± 0.19
RainDrop	9.82 ± 0.08	13.69 ± 0.34	14.92 ± 0.14	5.78 ± 0.22
Warpformer	5.94 ± 0.35	11.76 ± 0.13	2.79 ± 0.04	5.25 ± 0.05
mTAND	6.23 ± 0.24	11.1 ± 0.09	3.22 ± 0.07	5.33 ± 0.05
CRU	8.56 ± 0.26	11.07 ± 0.08	6.97 ± 0.78	6.09 ± 0.17
Neural Flow	7.2 ± 0.07	13.12 ± 0.05	4.05 ± 0.13	5.35 ± 0.05
GraFITi	5.11 ± 0.19	10.49 ± 0.13	3.03 ± 0.14	5.07 ± 0.03
ISTS-PLM	4.92 ± 0.05	10.08 ± 0.23	2.58 ± 0.03	5.23 ± 0.07
Hi-Patch	4.86 ± 0.03	9.92 ± 0.14	2.57 ± 0.02	5.14 ± 0.05
ADDP	3.860 ± 0.050	9.419 ± 0.020	2.287 ± 0.009	4.954 ± 0.040

Table 8: Dataset representations and forecasting tasks. Observation rates refer to the processed inputs. Input dimensions and prediction channels are distinguished in Section 3.1.

Dataset	**Input representation**	**Forecasting task**
PhysioNet	4,000 records; 41 input dimensions; 23.4% observed	First 24 hours to the subsequent 24 hours
Activity	12 coordinate channels; 10,739 windows	Five-step forecasting from four-second windows
USHCN	1,218-station collection; five processed channels	Next-month forecasting
MIMIC-III	96 processed variables; 8.7% observed	First 24 hours to the subsequent 24 hours

Table 9: PhysioNet adaptive-selection and view-weighting ablation.

Configuration	Test MSE	Increase over ADDP
Full ADDP	**0.003860**	–
Adaptive ranking + fixed temperature	0.003916	1.45%
Adaptive ranking + mean loss	0.003950	2.33%
Random dual views + mean loss	0.004107	6.40%

E.3 History-Dependent Routing Signals

Table 10 reports the routing-signal comparison shown in Figure 3(b).

Table 10: PhysioNet routing-signal comparison over repeated runs.

Routing signal	Test MSE (10^{-3})	Increase
History-derived γ	$\mathbf{3.86 \pm 0.05}$	–
Fixed $\gamma = 0.5$	4.85 ± 0.43	25.65%
Linear-decay γ	4.92 ± 0.45	27.46%
Random γ	5.41 ± 0.52	40.16%

E.4 Coordinating Adaptive Decisions

Table 11 gives the Activity values for Figure 3(c). Within this ablation, the shared-γ configuration achieves an MSE of 0.002280.

Table 11: Activity coordination ablation, with the shared-γ configuration as reference.

Configuration	Test MSE	Increase
Shared-γ full	**0.002280**	–
Independent signals	0.002331	2.24%
Scoring-only adaptation	0.002348	2.98%
Temperature-only adaptation	0.002356	3.33%
Core-only adaptation	0.002371	3.99%

E.5 PhysioNet Component Ablation

Table 12 reports the PhysioNet component ablations, using ADDP's main result as the reference.

Table 12: PhysioNet component analyses. Relative changes are computed against ADDP in Table 1.

Variant	**MSE** (10^{-3}) $\downarrow$	**Change**
ADDP	3.86 ± 0.05	–
Scoring and routing variants		
Single fixed scoring channel	4.74 ± 0.45	+22.8%
Replace energy with squared deviations	4.58 ± 0.42	+18.7%
Remove recency	4.42 ± 0.39	+14.5%
Remove continuity proxy	4.33 ± 0.40	+12.2%
SD-based replacement index	4.38 ± 0.41	+13.5%
Hard-threshold variant	4.35 ± 0.39	+12.7%
Architectural diagnostics		
Single branch with standard MSE	6.60	+71.0%
Alternative index using SD, mean, and min–max	4.40	+14.0%
Independent component-specific indices	4.37	+13.2%

E.6 Metric Ablation

We evaluate the five scoring terms on Activity. The complete score achieves 0.002287 ± 0.000009 MSE. Table 13 reports the five-term ablation.

Removing any term increases test MSE, with the largest changes for density, deviation, and recency.

E.7 Diagnostic Definitions

We examine whether reducing available history changes the optimization signal and attention distribution of an iTransformer on Activity. The first diagnostic measures the attention-parameter gradient

Table 13: Activity patch-scoring term ablation over five runs. MSE is multiplied by 10^3; changes are relative to the complete score.

Variant	MSE $\times 10^3$	MSE increase
Complete score	$\mathbf{2.287 \pm 0.009}$	–
Remove energy	2.350 ± 0.096	2.76%
Remove inverse variance	2.367 ± 0.073	3.50%
Remove density	2.416 ± 0.106	5.64%
Remove deviation	2.410 ± 0.102	5.38%
Remove recency	2.409 ± 0.103	5.33%

norm after one forward–backward pass at initialization. The second measures attention entropy after training on the full reference history. Table 14 reports the two measurements separately.

Table 14: Activity observation-removal diagnostics with iTransformer. Gradient norms are measured at initialization; attention entropy is measured during evaluation of a model trained on the full reference input. N_{eff} is the entropy-based effective attention count.

Retained	**Gradient norm**	H_{norm}	N_{eff}
100%	4.61 ± 0.78	0.994	11.82
60%	4.02 ± 0.60	0.992	11.76
40%	3.87 ± 0.65	0.995	11.85
20%	3.19 ± 0.77	0.710	5.84
10%	1.46 ± 0.60	0.214	1.70

Reducing the retained fraction from 100% to 10% lowers the initialization gradient norm from 4.61 to 1.46. Normalized attention entropy stays near 0.99 at 40%–100% retention, then falls to 0.710 at 20% and 0.214 at 10%. Lower entropy means more concentrated attention. These diagnostics characterize how history retention affects the initialization gradient norm and attention distribution.

For an attention distribution over $N = 12$ variable tokens, define

$$H(A) = -\sum_{j=1}^{N} a_j \log a_j, \qquad H_{\text{norm}} = \frac{H(A)}{\log N}. \tag{25}$$

The entropy-based effective attention count is

$$N_{\text{eff}} = \exp(H(A)) = N^{H_{\text{norm}}}. \tag{26}$$

Table 14 computes N_{eff} from the displayed normalized entropies. Exponentiating an averaged entropy gives a geometric summary of effective attention counts.

Both diagnostics use a four-layer, eight-head iTransformer with hidden dimension 256. Gradient norms are measured after one forward–backward pass at initialization and summarized over five random seeds. Attention entropy is evaluated after training on the full reference input. The evaluation input is then reduced to the retained fractions listed in Table 14.

F Supplementary Calibration and Evaluation Details

F.1 Full Retention Sweep

We vary the nominal conservative and aggressive retention targets on Activity. Table 15 reports the MSE for each configuration. On PhysioNet, the default $(0.8, 0.4)$ setting achieves MSE 0.003860 (Table 1). On Activity, the default $(0.8, 0.4)$ gives test MSE 0.002287 ± 0.000009. The $(0.9, 0.5)$ setting gives validation MSE 0.00253334 and test MSE 0.00242555. The nominal targets specify branch-level patch retention, while the realized shared-core size is determined by $\rho(\gamma)$. Table 16 reports the validation errors and MAE.

F.2 Full Routing-Scale Sweep

We vary the roughness normalization constant $c \in \{0.5, 1.0, 1.5, 2.0, 3.0\}$ on PhysioNet. Test MSE ranges from 0.003989 to 0.004095, with the lowest value at $c = 1.0$. The default $c = 1.5$ gives

Table 15: Activity sensitivity to nominal retention targets under the controlled sweep. Relative changes use the default $(0.8, 0.4)$ configuration as the reference.

(r_1, r_2)	**MSE** $\downarrow$	**Change**
$(0.7, 0.3)$	0.002552	$+11.57\%$
$(0.7, 0.5)$	0.002480	$+8.45\%$
$(0.8, 0.4)$	$\mathbf{0.002287} \pm 0.000009$	–
$(0.9, 0.3)$	0.002458	$+7.48\%$
$(0.9, 0.5)$	0.002426	$+6.06\%$

Table 16: Activity retention sweep without metric rescaling.

(r_1, r_2)	Validation MSE	Test MSE	Test MAE
$(0.7, 0.3)$	0.00243148	0.00255155	0.03079948
$(0.7, 0.5)$	0.00237760	0.00248023	0.03061680
$(0.8, 0.4)$	0.00245509	0.002287 ± 0.000009	0.03067239
$(0.9, 0.3)$	0.00236609	0.00245810	0.02989415
$(0.9, 0.5)$	0.00253334	0.00242555	0.02957404

0.004037. Performance varies modestly across the tested settings on PhysioNet. Table 17 lists the validation and test errors. Each configuration reports one run.

Table 17: PhysioNet sensitivity to roughness scale c.

c	Validation MSE	Test MSE
0.5	0.003918	0.004074
1.0	**0.003825**	**0.003989**
1.5	0.003896	0.004037
2.0	0.003884	0.004095
3.0	0.003909	0.004083

F.3 Interpreting the Smooth Worst-View Loss

For fixed positive τ_i, differentiating the sample objective gives $\partial L_i / \partial L_{ib} = w_{ib}$. For two views, their relative weighting has the simple form

$$w_{i1} = \sigma\left(\frac{L_{i1} - L_{i2}}{\tau_i}\right), \qquad w_{i2} = 1 - w_{i1}. \tag{27}$$

Equal losses receive equal weight, and a loss difference much smaller than τ_i gives nearly equal weights. As $\tau_i \to \infty$, the objective approaches the mean of the two losses. As $\tau_i \to 0^+$, it approaches their maximum.

The objective is an instance of exponential tilting (Li et al., 2021). Its variational form is

$$L_i = \max_{\mathbf{w} \in \Delta_2} \left\{ \sum_{b=1}^{2} w_b L_{ib} - \tau_i D_{\mathrm{KL}}(\mathbf{w} \| \mathbf{u}) \right\}, \qquad \mathbf{u} = (1/2, 1/2), \tag{28}$$

where Δ_2 is the probability simplex over the two views. To verify the identity, set $w_b^* = \exp(L_{ib}/\tau_i) / \sum_{b'} \exp(L_{ib'}/\tau_i)$. The expression in braces equals $L_i - \tau_i D_{\mathrm{KL}}(\mathbf{w} \| \mathbf{w}^*)$, which is maximized at $\mathbf{w}^*$. The objective therefore reweights the two constructed losses according to their relative magnitudes.

F.4 Metric Aggregation and Reporting

The per-sample training loss in Equation 1 is distinct from aggregation over a complete evaluation set. For MIMIC-III, let E_v and N_v denote total squared error and observed-target count for channel v. The reported metric is

$$\mathrm{MSE}_{\mathrm{entry}} = \frac{\sum_v E_v}{\sum_v N_v}. \tag{29}$$

The Activity score-term study uses channel-macro MSE:

$$\mathrm{MSE}_{\mathrm{channel}} = \frac{1}{|\{v : N_v > 0\}|} \sum_{v:N_v>0} \frac{E_v}{N_v}. \tag{30}$$

MAE uses the same aggregation with absolute errors.

Table 1 reports forecasting MSE across the four datasets. Table 12 uses the PhysioNet ADDP result $(3.86 \pm 0.05) \times 10^{-3}$ as its baseline, with relative changes computed as $(\mathrm{MSE}_{\mathrm{variant}}/0.003860 - 1) \times 100\%$. The Activity retention sweep is reported in Appendix F.1.

G LIMITATIONS

Dual-view training introduces additional training computation, and masking does not reduce attention complexity without token compaction. At inference, ADDP uses the original history and a single forecasting pass.

REPRODUCIBILITY STATEMENT

Section 3 defines the prediction task, routing index, constrained calibrators, view construction, and training objective. Appendix A provides notation and pseudocode. Appendix B lists forecasting-backbone configurations and backbone comparisons. Appendix E describes dataset tasks, and Appendix F.4 defines metric aggregation.

ETHICS STATEMENT

This work studies forecasting methodology using activity, climate, and de-identified clinical data. The use of MIMIC-III remains subject to its data-access conditions (Johnson et al., 2016). The evaluation focuses on predictive performance; clinical deployment and patient-subgroup analysis are outside the scope of this work.

AI USE STATEMENT

Generative AI tools assisted language revision, LaTeX organization, consistency checking, reference preparation, and pseudocode drafting. The authors are responsible for the final manuscript, citations, experimental results, and methodological descriptions.