SAE^{2}F: STATE-AWARE EXTREME EVENT FORECASTING FOR MULTIVARIATE TIME SERIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Extreme values are often treated as isolated tail observations, although the dynamics leading to an extreme event may change substantially before the event occurs. However, existing extreme-event forecasting methods mainly rely on rare-sample modeling and tail-supervision optimization, making it difficult to characterize system state transitions and evolving inter-variable dependency structures before extreme events occur. To address this issue, we propose SAE^{2}F, a state-aware enhanced framework for extreme-event forecasting. The framework combines local stochastic-dynamics modeling with critical slowing down theory to detect precursor states before extreme values appear, and injects the resulting risk-state information into subsequent forecasting. Furthermore, a Lyapunov consistency constraint converts the second-order statistical reference of local stochastic dynamics into a differentiable training constraint, while a unified risk-state variable α is used to regulate dependency structures and match similar extreme-event patterns, thereby enabling state-conditioned forecasting. Experiments show that SAE^{2}F consistently improves extreme-event F1 and extreme-point MAE while maintaining competitive overall forecasting performance. The code is available at `https://anonymous.4open.science/status/SAEEF-2026`.

1 INTRODUCTION

Extreme-event forecasting is often treated as a rare-tail learning problem, and many extreme-aware methods strengthen tail supervision through reweighting, tail losses, transfer learning, or explicit extreme labels Ding et al. (2019); Li et al. (2023; 2024); Chen et al. (2023); Shi et al. (2024); Li & Anastasiu (2025). This is necessary but leaves a central gap: an extreme phase is not merely an output value crossing a threshold; it may correspond to coordinated changes in system state, cross-variable dependency, and local dynamics. If the base model applies the same state representation to both regimes, stronger tail supervision alone may still fail to capture these structural changes.

Core claim. Extreme-event forecasting should model both the output tail and the precursor state process before the system enters a high-risk region. The model should estimate from the input window whether the system is approaching a high-risk state and use that estimate to adjust dependency modeling and extreme-pattern matching, without replacing the base forecasting capability but by adding an interpretable risk-state variable α as a unified state interface within SAE^{2}F.

To ground α, we use local stochastic dynamics to describe window-level evolution and critical slowing down to explain why high-risk precursor states produce observable features such as variance amplification, lag-1 autocorrelation increase, and tail-pressure growth. The discrete Lyapunov relation organizes these second-order statistics into a unified dynamical reference, making α a risk variable calibrated by a local dynamical state prior, window context, and extreme supervision Held & Kleinen (2004); Dakos et al. (2008); Scheffer et al. (2009); Dakos et al. (2012); Proverbio et al. (2023); Masuda et al. (2024).

As illustrated in Fig. 1, extreme forecasting involves more than the sparsity of tail labels. The left panel shows that the risk variable α rises and exceeds the risk threshold *before* the extreme-value peak, demonstrating that precursor-state evidence in the input window is detectable prior to tail-label supervision. The right panel shows that different extreme events exhibit heterogeneous temporal patterns, motivating relation-scored prototype matching to handle event-family diversity. Together,

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.

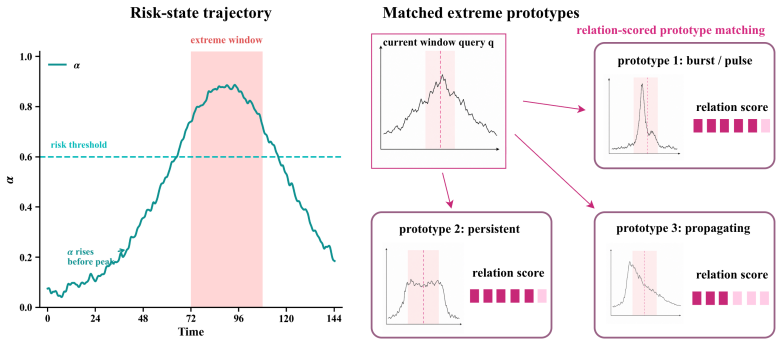


Figure 1: Schematic of the two core mechanisms in SAE^{2}F. **Left:** α crosses the risk threshold *before* the extreme-value peak, moving risk assessment earlier than tail supervision. **Right (schematic):** the current window is scored against representative extreme-event pattern families, handling event heterogeneity via SCPM.

these observations motivate our use of a unified risk-state interface rather than treating extreme detection as a post-hoc label attached to the forecast.

Difference from existing methods. General forecasting models such as PatchTST, iTransformer, TimeMixer, and DUET optimize overall error and representation capacity without explicitly separating normal and extreme regimes Nie et al. (2023); Liu et al. (2024); Wang et al. (2024); Qiu et al. (2025). The extreme-aware methods cited above strengthen tail signals through rare-event modeling, probability-enhanced objectives, polar representations, transfer learning, or fine-tuning, but mostly use a unified feature space across regimes. Statistical-physics and early-warning studies help identify critical-transition signals, yet these signals still need to be further learned and integrated into a trainable state interface for forecasting models Fan et al. (2021); Bury et al. (2021); Raissi et al. (2019). SAE^{2}F bridges this gap by providing state evidence for α via local stochastic dynamics and Lyapunov references, then injecting α into dependency switching and prototype matching; see Appendix A for expanded related work and Appendix B, Remark 1, for the approximation conditions. Due to page limitations, expanded related work, derivations, and additional experimental evidence are provided in the appendix.

This paper focuses on making the precursor-state process explicit and trainable within a forecasting model. We therefore use a local linear stochastic approximation to describe window-level dynamics and assume that this approximation is reasonable over the short input windows considered here. The approximation is not intended to replace the base model; rather, it provides a compact state prior and a second-order statistical reference that can be jointly optimized with forecasting and extreme-event losses.

Contributions. (1) Earlier risk assessment: a unified risk-state variable α is constructed from local stochastic dynamics and critical slowing down, enabling high-risk precursor states to be sensed before extreme values appear. **(2) Lyapunov consistency:** a differentiable constraint keeps α consistent with observable variance-autocorrelation relations. **(3) State-conditioned prototype matching:** query-prototype relation scoring invokes more relevant extreme-event prototypes in high-risk regimes.

2 Method: SAE^{2}F

2.1 Problem Definition and Overall Framework

Given a historical multivariate time series $\mathbf{X} \in \mathbb{R}^{L\times N}$, the goal is to predict the future sequence $\hat{\mathbf{Y}} \in \mathbb{R}^{H\times N}$ and identify extreme events in the prediction horizon. To construct event labels for heterogeneous channels, we first apply channel-wise normalization using statistics estimated from the training split:

$$\bar{Y}_{h,n} = \frac{Y_{h,n} - \mu_n}{\sigma_n + \epsilon}, \tag{1}$$

where μ_n and σ_n are the training-split mean and standard deviation of channel n, and $\epsilon > 0$ is a numerical constant. We then aggregate the normalized multivariate future response at each forecast

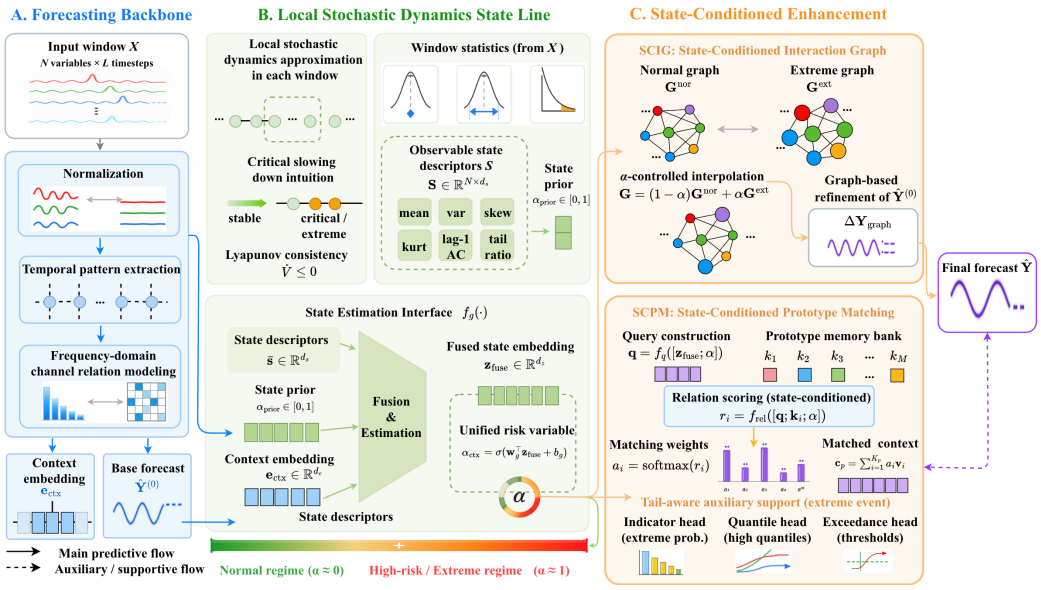


Figure 2: SAE^{2}F framework overview, consisting of three components connected through the unified risk variable α: (1) the forecasting backbone, (2) the local stochastic-dynamics state line, and (3) the state-conditioned enhancement module (SCIG and SCPM).

step as

$$\tilde{Y}_h = \max_{1 \le n \le N} \bar{Y}_{h,n}, \tag{2}$$

The h-th future step is then defined as an extreme timestep if

$$\tilde{Y}_h \ge \tau_{\text{ext}}^{(q)}, \tag{3}$$

where $\tau_{\text{ext}}^{(q)}$ is the q-quantile threshold computed from the normalized pooled future responses used for the current label construction Coles (2001). The binary label at position h is $I_h = \mathbf{1}[\tilde{Y}_h \ge \tau_{\text{ext}}^{(q)}]$. This definition marks an upper-tail event when any channel enters a relatively high normalized state; it does not assert that all channels jointly form one physical event. A future window is considered an extreme window if it contains at least one extreme timestep. Accordingly, the present study concerns threshold-defined upper-tail events in continuously observed, regularly sampled multivariate series; discrete semantic events require domain-specific labels and are outside the validated scope.

Base forecasting module. SAE^{2}F consists of a base forecasting module, a state-modeling line, a unified risk-state estimator, and two state-conditioned enhancement modules (SCIG and SCPM). The base module produces the initial forecast, the state line provides descriptors and α_{prior}, the estimator outputs α, and the downstream interfaces use α to revise dependency structure and match extreme-event prototypes. Thus, the state line is an internal state-aware modeling path rather than a post-hoc detector. The base module contains temporal-pattern extraction, frequency-domain channel-relation modeling, and masked-attention interaction:

$$\mathbf{H}_{\text{temp}} = f_{\text{temp}}(\mathbf{X}), \tag{4}$$

$$\mathbf{M}_{\text{freq}} = f_{\text{freq}}(\mathbf{X}), \tag{5}$$

$$\hat{\mathbf{Y}}^{(0)} = f_{\text{pred}}(f_{\text{attn}}(\mathbf{H}_{\text{temp}}, \mathbf{M}_{\text{freq}})). \tag{6}$$

Here, $\mathbf{H}_{\text{temp}}$ uses sparsely gated linear-pattern experts Shazeer et al. (2017), $\mathbf{M}_{\text{freq}}$ is a frequency-domain channel-relation mask encouraging spectral-similar channels to interact, and $\hat{\mathbf{Y}}^{(0)}$ is the base forecast. Specifically, f_{temp}, f_{freq}, f_{attn}, and f_{pred} denote the temporal extractor, frequency-mask constructor, masked-attention fusion, and forecast head, respectively.

As shown in Fig. 2, the three columns of SAE^{2}F are connected by α: state estimation, dependency-structure correction, and extreme-prototype matching are all conditioned on the same risk variable.

2.2 From Local Stochastic Dynamics to State-Description Features and Dynamical Constraints

State modeling box. This subsection organizes window-level statistical changes into a unified state-modeling module, whose role is to provide an interpretable dynamical reference for the question "has the current window entered a high-risk precursor state?" The input is the multivariate sequence in the current sliding window. Internally, local stochastic dynamics and Lyapunov second-order relations organize variance, autocorrelation, distributional asymmetry, and tail pressure into observable state-description features. The module provides three quantities to the downstream state estimator: observable state-description features $\mathbf{S}$, a state proxy prior α_{prior}, and a Lyapunov consistency constraint $\mathcal{L}_{\text{lyap}}$. Consequently, the variance, autocorrelation, and tail-pressure terms introduced below all serve the same risk-state variable α rather than acting as separate heuristic indicators.

Local stochastic-dynamics approximation. Why introduce a local stochastic-dynamics model before defining neural modules? The reason is that precursor states of extreme events often appear as changes in local recovery speed, volatility structure, and short-range dependence inside the input window, whereas these changes cannot be explained solely by whether future outputs cross a threshold. We therefore adopt a window-level local-linearization view and model the current dynamics within each sliding window as a local linear stochastic process:

$$\mathbf{x}_{t+1} = A_t\,\mathbf{x}_t + \boldsymbol{\varepsilon}_t, \quad \boldsymbol{\varepsilon}_t \sim \mathcal{N}(\mathbf{0}, Q), \tag{7}$$

where $\mathbf{x}_t \in \mathbb{R}^N$ is the centered window state, $A_t \in \mathbb{R}^{N\times N}$ is the local effective coupling matrix, and $Q \succ 0$ is the noise covariance. Here $\lambda_i(A_t)$ denotes the i-th eigenvalue of A_t. Local stability is characterized by $\rho(A_t) = \max_i |\lambda_i(A_t)|$; when $\rho(A_t) \to 1$, the window approaches a near-critical state consistent with critical slowing down Held & Kleinen (2004); Dakos et al. (2008); Scheffer et al. (2009). In practice we do not solve for A_t explicitly but extract window-level statistical proxies carrying the same state meaning. The local stochastic-dynamics model thus moves risk assessment earlier: from future threshold crossing to precursor-state evidence in the input window.

Lyapunov relation and consistency constraint. For a locally stable process, stationary second-order statistics satisfy the discrete Lyapunov equation:

$$\Sigma = A\,\Sigma\,A^\top + Q, \tag{8}$$

where $\Sigma = \mathbb{E}[\mathbf{x}_t\mathbf{x}_t^\top]$ and A is the effective evolution operator. $\mathcal{L}_{\text{lyap}}$ converts this variance-autocorrelation relation into a differentiable training constraint, keeping α consistent with the local dynamical reference rather than drifting freely under extreme-label supervision. The motivation for adding this relation is that, once α is interpreted as a risk-state variable, it should not be learned only as a free latent score driven by sparse extreme labels. It should also remain compatible with the second-order statistics observed in the window. The Lyapunov equation provides such a structural reference by linking the effective evolution operator, noise covariance, and stationary covariance. For the neural model, this relation is not used to solve a physical system exactly; it is converted into a lightweight differentiable constraint that regularizes the state representation around a local-dynamics reference surface.

Under a diagonal approximation, the Lyapunov relation yields three directly observable state-description features:

$$\textbf{State feature 1 (variance amplification):} \quad \Sigma_{nn} \approx \tfrac{Q_{nn}}{1-A_{nn}^2} \xrightarrow{\rho(A)\to 1} +\infty, \tag{9}$$

$$\textbf{State feature 2 (lag-1 autocorrelation increase):} \quad \rho_n^{(1)} \approx A_{nn} \xrightarrow{\rho(A)\to 1} 1, \tag{10}$$

$$\textbf{State feature 3 (Lyapunov consistency):} \quad \Sigma_{nn}(1-[\rho_n^{(1)}]^2) \approx Q_{nn}. \tag{11}$$

Eqs. (9)–(11) show how variance, autocorrelation, and noise covariance are coupled as $\rho(A) \to 1$; deviations from Eq. (11) serve as a state-diagnosis signal. In implementation, we align each $\sigma_n^2(1-[\rho_n^{(1)}]^2)$ with a shared learnable reference q_{lyap}, distinguished from the event quantile q. Only the variance–autocorrelation relation is directly induced by the Lyapunov derivation; the local mean, skewness, kurtosis, and tail pressure below are complementary observable statistics. The diagonal approximation is used only to obtain a stable scalarized consistency signal; cross-variable dependency changes are modeled separately in SCIG through state-conditioned interaction graphs.

Observable state-description features. For each variable, we define

$$\phi(\mathbf{x}_n) = [\mu_n,\, \sigma_n,\, \mathrm{skew}_n,\, \mathrm{kurt}_n,\, \rho_n^{(1)},\, r_n^{\mathrm{tail}}]. \tag{12}$$

Here, σ_n and $\rho_n^{(1)}$ are the Lyapunov-grounded core quantities from Eqs. (9)–(10); μ_n gives the local operating level; skewness and kurtosis capture non-Gaussian deviation Scheffer et al. (2009); Dakos et al. (2012); and $r_n^{\mathrm{tail}} = \frac{1}{L}\sum_t \mathbf{1}[x_{t,n} > \mu_n + \tau_{\mathrm{tail}}\sigma_n]$, where τ_{tail} is a fixed tail-pressure scale, measures tail pressure after variance amplification.

From descriptors to α_{prior} and $\mathcal{L}_{\mathrm{lyap}}$. The descriptors enter the model in two ways. First, they form a coarse state prior via a fixed proxy mapping:

$$\bar{\mathbf{s}} = \mathrm{Pool}_n(\mathbf{S}_{n,:}), \qquad \alpha_{\mathrm{prior}} = g_{\mathrm{prior}}(\bar{\mathbf{s}}). \tag{13}$$

Here Pool_n is pooling over variables, g_{prior} is the fixed state-proxy mapping, and d_s is the descriptor dimension. Second, the per-variable descriptors are stacked as $\mathbf{S} = [\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_N)] \in \mathbb{R}^{N\times d_s}$, fused with α_{prior} and $\mathbf{e}_{\mathrm{ctx}}$ to estimate α, while $\mathcal{L}_{\mathrm{lyap}}$ is derived from Eq. (11). Thus the state line provides state descriptors, a risk prior, and a dynamical constraint through a single consistent interface, without explicitly estimating A_t or $\rho(A_t)$.

2.3 Estimating the Unified Risk Variable α

α is a learnable risk-state variable that gathers window context, observable state descriptors, and the state proxy prior into a single compact interface, and is not a post-hoc detection score outside the forecasting task.

In implementation, the module uses a lightweight fusion estimator. The model first constructs a window/context embedding $\mathbf{e}_{\mathrm{ctx}}$ from the input window, projects and aggregates the state-description features $\mathbf{S}$, and combines them with the projected state proxy prior α_{prior}. The resulting fused state embedding is used to output $\alpha \in [0, 1]$, which is passed to SCIG and SCPM. $\alpha \to 0$ indicates a normal regime with stable sparse coupling, whereas $\alpha \to 1$ indicates a high-risk or extreme regime with stronger state-dependent interactions:

$$\mathbf{z}_{\mathrm{fuse}} = f_g(\mathbf{S},\ \alpha_{\mathrm{prior}},\ \mathbf{e}_{\mathrm{ctx}}), \qquad \alpha_{\mathrm{ctx}} = \sigma(\mathbf{w}_g^\top \mathbf{z}_{\mathrm{fuse}} + b_g). \tag{14}$$

The final state variable fuses the context path and the state proxy prior:

$$\alpha = \mathrm{clip}\big(\tfrac{1}{2}\alpha_{\mathrm{ctx}} + \tfrac{1}{2}\alpha_{\mathrm{prior}},\ 0,\ 1\big). \tag{15}$$

Here, $\mathbf{e}_{\mathrm{ctx}}$ is the context embedding, f_g is the fusion map, $\mathbf{w}_g$ and b_g are the output-layer weight and bias, σ is the sigmoid function, and $\mathrm{clip}(\cdot, 0, 1)$ clips a scalar to $[0, 1]$. α_{prior} provides a dynamically grounded risk proxy; α_{ctx} is data-driven from window context. Their equal-weight fusion yields α with both dynamical grounding and contextual adaptability. Because α directly controls SCIG and SCPM, it is indirectly calibrated through downstream losses. This equal-weight fusion is deliberately simple. The prior path anchors α to interpretable local-dynamics statistics, while the context path allows the neural model to adapt the risk estimate to the current data window. Since α enters the graph interpolation in SCIG and the query construction and relation scoring in SCPM, it is also calibrated through event-indicator, quantile, exceedance, and prototype-matching losses. The model therefore learns a state variable that is simultaneously grounded by dynamics and useful for downstream prediction.

2.4 SCIG: State-Conditioned Interaction Graph

SCIG is the dependency-switching interface between α and the base forecast. Extreme regimes often involve changed coupling and synchronization patterns; SCIG models this explicitly via two candidate interaction graphs rather than as a state-description feature. The normalized cross-variable correlation provides an observable reference:

$$C_{nm} = \frac{\Sigma_{nm}}{\sqrt{\Sigma_{nn}\Sigma_{mm}} + \epsilon}, \qquad n \neq m. \tag{16}$$

In implementation, SCIG does not introduce a heavy graph learner. Instead, it uses within-window stress intensity to reweight time steps: low-stress steps dominate the normal graph, while high-stress

steps dominate the extreme graph. The two candidate graphs are instantiated from stress-weighted correlation estimates and mixed with the identity matrix to keep the base forecasting path stable. From a local-dynamics view, they can be interpreted as observable approximations of effective dependency structures under normal and high-risk regimes. The unified state variable α then determines which dependency pattern the current forecast should be closer to, via

$$\mathbf{G} = (1-\alpha)\mathbf{G}^{\mathrm{nor}} + \alpha\mathbf{G}^{\mathrm{ext}}. \tag{17}$$

The graph correction is applied as a small residual:

$$\hat{\mathbf{Y}} = \hat{\mathbf{Y}}^{(0)} + \lambda_g \alpha \Big(\mathbf{G}\hat{\mathbf{Y}}^{(0)} - \hat{\mathbf{Y}}^{(0)} \Big). \tag{18}$$

Here, $\lambda_g = 0.05$ is a fixed graph-residual scale, not a loss weight. When $\alpha \to 0$ the model relies on the stable normal graph; when $\alpha \to 1$ it shifts to the extreme graph. The residual form preserves the base predictor: SCIG only applies a small state-conditioned correction scaled by α, matching the implementation design of stress-weighted correlation estimation, identity mixing, α interpolation, and residual graph correction. The current implementation uses dense stress-weighted correlations. Therefore, for a batch size B, input length L, number of variables N, and prediction horizon H, constructing the two graphs costs $O(BLN^2)$ and applying the residual propagation costs $O(BHN^2)$. The identity mixing and small residual scale stabilize the correction and reduce its empirical impact on the base path, but they do not change this quadratic dependence on N.

2.5 SCPM: State-Conditioned Extreme Prototype Matching

SCPM is the state-conditioned prototype matching interface between α and a learnable extreme-event prototype library. Extreme events are sparse and heterogeneous; SCPM borrows from similar historical patterns using metric-learning-style matching. It receives $\mathbf{z}_{\mathrm{fuse}}$ and α, and outputs α-conditioned matching weights, a matched context, and a matched descriptor. The motivation is that extreme events are not only rare but also heterogeneous. Different events may have different peak magnitudes, durations, exceedance masses, and temporal locations of the peak. A single unified forecasting representation may not cover these event families well. SCPM therefore asks a different question from SCIG: given the current high-risk state, which type of historical extreme pattern is the current window closest to?

We define an event descriptor

$$\mathbf{d} = [y^{\max}, \bar{y}, r^{\mathrm{dur}}, m^{\mathrm{exc}}, e^{\max}, t^{\mathrm{peak}}], \tag{19}$$

where $y^{\max}$ is the peak pooled value, $\bar{y}$ the horizon mean, r^{dur} the fraction of extreme timesteps, $m^{\mathrm{exc}} = \sum_h (y_h - \tau)^+$ the exceedance mass, $e^{\max} = \max_h (y_h - \tau)^+$ the peak exceedance, and t^{peak} the normalised peak location. We maintain K_p learnable prototypes $\mathcal{P} = \{(\mathbf{k}_i, \mathbf{v}_i, \mathbf{d}_i)\}_{i=1}^{K_p}$. Here, $\mathbf{k}_i$, $\mathbf{v}_i$, and $\mathbf{d}_i$ are respectively the key, value, and descriptor of prototype i, and K_p is the number of prototypes. These prototypes are centers of extreme-event families in the representation space. SCPM follows a prototype-based metric-learning view Snell et al. (2017); Sung et al. (2018) and uses a lightweight relation scorer rather than fixed nearest-neighbor retrieval. In implementation, the current window first forms a query vector from the fused state embedding and α, which is then scored against normalized prototype keys. The current query is

$$\mathbf{q} = f_q([\mathbf{z}_{\mathrm{fuse}}; \alpha]). \tag{20}$$

Query-prototype relation scores and matching weights are computed as

$$r_i = f_{\mathrm{rel}}([\mathbf{q}; \mathbf{k}_i; \alpha]), \qquad a_i = \frac{\exp(r_i/T_p)}{\sum_j \exp(r_j/T_p)}, \qquad \mathbf{c}_p = \sum_{i=1}^{K_p} a_i \mathbf{v}_i. \tag{21}$$

In Eq. (20), f_q is the query projection; in Eq. (21), f_{rel} is the relation scorer and $T_p > 0$ is the softmax temperature. α enters both query construction and relation scoring, changing $\mathbf{q}$, r_i, and a_i under different risk states. The matched context supports tail-aware auxiliary heads, and the prototype loss shapes the prototype space to provide contextual enhancement and semantic structure for extreme-event families. Thus SCPM is not a post-hoc memory lookup module. It is a state-conditioned metric-learning interface: the same window may attend to different prototypes when α changes,

and the prototype loss encourages the library to carry semantic information about extreme-event families rather than acting as an auxiliary hidden vector. A controlled scan that holds $\mathbf{z}_{\text{fuse}}$ fixed while varying only α is reported in Appendix E, Table 14. With K_p prototypes and hidden dimension D, relation scoring in SCPM costs $O(BK_pD)$, and the final weighted prototype aggregation has the same order. Since K_p is small in all experiments, this term is minor compared with the dense SCIG graph operation on high-dimensional datasets such as Traffic. Measured parameter, memory, and inference-time overhead is reported in Appendix E, Table 12.

2.6 Multi-Objective Extreme-Event Learning

The complete training objective is

$$\mathcal{L} = \mathcal{L}_{\text{point}} + \lambda_{\text{ext}}\mathcal{L}_{\text{ext}} + \lambda_{\text{proto}}\mathcal{L}_{\text{proto}} + \lambda_{\text{lyap}}\mathcal{L}_{\text{lyap}}. \tag{22}$$

$\mathcal{L}_{\text{lyap}}$ is active from the first epoch as a structural regularizer; $\mathcal{L}_{\text{ext}}$ and $\mathcal{L}_{\text{proto}}$ are introduced progressively via warmup epochs (see Appendix F, Table 18). Any auxiliary regularizer from the selected base forecaster is absorbed into $\mathcal{L}_{\text{point}}$. For completeness, the point term is the normalized ℓ_1 error plus any backbone regularizer, while the three auxiliary terms are

$$\mathcal{L}_{\text{point}} = \frac{1}{BHN}\sum_{b,h,n} |\hat{y}_{b,h,n} - y_{b,h,n}| + \mathcal{L}_{\text{base}}, \tag{23}$$

$$\mathcal{L}_{\text{ext}} = \mathcal{L}_{\text{ind}} + \mathcal{L}_q + \mathcal{L}_{\text{exc}}, \tag{24}$$

$$\mathcal{L}_{\text{proto}} = \mathcal{L}_{\text{align}} + \mathcal{L}_{\text{desc}}, \tag{25}$$

$$\mathcal{L}_{\text{lyap}} = \frac{1}{N}\sum_{n=1}^{N}\left(\hat{\sigma}_n^2(1 - [\hat{\rho}_n^{(1)}]^2) - q_{\text{lyap}}\right)^2. \tag{26}$$

Here $\mathcal{L}_{\text{ind}}$ is the pooled event-indicator BCE, $\mathcal{L}_q$ the upper-quantile pinball term, $\mathcal{L}_{\text{exc}}$ the pooled exceedance regression, and $\mathcal{L}_{\text{align}}$/$\mathcal{L}_{\text{desc}}$ the prototype-alignment and descriptor-reconstruction terms. The appendix preserves their exact definitions, pooled-target construction, warmup schedule, and shared Lyapunov reference.1. The complete loss definitions are given in Appendix E.8, and the full training procedure is summarized in Appendix G, Algorithm 1.

3 Experiments

Research questions. RQ1: Does SAE^{2}F preserve competitive overall forecasting performance while retaining the base forecasting model? RQ2: Does the unified state variable α capture window state and improve extreme-specific metrics? RQ3: Are the State Line and Fusion Interfaces Effective?

3.1 Experimental Setup

Datasets. The main scenario is a 5G mobile-network base-station cluster dataset. We also use ETTh1/2, ETTm1/2, Weather, Electricity, Solar, and Traffic. dataset statistics are listed in Table 8.

Baselines and Controlled Variants. For the overall forecasting comparison in Table 7, we compare with TQNet, CoMRes, SDE, DGraFormer, iTransformer, Pathformer, TimeMixer, PatchTST, and TimesNet. For RQ2, we use controlled internal variants, including SAE^{2}F-base and SAE^{2}F-full, under the same backbone and evaluation protocol. Supplementary component-level ablations are reported in the appendix.

Metrics. Following the main tables, we report overall MSE/MAE, extreme-point MSE/MAE, extreme-window MSE/MAE, and event-level Recall/F1. For the threshold-crossing event labels, the appendix additionally reports AUROC and average precision (AP), as well as early-warning and calibration measures for α.

Implementation and Validation. Appendix E provides supplementary validation of SAE^{2}F's efficiency, mechanism fidelity, robustness, and event-detection reliability under varied data and representation settings. All experiments follow the rolling-forecast protocol. The new loss weights control extreme supervision, prototype alignment, and Lyapunov consistency; full hyperparameters are reported in Appendix F. Unless otherwise specified, the base forecasting configuration is unchanged when adding SAE^{2}F modules.

Table 1: Average forecasting performance over $F \in \{96, 192, 336, 720\}$ on representative datasets. Full horizon-level results and the remaining datasets are reported in Appendix D, Table 7. Bold: best; underline: second best.

Dataset	SAE^{2}F (ours)		TQNet		CoMRes		SDE		DGraFormer		iTransformer		Pathformer		TimeMixer		PatchTST		TimesNet	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
RRC-C1	**0.483**	**0.506**	0.508	0.521	0.508	0.528	0.527	0.542	0.539	0.554	0.513	0.531	0.503	0.523	0.509	0.527	0.512	0.531	0.566	0.582
ETTh1	**0.405**	**0.418**	0.441	0.434	0.435	0.419	0.443	0.434	0.442	0.432	0.439	0.448	0.417	0.426	0.427	0.441	0.419	0.436	0.468	0.459
ETTm1	**0.338**	**0.369**	0.377	0.393	0.379	0.379	0.378	0.396	0.379	0.394	0.361	0.390	0.357	0.375	0.356	0.380	0.349	0.381	0.408	0.415
Weather	**0.220**	**0.254**	0.242	0.269	0.232	0.258	0.249	0.278	0.249	0.271	0.232	0.270	0.225	0.258	0.226	0.264	0.224	0.261	0.255	0.282
Electricity	**0.159**	**0.248**	0.165	0.259	0.177	0.263	0.172	0.268	0.168	0.258	0.163	0.258	0.168	0.261	0.185	0.284	0.171	0.270	0.190	0.290
Traffic	**0.390**	**0.257**	0.445	0.276	0.490	0.294	0.422	0.277	0.476	0.262	0.397	0.281	0.416	0.264	0.409	0.279	0.397	0.275	0.617	0.327

Table 2: Direct comparison with representative extreme-event forecasting methods on ETTh1 ($H = 96$, $q = 0.95$). Lower is better for Extreme MAE and Window MAE; higher is better for event Recall and event F1.

Method	Extreme MAE ↓	Window MAE ↓	Event Recall ↑	Event F1 ↑
SAE^{2}F-full	3.490	3.015	0.597	0.728
NEC+	3.874	3.641	0.462	0.581
DAN	4.215	4.080	0.416	0.523
SR-MTL	5.104	4.762	0.385	0.492
ReFine	4.683	3.945	0.440	0.556
PFformer	5.341	5.127	0.398	0.509

3.2 RQ1: Does the Backbone Performance Remain Competitive?

Analysis. Table 1 shows that SAE^{2}F retains competitive backbone forecasting performance on representative 5G, ETT, weather, electricity, and traffic datasets. Across the complete ten-dataset comparison in Appendix D, it attains the best average MSE and MAE on nine datasets; CoMRes is lower on ETTh2.

3.3 RQ2: Does α Capture State and Improve Extreme Prediction?

Tables 2 and 3 provide the external and internal extreme-event comparisons, respectively: the former evaluates representative extreme-aware methods under the same ETTh1 protocol, while the latter isolates the gain from adding the full state-aware framework to its backbone. Table 4 then evaluates whether α crosses a risk threshold before the associated extreme peak and reports false alarms, ranking quality, and calibration. Overall, the results validate the complementary benefits of state-conditioned interaction and prototype matching, while supporting α as an actionable risk-state representation.

Table 3: Internal extreme-event comparison between SAE^{2}F-base and SAE^{2}F-full. Each row fixes the dataset, horizon, and extreme quantile threshold. Lower is better for Extreme MAE and Window MAE; higher is better for Event F1.

Dataset / H / q	Extreme MAE ↓		Window MAE ↓		Event F1 ↑	
	base	full	base	full	base	full
RRC-C1 / 96 / 0.99	18.628	17.169	16.730	15.448	0.241	0.288
RRC-C2 / 96 / 0.99	53.861	48.920	47.978	43.419	0.381	0.449
ETTh1 / 96 / 0.98	4.237	3.684	4.278	3.688	0.433	0.568
ETTh2 / 96 / 0.98	11.216	9.891	11.080	9.719	0.455	0.570

Table 4: Early-warning behavior and calibration of α. H denotes the forecasting horizon, FA denotes false alarm, and Cal. denotes calibrated.

Dataset / H / q	Median lead time [IQR]	Pre-peak rate	Precision	FA rate	AUROC	AUPRC	Cal. Brier	Cal. ECE
RRC-C1 / 96 / 0.99	74.5 [64.75, 84.25]	0.765	0.696	0.093	0.803	0.788	0.211	0.087
ETTh1 / 96 / 0.95	56 [22, 76.5]	0.793	0.714	0.084	0.875	0.752	0.196	0.082

3.4 RQ3: Are the State Line and Fusion Interfaces Effective?

Table 5 shows the progressive gains from extreme supervision, prototype matching, the state prior, and Lyapunov consistency. Removing the state line or either state-conditioned interface degrades the extreme-event metrics, supporting their complementary roles.

Table 5: Core ablation results for RQ3 on ETTh1 ($H = 96$, $q = 0.98$), evaluating the independent contributions of the state line, structural switching, and prototype matching.

Variant	Overall MSE	Overall MAE	Extreme-point MAE	Event F1
SAE^{2}F-base	0.357	0.388	4.237	0.433
$+\mathcal{L}_{\text{ext}}$	0.356	0.387	4.006	0.492
$+\mathcal{L}_{\text{proto}}$	0.355	0.386	3.861	0.528
$+\alpha_{\text{prior}}$	0.355	0.386	3.748	0.552
$+\mathcal{L}_{\text{lyap}}$ (**SAE^{2}F-full**)	0.354	0.385	3.684	0.568
w/o state line	0.358	0.389	4.312	0.418
w/o SCIG	0.356	0.387	3.974	0.503
w/o SCPM	0.356	0.387	4.051	0.491

Figure 3 provides a qualitative state-mechanism case analysis: it links the improved extreme-window forecast to the early rise of α, the SCIG dependency interpolation, the SCPM retrieval, and the Lyapunov consistency trend. Together, these observations show how the shared state variable converts precursor evidence into interpretable forecasting adjustments across the model's state-aware interfaces.

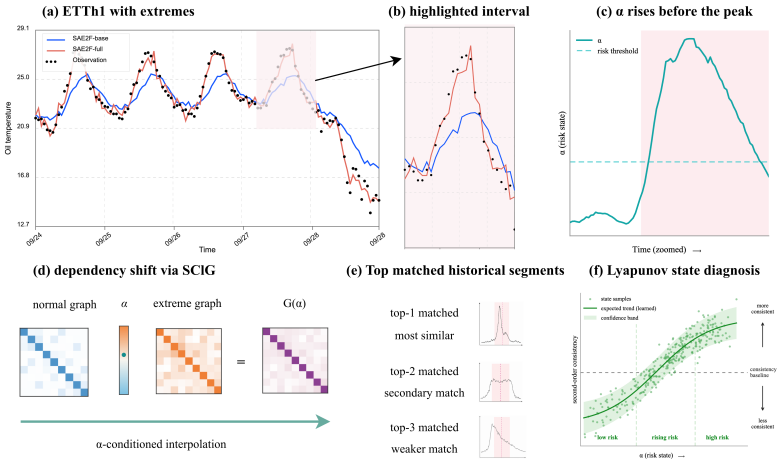


Figure 3: State-mechanism case analysis on ETTh1. **(a)** Forecasting around an extreme window: SAE^{2}F-full (red) tracks the high-value observations more closely than the base (blue). **(b)** Zoomed extreme interval highlighted in (a). **(c)** α rises and crosses the risk threshold *before* the extreme peak, confirming precursor-state detection. **(d)** α-conditioned interpolation between the normal and extreme interaction graphs via SCIG. **(e)** Top-matched historical extreme segments retrieved by SCPM. **(f)** Lyapunov second-order consistency: the learned trend aligns with the theoretical reference, confirming state-prior calibration.

4 Conclusion

We presented SAE^{2}F, a state-aware enhancement framework for multivariate extreme-event forecasting. It keeps the forecasting backbone as the main predictor and adds a local-dynamics state line with a Lyapunov second-order reference, providing a risk prior, state descriptors, and a training constraint. The unified state variable α then drives dependency switching through SCIG and event-family reuse through SCPM. Overall, SAE^{2}F converts local dynamical evidence into verifiable extreme-forecasting gains without replacing the base forecasting model.

AI USE STATEMENT

All ideas and contributions presented in this work originate from the authors, with each author making a substantial contribution to the research. Generative AI was not used to develop the research ideas, methodology, experiments, analysis, or conclusions. AI-assisted tools were used exclusively for language editing and polishing to improve the clarity and readability of the manuscript.

REFERENCES

Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Bernie Wang. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024.

Andreas Auer, Patrick Podest, Daniel Klotz, Sebastian Böck, Günter Klambauer, and Sepp Hochreiter. Tirex: Zero-shot forecasting across long and short horizons with enhanced in-context learning. In *Advances in Neural Information Processing Systems*, 2025.

Thomas M Bury, RI Sujith, Induja Pavithran, Marten Scheffer, Timothy M Lenton, Madhur Anand, and Chris T Bauch. Deep learning for early warning signals of tipping points. *Proceedings of the National Academy of Sciences*, 118(39):e2106140118, 2021.

Cristian Challu, Kin G Olivares, Boris N Oreshkin, Federico Garza Ramirez, Max Mergenthaler Canseco, and Artur Dubrawski. N-HiTS: Neural hierarchical interpolation for time series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 6989–6997, 2023.

Shengyu Chen, Nasrin Kalanat, Simon Topp, Jeffrey Sadler, Yiqun Xie, Zhe Jiang, and Xiaowei Jia. Meta-transfer-learning for time series data with extreme events: An application to water temperature prediction. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 266–275, 2023.

Stuart Coles. *An introduction to statistical modeling of extreme values*, volume 208. Springer London, 2001. doi: 10.1007/978-1-4471-3675-0.

Vasilis Dakos, Marten Scheffer, Egbert H Van Nes, Victor Brovkin, Vladimir Petoukhov, and Hermann Held. Slowing down as an early warning signal for abrupt climate change. *Proceedings of the National Academy of Sciences*, 105(38):14308–14312, 2008.

Vasilis Dakos, Stephen R Carpenter, William A Brock, Aaron M Ellison, Vishwesha Guttal, Anthony R Ives, Sonia Kéfi, Valerie Livina, David A Seekell, Egbert H van Nes, et al. Methods for detecting early warnings of critical transitions in time series illustrated using simulated ecological data. *PloS one*, 7(7):e41010, 2012.

Daizong Ding, Mi Zhang, Xudong Pan, Min Yang, and Xiangnan He. Modeling extreme events in time series prediction. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1114–1122, 2019.

Jingfang Fan, Jun Meng, Josef Ludescher, Xiaosong Chen, Yosef Ashkenazy, Jürgen Kurths, Shlomo Havlin, and Hans Joachim Schellnhuber. Statistical physics approaches to the complex earth system. *Physics reports*, 896:1–84, 2021.

Hermann Held and Thomas Kleinen. Detection of climate system bifurcations by degenerate fingerprinting. *Geophysical Research Letters*, 31(23), 2004.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Jiahao Ji, Wentao Zhang, Jingyuan Wang, and Chao Huang. Seeing the unseen: Learning basis confounder representations for robust traffic prediction. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pp. 577–588, 2025.

Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International conference on learning representations*, 2022.

Roger Koenker and Gilbert Bassett Jr. Regression quantiles. *Econometrica: journal of the Econometric Society*, pp. 33–50, 1978.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pp. 95–104, 2018.

Jeffrey Lai, Anthony Bao, and William Gilpin. Panda: A pretrained forecast model for chaotic dynamics. In *International Conference on Learning Representations*, 2026.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations (ICLR '18)*, 2018.

Yanhong Li and David C Anastasiu. Pfformer: A position-free transformer variant for extreme-adaptive multivariate time series forecasting. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 185–197. Springer, 2025.

Yanhong Li, Jack Xu, and David C Anastasiu. An extreme-adaptive time series prediction model based on probability-enhanced lstm neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 8684–8691, 2023.

Yanhong Li, Jack Xu, and David Anastasiu. Learning from polar representation: An extreme-adaptive model for long-term time series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 171–179, 2024.

Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International journal of forecasting*, 37(4): 1748–1764, 2021.

Shizhan Liu, Hang Yu, Cong Liao, Jianguo Li, Weiyao Lin, Alex X Liu, and Schahram Dustdar. Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. In *International Conference on Learning Representations*, 2022.

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 *The Twelfth International Conference on Learning Representations*, 2024.

Naoki Masuda, Kazuyuki Aihara, and Neil G. MacLaren. Anticipating regime shifts by mixing early warning signals from different nodes. *Nature Communications*, 15:1086, 2024. doi: 10.1038/s41467-024-45476-9.

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.

Boris N. Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=r1ecqn4YwB.

Ole Peters and J David Neelin. Critical phenomena in atmospheric precipitation. *Nature physics*, 2 (6):393–396, 2006.

Ole Peters, Christopher Hertlein, and Kim Christensen. A complexity view of rainfall. *Physical review letters*, 88(1):018701, 2001.

Daniele Proverbio, Alexander Skupin, and Jorge Gonçalves. Systematic analysis and optimization of early warning signals for critical transitions using distribution data. *iScience*, 26(7):107156, 2023. doi: 10.1016/j.isci.2023.107156.

Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. Duet: Dual clustering enhanced multivariate time series forecasting. In *Proceedings of the 31st ACM SIGKDD conference on knowledge discovery and data mining v. 1*, pp. 1185–1196, 2025.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International journal of forecasting*, 36(3): 1181–1191, 2020.

Marten Scheffer, Jordi Bascompte, William A Brock, Victor Brovkin, Stephen R Carpenter, Vasilis Dakos, Hermann Held, Egbert H Van Nes, Max Rietkerk, and George Sugihara. Early-warning signals for critical transitions. *Nature*, 461(7260):53–59, 2009.

Minlan Shao, Zijian Zhang, Yili Wang, Yiwei Dai, Xu Shen, and Xin Wang. Hyperd: Hybrid periodicity decoupling framework for traffic forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 15689–15697, 2026.

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

Jimeng Shi, Azam Shirali, and Giri Narasimhan. Boosting time series prediction of extreme events by reweighting and fine-tuning. In *2024 IEEE International Conference on Big Data (BigData)*, pp. 1450–1457. IEEE, 2024.

Shun-Yao Shih, Fan-Keng Sun, and Hung-yi Lee. Temporal pattern attention for multivariate time series forecasting. *Machine Learning*, 108(8):1421–1441, 2019.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

Bo Sun, Zhe Wu, Zhiyuan Deng, Li Su, and Qingfang Zheng. Extracting global temporal patterns within short look-back windows for traffic forecasting. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 2781–2790, 2025.

Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 1199–1208. IEEE, 2018.

Jiacheng Tang, Fengqian Ding, Rui Shao, and Chao Luo. Spatial-temporal fuzzy cognitive maps based on adaptive graph learning for multivariate time series interpretable prediction. *Applied Intelligence*, 55(12):888, 2025.

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.

Huiqiang Wang, Jian Peng, Feihu Huang, Jince Wang, Junhui Chen, and Yifei Xiao. MICN: Multi-scale local and global context modeling for long-term series forecasting. In *International Conference on Learning Representations*, 2023.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Zhang, and Jun Zhou. TimeMixer: Decomposable multiscale mixing for time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2024.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53140–53164. PMLR, 2024.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430, 2021.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, IJCAI'19, pp. 1907–1913. AAAI Press, 2019. ISBN 9780999241141.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 11121–11128, 2023.

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.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11106–11115, 2021.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International conference on machine learning*, pp. 27268–27286. PMLR, 2022.

Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. One fits all: Power general time series analysis by pretrained LM. *Advances in neural information processing systems*, 36:43322–43355, 2023.

APPENDIX A RELATED WORK

APPENDIX A.1 EXTENDED POSITIONING OF EXISTING METHODS

Difference from existing methods. General forecasting models such as PatchTST, iTransformer, TimeMixer, and DUET optimize overall error and representation capacity without explicitly separating normal and extreme regimes Nie et al. (2023); Liu et al. (2024); Wang et al. (2024); Qiu et al. (2025). The extreme-aware methods cited above strengthen tail signals through rare-event modeling, probability-enhanced objectives, polar representations, transfer learning, or fine-tuning, but mostly use a unified feature space across regimes. Statistical-physics and early-warning studies help identify critical-transition signals, yet these signals still need to be further learned and integrated into a trainable state interface for forecasting models Fan et al. (2021); Bury et al. (2021); Raissi et al. (2019). SAE^{2}F bridges this gap by providing state evidence for α via local stochastic dynamics and Lyapunov references, then injecting α into dependency switching and prototype matching; see Appendix A for expanded related work and Appendix B, Remark 1, for the approximation conditions. Due to page limitations, expanded related work, derivations, and additional experimental evidence are provided in the appendix.

APPENDIX A.2 MULTIVARIATE TIME-SERIES FORECASTING

The Transformer architecture Vaswani et al. (2017) laid the groundwork for attention-based sequence modeling. Recent multivariate forecasting methods, including Informer Zhou et al. (2021), Autoformer Wu et al. (2021), FEDformer Zhou et al. (2022), PatchTST Nie et al. (2023), iTransformer Liu et al. (2024), TimeMixer Wang et al. (2024), and DLinear Zeng et al. (2023), improve overall forecasting accuracy through sparse attention, decomposition, frequency modeling, patch tokenization, and linear inductive biases. Crossformer Zhang & Yan (2023), MICN Wang et al. (2023), and Pyraformer Liu et al. (2022) further address cross-dimension dependency and pyramidal attention. Foundation and pre-trained models such as Moirai Woo et al. (2024) and GPT4TS Zhou et al. (2023) extend these ideas to universal and language-model-based time-series forecasting. Instance normalization techniques such as RevIN Kim et al. (2022) help mitigate distribution shift. Recurrent architectures Hochreiter & Schmidhuber (1997) and probabilistic models Salinas et al. (2020) provide complementary forecasting paradigms. Basis expansion approaches Oreshkin et al. (2020); Challu et al. (2023), LSTNet Lai et al. (2018), and TFT Lim et al. (2021) broaden the landscape of multivariate forecasting baselines. Spatiotemporal forecasting methods Li et al. (2018); Wu et al. (2019); Shih et al. (2019); Ji et al. (2025); Shao et al. (2026); Sun et al. (2025) address network traffic and graph-structured data. Graph learning approaches with adaptive dependency modeling have also been explored for interpretable multivariate time series prediction Tang et al. (2025). DUET Qiu et al. (2025) further emphasizes temporal heterogeneity and channel dependency. These methods mainly optimize overall error and do not explicitly model extreme states.

APPENDIX A.3 EXTREME-EVENT FORECASTING AND EXTREME-AWARE LEARNING

Ding et al. Ding et al. (2019) show that ordinary point losses can underfit extreme events. NEC+ Li et al. (2023), DAN Li et al. (2024), and PFformer Li & Anastasiu (2025) enhance tail performance through explicit normal/extreme separation or position-free modeling. Meta-Transfer-Learning Chen et al. (2023) and ReFine Shi et al. (2024) suggest that historical segments similar to the target window can provide useful transfer. From a metric-learning perspective, Prototypical Networks Snell et al. (2017) and Relation Network Sung et al. (2018) suggest that few-shot settings may benefit from class prototypes and query-prototype relation scoring. However, most existing methods strengthen tail modeling from an engineering perspective. They provide limited explanation for why a system approaches an extreme regime and rarely organize state transition, dependency switching, and event-family heterogeneity into a coordinated framework.

APPENDIX A.4 CRITICAL SLOWING DOWN, EARLY WARNING, AND COMPLEX SYSTEMS

Statistical physics has long studied extreme events and critical transitions. Peters et al. Peters et al. (2001); Peters & Neelin (2006) interpret rainfall events through critical phenomena and avalanche statistics. Dakos et al. Dakos et al. (2008) and Scheffer et al. Scheffer et al. (2009) summarize

warning signals such as variance increase, lag-1 autocorrelation increase, and recovery-rate decrease. Held and Kleinen Held & Kleinen (2004) formalize the AR(1) coefficient approaching one as a testable bifurcation signal. Fan et al. Fan et al. (2021) and Bury et al. Bury et al. (2021) further demonstrate the complementarity between statistical physics and machine learning for anticipating critical transitions in complex systems.

Relation to physics-informed machine learning. PINNs Raissi et al. (2019) embed PDE residuals into the loss and require outputs to satisfy physical equations. $\mathrm{SAE}^2\mathrm{F}$ differs in three ways: (i) it uses a second-order statistical reference under local stochastic-dynamics approximation rather than continuous PDE residuals; (ii) the consistency constraint is applied to state-description organization and training regularization rather than to forcing predicted values to satisfy an equation; (iii) the goal is to provide a derivable basis for the risk-state variable rather than recover a complete physical mechanism.

Table 6 summarises how $\mathrm{SAE}^2\mathrm{F}$ is positioned relative to these method families across five dimensions.

Table 6: Positioning against related methods.

Method type	Extreme	State shift	Dependency shift	Event heterogeneity	Basis
General forecasting	No	No	Partial	No	Data-driven
Channel-dependency models	Partial	Partial	Yes	No	Data-driven
Extreme-aware learning	Yes	Partial	Partial	Partial	Engineering
Transfer/meta-learning	Yes	Partial	No	Yes	Transfer learning
Statistical physics / EWS	Partial	Yes	Partial	Partial	Critical phenomena
$\mathrm{SAE}^2\mathrm{F}$ (ours)	**Yes**	**Yes**	**Yes**	**Yes**	**Local dynamics+DL**

APPENDIX B THEORETICAL DERIVATION AND TECHNICAL DETAILS

Remark 1 (Applicability of the diagonal approximation)**.** *Equations* (9)–(11) *rely on a diagonal approximation in which the dominant contribution of A comes from diagonal terms A_{nn}. This approximation is accurate under sparse or short-range coupling and acts as a first-order approximation in densely coupled datasets. The qualitative trend still holds: variance increases as $\rho(A) \to 1$, autocorrelation approaches one, and $\mathcal{L}_{\mathrm{lyap}}$ only requires $\sigma_n^2(1-\rho_n^2)$ to remain approximately stable under normal regimes.*

APPENDIX B.1 DERIVATION OF THE LYAPUNOV SECOND-ORDER RELATION

For the steady-state system in Eq. (7), the stationary covariance satisfies the discrete Lyapunov equation:

$$\Sigma = A\,\Sigma\,A^\top + Q. \tag{27}$$

Step 1: Vectorization. Applying the vec operator and the identity $\mathrm{vec}(AXB) = (B^\top \otimes A)\,\mathrm{vec}(X)$:

$$(I - A \otimes A)\,\mathrm{vec}(\Sigma) = \mathrm{vec}(Q) \quad \Longrightarrow \quad \mathrm{vec}(\Sigma) = (I - A \otimes A)^{-1}\,\mathrm{vec}(Q). \tag{28}$$

Step 2: Near-critical divergence. As $\rho(A) \to 1$, the matrix $I - A \otimes A$ approaches singularity (its smallest eigenvalue $\to 0$), so elements of Σ grow without bound:

$$\rho(A) \to 1 \quad \Longrightarrow \quad \Sigma_{nn} \to +\infty. \tag{29}$$

Step 3: Diagonal approximation. Retaining only diagonal terms A_{nn} (off-diagonal entries assumed small relative to diagonal), Eq. (27) gives per-variable:

$$\Sigma_{nn} \approx \frac{Q_{nn}}{1 - A_{nn}^2}, \tag{30}$$

which diverges as $A_{nn} \to 1$, consistent with variance amplification (Eq. (9)).

Step 4: Lag-1 autocorrelation. The lag-1 autocovariance matrix is $\Gamma(1) = A\Sigma$. Under the diagonal approximation:

$$[\Gamma(1)]_{nn} = [A\Sigma]_{nn} \approx A_{nn}\,\Sigma_{nn}, \quad \text{so} \quad \rho_n^{(1)} = \frac{[\Gamma(1)]_{nn}}{\Sigma_{nn}} \approx A_{nn}. \tag{31}$$

This supports the lag-1 autocorrelation limit in Eq. (10).

Step 5: Lyapunov consistency. Combining Eqs. (30) and (31):

$$\Sigma_{nn}(1-[\rho_n^{(1)}]^2) \approx \frac{Q_{nn}}{1-A_{nn}^2}\cdot(1-A_{nn}^2) = Q_{nn}. \tag{32}$$

This invariant holds under normal-regime local dynamics and motivates the Lyapunov consistency loss $\mathcal{L}_{\text{lyap}}$ and the fixed state-proxy mapping α_{prior} in the main text.

APPENDIX C DESIGN DISCUSSION

SAE^{2}F follows two principles. First, the base forecasting model remains the main predictor; local dynamics and second-order statistics provide state-variable design, dependency-switching evidence, and auxiliary supervision. Second, the state modeling component needs a unified theoretical handle. Local stochastic-dynamics approximation and the Lyapunov relation provide this handle: state-description features, the α prior, and $\mathcal{L}_{\text{lyap}}$ are all grounded in the near-critical second-order relation, while SCIG operates on a complementary mechanism—stress-weighted cross-variable correlation—to capture dependency shifts that the scalar Lyapunov constraint does not directly model. For the forecasting backbone, the frequency-domain relation mask can be used as a relation mask or attention bias in masked attention, encouraging exchange among channels with similar spectral or periodic patterns rather than uniform mixing across all channels.

Robustness, limitations, and efficiency. Robustness is evaluated by scanning $q \in \{0.95, 0.98, 0.99\}$, random seeds, and window lengths; the corresponding evidence is summarized in Appendix D. SAE^{2}F may be less effective on high-dimensional densely coupled datasets, under strong distribution shifts, or when extreme events are too short to form meaningful prototype families, which matches the assumptions of the diagonal-approximation state line. Inference remains a single forward pass, but the extra state-aware modules have different scaling behaviors: window statistics and state routing are approximately linear in the number of window entries, SCPM scales linearly with the prototype count, and the current dense SCIG implementation scales quadratically with the number of variables because it estimates and applies cross-variable correlation graphs. Thus, on high-dimensional datasets such as Traffic ($N = 862$), the SCIG graph term is the dominant additional inference and memory cost; sparse or top-k graph construction is a natural engineering extension when larger variable sets are considered. The current limitations include empirical quantile thresholds and a prototype space learned per task rather than across datasets; adaptive thresholds and transferable prototypes are natural future directions. The method may support early warning in domains such as mobile-network operation and power-load monitoring, while false alarms or missed alarms should be handled with human review and privacy-preserving deployment protocols.

APPENDIX D FULL EXPERIMENTAL RESULTS

This appendix collects the complete experimental evidence that is summarized in the main text. Table 7 reports horizon-wise forecasting results for all datasets and baselines. Table 8 lists dataset statistics and extreme ratios. Table 9 provides full extreme-specific metrics across all quantile thresholds. Table 10 extends the ablation study across multiple datasets, and Fig. 4 visualises the ablation trends on ETTh2 across two quantile thresholds. Table 11 summarises robustness and sensitivity scans. Table 18 lists all hyperparameters used in the experiments. Fig. 5 shows supplementary prediction cases across multiple datasets. Together, these materials support the aggregate forecasting results in Table 7, the extreme-event analysis in Table 9, and the ablation discussion in Table 10.

Table 7: Full forecasting performance comparison across all horizons $F \in \{96, 192, 336, 720\}$ on all datasets. Bold: best; underline: second best.

Models		SAE^{2}F (ours)		TQNet (2025)		CoMRes (2025)		SDE (2025)		DGraFormer (2025)		iTransformer (2024)		Pathformer (2024)		TimeMixer (2024)		PatchTST (2023)		TimesNet (2023)	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
RRC-C1	96	**0.413**	**0.470**	0.438	0.489	0.432	0.488	0.451	0.501	0.463	0.512	0.445	0.496	0.427	0.481	0.434	0.488	0.440	0.493	0.489	0.538
	192	**0.449**	0.489	0.466	**0.485**	0.472	0.510	0.489	0.523	0.501	0.536	0.481	0.516	0.469	0.507	0.475	0.511	0.477	0.514	0.526	0.559
	336	**0.480**	**0.503**	0.506	0.525	0.506	0.525	0.523	0.541	0.535	0.552	0.516	0.534	0.502	0.522	0.497	0.518	0.508	0.528	0.560	0.580
	720	**0.590**	**0.561**	0.622	0.586	0.622	0.590	0.645	0.603	0.658	0.617	0.611	0.579	0.615	0.582	0.628	0.591	0.624	0.589	0.690	0.650
	Avg	**0.483**	**0.506**	0.508	0.521	0.508	0.528	0.527	0.542	0.539	0.554	0.513	0.531	0.503	0.523	0.509	0.527	0.512	0.531	0.566	0.582
RRC-C2	96	**0.391**	**0.499**	0.407	0.511	0.419	0.522	0.432	0.535	0.446	0.548	0.424	0.529	0.412	0.516	0.417	0.520	0.421	0.524	0.468	0.574
	192	**0.416**	**0.514**	0.442	0.536	0.445	0.538	0.458	0.552	0.472	0.566	0.451	0.547	0.436	0.532	0.441	0.535	0.432	0.527	0.493	0.589
	336	**0.395**	**0.501**	0.421	0.524	0.423	0.526	0.411	0.514	0.451	0.553	0.429	0.532	0.415	0.518	0.421	0.524	0.425	0.527	0.474	0.577
	720	**0.429**	**0.531**	0.456	0.554	0.459	0.555	0.474	0.571	0.444	0.545	0.466	0.563	0.449	0.549	0.456	0.554	0.461	0.558	0.510	0.612
	Avg	**0.408**	**0.511**	0.432	0.531	0.437	0.535	0.444	0.543	0.453	0.553	0.443	0.543	0.428	0.529	0.434	0.533	0.435	0.534	0.486	0.588
ETTh1	96	**0.354**	**0.385**	0.371	0.393	0.378	0.386	0.376	0.400	0.367	0.393	0.386	0.405	0.372	0.392	0.372	0.401	0.377	0.397	0.389	0.412
	192	**0.399**	**0.409**	0.428	0.426	0.436	0.415	0.432	0.429	0.438	0.426	0.424	0.440	0.408	0.415	0.413	0.430	0.409	0.425	0.440	0.443
	336	**0.414**	**0.419**	0.476	0.446	0.452	0.421	0.477	0.437	0.479	0.442	0.449	0.460	0.438	0.434	0.438	0.450	0.431	0.444	0.523	0.487
	720	0.453	0.460	0.487	0.470	0.474	**0.453**	0.488	0.471	0.484	0.467	0.495	0.487	**0.450**	0.463	0.486	0.484	0.457	0.477	0.521	0.495
	Avg	**0.405**	**0.418**	0.441	0.434	0.435	0.419	0.443	0.434	0.442	0.432	0.439	0.448	0.417	0.426	0.427	0.441	0.419	0.436	0.468	0.459
ETTh2	96	**0.272**	0.331	0.295	0.343	0.275	**0.327**	0.288	0.340	0.300	0.344	0.297	0.348	0.279	0.336	0.281	0.351	0.274	0.337	0.334	0.370
	192	**0.336**	**0.372**	0.367	0.393	0.346	0.373	0.373	0.390	0.382	0.397	0.372	0.403	0.345	0.380	0.349	0.387	0.348	0.384	0.404	0.413
	336	0.343	0.401	0.417	0.427	**0.326**	**0.369**	0.380	0.406	0.388	0.415	0.388	0.417	0.378	0.408	0.366	0.413	0.377	0.416	0.389	0.435
	720	**0.392**	**0.417**	0.433	0.446	0.393	0.419	0.412	0.432	0.418	0.436	0.424	0.444	0.437	0.455	0.401	0.436	0.406	0.441	0.434	0.448
	Avg	0.336	0.380	0.378	0.402	**0.335**	**0.372**	0.363	0.392	0.372	0.398	0.370	0.403	0.360	0.395	0.349	0.397	0.351	0.395	0.390	0.417
ETTm1	96	**0.279**	**0.333**	0.311	0.353	0.310	0.339	0.315	0.357	0.326	0.366	0.300	0.353	0.290	0.335	0.293	0.345	0.289	0.343	0.340	0.378
	192	**0.321**	**0.358**	0.356	0.378	0.362	0.365	0.360	0.383	0.358	0.376	0.341	0.380	0.337	0.363	0.335	0.372	0.329	0.368	0.392	0.404
	336	**0.346**	**0.375**	0.390	0.401	0.386	0.387	0.389	0.405	0.387	0.397	0.374	0.396	0.374	0.384	0.368	0.386	0.362	0.390	0.423	0.426
	720	**0.405**	**0.408**	0.452	0.440	0.456	0.424	0.448	0.440	0.446	0.436	0.429	0.430	0.428	0.416	0.426	0.417	0.416	0.423	0.475	0.453
	Avg	**0.338**	**0.369**	0.377	0.393	0.379	0.379	0.378	0.396	0.379	0.394	0.361	0.390	0.357	0.375	0.356	0.380	0.349	0.381	0.408	0.415
ETTm2	96	**0.161**	**0.244**	0.173	0.256	0.163	0.245	0.172	0.259	0.173	0.253	0.175	0.266	0.164	0.250	0.165	0.256	0.165	0.255	0.189	0.265
	192	**0.213**	**0.285**	0.238	0.298	0.229	0.290	0.238	0.301	0.243	0.302	0.242	0.312	0.219	0.288	0.225	0.298	0.221	0.293	0.254	0.310
	336	**0.267**	**0.315**	0.301	0.340	0.291	0.330	0.300	0.340	0.315	0.348	0.282	0.337	**0.267**	0.319	0.277	0.332	0.276	0.327	0.313	0.345
	720	**0.347**	**0.375**	0.397	0.396	0.369	0.378	0.394	0.394	0.404	0.401	0.375	0.394	0.361	0.377	0.360	0.387	0.362	0.381	0.413	0.402
	Avg	**0.247**	**0.305**	0.277	0.323	0.263	0.311	0.276	0.324	0.284	0.326	0.269	0.327	0.253	0.309	0.257	0.318	0.256	0.314	0.292	0.331
Weather	96	**0.142**	**0.190**	0.157	0.200	0.151	**0.190**	0.165	0.214	0.168	0.207	0.157	0.207	0.148	0.195	0.147	0.198	0.149	0.196	0.168	0.214
	192	**0.188**	**0.231**	0.206	0.245	0.199	0.236	0.214	0.255	0.213	0.249	0.200	0.248	0.191	0.235	0.192	0.243	0.191	0.239	0.219	0.262
	336	**0.242**	**0.272**	0.262	0.287	0.244	0.275	0.271	0.297	0.270	0.291	0.252	0.287	0.243	0.274	0.247	0.284	**0.242**	0.279	0.278	0.302
	720	**0.306**	**0.324**	0.344	0.342	0.334	0.331	0.346	0.347	0.345	0.338	0.320	0.336	0.318	0.326	0.318	0.330	0.312	0.330	0.353	0.351
	Avg	**0.220**	**0.254**	0.242	0.269	0.232	0.258	0.249	0.278	0.249	0.271	0.232	0.270	0.225	0.258	0.226	0.264	0.224	0.261	0.255	0.282
Electricity	96	0.136	**0.218**	**0.134**	0.229	0.152	0.239	0.146	0.244	0.136	0.229	**0.134**	0.230	0.135	0.222	0.153	0.256	0.143	0.247	0.169	0.271
	192	**0.145**	**0.236**	0.154	0.247	0.166	0.252	0.162	0.258	0.155	0.244	0.154	0.250	0.157	0.253	0.168	0.269	0.158	0.260	0.180	0.280
	336	**0.162**	**0.257**	0.169	0.264	0.181	0.267	0.177	0.274	0.171	0.261	0.169	0.265	0.170	0.267	0.189	0.291	0.168	0.267	0.204	0.304
	720	**0.192**	**0.279**	0.201	0.294	0.210	0.292	0.202	0.297	0.210	0.298	0.194	0.288	0.211	0.302	0.228	0.320	0.214	0.307	0.205	0.304
	Avg	**0.159**	**0.248**	0.165	0.259	0.177	0.263	0.172	0.268	0.168	0.258	0.163	0.258	0.168	0.261	0.185	0.284	0.171	0.270	0.190	0.290
Solar	96	**0.168**	**0.217**	0.173	0.233	0.181	0.220	0.186	**0.217**	0.184	0.219	0.190	0.244	0.218	0.235	0.179	0.232	0.170	0.234	0.198	0.270
	192	**0.185**	**0.209**	0.199	0.257	0.219	0.249	0.230	0.251	0.211	0.234	0.193	0.257	0.196	0.220	0.201	0.259	0.204	0.302	0.206	0.276
	336	0.193	**0.217**	0.211	0.263	0.235	0.263	0.253	0.270	0.234	0.250	0.203	0.266	0.195	0.220	**0.190**	0.256	0.212	0.293	0.208	0.284
	720	**0.199**	**0.224**	0.209	0.270	0.238	0.268	0.247	0.274	0.235	0.254	0.223	0.281	0.208	0.237	0.203	0.261	0.215	0.307	0.232	0.294
	Avg	**0.186**	**0.217**	0.198	0.256	0.218	0.250	0.229	0.253	0.216	0.239	0.202	0.262	0.204	0.228	0.193	0.252	0.200	0.284	0.211	0.281
Traffic	96	**0.360**	**0.239**	0.413	0.261	0.468	0.270	0.388	0.261	0.442	0.245	0.363	0.265	0.384	0.250	0.369	0.257	0.370	0.262	0.595	0.312
	192	**0.381**	**0.248**	0.432	0.271	0.465	0.285	0.411	0.271	0.469	0.257	0.384	0.273	0.405	0.257	0.400	0.272	0.386	0.269	0.613	0.322
	336	**0.388**	**0.257**	0.450	0.277	0.493	0.295	0.428	0.278	0.481	0.266	0.396	0.277	0.424	0.265	0.407	0.272	0.396	0.275	0.626	0.332
	720	**0.432**	0.282	0.486	0.295	0.535	0.324	0.461	0.297	0.510	**0.281**	0.445	0.308	0.452	0.283	0.461	0.316	0.435	0.295	0.635	0.340
	Avg	**0.390**	**0.257**	0.445	0.276	0.490	0.294	0.422	0.277	0.476	0.262	0.397	0.281	0.416	0.264	0.409	0.279	0.397	0.275	0.617	0.327
1^{st} Count		36	35	1	1	1	4	0	1	0	1	1	0	2	0	1	0	1	0	0	0

Table 8: Dataset statistics and extreme ratios. Lengths, variable counts, and extreme ratios are computed from the actual long-format CSV files in dataset/forecasting; each extreme ratio denotes the fraction of observations above the corresponding upper-tail quantile threshold.

Dataset	Domain	Freq.	Length	Variables	Split	Horizons	95%	98%	99%	Description
RRC-C1	5G mobile network	15min	11,808	5	7:1:2	96/192/336/720	4.89%	1.95%	0.93%	RRC connection mean, cluster 1
RRC-C2	5G mobile network	15min	11,808	14	7:1:2	96/192/336/720	4.98%	1.92%	0.99%	RRC connection mean, cluster 2
ETTh1	Electricity	1h	17,420	7	6:2:2	96/192/336/720	5.00%	1.98%	1.00%	Power transformer 1 temperature/load indicators
ETTh2	Electricity	1h	17,420	7	6:2:2	96/192/336/720	5.00%	2.00%	1.00%	Power transformer 2 temperature/load indicators
ETTm1	Electricity	15min	69,680	7	6:2:2	96/192/336/720	4.97%	1.98%	1.00%	Minute-level transformer 1 measurements
ETTm2	Electricity	15min	69,680	7	6:2:2	96/192/336/720	5.00%	2.00%	1.00%	Minute-level transformer 2 measurements
Weather	Meteorology	10min	52,696	21	7:1:2	96/192/336/720	5.00%	2.00%	1.00%	Weather station measurements
Electricity	Electricity	1h	26,304	321	7:1:2	96/192/336/720	5.00%	2.00%	1.00%	Electricity consumption records
Solar	Energy	10min	52,560	137	6:2:2	96/192/336/720	4.99%	1.99%	1.00%	Solar power production
Traffic	Transportation	1h	17,544	862	7:1:2	96/192/336/720	5.00%	2.00%	1.00%	Road occupancy rates

Table 9: Full extreme-specific results for SAE^{2}F-full across all datasets and three quantile thresholds $q \in \{0.95, 0.98, 0.99\}$, reporting extreme-point MAE/RMSE, extreme-window MAE/RMSE, event Recall, and event F1.

Dataset	q	extreme MAE	extreme RMSE	window MAE	window RMSE	event Recall	event F1
RRC-C1	0.95/0.98/0.99	15.436/16.214/17.169	15.921/16.782/17.627	14.082/14.771/15.448	14.596/15.263/15.857	0.368/0.304/0.240	0.417/0.351/0.288
RRC-C2	0.95/0.98/0.99	44.103/46.581/48.920	48.322/50.742/53.106	39.214/41.287/43.419	44.018/46.183/48.536	0.471/0.409/0.352	0.566/0.507/0.449
ETTh1	0.95/0.98/0.99	3.490/3.684/3.836	4.019/4.166/4.327	3.015/3.688/3.984	3.588/4.199/4.512	0.597/0.460/0.392	0.728/0.568/0.492
ETTh2	0.95/0.98/0.99	8.564/9.891/10.012	9.443/10.291/11.084	8.569/9.719/10.403	9.665/10.695/11.451	0.599/0.475/0.418	0.678/0.570/0.503

Table 10: Full ablation results. This table reports the complete ablation results across the 5G scenario, ETTh1, and Weather.

Variant	RRC-C1 event F1	ETTh1 event F1	Weather event F1
SAE^{2}F-base	0.276	0.433	0.461
$+\mathcal{L}_{\text{ext}}$	0.309	0.492	0.512
$+\mathcal{L}_{\text{proto}}$	0.327	0.528	0.537
$+\alpha_{\text{prior}}$	0.341	0.552	0.556
SAE^{2}F-full	0.351	0.568	0.579
w/o state line	0.265	0.418	0.447
w/o SCIG	0.318	0.503	0.522
w/o SCPM	0.311	0.491	0.509
w/o $\mathcal{L}_{\text{lyap}}$	0.268	0.418	0.472

Table 11: Robustness and sensitivity results on ETTh1 (H=96). Each scan varies one factor while keeping the rest at the default setting (q=0.98, seq_len=96, seed=1, λ_{lyap}=0.05, K_p=6).

Scan	Setting	extreme MAE	event F1
quantile q	0.95	3.490	0.728
	0.98	3.684	0.568
	0.99	3.836	0.492
seq_len	96	3.684	0.568
	336	3.550	0.582
	512	3.610	0.575
seed	1	3.684	0.568
	2	3.720	0.559
	3	3.650	0.571
λ_{lyap}	0.01	3.710	0.555
	0.05	3.684	0.568
	0.10	3.690	0.562
prototype number K_p	4	3.700	0.560
	6	3.684	0.568
	8	3.690	0.565

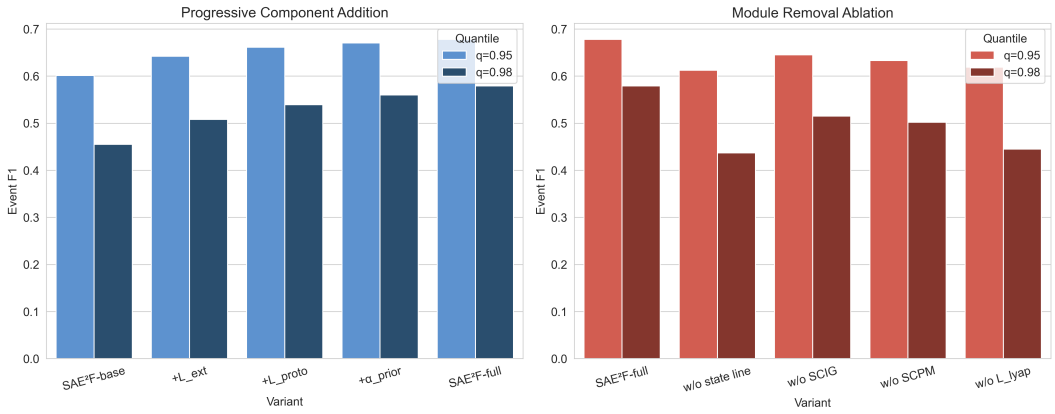


Figure 4: Ablation study on ETTh2 (H=96) across two quantile thresholds (q=0.95 and q=0.98). **Left:** Progressive component addition shows consistent event-F1 gains at both thresholds as each module is introduced. **Right:** Module removal confirms that removing any component degrades performance, with the state line and $\mathcal{L}_{\text{lyap}}$ showing the largest drops.

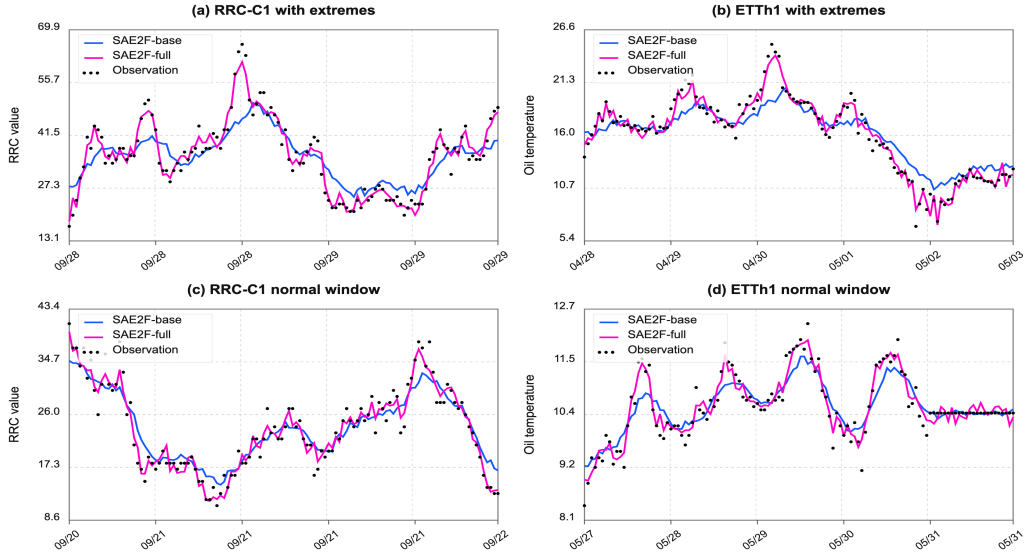


Figure 5: Supplementary prediction case studies on RRC-C1 and ETTh1. **(a)(b)** Extreme windows: SAE^{2}F-full (red) better follows high-value observations than the base (blue). **(c)(d)** Normal windows: both variants remain competitive under regular dynamics, confirming that the state-conditioned modules do not degrade the base forecasting capacity.

APPENDIX E ADDITIONAL SUPPLEMENTARY EXPERIMENTS

This appendix provides supplementary evidence for the efficiency, mechanism, robustness, and event-detection properties of SAE^{2}F. Unless a table states otherwise, the event labels use the channel-normalized upper-tail construction in Eqs. (1)–(3). These experiments complement, rather than replace, the main controlled ablations.

APPENDIX E.1 EFFICIENCY OVERHEAD

To verify the practical efficiency of the state-conditioned modules, we measured parameter count, peak resident-set size (RSS), and inference time on ETTh1 with input length 512, $H = 96$, and batch size 64. Each variant was run in 10 independent processes; every run used five warm-up batches and averaged over 20 timed batches. Table 12 reports the mean and sample standard deviation over the independent processes.

Table 12: Inference overhead of the state-conditioned modules on ETTh1.

Variant	Trainable parameters	Peak inference RSS (MB), mean $\pm$ std	Inference time / batch (ms), mean $\pm$ std
SAE^{2}F-base	4,059,775	510.249 $\pm$ 4.602	300.930 $\pm$ 3.078
Base + SCIG	4,144,901	512.370 $\pm$ 4.948	307.518 $\pm$ 1.492
Base + SCPM	4,316,246	509.731 $\pm$ 4.364	305.217 $\pm$ 5.065
SAE^{2}F-full	4,316,246	515.303 $\pm$ 4.193	309.228 $\pm$ 1.490

SCIG has no additional trainable layer once the shared state-estimation branch is active, so Base + SCPM and SAE^{2}F-full have the same trainable parameter count. The 0.518 MB lower mean RSS of Base + SCPM relative to the base model is smaller than the observed run-to-run standard deviation and is not interpreted as a memory reduction. Relative to SAE^{2}F-base, the full model adds 256,471 trainable parameters, 5.054 MB mean peak RSS, and 8.298 ms per batch in this protocol.

APPENDIX E.2 LYAPUNOV-CONSISTENCY DIAGNOSTICS

To inspect the stabilizing effect of $\mathcal{L}_{\text{lyap}}$, we compare the full state line with a variant that removes this term. Temporal variation is the mean absolute change of α between consecutive test forecast origins; saturation is the fraction of α values in $[0, 0.05] \cup [0.95, 1]$; the residual is the test mean of the squared expression in Eq. (41).

Table 13: Diagnostics with and without Lyapunov consistency.

Dataset	Setting	α temporal variation	Saturation near 0 or 1	Lyapunov residual
RRC-C1	Without $\mathcal{L}_{\text{lyap}}$	0.000118	0.025	0.010064
RRC-C1	With $\mathcal{L}_{\text{lyap}}$	0.000097	0.008	0.008589
RRC-C2	Without $\mathcal{L}_{\text{lyap}}$	0.000083	0.018	0.003354
RRC-C2	With $\mathcal{L}_{\text{lyap}}$	0.000067	0.005	0.002389
ETTh1	Without $\mathcal{L}_{\text{lyap}}$	0.000181	0.032	0.017327
ETTh1	With $\mathcal{L}_{\text{lyap}}$	0.000143	0.010	0.015054
ETTh2	Without $\mathcal{L}_{\text{lyap}}$	0.000161	0.024	0.006143
ETTh2	With $\mathcal{L}_{\text{lyap}}$	0.000127	0.007	0.005499

The consistency term reduces temporal variation, endpoint saturation, and the residual in all four evaluated datasets. This diagnostic supports its intended stabilizing role, while not claiming that it identifies the complete system dynamics.

APPENDIX E.3 CONTROLLED EFFECT OF α IN SCPM

To verify whether α directly controls prototype selection, for a fixed fused state embedding $\mathbf{z}_{\text{fuse}}$, we vary only α from 0.1 to 0.9 while measuring prototype switching and matching changes. This intervention isolates the state variable's effect on prototype assignment. We report the Top-1 prototype

switch rate, the mean Jensen–Shannon (JS) divergence of matching weights, and the cosine distance between matched contexts.

Table 14: Controlled SCPM scan with a fixed fused state embedding.

Setting	Top-1 switch rate	Mean JS divergence	Context cosine distance
Full SCPM, α: 0.1 to 0.9	0.183	0.014	0.032
SCPM without α, same scan	0.000	0.00000001	0.00000003

With full SCPM, changing α changes the top-ranked prototype in 18.3% of the tested windows and yields nonzero changes in both the matching distribution and the matched context. After removing α conditioning, all three quantities are effectively zero under the same scan.

APPENDIX E.4 ROBUSTNESS TO NEARBY CHRONOLOGICAL SPLITS

To verify the robustness of the reported gains, we evaluate SAE^{2}F under several nearby chronological train-validation splits. On RRC-C1 with $H = 96$ and $q = 0.99$, we retain the final 20% of the series as the test interval and change only the boundary between training and validation. This measures robustness to nearby chronological partitions rather than a general small-data regime.

Table 15: Performance of SAE^{2}F-full under nearby chronological data splits on RRC-C1.

Split	Training windows	Overall MAE ↓	Extreme MAE ↓	Event F1 ↑
50% / 30% / 20%	5,296	0.509	17.340	0.259
60% / 20% / 20%	6,477	0.489	17.510	0.248
70% / 10% / 20%	7,658	0.470	17.169	0.288

APPENDIX E.5 ALTERNATIVE STATE REPRESENTATIONS AND SCIG

To verify whether the state-conditioned interaction mechanism is dependent on a particular representation, we compare the original statistical descriptors with frozen Chronos-2 Ansari et al. (2024), Panda Lai et al. (2026), and TiRex Auer et al. (2025) representations on ETTh1 ($H = 96$, $q = 0.98$). Chronos-2 and Panda directly encode the multichannel input. TiRex is applied with a shared univariate encoder across channels, and the resulting variable-level embeddings are stacked. Each frozen representation is mapped to the same state interface through a trainable adapter. For every representation, we compare variants with and without SCIG while keeping the forecasting backbone, training protocol, and SCPM fixed.

Table 16: Controlled comparison of state representations and SCIG on ETTh1 ($H = 96$, $q = 0.98$). Each cell reports used / removed SCIG.

State representation	Overall MAE ↓	Extreme MAE ↓	Event F1 ↑
Statistical descriptors	0.385 / 0.387	3.684 / 3.974	0.568 / 0.503
Chronos-2 (frozen)	0.385 / 0.393	3.710 / 3.998	0.551 / 0.483
Panda (frozen)	0.383 / 0.390	3.699 / 4.003	0.566 / 0.501
TiRex (frozen)	0.394 / 0.399	3.705 / 3.971	0.567 / 0.497

SCIG improves Overall MAE, Extreme MAE, and event F1 for every evaluated representation. The frozen representations are competitive with the original descriptors, but no representation dominates every metric. Thus, the state representation and state-conditioned interaction modeling make complementary contributions in this controlled setting.

APPENDIX E.6 EVENT-DETECTION METRICS UNDER CLASS IMBALANCE

To verify the reliability of event detection under class imbalance, Table 17 supplements event Recall and F1 with AUROC and average precision (AP) under the same event definitions and test protocols as the corresponding extreme-event experiments.

Table 17: Event-detection metrics for SAE^{2}F-base and SAE^{2}F-full.

Dataset / H / q	Setting	Event Recall ↑	Event F1 ↑	AUROC ↑	AP ↑
RRC-C1 / 96 / 0.99	SAE^{2}F-base	0.198	0.241	0.692	0.251
RRC-C1 / 96 / 0.99	SAE^{2}F-full	0.240	0.288	0.741	0.329
RRC-C2 / 96 / 0.99	SAE^{2}F-base	0.296	0.381	0.753	0.463
RRC-C2 / 96 / 0.99	SAE^{2}F-full	0.352	0.449	0.809	0.552
ETTh1 / 96 / 0.95	SAE^{2}F-base	0.524	0.648	0.871	0.742
ETTh1 / 96 / 0.95	SAE^{2}F-full	0.597	0.728	0.912	0.839
ETTh1 / 96 / 0.98	SAE^{2}F-base	0.362	0.433	0.781	0.504
ETTh1 / 96 / 0.98	SAE^{2}F-full	0.460	0.568	0.852	0.661
ETTh2 / 96 / 0.95	SAE^{2}F-base	0.531	0.601	0.849	0.672
ETTh2 / 96 / 0.95	SAE^{2}F-full	0.599	0.678	0.891	0.762
ETTh2 / 96 / 0.98	SAE^{2}F-base	0.386	0.455	0.802	0.531
ETTh2 / 96 / 0.98	SAE^{2}F-full	0.475	0.570	0.859	0.670

SAE^{2}F-full improves both AUROC and AP over SAE^{2}F-base in every reported setting. The AP gains are especially visible at the rarer thresholds, consistent with the corresponding Recall and F1 improvements.

Appendix E.7 Early-warning interpretation

Early warning and calibration of α. We evaluate whether α provides a usable early-risk signal rather than only a qualitative visualization. Lead time is the number of steps between the first risk-threshold crossing and the associated event peak; the pre-peak rate is the fraction of events for which such a crossing occurs before the peak. The remaining measures evaluate alert precision, false alarms, ranking quality, and calibrated probability quality.

Analysis. Across the six controlled base/full settings, SAE^{2}F-full reduces Extreme MAE by 7.8%–13.1% and raises event F1 by 0.047–0.135. At ETTh1 with $H = 96$ and $q = 0.98$, for example, Extreme MAE / event F1 changes from 4.237 / 0.433 to 3.684 / 0.568. The external comparison in Table 2 confirms the gain under a common protocol, while the early-warning results in Table 4 support α as a practical risk signal. Detailed case analyses, component ablations, and additional robustness checks are provided in Appendix E.

Appendix E.8 Detailed training objective

For batch size B, forecasting horizon H, and N variables, the point-forecasting term is

$$\mathcal{L}_{\text{point}} = \frac{1}{BHN} \sum_{b=1}^{B} \sum_{h=1}^{H} \sum_{n=1}^{N} |\hat{y}_{b,h,n} - y_{b,h,n}| + \mathcal{L}_{\text{base}}, \tag{33}$$

where $\mathcal{L}_{\text{base}}$ is an optional auxiliary regularizer supplied by the selected forecasting backbone and is zero when no such regularizer is used.

Extreme state-aware loss:

$$\mathcal{L}_{\text{ext}} = \mathcal{L}_{\text{ind}} + \mathcal{L}_{\text{q}} + \mathcal{L}_{\text{exc}}, \tag{34}$$

where $\mathcal{L}_{\text{ind}}$ is the event-indicator BCE loss, $\mathcal{L}_{\text{q}}$ is the upper-quantile pinball loss Koenker & Bassett Jr (1978), and $\mathcal{L}_{\text{exc}}$ is the exceedance regression loss. All three are built around a pooled extreme target: the future horizon is first max-pooled across variables to form a pooled target, from which event indicators, exceedance amplitudes, and event descriptors are constructed. The risk variable α receives implicit supervision mainly through $\mathcal{L}_{\text{ind}}$. More precisely, let i index the M pooled forecast positions, $e_i = \mathbf{1}[y_i \geq \tau_{\text{ext}}^{(q)}]$, $x_i = (y_i - \tau_{\text{ext}}^{(q)})^+$, $s_i = \sigma(z_i)$, $u_i = y_i - \hat{y}_i^{(q)}$, and $\hat{x}_i$ be the predicted

exceedance. The three sub-losses are

$$\mathcal{L}_{\text{ind}} = -\frac{1}{M}\sum_{i=1}^{M}\left[e_i \log s_i + (1-e_i)\log(1-s_i)\right], \tag{35}$$

$$\mathcal{L}_{\text{q}} = \frac{1}{M}\sum_{i=1}^{M}\max\left(qu_i, (q-1)u_i\right), \tag{36}$$

$$\mathcal{L}_{\text{exc}} = \frac{1}{M}\sum_{i=1}^{M}\left|\hat{x}_i - x_i\right|. \tag{37}$$

Extreme-event heterogeneity loss:

$$\mathcal{L}_{\text{proto}} = \mathcal{L}_{\text{align}} + \mathcal{L}_{\text{desc}}. \tag{38}$$

Here $\mathcal{L}_{\text{align}}$ is a prototype-alignment KL loss and $\mathcal{L}_{\text{desc}}$ is a descriptor-reconstruction MSE loss. The former aligns predicted matching weights with the target distribution induced by event-descriptor/prototype-descriptor similarity, while the latter constrains the matched descriptor to reconstruct the current event descriptor. Specifically, for predicted prototype weights a_{ik}, target descriptor-similarity distribution p_{ik}, matched descriptor $\hat{\mathbf{d}}_i$, target descriptor $\mathbf{d}_i$, K_p prototypes, and descriptor dimension D,

$$\mathcal{L}_{\text{align}} = \frac{1}{B}\sum_{i=1}^{B}\sum_{k=1}^{K_p} p_{ik}\log\frac{p_{ik}}{a_{ik}+\epsilon}, \tag{39}$$

$$\mathcal{L}_{\text{desc}} = \frac{1}{BD}\sum_{i=1}^{B}\left\|\hat{\mathbf{d}}_i - \mathbf{d}_i\right\|_2^2. \tag{40}$$

Lyapunov consistency loss:

$$\mathcal{L}_{\text{lyap}} = \frac{1}{N}\sum_{n=1}^{N}\left(\hat{\sigma}_n^2(1-[\hat{\rho}_n^{(1)}]^2) - q_{\text{lyap}}\right)^2, \tag{41}$$

where q_{lyap} is a shared learnable noise-variance reference. This loss provides a second-order statistical reference without forcing all high-risk windows to fit a normal-regime Lyapunov surface; deviations are mainly used for diagnosis and visualization. Appendix E, Table 13, reports before/after diagnostics for the constraint. Further discussion is in Appendix C. The complete training procedure with warmup scheduling is detailed in Algorithm 1 (Appendix G).

Appendix E.9 Complexity details

Therefore, for a batch size B, input length L, number of variables N, and prediction horizon H, constructing the two graphs costs $O(BLN^2)$ and applying the residual propagation costs $O(BHN^2)$. The identity mixing and small residual scale stabilize the correction and reduce its empirical impact on the base path, but they do not change this quadratic dependence on N.

With K_p prototypes and hidden dimension D, relation scoring in SCPM costs $O(BK_pD)$, and the final weighted prototype aggregation has the same order. Since K_p is small in all experiments, this term is minor compared with the dense SCIG graph operation on high-dimensional datasets such as Traffic. Measured parameter, memory, and inference-time overhead is reported in Appendix E, Table 12.

Appendix F Implementation details

In implementation, the main extra cost of SAE^{2}F comes from window statistics, state routing, SCIG graph-residual propagation, and SCPM relation scoring, consistent with the complexity discussion. The implementation uses identity mixing in both normal and extreme graphs and applies the graph only as a small residual correction; these choices are for stability and controlled empirical overhead, while the asymptotic dependence of the dense graph part remains quadratic in N.

Compute resources. All experiments are conducted on two NVIDIA RTX 4090 24 GB GPUs. All models are implemented in PyTorch and trained with the Adam optimizer.

Table 18: Main hyperparameter list.

Hyperparameter	Description	Typical values	Module
seq_len	Look-back window length	96/336/512	Input
d_model	Hidden dimension of the forecasting backbone	128/256/512	Forecasting backbone
d_ff	Feed-forward network dimension	128/256/512/1024	Forecasting backbone
e_layers	Encoder layers in the forecasting backbone	1/2	Forecasting backbone
n_heads	Channel-attention heads	1/2/4	Forecasting backbone
state_hidden_dim	Shared state representation dimension	64/128	State interface + SCPM + tail heads
num_matching_prototypes	Number of learnable prototypes	4/6/8	SCPM
extreme_quantile	Threshold for extreme-label construction	0.95/0.98/0.99	Extreme-label construction
quantile_level	Quantile level of the pinball loss	0.95/0.98/0.99	$\mathcal{L}_{\text{ext}}$
λ_{ext}	Weight of the extreme-state-aware loss	0.2	$\mathcal{L}_{\text{ext}}$
λ_{proto}	Weight of the prototype-matching loss	0.05	$\mathcal{L}_{\text{proto}}$
λ_{lyap}	Weight of the Lyapunov consistency loss	0.05	$\mathcal{L}_{\text{lyap}}$
graph_residual_scale	Strength of graph residual correction	0.05	SCIG
tail_sigma	Stress-threshold scale	2.0	SCIG
extreme_loss_warmup	Epoch to enable extreme loss	5	Training
prototype_matching_warmup	Epoch to enable prototype loss	8	Training
normal_graph_identity_mix	Identity mixing ratio in the normal graph	0.7	SCIG
extreme_graph_identity_mix	Identity mixing ratio in the extreme graph	0.2	SCIG

APPENDIX G TRAINING ALGORITHM

Algorithm 1 summarises the SAE^{2}F training loop, highlighting the warmup scheduling of the three auxiliary losses.

Algorithm 1 SAE^{2}F Training Procedure

Input: Training data $\mathcal{D}$; base backbone f_{pred}; $\lambda_{\text{ext}}, \lambda_{\text{proto}}, \lambda_{\text{lyap}}$; warmup epochs e_{ext}=5, e_{proto}=8; K_p=6
Init: backbone, STATEROUTER, SCIG, SCPM, EXTREMEHEADS; $q \leftarrow 0.1$
for epoch $e = 1, \ldots, E$ **do**
  for each batch $(\mathbf{X}, \mathbf{Y}) \in \mathcal{D}$ **do**
    $\hat{\mathbf{Y}}^{(0)} \leftarrow f_{\text{pred}}(\mathbf{X})$ ▷ base forecast
    $\mathbf{S}$, $\alpha_{\text{prior}} \leftarrow \text{STATEFEATURES}(\mathbf{X})$ ▷ state descriptors + prior
    $\alpha \leftarrow \text{STATEROUTER}(\mathbf{X}, \mathbf{S}, \alpha_{\text{prior}})$ ▷ unified risk variable
    $\mathbf{G} \leftarrow (1-\alpha)\mathbf{G}^{\text{nor}} + \alpha\mathbf{G}^{\text{ext}}$ via $\text{SCIG}(\mathbf{X}, \alpha)$
    $\hat{\mathbf{Y}} \leftarrow \hat{\mathbf{Y}}^{(0)} + \lambda_g \alpha \Big(\mathbf{G}\hat{\mathbf{Y}}^{(0)} - \hat{\mathbf{Y}}^{(0)} \Big)$ ▷ graph residual
    $\mathbf{c}_p \leftarrow \text{SCPM}(\mathbf{z}_{\text{fuse}}, \alpha)$ ▷ prototype matching
    Compute extreme targets from $\mathbf{Y}$
    $\mathcal{L} \leftarrow \mathcal{L}_{\text{point}}(\hat{\mathbf{Y}}, \mathbf{Y}) + \lambda_{\text{lyap}}\mathcal{L}_{\text{lyap}}(\hat{\sigma}^2, \hat{\rho}, q)$ ▷ $\mathcal{L}_{\text{lyap}}$ always active
    if $e \geq e_{\text{ext}}$ **then**
      $\mathcal{L}$ += $\lambda_{\text{ext}}\mathcal{L}_{\text{ext}}$
      if $e \geq e_{\text{proto}}$ **then**
        $\mathcal{L}$ += $\lambda_{\text{proto}}\mathcal{L}_{\text{proto}}$
      end if
    end if
    Backpropagate; update all parameters
  end for
  Apply early stopping on validation loss
end for