PHYTM: A PHYSICS-INFORMED TRI-MODAL TRANSFORMER-MAMBA FRAMEWORK FOR TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

With the growing volume of time series data across real-world applications, accurate forecasting has become essential for understanding temporal dynamics and predicting future trends. However, existing approaches often overlook the physical processes underlying observed time series, resulting in limited incorporation of physical dynamics under noisy observations, weak temporal-physical awareness in attention-based components, and insufficient dynamical constraints in State-Space Models (SSMs) that may lead to spectral drift and structural distortion over long forecasting horizons. To address these challenges, PhyTM is proposed as a physics-informed tri-modal forecasting framework that incorporates physical priors into multimodal temporal modeling and improves long-term forecasting stability. First, a Dynamic Mode Decomposition (DMD)-based representation module is introduced to embed time series into a delayed state space and extract spectrally consistent dynamical modes for temporal evolution modeling. Second, a physics-informed dual-stream attention Transformer is designed for textual temporal representations by integrating semantic attention with DMD-derived physical priors to enhance temporal-physical awareness beyond conventional positional encoding. Third, a physics-structured SSM is developed for visual temporal representations to enforce physically consistent state transitions and patch-level evolution while maintaining linear-time efficiency. Finally, extensive experiments on long-term and short-term forecasting benchmarks verify the effectiveness of PhyTM, which achieves a 23.4% Mean Squared Error (MSE) reduction on Exchange at $H = 720$ and a 12.7% Overall Weighted Average (OWA) reduction on M4-Others. Code and data are available at `https://anonymous.4open.science/r/PhyTM-2026-E33D`.

1 INTRODUCTION

Time series forecasting plays a fundamental role in understanding and predicting the evolution of real-world systems, including energy, traffic, and weather (Guo et al., 2026). Recent deep forecasting models have achieved strong performance by learning statistical dependencies from historical observations. However, observed time series are often driven by underlying dynamical processes, while purely data-driven models rarely characterize the physical dynamics governing temporal evolution (Park et al., 2026; Chen et al., 2024b). Physics-informed learning bridges this gap by incorporating physical knowledge into data-driven forecasting (Karniadakis et al., 2021). Beyond explicit governing equations, such knowledge can also be extracted from observations through dynamical-system analysis. In particular, DMD decomposes temporal evolution into dominant modes characterized by decay or growth rates and oscillation frequencies (Takeishi et al., 2017; Liu et al., 2023), providing a data-driven physical prior for physics-informed forecasting.

A key challenge in time series forecasting is to model long-range dependencies while preserving the dynamics of temporal evolution. Recurrent and convolutional models capture sequential dependencies with different tradeoffs in memory and parallelism (Bai et al., 2018). Transformer-based forecasters further model long-range dependencies through attention and related mechanisms (Nie et al., 2023; Zhou et al., 2022), while structured state-space models such as S4 and Mamba provide efficient long-sequence modeling (Gu et al., 2022b; Gu & Dao, 2024). Despite their strong ability to capture

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.

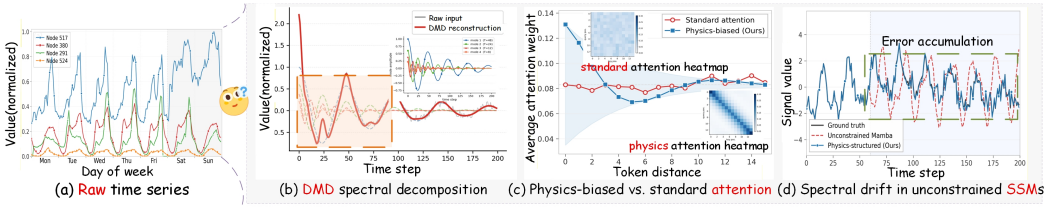


Figure 1: Physics-aware temporal analysis. (a) Raw time series visualization. (b) DMD spectral decomposition of dominant dynamical modes. (c) Standard attention vs. physics-informed dual-stream attention. (d) Spectral drift in unconstrained SSMs.

complex temporal dependencies from observations, these architectures do not explicitly ensure that learned representations follow the physical dynamics of the observed system (Hu et al., 2026). This motivates incorporating spectral priors to constrain temporal modeling toward physically consistent dynamics (Takeishi et al., 2017; Liu et al., 2023).

Many forecasting methods operate directly on numerical observations, using learned temporal or cross-variable dependencies to construct predictions (Zeng et al., 2023; Liu et al., 2024b). However, these observations may be generated by latent dynamical systems. As illustrated in Fig. 1(b), a local dynamical approximation can be characterized by dominant modes associated with decay rates and oscillation frequencies. Learning suitable observables for Koopman analysis provides a principled way to extract such structure from data (Takeishi et al., 2017). This motivates testing whether dynamical representations improve long-horizon forecasting and the consistency of learned temporal features. Meanwhile, textual and visual representations offer complementary views of the same series. Prompt-based forecasting uses natural-language representations (Xue & Salim, 2023; Jin et al., 2024), whereas visual approaches exploit image-like temporal structure (Chen et al., 2025; Chang et al., 2026). These results motivate combining representations, but agreement across modalities is not an automatic consequence of predictive training. In PhyTM, we use a shared dynamical representation to guide the attention and state-space components, targeting the inconsistencies illustrated in Fig. 1(c)-(d). Our design addresses three issues.

❶ ***Insufficient explicit physical-dynamics modeling.*** Predictive backbones need not extract interpretable dynamical modes, motivating the incorporation of data-derived physical representations.

❷ ***Limited temporal-physical awareness in attention.*** Temporal order and statistical similarity alone do not specify physical evolution, motivating attention conditioned on dynamical information.

❸ ***Insufficient alignment between learned states and observed dynamics.*** Efficient state transitions need not match the spectral structure of the observations, motivating explicit dynamical constraints.

Guided by these observations, we propose PhyTM, a physics-informed tri-modal framework for time series forecasting. To the best of our knowledge, this is the first tri-modal framework to enforce physical constraints in both attention and state-space components, bridging data-driven modeling and physical dynamics for multimodal forecasting. The key contributions are summarized as follows.

① ***DMD-based physical information modeling.*** A DMD-based physical information modeling module is introduced, embedding time series into a delayed state space and extracting physically interpretable dynamical representations via spectral decomposition for explicit evolution modeling.

② ***Physics-informed dual-stream attention Transformer for textual time series.*** A dual-stream attention mechanism is designed to jointly model semantic dependencies and physical dynamics by integrating content attention with DMD-derived physical signals, overcoming the limitations of standard positional encoding.

③ ***Physics-structured SSM for visual time series.*** A physics-structured SSM is developed by incorporating physical constraints into state transitions, enforcing patch-level evolution consistent with spectral dynamics while preserving linear-time efficiency.

④ ***Comprehensive experimental validation.*** Extensive experiments demonstrate that PhyTM achieves competitive forecasting performance on multiple benchmarks, achieving up to 23.4% MSE reduction on long-term forecasting and 12.7% OWA reduction on short-term forecasting.

2 Related Work

Physics-Informed Temporal Modeling. Physics-informed learning extends beyond explicit governing-equation constraints to structural priors derived from dynamical systems (Karniadakis et al., 2021). Koopman theory provides a spectral representation of nonlinear dynamics, while DMD estimates such structure from observations (Takeishi et al., 2017). The resulting modes characterize dominant growth, decay, and oscillation dynamics, providing data-derived dynamical priors. For time-series forecasting, Koopa (Liu et al., 2023) demonstrates the utility of Koopman representations for non-stationary dynamics. Following this perspective, PhyTM incorporates DMD-derived spectral dynamics into attention and state transitions as structural priors for multimodal forecasting.

Multimodal Time Series Representation Learning. Recent forecasting methods increasingly exploit representations beyond raw numerical sequences. Language-based approaches transform or reprogram time series through textual representations (Xue & Salim, 2023; Jin et al., 2024), while VisionTS (Chen et al., 2025) formulates forecasting as visual reconstruction. Other studies explore observed modalities and multimodal alignment (Chacon Maldonado et al., 2025; Zhou et al., 2025). These methods demonstrate the value of complementary representations, but agreement among modalities does not necessarily ensure consistency with underlying temporal dynamics. PhyTM coordinates numerical, textual, and visual representations through a DMD-derived dynamical prior.

3 Methodology

PhyTM derives a physical prior from DMD-estimated decay and oscillation dynamics to guide textual attention and visual state transitions, thereby promoting dynamically structured representations across modalities. The forecasting problem and notation are formally defined in Appendix A.

3.1 Overall Framework

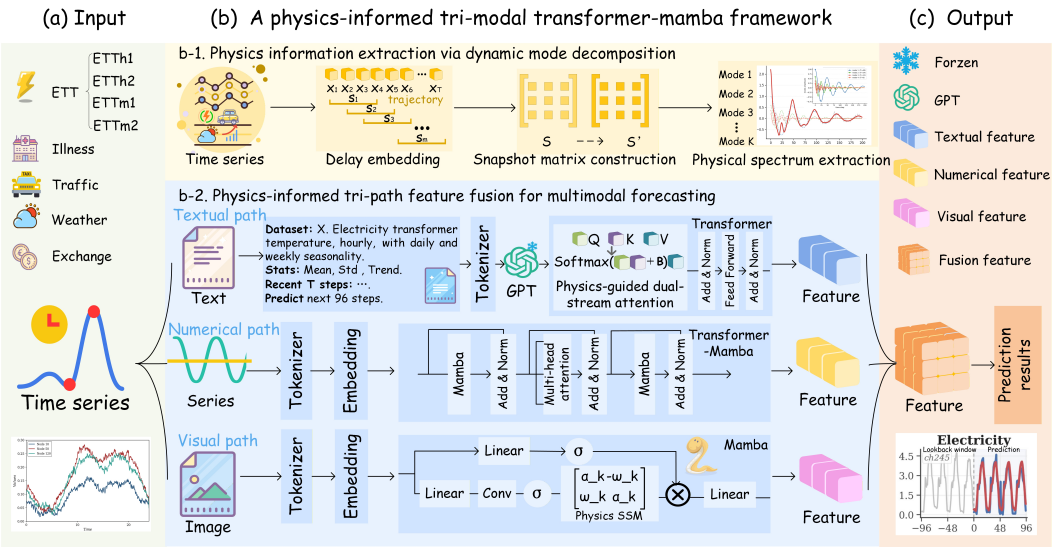


Figure 2: PhyTM model architecture. Module (a) inputs multivariate time series from diverse domains; Module (b-1) performs physics information extraction via DMD; Module (b-2) conducts physics-informed tri-path feature fusion over textual, numerical, and visual representations; Module (c) fuses the three modal features and generates the final prediction results.

As shown in Figure 2, PhyTM is a physics-informed tri-modal framework for time series forecasting. It first extracts DMD-based dynamical priors from observed time series and then incorporates them into textual and visual representations through physics-informed Transformer and state-space modeling. The resulting textual, visual, and numerical representations are finally fused for forecasting.

3.2 PHYSICS INFORMATION EXTRACTION VIA DMD

DMD is estimated once using training data only. Residual corrections are learned during training; all parameters remain fixed during validation and testing. Details appear in Appendix C.5.

Delay-Embedded State Construction Raw time series $\mathbf{x}_t$ conflate latent dynamical processes along a single temporal axis, providing no explicit notion of system state. Delay embedding lifts each observation into a higher-dimensional space, where concatenating d_{emb} consecutive steps provides a richer state representation for characterizing latent temporal dynamics. Formally, the embedded state at time t is constructed by stacking d_{emb} consecutive observations:

$$\mathbf{s}_t = [\mathbf{x}_t^\top, \mathbf{x}_{t-1}^\top, \ldots, \mathbf{x}_{t-d_{\text{emb}}+1}^\top]^\top, \tag{1}$$

where d_{emb} denotes the embedding depth. Sliding this window over the DMD training sequence of length T_{DMD} yields $M = T_{\text{DMD}} - d_{\text{emb}} + 1$ states, reindexed chronologically as $\{\mathbf{s}_j\}_{j=1}^{M}$ for snapshot construction. Here, T_{DMD} differs from the forecasting lookback L.

Snapshot Matrix Construction The transition structure between consecutive states is the core quantity that DMD is designed to identify. To expose this structure, the embedded trajectory $\{\mathbf{s}_t\}_{t=1}^{M}$ is organized into two temporally shifted snapshot matrices:

$$\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_{M-1}], \qquad \mathbf{S}' = [\mathbf{s}_2, \mathbf{s}_3, \ldots, \mathbf{s}_M], \tag{2}$$

where the k-th columns of $\mathbf{S}$ and $\mathbf{S}'$ constitute a matched current-future state pair. This paired construction encodes all observed one-step transitions and reformulates temporal evolution as the estimation of a single linear operator $\mathbf{A}$ satisfying $\mathbf{S}' \approx \mathbf{A}\mathbf{S}$.

Operator Estimation via Low-Rank Projection Direct least-squares estimation of $\mathbf{A}$ is computationally prohibitive and noise-sensitive in the original high-dimensional space. To overcome this, we identify the dominant low-rank subspace of $\mathbf{S}$ via truncated SVD: $\mathbf{S} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$, retaining the leading r singular components that capture the most energetically significant temporal structures. By projecting the state transition onto this subspace, the reduced-order evolution operator is obtained as:

$$\tilde{\mathbf{A}} = \mathbf{U}^\top \mathbf{S}' \mathbf{V} \mathbf{\Sigma}^{-1}, \tag{3}$$

which preserves the evolution patterns while improving numerical stability and suppressing noise.

Physical Spectrum Extraction Although $\tilde{\mathbf{A}}$ captures the dominant dynamics, its matrix entries are not physically interpretable. The underlying dynamical modes are exposed via eigendecomposition:

$$\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\mathbf{\Lambda}, \qquad \mathbf{\Lambda} = \operatorname{diag}(\mu_1, \ldots, \mu_r), \tag{4}$$

where each discrete-time eigenvalue μ_k encodes one dynamical mode. Under the continuous-time assumption $\mu_k = \exp\big((\alpha_k + i\omega_k)\Delta t\big)$, the physical parameters follow directly:

$$\alpha_k = \frac{\ln|\mu_k|}{\Delta t}, \qquad \omega_k = \frac{\arg(\mu_k)}{\Delta t}, \tag{5}$$

where α_k is the growth or decay rate and ω_k is the oscillation frequency of mode k. Because these parameters derive from the energy-dominant subspace identified by SVD, they characterize dominant dynamical patterns while reducing the influence of low-energy variations. However, α_k and ω_k are extracted under a global linear assumption and cannot capture nonlinear deviations or adapt to input samples. Learnable residual corrections are therefore introduced to adapt the extracted parameters while preserving stable state dynamics:

$$\tilde{\alpha}_k = -\exp(\ell_k + \Delta\alpha_k), \qquad \tilde{\omega}_k = \omega_k + \Delta\omega_k, \tag{6}$$

where $\ell_k = \log(-\alpha_k^{(0)})$ is fixed for a stable DMD reference $\alpha_k^{(0)} < 0$ (Appendix C.5). Zero residuals recover this reference; the parameterization ensures $\tilde{\alpha}_k < 0$. Effective parameters are shared across windows and remain fixed at inference.

3.3 PHYSICS-INFORMED TRI-PATH FEATURE FUSION FOR MULTIMODAL FORECASTING

The tri-path architecture fuses textual, visual, and numerical representations, with DMD-derived priors guiding the textual and visual paths.

3.3.1 PHYSICS-GUIDED DUAL-STREAM ATTENTION FOR TEXTUALIZED TIME SERIES

The textual path converts time series into natural language descriptions and encodes them with a physics-informed Transformer. A DMD-derived physical prior is incorporated into attention logits as a structured relative-time bias, jointly modeling semantic dependencies and dynamical structure.

Textual Modality Construction For each channel, a structured prompt combines dataset metadata, lookback statistics, chronological observations, and the forecasting task. A frozen GPT-2 encoder produces contextual token states, which are mean-pooled over each observation's token span and linearly projected to d_f dimensions. This yields $\mathbf{E}^{(n)} \in \mathbb{R}^{L\times d_f}$ per channel and $\mathbf{E} \in \mathbb{R}^{N\times L\times d_f}$ across channels, preserving the time-step alignment required by the physical bias. Prompt construction and token alignment are detailed in Appendix B.1.

Physical Bias Construction While $\mathbf{E}$ captures semantic temporal information, standard attention has no awareness of the underlying physical dynamics. To address this, a DMD-derived bias matrix $\mathbf{B}$ is constructed from the refined physical parameters $\{(\tilde{\alpha}_k, \tilde{\omega}_k)\}_{k=1}^{K}$ and injected as a relative-position bias along the temporal axis:

$$\mathbf{B}[i,j] = \sum_{k=1}^{K} \beta_k \cdot \exp\big(\tilde{\alpha}_k \cdot \tau_{ij}\big) \cdot \cos\big(\tilde{\omega}_k \cdot \tau_{ij}\big), \tag{7}$$

where i, j index the L observed time steps, $\tau_{ij} = |i-j|\Delta t$ is their physical time lag, Δt is the sampling interval, and β_k controls the contribution of mode k. The bias $\mathbf{B} \in \mathbb{R}^{L\times L}$ is shared across channels and heads. Its training-derived spectrum and learned corrections are fixed at inference; only content attention varies with the input window.

Physics-Informed Dual-Stream Attention For each channel n and attention head, the query, key, and value matrices $\mathbf{Q}^{(n)}, \mathbf{K}^{(n)}, \mathbf{V}^{(n)} \in \mathbb{R}^{L\times d_h}$ are projected from $\mathbf{E}^{(n)}$, where d_h is the head dimension. Both the content logits and the physical bias therefore have shape $L \times L$, with their rows and columns referring to the same observed time steps. The attention adopts a dual-stream mechanism that jointly models semantic dependencies through scaled dot-product attention and DMD-derived dynamical affinity through the bias $\mathbf{B}$, with both streams fused within a single softmax operation:

$$\mathrm{Attn}^{(n)}_{\mathrm{phys}} = \mathrm{softmax}\left(\frac{\mathbf{Q}^{(n)}(\mathbf{K}^{(n)})^{\top}}{\sqrt{d_h}} + \mathbf{B}\right)\mathbf{V}^{(n)}. \tag{8}$$

By biasing attention toward time steps that are dynamically related rather than merely positionally adjacent, this formulation allows the model to simultaneously capture semantic content and physical temporal structure within a single attention operation.

For each channel, the encoder output is mean-pooled over the L time steps and projected to the shared feature dimension. Stacking all channels yields the final textual path representation $\mathbf{Z}_{\mathrm{txt}} \in \mathbb{R}^{N\times d_f}$.

3.3.2 PHYSICS-STRUCTURED SSM FOR IMAGIFIED TIME SERIES

The visual path converts time series into GAF images and encodes them with a physics-structured Mamba. The DMD-derived physical prior parameterizes state blocks with effective decay and oscillation dynamics, constraining latent state evolution.

Visual Modality Construction Each channel is transformed into a Gramian Angular Field (GAF) image that represents pairwise temporal relationships. The image is divided into non-overlapping patches, which are flattened, linearly projected, and serialized into the embedding sequence $\mathbf{E}_{\mathrm{img}}$ for the physics-structured Mamba encoder. Implementation details are provided in Appendix B.2.

Physics-Structured State Matrix To process $\mathbf{E}_{\mathrm{img}}$ with physically consistent state transitions, the state matrix is constructed from the effective DMD-initialized parameters $\{(\tilde{\alpha}_k, \tilde{\omega}_k)\}_{k=1}^{K}$, rather than relying on unconstrained learnable parameters that offer no spectral guarantees and can cause drift over long horizons. Each dynamical mode k is encoded as a 2×2 rotation-decay block:

$$\mathbf{A}_k = \begin{bmatrix} \tilde{\alpha}_k & -\tilde{\omega}_k \\ \tilde{\omega}_k & \tilde{\alpha}_k \end{bmatrix}, \tag{9}$$

where $\tilde{\alpha}_k$ governs the decay rate and remains negative by construction, while $\tilde{\omega}_k$ governs the oscillation frequency of mode k. The full state matrix is assembled as:

$$\mathbf{A} = \mathrm{diag}(\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_K), \tag{10}$$

whose eigenvalues $\lambda_k^{\pm} = \tilde{\alpha}_k \pm i\tilde{\omega}_k$ match the effective model parameters. Here, K counts state blocks and $d_{\mathrm{state}} = 2K$, independently of the retained DMD rank r.

Physics-Informed State-Space Discretization Given the state matrix $\mathbf{A}$, zero-order hold discretization with patch-domain transition scale Δ_p yields $\bar{\mathbf{A}} = \exp(\mathbf{A}\Delta_p)$. Each mode admits the closed-form transition:

$$\bar{\mathbf{A}}_k = e^{\tilde{\alpha}_k \Delta_p} \begin{bmatrix} \cos(\tilde{\omega}_k \Delta_p) & -\sin(\tilde{\omega}_k \Delta_p) \\ \sin(\tilde{\omega}_k \Delta_p) & \cos(\tilde{\omega}_k \Delta_p) \end{bmatrix}. \tag{11}$$

Since $\tilde{\alpha}_k < 0$, each block is a scaled rotation with $\|\bar{\mathbf{A}}_k\|_2 = \rho(\bar{\mathbf{A}}_k) = e^{\tilde{\alpha}_k \Delta_p} < 1$ for $\Delta_p > 0$, ensuring contraction of the unforced state dynamics. Here, Δ_p indexes latent patch transitions rather than the original sampling interval.

The physics-structured Mamba encoder processes the patch embeddings $\mathbf{E}_{\mathrm{img}}$ sequentially. At patch position t, mode k evolves as:

$$\mathbf{h}_t^{(k)} = \bar{\mathbf{A}}_k \, \mathbf{h}_{t-1}^{(k)} + \mathbf{b}_k \, u_t^{(k)}, \tag{12}$$

where $u_t^{(k)}$ is the projection of the t-th patch embedding and $\mathbf{b}_k$ is an input vector. The mode states are concatenated into $\mathbf{h}_t = [(\mathbf{h}_t^{(1)})^\top, \ldots, (\mathbf{h}_t^{(K)})^\top]^\top$, encoding the evolution of visual features.

The resulting states are mean-pooled over patch positions and projected to d_f dimensions for each channel, yielding $\mathbf{Z}_{\mathrm{img}} \in \mathbb{R}^{N \times d_f}$.

3.3.3 Numerical Path

The numerical path processes raw observations $\mathbf{X}^{\mathrm{Num}}$ without modality conversion. Each channel is patched and projected into $\mathbf{E}_{\mathrm{num}}$, which is processed by a Transformer-Mamba encoder:

$$\mathbf{H}_{\mathrm{num}} = \text{Transformer-Mamba}(\mathbf{E}_{\mathrm{num}}). \tag{13}$$

The output $\mathbf{H}_{\mathrm{num}}$ is aggregated over the patch dimension and projected to the shared feature dimension d_f, yielding the numerical path representation $\mathbf{Z}_{\mathrm{num}} \in \mathbb{R}^{N \times d_f}$.

3.4 Tri-Path Feature Fusion

The three paths yield complementary representations that capture semantic, structural, and numerical aspects of the input time series. To consolidate these into a unified prediction, the three representations are concatenated along the feature dimension: $\mathbf{Z}_{\text{fuse}} = \left[\mathbf{Z}_{\text{txt}} \,|\, \mathbf{Z}_{\text{num}} \,|\, \mathbf{Z}_{\text{img}}\right] \in \mathbb{R}^{N \times 3d_f}$. A lightweight two-layer MLP prediction head maps the fused representation to the forecast horizon:

$$\widehat{\mathbf{Y}} = \mathbf{W}_2 \cdot \mathrm{GELU}\big(\mathbf{W}_1 \mathbf{Z}_{\text{fuse}}^\top + \mathbf{b}_1\big) + \mathbf{b}_2, \tag{14}$$

where $\mathbf{W}_1 \in \mathbb{R}^{d_h \times 3d_f}$ and $\mathbf{W}_2 \in \mathbb{R}^{H \times d_h}$ are learnable parameters. The output is inverse-normalized using statistics from the numerical path to recover the original scale.

3.5 Training Objective

The training objective combines a forecasting loss with physics regularization. The forecasting loss is the mean squared error over all variables and horizons: $\mathcal{L}_{\mathrm{pred}} = \frac{1}{H \cdot N} \sum_{h=1}^{H} \sum_{n=1}^{N} \left(\widehat{y}_h^{(n)} - y_h^{(n)}\right)^2$, where $\widehat{y}_h^{(n)}$ and $y_h^{(n)}$ denote the predicted and ground-truth values for variable n at horizon h. To constrain the learned corrections, an L_2 penalty is imposed on the residual offsets: $\mathcal{L}_{\mathrm{phys}} = \sum_{k=1}^{K} \left(\|\Delta\alpha_k\|^2 + \|\Delta\omega_k\|^2\right)$. The final objective is:

$$\mathcal{L} = \mathcal{L}_{\mathrm{pred}} + \lambda_{\mathrm{phys}} \mathcal{L}_{\mathrm{phys}}, \tag{15}$$

where λ_{phys} balances forecasting accuracy and adherence to the physical prior. Experimental results indicate that $\lambda_{\mathrm{phys}} = 0.2$ achieves a favorable trade-off between optimization stability and regularization effectiveness while avoiding over-constraining nonlinear dynamics modeling.

4 EXPERIMENTS

PhyTM is evaluated against nine baselines on long-term and short-term forecasting tasks, with key results summarized in Figure 3.

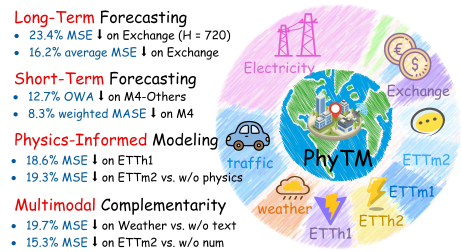


Figure 3: Evaluation summary of PhyTM.

Forecasting setup. Long-term forecasting covers eight benchmarks: ETTh1, ETTh2, ETTm1, ETTm2, Weather, Electricity, Traffic, and Exchange, using $L = 96$, $H \in \{96, 192, 336, 720\}$, and MSE/MAE metrics. Short-term forecasting is evaluated on the M4 benchmark (Makridakis et al., 2020) using SMAPE, MASE, and OWA. Train-validation-test splits are 6:2:2 for ETT and 7:1:2 for the other long-term benchmarks.

Implementation setup. All methods use consistent forecasting settings for fair comparison. PhyTM uses $K = 8$ physical modes across datasets and a hybrid encoder containing two Transformer layers and one Mamba layer. Ablation studies assess the contributions of physics-informed components and multimodal representations. Dataset, baseline, and implementation settings are provided in Appendix C.

4.1 OVERALL PERFORMANCE

Long-Term Forecasting. For long-term forecasting, Table 1 summarizes the average results over four forecast horizons, showing that PhyTM achieves competitive or superior performance across most datasets. Per-horizon results are provided in Appendix D. Notably, PhyTM reduces the average MSE on Exchange by 16.2% over the best baseline, with the reduction reaching 23.4% at $H = 720$. Consistent gains on ETTh2, ETTm1, ETTm2, and Weather demonstrate the effectiveness of DMD-derived physical priors across diverse temporal dynamics. In contrast, the more moderate performance on Traffic and Electricity reflects their higher dynamical complexity, as analyzed in Appendix H. Nevertheless, all datasets use the same $K = 8$ physical modes to ensure a unified comparison.

Table 1: Long-term forecasting results averaged over four forecast horizons $H \in \{96, 192, 336, 720\}$ with lookback length $L = 96$. **Bold**: best, Underline: second best.

Dataset	PhyTM		AURORA		AutoHFormer		TimeDRL		CycleNet		FilterNet		iTransformer		PatchTST		FITS		DLinear	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	0.433	0.432	0.438	0.435	0.442	0.446	0.441	0.438	**0.432**	**0.427**	0.440	0.432	0.454	0.448	0.453	0.446	0.451	0.440	0.456	0.452
ETTh2	**0.368**	**0.394**	0.378	0.401	0.375	0.405	0.378	0.405	0.383	0.404	0.378	0.404	0.383	0.407	0.385	0.410	0.383	0.408	0.509	0.445
ETTm1	**0.379**	**0.393**	0.386	0.402	0.385	0.404	0.382	0.402	0.383	0.397	0.384	0.398	0.407	0.410	0.396	0.406	0.415	0.408	0.403	0.407
ETTm2	**0.264**	**0.308**	0.284	0.324	0.272	0.313	0.279	0.324	0.269	0.315	0.276	0.322	0.288	0.332	0.287	0.330	0.286	0.328	0.350	0.401
Weather	**0.244**	**0.242**	0.249	0.275	0.247	0.251	**0.244**	0.270	0.245	0.271	0.245	0.272	0.258	0.278	0.256	0.279	0.249	0.276	0.265	0.317
Electricity	0.215	0.291	0.194	0.281	0.186	0.273	0.204	0.292	0.197	0.289	0.187	0.275	**0.178**	**0.270**	0.193	0.280	0.193	0.290	0.212	0.300
Exchange	**0.269**	**0.372**	0.336	0.403	0.321	0.399	0.349	0.412	0.330	0.407	0.326	0.422	0.335	0.416	0.336	0.423	0.335	0.420	0.345	0.416
Traffic	0.601	0.389	0.520	0.313	0.481	0.307	0.462	**0.276**	0.484	0.303	0.463	0.310	**0.428**	0.282	0.479	0.286	0.627	0.376	0.625	0.383

Short-Term Forecasting. For short-term forecasting, Table 2 summarizes the overall results on M4, where PhyTM achieves the best MASE and OWA. Complete results for individual subsets are provided in Appendix E. Specifically, PhyTM reduces the overall MASE by 8.3% compared with the best-performing baseline. On the Others subset, it further reduces OWA by 12.7% and MASE by 8.2% relative to the respective best baselines. These results demonstrate the effectiveness of PhyTM across diverse short-term forecasting settings.

Table 2: Short-term forecasting results on the M4 benchmark, with overall results summarized across all subsets. **Bold**: best, Underline: second best.

Metric	PhyTM	AURORA	AutoHFormer	TimeDRL	CycleNet	FilterNet	iTransformer	PatchTST	FITS	DLinear
SMAPE	12.230	12.433	12.353	12.487	12.887	12.274	12.685	**11.991**	12.454	13.641
MASE	**1.462**	1.734	1.634	1.760	1.860	1.595	1.770	1.671	1.729	2.097
OWA	**0.863**	0.929	0.914	0.943	0.980	0.896	0.956	0.894	0.924	1.043

4.2 Hyperparameter Sensitivity

The sensitivity of PhyTM to the number of DMD modes K, embedding dimension d_{model}, and delay-embedding length d_{emb} is evaluated on the four ETT datasets at $H = 96$. Each parameter is varied independently while the others remain fixed, with performance measured by percentage deviations in MSE and MAE from the optimal setting. Figure 4 shows that $K = 8$ and $d_{\text{emb}} = 8$ achieve the best performance, while increasing d_{model} beyond 128 provides limited additional gains. Reducing the dimension to 32 increases MSE by up to 30% on ETTh1, highlighting the importance of sufficient representation capacity. The gradual changes around the optimal settings support the robustness of PhyTM to moderate hyperparameter variations.

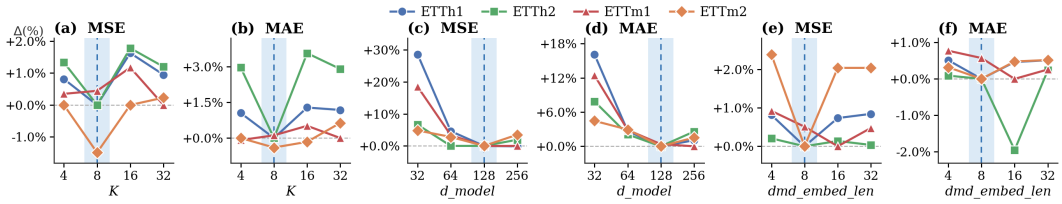


Figure 4: Hyperparameter sensitivity analysis with H=96.

4.3 Ablation Study

Ablation on Physics Components. Four variants assess the physical priors while retaining all three modality paths: ① disables DMD priors and replaces both physics-informed encoders with standard counterparts; ② replaces only Phys-Mamba with standard Mamba; ③ replaces only Phys-Transformer with a standard Transformer; and ④ is the complete model. Variants ② and ③ retain DMD and the other physics-informed encoder. Table 3 summarizes the results, with complete results in Appendix F.1. The complete model reduces MSE by 18.6% on ETTh1 and 19.3% on ETTm2 relative to ①. Replacing either encoder also degrades performance, supporting their complementary contributions.

Ablation on Tri-Modal Structure. Four variants assess modality contributions while retaining the physics components associated with the remaining paths: ① removes the textual path; ② removes the visual path; ③ removes the numerical path; and ④ is the complete model. Variants ①–③ exclude the removed path from fusion and retain the other two paths unchanged. Table 3 summarizes the results, with complete results in Appendix F.2. The complete model reduces MSE by 19.7% on Weather relative to ① and by 15.3% on ETTm2 relative to ③. Removing any path generally degrades performance, supporting the complementary value of the three modalities.

Table 3: Ablation studies of the physics-informed components and tri-modal structure at prediction horizon $H = 96$.

Physics Components							Tri-modal Structure						
Case	DMD	Phys-Trans.	Phys-Mamba	ETTh1	ETTm2	Weather	Case	Text	Image	Num	ETTh1	ETTm2	Weather
①	✗	✗	✗	0.452	0.212	0.179	①	✗	✓	✓	0.420	0.197	0.203
②	✓	✓	✗	0.449	0.186	0.169	②	✓	✗	✓	0.418	0.189	0.181
③	✓	✗	✓	0.376	0.176	0.178	③	✓	✓	✗	0.415	0.202	0.185
④	✓	✓	✓	**0.368**	**0.171**	**0.163**	④	✓	✓	✓	**0.368**	**0.171**	**0.163**

4.4 Physical Interpretability

To examine how the DMD-derived physical prior is reflected in learned temporal dynamics, we analyze its decay and oscillation characteristics in the temporal and spectral domains. Additional dynamical-complexity analysis and prediction visualizations are provided in Appendices H and I, respectively.

Temporal-Domain Analysis. Figure 5 visualizes the basis functions $e^{\tilde{\alpha}_k t}\cos(\tilde{\omega}_k t)$ using the shared learned parameters of each mode. Differences in oscillation frequency and attenuation reveal multiple temporal scales: slowly decaying modes represent persistent dynamics, while rapidly decaying modes describe shorter-lived variations. ETTh1 and ETTh2 exhibit pronounced oscillations, whereas ETTm2 displays smoother, rapidly attenuating patterns. These differences illustrate how the learned parameters adapt the dynamical basis to dataset-specific temporal characteristics.

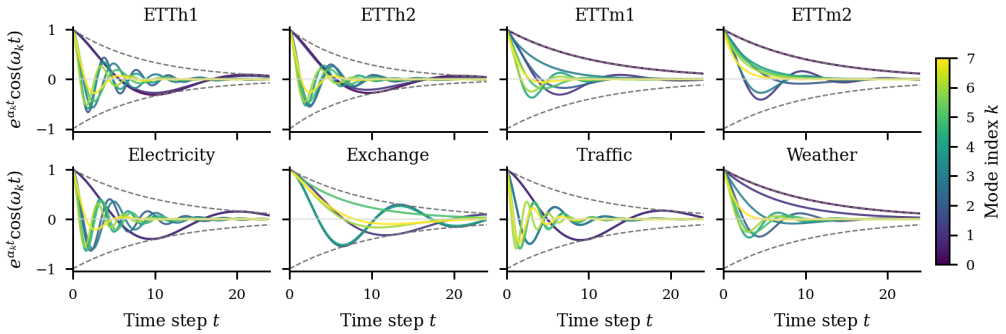
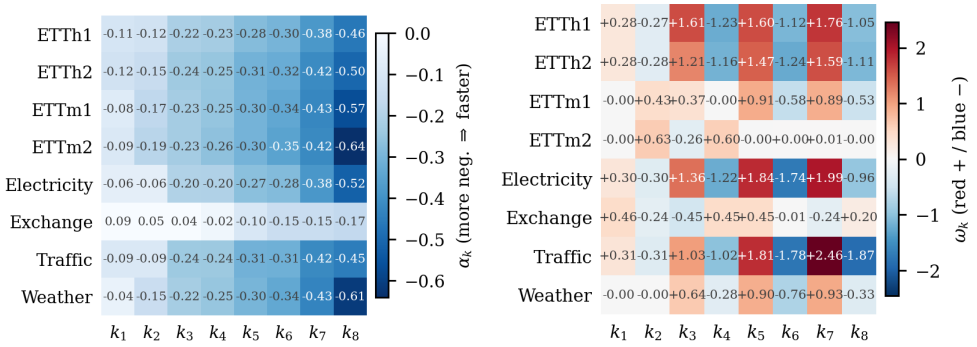


Figure 5: Temporal profiles of learned dynamical modes across eight datasets.

Spectral-Domain Analysis. Figure 6 visualizes the rate and angular frequency parameters of individual modes, providing a spectral view of their temporal behavior. More negative rates indicate faster attenuation, while larger absolute angular frequencies correspond to faster oscillations. Raw growth rates α_k are distinct from the strictly negative effective rates $\tilde{\alpha}_k$. Opposite frequency signs alone do not identify distinct oscillation scales, since the cosine basis depends on $|\omega_k|$. Appendix C.5 clarifies the parameter and dimension conventions.



(a) Growth/decay rates α_k. (b) Angular frequencies ω_k.

Figure 6: Spectral parameters across datasets.

5 Conclusion

This paper presents PhyTM, a physics-informed tri-modal forecasting framework that integrates DMD-based physical representations, physics-informed dual-stream attention, and physics-structured state-space modeling within a unified architecture. By incorporating physical priors into textual and visual temporal representations, PhyTM effectively enhances physical dynamics awareness and forecasting stability. Extensive experiments on both long-term and short-term forecasting benchmarks demonstrate that PhyTM achieves competitive forecasting performance across diverse datasets and forecasting horizons, validating the effectiveness of integrating physical dynamics into multimodal temporal modeling. While the current framework employs DMD-derived physical priors to characterize temporal evolution, future work will explore more expressive complex temporal dynamics and real-world heterogeneous data sources to further advance physics-informed temporal representation learning and forecasting in real-world applications.

AI USE STATEMENT

In this work, we used generative AI tools for language polishing and improving the readability of the manuscript. We have not used generative AI tools for developing theoretical or conceptual frameworks, formulating mathematical claims, designing or implementing research methods or experiments, analyzing data, or interpreting experimental results, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for refining the organization and presentation of the manuscript. We have reviewed all AI-assisted work. All AI-assisted content was carefully checked and revised by the authors to ensure accuracy and consistency with the actual research. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Lei Bai, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. Adaptive Graph Convolutional Recurrent Network for Traffic Forecasting. In *Advances in Neural Information Processing Systems*, volume 33, pp. 17804–17815, 2020.

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

Ricardo Buitrago Ruiz, Tanya Marwah, Albert Gu, and Andrej Risteski. On the benefits of memory for modeling time-dependent pdes. In *International Conference on Learning Representations*, volume 2025, pp. 54972–55002, 2025.

Andres Manuel Chacon Maldonado, Gualberto Asencio Cortes, and Alicia Troncoso. A multimodal hybrid deep learning approach for pest forecasting using time series and satellite images. *Information Fusion*, 124:103350, 2025.

Cristian Challu, Kin G. Olivares, Boris N. Oreshkin, Federico Garza Ramirez, Max Mergenthaler Canseco, and Artur Dubrawski. NHITS: Neural Hierarchical Interpolation for Time Series Forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 6989–6997, 2023.

Ching Chang, Chiaotung Chan, Weiyao Wang, Wenchih Peng, and Tienfu Chen. TimeDRL: Disentangled representation learning for multivariate time-series. In *IEEE International Conference on Data Engineering (ICDE)*, pp. 625–638, 2024.

Wenxia Chang, Chao Zhang, Wentao Li, and Deyu Li. MIGC-CMamba: Cross-domain mamba with multi-scale imaging and granular-ball computing for traffic flow prediction. In *Proceedings of the ACM Web Conference*, pp. 7167–7177, 2026.

Mouxiang Chen, Lefei Shen, Zhuo Li, Xiaoyun Joy Wang, Jianling Sun, and Chenghao Liu. VisionTS: Visual masked autoencoders are free-lunch zero-shot time series forecasters. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 8979–9007. ICML, 2025.

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*, volume 2024, pp. 10527–10545, 2024a.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K. Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Xiaodan Chen, Xiucheng Li, Xinyang Chen, and Zhijun Li. Structured Matrix Basis for Multivariate Time Series Forecasting with Interpretable Dynamics. In *Advances in Neural Information Processing Systems*, volume 37, pp. 24326–24349, 2024b.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 1724–1734, 2014.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized Models and Efficient Algorithms Through Structured State Space Duality. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 10041–10071, 2024.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 10148–10167, 2024.

Jiaxiang Dong, Haixu Wu, Yuxuan Wang, Yun-Zhong Qiu, Li Zhang, Jianmin Wang, and Mingsheng Long. TimeSiam: A pre-training framework for siamese time-series modeling. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 11412–11436, 2024.

Hatice Vildan Dudukcu, Murat Taskiran, Zehra Gulru Cam Taskiran, and Tulay Yildirim. Temporal convolutional networks with Rnn approach for chaotic time series prediction. *Applied Soft Computing*, 133:109945, 2023.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A Family of Open Time-series Foundation Models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 16115–16152, 2024.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G. Wilson. Large Language Models Are Zero-Shot Time Series Forecasters. In *Advances in Neural Information Processing Systems*, volume 36, pp. 19622–19635, 2023.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *Conference on Language Modeling*, 2024.

Albert Gu, Karan Goel, Ankit Gupta, and Christopher Ré. On the Parameterization and Initialization of Diagonal State Space Models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 35971–35983, 2022a.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022b.

Qi Guo, Baolin Zhao, Mingchen Song, and Guoqiang Zhong. A survey of deep learning for time series forecasting: taxonomy, analysis and future directions. *IEEE Transactions on Knowledge and Data Engineering*, 38(5):2541–2560, 2026.

Lu Han, Han-Jia Ye, and De-Chuan Zhan. SIN: Selective and interpretable normalization for long-term time series forecasting. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 17437–17453, 2024.

Jiaxi Hu, Disen Lan, Ziyu Zhou, Gefeng Luo, Qingsong Wen, and Yuxuan Liang. How to train your mamba for time series forecasting. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 476–485, 2026.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. In *International Conference on Learning Representations*, volume 2024, pp. 23857–23880, 2024.

George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3:422–440, 2021.

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.

Zhichen Lai, Dalin Zhang, Huan Li, Christian S Jensen, Hua Lu, and Yan Zhao. Lightcts: A lightweight framework for correlated time series forecasting. *Proceedings of the ACM on Management of Data*, 1(2):1–26, 2023.

Wenbing Li, Hang Zhou, Junqing Yu, Zikai Song, and Wei Yang. Coupled Mamba: Enhanced Multi-modal Fusion with Coupled State Space Model. In *Advances in Neural Information Processing Systems*, volume 37, pp. 59808–59832, 2024.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In *International Conference on Learning Representations*, 2018.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier Neural Operator for Parametric Partial Differential Equations. In *International Conference on Learning Representations*, 2021.

Shengsheng Lin, Weiwei Lin, Xinyi Hu, Wentai Wu, Ruichao Mo, and Haocheng Zhong. Cyclenet: Enhancing time series forecasting through modeling periodic patterns. *Advances in Neural Information Processing Systems*, 37:106315–106345, 2024.

Shengsheng Lin, Weiwei Lin, Wentai Wu, Haojun Chen, and CL Philip Chen. SparseTSF: Lightweight and robust time series forecasting via sparse modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(1):170–183, 2026.

Haoxin Liu, Shangqing Xu, Zhiyuan Zhao, Lingkai Kong, Harshavardhan Kamarthi, Aditya B. Sasanur, Megha Sharma, Jiaming Cui, Qingsong Wen, Chao Zhang, and B. A. Prakash. Time-MMD: Multi-Domain Multimodal Dataset for Time Series Analysis. In *Advances in Neural Information Processing Systems*, volume 37, pp. 77888–77933, 2024a.

Peiyuan Liu, Beiliang Wu, Yifan Hu, Naiqi Li, Tao Dai, Jigang Bao, and Shu-Tao Xia. Time-Bridge: Non-stationarity matters for long-term time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 39815–39840, 2025a.

Yong Liu, Chenyu Li, Jianmin Wang, and Mingsheng Long. Koopa: Learning non-stationary time series dynamics with koopman predictors. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

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*, pp. 11116–11140, 2024b.

Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Timer: Generative Pre-trained Transformers Are Large Time Series Models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 32369–32399, 2024c.

Yong Liu, Guo Qin, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Timer-XL: Long-context transformers for unified time series forecasting. In *International Conference on Learning Representations*, volume 2025, pp. 83982–84006, 2025b.

Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. VMamba: Visual State Space Model. In *Advances in Neural Information Processing Systems*, volume 37, pp. 103031–103063, 2024d.

Donghao Luo and Xue Wang. ModernTCN: A modern pure convolution structure for general time series analysis. In *International conference on learning representations*, volume 2024, pp. 31728–31770, 2024.

Yingtao Luo, Qiang Liu, Yuntian Chen, Wenbo Hu, Tian Tian, and Jun Zhu. Physics-guided discovery of highly nonlinear parametric partial differential equations. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1595–1607, 2023.

Yingtao Luo, Shikai Fang, Binqing Wu, Qingsong Wen, and Liang Sun. Physics-guided learning of meteorological dynamics for weather downscaling and forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2010–2020, 2025.

Qianli Ma, Zhen Liu, Zhenjing Zheng, Ziyang Huang, Siying Zhu, Zhongzhong Yu, and James T Kwok. A survey on time-series pre-trained models. *IEEE Transactions on Knowledge and Data Engineering*, 36(12):7536–7555, 2024.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. The m4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1):54–74, 2020.

James Morrill, Cristopher Salvi, Patrick Kidger, and James Foster. Neural rough differential equations for long time series. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 7829–7838, 2021.

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.

Antonio Orvieto, Samuel L. Smith, Albert Gu, Anushan Fernando, Caglar Gulcehre, Razvan Pascanu, and Soham De. Resurrecting recurrent neural networks for long sequences. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 26670–26698, 2023.

Youngjin Park, Anh Tong, Sehyun Lee, Jihyeon Seong, Qin Xie, and Jaesik Choi. Towards transparent time series analysis: Exploring methods and enhancing interpretability. *ACM Computing Surveys*, 58(9):240, 2026.

Xiangfei Qiu, Jilin Hu, Lekui Zhou, Xingjian Wu, Junyang Du, Buang Zhang, Chenjuan Guo, Aoying Zhou, Christian S. Jensen, Zhenli Sheng, and Bin Yang. TFB: Towards comprehensive and fair benchmarking of time series forecasting methods. *Proceedings of the VLDB Endowment*, 17(9): 2363–2377, 2024.

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.

Syama Sundar Rangapuram, Matthias W. Seeger, Jan Gasthaus, Lorenzo Stella, Yuyang Wang, and Tim Januschowski. Deep state space models for time series forecasting. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 8857–8868, 2021.

Xiaoming Shi, Shiyu Wang, Yuqi Nie, Dianqi Li, Zhou Ye, Qingsong Wen, and Ming Jin. Time-MoE: Billion-scale time series foundation models with mixture of experts. In *International Conference on Learning Representations*, volume 2025, pp. 34635–34667, 2025.

Jimmy T. H. Smith, Andrew Warrington, and Scott W. Linderman. Simplified state space layers for sequence modeling. In *International Conference on Learning Representations*, 2023.

Naoya Takeishi, Yoshinobu Kawahara, and Takehisa Yairi. Learning koopman invariant subspaces for dynamic mode decomposition. *Advances in Neural Information Processing Systems*, 30, 2017.

Mingtian Tan, Mike A. Merrill, Vinayak Gupta, Tim Althoff, and Thomas Hartvigsen. Are Language Models Actually Useful for Time Series Forecasting? In *Advances in Neural Information Processing Systems*, volume 37, pp. 60162–60191, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Bingjie Wang, Chao Zhang, Wentao Li, and Deyu Li. MF3: Multimodal federated learning with dual-path mamba-transformer for metro flow prediction. In *Proceedings of the ACM Web Conference*, pp. 5451–5462, 2026.

Hao Wang, Lichen Pan, Yuan Shen, Zhichao Chen, Degui Yang, Yifei Yang, Sen Zhang, Xinggao Liu, Haoxuan Li, and Dacheng Tao. FreDF: Learning to forecast in the frequency domain. In *International Conference on Learning Representations*, volume 2025, pp. 6893–6922, 2025a.

Huqiang 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 *International Conference on Learning Representations*, volume 2024, pp. 38626–38652, 2024.

Shiyu Wang, Jiawei Li, Xiaoming Shi, Zhou Ye, Baichuan Mo, Wenze Lin, Shengtong Ju, Zhixuan Chu, and Ming Jin. TimeMixer++: A general time series pattern machine for universal predictive analysis. In *International Conference on Learning Representations*, volume 2025, pp. 16980–17016, 2025b.

Gerald Woo, Chenghao Liu, Doyen Sahoo, Akshat Kumar, and Steven C. H. Hoi. CoST: Contrastive learning of disentangled seasonal-trend representations for time series forecasting. In *International Conference on Learning Representations*, 2022.

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, pp. 53140–53164, 2024.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis. In *International Conference on Learning Representations*, 2023.

Xingjian Wu, Jianxin Jin, Wanghui Qiu, Peng Chen, Yang Shu, Bin Yang, and Guo Guo. Aurora: Towards universal generative multimodal time series forecasting. In *International Conference on Learning Representations*, volume 2026, pp. 103239–103277, 2026.

Zhijian Xu, Ailing Zeng, and Qiang Xu. FITS: Modeling time series with $10k$ parameters. In *International Conference on Learning Representations*, pp. 26295–26318, 2024.

Hao Xue and Flora D Salim. Promptcast: A new prompt-based learning paradigm for time series forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 36(11):6851–6864, 2023.

Kun Yi, Jingru Fei, Qi Zhang, Hui He, Shufeng Hao, Defu Lian, and Wei Fan. Filternet: Harnessing frequency filters for time series forecasting. *Advances in Neural Information Processing Systems*, 37:55115–55140, 2024.

Xinyu Yuan and Yan Qiao. Diffusion-ts: Interpretable diffusion for general time series generation. In *International Conference on Learning Representations*, volume 2024, pp. 41582–41610, 2024.

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.

Kexin Zhang, Qingsong Wen, Chaoli Zhang, Rongyao Cai, Ming Jin, Yong Liu, James Y Zhang, Yuxuan Liang, Guansong Pang, Dongjin Song, et al. Self-supervised learning for time series analysis: Taxonomy, progress, and prospects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(10):6775–6794, 2024.

Qianru Zhang, Honggang Wen, Ming Li, Dong Huang, Siuming Yiu, Christian S. Jensen, and Pietro Lio. AutoHFormer: Efficient hierarchical autoregressive transformer for time series prediction. In *IEEE International Conference on Data Engineering (ICDE)*, 2026.

Yunhao Zhang and Junchi Yan. Crossformer: Transformer Utilizing Cross-Dimension Dependency for Multivariate Time Series Forecasting. In *International Conference on Learning Representations*, 2023.

Jingyuan Zhao, Fulin Chu, Lili Xie, Yunhong Che, Yuyan Wu, and Andrew F Burke. A survey of transformer networks for time series forecasting. *Computer Science Review*, 60:100883, 2026.

Zhiyuan Zhao, Xueying Ding, and B. Aditya Prakash. PINNsFormer: A transformer-based framework for physics-informed neural networks. In *International Conference on Learning Representations*, 2024.

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.

Shiqiao Zhou, Holger Schoner, Huanbo Lyu, Edouard Fouche, and Shuo Wang. BALM-TSF: Balanced multimodal alignment for llm-based time series forecasting. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 4498–4508, 2025.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 27268–27286. ICML, 2022.

Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 62429–62442, 2024.

A Problem Definition

Time series data are typically observed as sequential measurements of evolving real-world systems, where they are continuously collected from diverse monitoring and sensing applications. This section formally defines the multivariate time series forecasting problem with optional multimodal inputs and physical priors.

Definition 1 (Multivariate Time Series Forecasting). A multivariate time series is represented as $\mathbf{X} \in \mathbb{R}^{T\times N}$, where T is the total sequence length and N is the number of variables. Each observation $\mathbf{x}_t \in \mathbb{R}^N$ at time step t represents the values of all N variables. Given a lookback window of length L ending at time t, the input segment is denoted as $\mathbf{X}_t = [\mathbf{x}_{t-L+1}, \ldots, \mathbf{x}_t] \in \mathbb{R}^{L\times N}$. Together with optional auxiliary modalities derived from $\mathbf{X}_t$, the goal is to learn a mapping $\mathcal{F}$ to predict the next H steps:

$$[\mathbf{x}_{t-L+1}, \ldots, \mathbf{x}_t] \xrightarrow{\mathcal{F}} [\hat{\mathbf{x}}_{t+1}, \ldots, \hat{\mathbf{x}}_{t+H}], \tag{16}$$

where $\hat{\mathbf{x}}_{t+h}$ should be close to the ground truth $\mathbf{x}_{t+h}$ under mean squared error.

Multivariate time series are often associated with heterogeneous contextual information. To fully exploit complementary temporal characteristics, the input window $\mathbf{X}_t$ is represented in three modalities.

Definition 2 (Multimodal Time Series). Given the original time series $\mathbf{X}$, three complementary modalities are constructed:

- **Numerical modality** $\mathbf{X}^{\mathrm{Num}}$ is a series of raw observation sequences derived from $\mathbf{X}$, preserving the original temporal structure for direct sequence modeling.
- **Textual modality** $\mathbf{X}^{\mathrm{Text}}$ is a collection of structured natural language prompts obtained by independently constructing a prompt for each channel, where each prompt incorporates dataset-level metadata, channel-specific statistics, and the forecasting task specification.
- **Visual modality** $\mathbf{X}^{\mathrm{Image}}$ is a set of Gramian Angular Field (GAF) images obtained by independently transforming each channel of $\mathbf{X}$, where each image encodes temporal correlations as a structured representation via angular relationships between observations.

All three modalities share the same temporal axis and provide complementary projections of the same underlying dynamical process.

Multimodal information provides contextual signals aligned with the time series, and physical information is further introduced to characterize the underlying temporal evolution of the observed sequence.

Definition 3 (Physics-Informed Representation). The delay-embedded states $\mathbf{s}_t = [\mathbf{x}_t^\top, \mathbf{x}_{t-1}^\top, \ldots, \mathbf{x}_{t-d_{\mathrm{emb}}+1}^\top]^\top$ are approximated by a linear evolution operator $\mathbf{A}$:

$$\mathbf{s}_{t+1} \approx \mathbf{A}\,\mathbf{s}_t. \tag{17}$$

The DMD framework estimates $\mathbf{A}$ from historical data and decomposes it into K dominant dynamical modes, yielding a physics-informed representation $\Phi(\mathbf{X}) = \{\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_K\}$, where each mode $\boldsymbol{\phi}_k$ is characterized by a growth/decay rate α_k and oscillation frequency ω_k.

B Modality Construction Details

PhyTM constructs textual and visual modalities from the original numerical time series to capture complementary temporal information. The textual modality uses structured prompts, while the visual modality encodes pairwise temporal relationships through GAF images.

B.1 Textual Prompt Construction and Temporal Alignment

The textual modality $\mathbf{X}^{\mathrm{Text}}$ is constructed by building a structured natural language prompt for each channel of the original time series $\mathbf{X}$. Each prompt incorporates dataset-level metadata (domain,

sampling frequency, and seasonality), channel-specific statistics (mean μ_n, standard deviation σ_n, and trend τ_n over the lookback window), recent observations, and the forecasting task specification. A typical prompt takes the form:

> *"**Dataset:** electricity transformer temperature, hourly, with daily and weekly seasonality. **Stats:** mean = μ_n, std = σ_n, trend = τ_n. **Recent steps:** $x_{t-L+1}^{(n)}, \ldots, x_t^{(n)}$. **Task:** predict the next H steps."*

Since each prompt is constructed from channel-specific statistics and observations, prompts differ across channels and yield distinct semantic representations. Each prompt is tokenized and passed through a frozen GPT-2 encoder to obtain contextual token states. To preserve temporal resolution, the hidden states of the tokens corresponding to each observation are mean-pooled, rather than using only the final token of the entire prompt. Tokenizer character offsets identify the token span of each numeric field, allowing observations represented by multiple tokens to be aligned with their original time steps.

Each pooled vector is mapped by a shared learnable linear layer to the feature dimension d_f. Stacking these vectors in chronological order yields $\mathbf{E}^{(n)} \in \mathbb{R}^{L \times d_f}$ for channel n, where L is the lookback length. Across all N channels, the resulting representation is $\mathbf{E} \in \mathbb{R}^{N \times L \times d_f}$. Metadata and statistics precede the observations in the prompt, providing context under GPT-2's causal attention mask. The physics-informed Transformer then applies temporal attention independently within each channel using shared parameters.

B.2 GAF Image Construction and Patch Embedding

The visual modality $\mathbf{X}^{\text{Image}}$ transforms each channel of the original time series $\mathbf{X}$ into a GAF image, which encodes pairwise temporal correlations as a spatially structured 2D representation. For each channel, observations are first normalized to $[-1, 1]$ to obtain $\tilde{x}_t$, then mapped to polar angles via $\varphi_t = \arccos(\tilde{x}_t)$. The GAF image is constructed entry-wise as:

$$G[i, j] = \cos(\varphi_i + \varphi_j), \tag{18}$$

where each entry $G[i, j]$ encodes the angular relationship between time steps i and j, preserving temporal ordering while exposing periodic and structural patterns that are difficult to identify directly from raw sequences. Each image is then divided into non-overlapping spatial patches, which are serialized into an ordered sequence of visual tokens with each patch independently flattened and linearly projected, yielding the patch embedding sequence $\mathbf{E}_{\text{img}}$.

C Experimental Setup

This section provides the detailed experimental setup for evaluating PhyTM, including benchmark datasets, evaluation metrics, baseline methods, and implementation details. The experiments cover both long-term and short-term forecasting under consistent evaluation settings to ensure a comprehensive comparison.

C.1 Datasets

The experiments cover both long-term and short-term forecasting benchmarks with diverse domains, sampling frequencies, data scales, and temporal granularities. Detailed statistics are provided below.

Long-Term Forecasting. For long-term forecasting, PhyTM is evaluated on eight widely used benchmark datasets: ETTh1, ETTh2, ETTm1, ETTm2, Weather, Electricity, Traffic, and Exchange. The four ETT datasets contain electricity transformer temperature measurements at hourly or 15-minute intervals, while Weather, Electricity, Traffic, and Exchange cover meteorological, energy, transportation, and financial domains, respectively. These datasets vary substantially in the number of variables, sequence length, and sampling frequency, providing diverse temporal dynamics for evaluating long-horizon forecasting. Following prior work (Qiu et al., 2024), the ETT datasets use a 6:2:2 train-validation-test split, while the remaining datasets follow a 7:1:2 split. The input length is fixed at $L = 96$, with prediction horizons $H \in \{96, 192, 336, 720\}$. Detailed statistics are summarized in Table 4.

Table 4: Statistics of benchmark datasets used for long-term forecasting. The dataset split denotes the ratio of samples in the training, validation, and test sets, respectively.

Dataset	Domain	Channels	Timesteps	Frequency	Split	Input Length	Prediction Length
ETTh1	Electricity	7	17,420	1 hour	6:2:2	96	{96, 192, 336, 720}
ETTh2	Electricity	7	17,420	1 hour	6:2:2	96	{96, 192, 336, 720}
ETTm1	Electricity	7	69,680	15 min	6:2:2	96	{96, 192, 336, 720}
ETTm2	Electricity	7	69,680	15 min	6:2:2	96	{96, 192, 336, 720}
Weather	Meteorology	21	52,696	10 min	7:1:2	96	{96, 192, 336, 720}
Electricity	Energy	321	26,304	1 hour	7:1:2	96	{96, 192, 336, 720}
Traffic	Transportation	862	17,544	1 hour	7:1:2	96	{96, 192, 336, 720}
Exchange	Finance	8	7,588	Daily	7:1:2	96	{96, 192, 336, 720}

Table 5: Statistics of the M4 dataset used for short-term forecasting, broken down by sampling frequency and domain. Forecast horizons follow the original M4 competition protocol.

Frequency	Demographic	Finance	Industry	Macro	Micro	Other	Total	Horizon
Yearly	1,088	6,519	3,716	3,903	6,538	1,236	23,000	6
Quarterly	1,858	5,305	4,637	5,315	6,020	865	24,000	8
Monthly	5,728	10,987	10,017	10,016	10,975	277	48,000	18
Weekly	24	164	6	41	112	12	359	13
Daily	10	1,559	422	127	1,476	633	4,227	14
Hourly	0	0	0	0	0	414	414	48
Total	8,708	24,534	18,798	19,402	25,121	3,437	100,000	–

Short-Term Forecasting. For short-term forecasting, PhyTM is evaluated on the M4 dataset, which contains 100,000 time series from six domains and six sampling frequencies: Yearly, Quarterly, Monthly, Weekly, Daily, and Hourly. Unlike the fixed forecasting horizons used in long-term forecasting, M4 adopts frequency-specific horizons following the original competition protocol, ranging from 6 steps for Yearly series to 48 steps for Hourly series. This benchmark therefore provides a complementary evaluation across heterogeneous temporal granularities and forecasting horizons. Detailed statistics are summarized in Table 5.

C.2 Evaluation Metrics

Forecasting performance is evaluated using the standard metrics associated with the long-term and short-term forecasting protocols. For long-term forecasting, MSE and MAE are adopted. For short-term forecasting on M4, sMAPE, MASE, and OWA are reported. Lower values indicate better forecasting performance for all metrics.

MSE. MSE measures the average squared deviation between the predicted and ground-truth values:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2, \tag{19}$$

where y_i and $\hat{y}_i$ denote the ground-truth and predicted values, respectively, and N is the total number of evaluated observations. MSE assigns larger penalties to large forecasting errors.

MAE. MAE measures the average absolute deviation between predictions and ground truth:

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|y_i - \hat{y}_i\right|. \tag{20}$$

Compared with MSE, MAE provides a direct measure of the average prediction error and is less sensitive to large deviations.

sMAPE. For short-term forecasting on M4, sMAPE measures the relative forecasting error while accounting for the scale of both the prediction and the ground-truth value:

$$\mathrm{sMAPE} = \frac{200}{H} \sum_{i=1}^{H} \frac{|y_i - \hat{y}_i|}{|y_i| + |\hat{y}_i|}, \tag{21}$$

where H denotes the forecasting horizon.

MASE. MASE evaluates forecasting errors relative to a seasonal naive benchmark:

$$\mathrm{MASE} = \frac{\frac{1}{H} \sum_{i=1}^{H} |y_i - \hat{y}_i|}{\frac{1}{T-m} \sum_{t=m+1}^{T} |y_t - y_{t-m}|}, \tag{22}$$

where T denotes the length of the in-sample series and m is its seasonal period. The denominator corresponds to the in-sample error of the seasonal naive forecasting method, making MASE comparable across series with different scales.

OWA. OWA is the primary aggregate metric adopted by the M4 competition and combines the relative sMAPE and MASE with respect to the Naive2 benchmark:

$$\mathrm{OWA} = \frac{1}{2} \left(\frac{\mathrm{sMAPE}}{\mathrm{sMAPE}_{\mathrm{Naive2}}} + \frac{\mathrm{MASE}}{\mathrm{MASE}_{\mathrm{Naive2}}} \right). \tag{23}$$

An OWA value below 1 indicates that the forecasting method performs better than the Naive2 benchmark on the combined sMAPE and MASE criterion.

C.3 Baselines

PhyTM is compared with nine representative forecasting methods covering linear, Transformer-based, and foundation-model architectures. These baselines encompass different temporal modeling paradigms, including time-domain decomposition, frequency-domain modeling, patch-based representation, attention-based dependency modeling, representation learning, and large-scale pretrained forecasting. Table 6 summarizes their publication year, architectural category, main characteristics, and publicly available implementations.

DLinear. DLinear (Zeng et al., 2023) is a lightweight linear forecasting model that decomposes time series into trend and remainder components and applies linear projection to the decomposed components for prediction. Its simple architecture provides an efficient baseline for evaluating whether more complex temporal modeling mechanisms bring additional forecasting benefits.

FilterNet. FilterNet (Yi et al., 2024) performs time-series representation learning through structured channel mixing and frequency filtering. By exploiting frequency-domain characteristics and interactions among variables, it provides an efficient alternative to attention-based forecasting architectures.

CycleNet. CycleNet (Lin et al., 2024) explicitly models recurrent periodic patterns in time series. It exploits cyclic structures to characterize repeated temporal behavior and improve forecasting when observations exhibit pronounced periodicity.

FITS. FITS (Xu et al., 2024) is a lightweight frequency-domain forecasting method that performs interpolation on frequency components. Its compact architecture enables efficient prediction with a small number of trainable parameters.

iTransformer. iTransformer (Liu et al., 2024b) applies the Transformer architecture over the variate dimension, treating individual variables as tokens to capture multivariate correlations while representing temporal observations as their features.

PatchTST. PatchTST (Nie et al., 2023) segments time series into local patches and employs channel-independent Transformer encoding. The patching strategy provides a compact representation of local temporal patterns and facilitates the modeling of long-range dependencies.

TimeDRL. TimeDRL (Chang et al., 2024) learns disentangled temporal representations through dual-level contrastive objectives. It aims to capture robust temporal characteristics that can support downstream time-series forecasting.

AutoHFormer. AutoHFormer (Zhang et al., 2026) models temporal dependencies through automatic hierarchical attention within a Transformer framework, allowing temporal information at different levels to be incorporated into forecasting representations.

AURORA. AURORA (Wu et al., 2026) is a large-scale pretrained foundation model for time-series forecasting. It incorporates multimodal information to support forecasting across heterogeneous datasets and application domains.

Table 6: Detailed summary of baseline models used in our experiments.

Model	Year	Type	Description	Code
DLinear	2023	Linear	Decomposes time series into trend and remainder components before linear projection for forecasting.	https://github.com/cure-lab/LTSF-Linear
FilterNet	2024	Linear	Performs feature extraction through structured channel mixing and frequency filtering.	https://github.com/aikunyi/FilterNet
CycleNet	2024	Linear	Explicitly models periodic cycles to capture recurrent temporal dependencies.	https://github.com/ACAT-SCUT/CycleNet
FITS	2024	Linear	Performs lightweight forecasting through frequency-domain interpolation with a small number of parameters.	https://github.com/VEWOXIC/FITS
iTransformer	2024	Transformer	Applies self-attention across the variate dimension to model multivariate correlations.	https://github.com/thuml/iTransformer
PatchTST	2023	Transformer	Tokenizes time series into local patches with channel-independent Transformer encoding.	https://github.com/yuqinie98/PatchTST
TimeDRL	2024	Transformer	Learns disentangled temporal representations through dual-level contrastive objectives for robust time-series modeling.	https://github.com/blacksnail789521/TimeDRL
AutoHFormer	2026	Transformer	Models temporal dependencies through automatic hierarchical attention within a Transformer framework.	https://github.com/CoderPowerBeyond/AutoHFormer
AURORA	2026	Foundation	Incorporates multimodal inputs within a large-scale pretrained model for universal time-series forecasting across diverse domains.	https://github.com/decisionintelligence/Aurora

C.4 Implementation Details

All experiments are conducted on an NVIDIA RTX 5880 Ada Generation GPU with 48 GB memory using PyTorch 2.1. The forecasting, model, and training configurations are summarized as follows.

- *Forecasting settings:* For long-term forecasting, the lookback window is fixed at $L = 96$, and four prediction horizons $H \in \{96, 192, 336, 720\}$ are evaluated. For short-term forecasting, experiments are conducted on the M4 benchmark (Makridakis et al., 2020), with forecasting horizons determined by the corresponding temporal granularities.

- *Model configuration:* For the DMD module, the number of physical modes is fixed at $K = 8$ across all datasets. The hybrid encoder consists of 2 Transformer layers and 1 Mamba layer, with embedding dimension $d_{\text{model}} = 128$ and $n_{\text{heads}} = 8$ attention heads.
- *Training configuration:* All models are trained using the Adam optimizer with a learning rate of 1×10^{-4}, weight decay of 1×10^{-4}, batch size of 16, and dropout rate of 0.1. Training is conducted for at most 30 epochs, with early stopping applied when the validation loss does not improve for 8 consecutive epochs.

C.5 DMD Estimation and Parameter Conventions

Training-only Estimation. The DMD spectrum is estimated once from the training portion of each dataset and retained as a fixed physical reference. Specifically, a training sequence of length T_{DMD} is used to construct the snapshot matrices and obtain the raw DMD spectrum, while the residual corrections are jointly optimized with the forecasting model. DMD is neither recomputed for individual forecasting windows nor fitted on validation or test data. Accordingly, all DMD-derived and learned parameters remain fixed during validation and testing, where inference requires forward propagation only. Here, T_{DMD} denotes the sequence length used for DMD fitting and is distinct from both the total dataset length T and the forecasting lookback length L.

Raw and Effective Parameters. The DMD procedure yields raw growth/decay rates α_k and angular frequencies ω_k as a fixed data-derived spectral reference. To construct contractive state transitions, we define a stable DMD reference and its log-decay coordinate by

$$\alpha_k^{(0)} = \begin{cases} -|\alpha_k|, & \alpha_k \neq 0, \\ -\varepsilon_{\text{decay}}, & \alpha_k = 0, \end{cases} \qquad \ell_k = \log(-\alpha_k^{(0)}),$$

where $\varepsilon_{\text{decay}} > 0$ is a fixed small decay rate in the same units as α_k, used only to avoid a zero decay magnitude and an undefined logarithm. Negative raw rates are preserved, whereas positive raw growth rates are reflected to negative rates of equal magnitude. Thus, the stable reference encodes a stability constraint, not a claim that growing raw DMD modes describe decaying dynamics. The effective parameters are

$$\tilde{\alpha}_k = -\exp(\ell_k + \Delta\alpha_k), \qquad \tilde{\omega}_k = \omega_k + \Delta\omega_k.$$

Here, ℓ_k and ω_k are fixed after DMD estimation, while the residuals are initialized to zero and learned jointly with the forecasting model. Consequently,

$$\tilde{\alpha}_k = \alpha_k^{(0)} \exp(\Delta\alpha_k), \qquad \Delta\alpha_k = 0 \implies \tilde{\alpha}_k = \alpha_k^{(0)}.$$

The decay residual therefore provides a multiplicative adjustment of the stable reference magnitude, and its regularization acts in the log-decay coordinate. The exponential parameterization ensures $\tilde{\alpha}_k < 0$ throughout optimization; independent residuals need not preserve the ordering of decay magnitudes after training. Thus, (α_k, ω_k) denote raw DMD estimates, $\alpha_k^{(0)}$ denotes the stable reference rate, and $(\tilde{\alpha}_k, \tilde{\omega}_k)$ are the effective parameters used in both the attention bias and the state-space transition. The learned corrections are shared across input windows of the same trained model, so the effective spectrum remains fixed at inference and requires no window-wise parameter averaging.

DMD Spectrum and State-Space Structure. The retained DMD rank r and the number of model state blocks K describe different quantities. The former determines the low-rank snapshot approximation used for DMD estimation, whereas the latter specifies the number of 2×2 rotation–decay blocks in the physics-structured state matrix, yielding $d_{\text{state}} = 2K$; therefore, $r = 2K$ is not required. For a single N-channel training sequence, the snapshot rank satisfies

$$\operatorname{rank}(\mathbf{S}) \leq \min(N d_{\text{emb}}, T_{\text{DMD}} - d_{\text{emb}}),$$

which depends on the training snapshot matrix rather than the length of an individual validation or test window. Moreover, because the snapshot matrix is real-valued, nonreal DMD eigenvalues occur in complex-conjugate pairs. For equal decay rates, opposite frequencies yield the same cosine basis, and their corresponding real state blocks satisfy

$$\mathbf{A}_k(-\omega) = \mathbf{R}\mathbf{A}_k(\omega)\mathbf{R}^{-1}, \qquad \mathbf{R} = \operatorname{diag}(1, -1).$$

Hence, the sign of ω_k alone does not represent an additional oscillation timescale, and K state blocks do not necessarily correspond to K distinct oscillation frequencies. Learned residual corrections may separate branches originating from a conjugate pair, but their distinction should be assessed from the effective parameters $(\tilde{\alpha}_k, \tilde{\omega}_k)$ rather than from frequency signs alone. Finally, $\tilde{\alpha}_k < 0$ guarantees contraction of the unforced state transition associated with each block, but does not imply stability of the entire forecasting network.

D Full Long-Term Forecasting Results

To compare performance across different datasets and prediction horizons, PhyTM is evaluated against nine representative baselines using lookback window $L = 96$ and four prediction horizons $H \in \{96, 192, 336, 720\}$ on eight datasets spanning energy, weather, financial, and traffic domains. Comprehensive results are summarized in Table 7.

Table 7: Full results with lookback length $L = 96$ and $H \in \{96, 192, 336, 720\}$. **Bold**: best, Underline: second best. The **Avg** row reports the mean of MSE/MAE over the four horizons.

Dataset	H	PhyTM		AURORA		AutoHFormer		TimeDRL		CycleNet		FilterNet		iTransformer		PatchTST		FITS		DLinear	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	**0.368**	**0.390**	0.376	0.397	0.374	0.398	0.371	0.394	0.378	0.391	0.375	0.394	0.386	0.405	0.394	0.406	0.386	0.396	0.386	0.400
	192	**0.421**	0.419	0.427	**0.416**	0.431	0.428	0.429	0.427	0.426	0.419	0.436	0.422	0.441	0.436	0.440	0.435	0.436	0.423	0.437	0.432
	336	**0.460**	0.444	0.463	0.451	0.472	0.448	0.478	0.452	0.464	**0.439**	0.476	0.443	0.487	0.458	0.491	0.462	0.478	0.444	0.481	0.459
	720	0.483	0.474	0.486	0.475	0.489	0.511	0.487	0.478	**0.461**	**0.460**	0.474	0.469	0.503	0.491	0.487	0.479	0.502	0.495	0.519	0.516
	Avg	0.433	0.432	0.438	0.435	0.442	0.446	0.441	0.438	**0.432**	**0.427**	0.440	0.432	0.454	0.448	0.453	0.446	0.451	0.440	0.456	0.452
ETTh2	96	0.293	0.331	0.287	0.338	**0.281**	**0.291**	0.291	0.344	0.285	0.335	0.292	0.343	0.297	0.349	0.288	0.340	0.295	0.350	0.333	0.387
	192	**0.361**	**0.386**	0.376	0.393	0.375	0.398	0.369	0.394	0.373	0.391	0.369	0.395	0.380	0.400	0.376	0.395	0.381	0.396	0.477	0.476
	336	**0.407**	**0.421**	0.414	0.426	0.421	0.489	0.419	0.429	0.421	0.433	0.420	0.432	0.428	0.432	0.440	0.451	0.426	0.438	0.594	0.451
	720	**0.412**	**0.438**	0.436	0.448	0.421	0.443	0.434	0.453	0.453	0.458	0.430	0.446	0.427	0.445	0.436	0.453	0.431	0.446	0.631	0.467
	Avg	**0.368**	**0.394**	0.378	0.401	0.375	0.405	0.378	0.405	0.383	0.404	0.378	0.404	0.383	0.407	0.385	0.410	0.383	0.408	0.509	0.445
ETTm1	96	0.317	**0.352**	0.327	0.364	0.328	0.371	**0.314**	0.355	0.319	0.360	0.318	0.358	0.334	0.368	0.329	0.365	0.355	0.375	0.345	0.372
	192	0.362	0.379	0.364	0.388	0.368	0.387	**0.357**	**0.378**	0.367	0.383	0.364	0.383	0.377	0.391	0.380	0.394	0.392	0.393	0.380	0.389
	336	0.394	**0.403**	0.402	0.417	**0.391**	0.412	0.396	0.427	0.398	**0.403**	0.396	0.406	0.426	0.420	0.400	0.410	0.424	0.414	0.413	0.413
	720	**0.443**	**0.438**	0.451	0.439	0.452	0.447	0.461	0.447	0.447	0.441	0.456	0.444	0.491	0.459	0.475	0.453	0.487	0.449	0.474	0.453
	Avg	**0.379**	**0.393**	0.386	0.402	0.385	0.404	0.382	0.402	0.383	0.397	0.384	0.398	0.407	0.410	0.396	0.406	0.415	0.408	0.403	0.407
ETTm2	96	**0.171**	**0.249**	0.189	0.267	0.184	0.251	0.178	0.259	0.174	0.252	0.174	0.257	0.180	0.264	0.184	0.264	0.183	0.266	0.193	0.292
	192	**0.223**	**0.282**	0.231	0.295	0.227	0.292	0.238	0.298	0.229	0.290	0.240	0.300	0.250	0.309	0.246	0.306	0.247	0.305	0.284	0.362
	336	**0.282**	0.322	0.309	0.327	0.297	**0.321**	0.302	0.341	0.284	0.327	0.297	0.339	0.311	0.348	0.308	0.346	0.307	0.342	0.369	0.427
	720	**0.380**	**0.378**	0.406	0.405	0.381	0.386	0.398	0.397	0.389	0.391	0.392	0.393	0.412	0.407	0.409	0.402	0.407	0.399	0.554	0.522
	Avg	**0.264**	**0.308**	0.284	0.324	0.272	0.313	0.279	0.324	0.269	0.315	0.276	0.322	0.288	0.332	0.287	0.330	0.286	0.328	0.350	0.401
Weather	96	0.163	**0.178**	0.168	0.206	0.167	0.185	0.163	0.201	0.168	0.203	**0.162**	0.207	0.174	0.214	0.176	0.217	0.166	0.213	0.196	0.255
	192	**0.201**	0.221	0.215	0.253	0.211	**0.217**	0.207	0.246	0.207	0.247	0.210	0.250	0.221	0.254	0.221	0.256	0.213	0.254	0.237	0.296
	336	0.272	**0.263**	0.263	0.289	0.267	0.289	**0.262**	0.289	**0.262**	0.289	0.265	0.290	0.278	0.296	0.275	0.296	0.269	0.294	0.283	0.335
	720	**0.338**	**0.307**	0.349	0.350	0.341	0.312	0.345	0.342	0.344	0.344	0.342	0.340	0.358	0.347	0.352	0.346	0.346	0.343	0.345	0.381
	Avg	**0.244**	**0.242**	0.249	0.275	0.247	0.251	**0.244**	0.270	0.245	0.271	0.245	0.272	0.258	0.278	0.256	0.279	0.249	0.276	0.265	0.317
Electricity	96	0.207	0.287	0.155	0.249	0.166	0.255	0.179	0.280	0.171	0.269	0.158	0.252	**0.148**	**0.240**	0.164	0.255	0.166	0.270	0.197	0.282
	192	0.211	0.289	0.183	0.272	0.173	0.261	0.195	0.281	0.183	0.279	0.174	0.262	**0.162**	**0.253**	0.178	0.264	0.178	0.274	0.196	0.285
	336	0.223	0.291	0.204	0.283	0.182	0.274	0.210	0.280	0.194	0.289	0.188	0.275	**0.178**	**0.269**	0.195	0.279	0.191	0.284	0.209	0.301
	720	**0.219**	**0.298**	0.234	0.318	0.221	0.302	0.233	0.327	0.238	0.320	0.226	0.310	0.225	0.317	0.236	0.320	0.238	0.330	0.245	0.333
	Avg	0.215	0.291	0.194	0.281	0.186	0.273	0.204	0.292	0.197	0.289	0.187	0.275	**0.178**	**0.270**	0.193	0.280	0.193	0.290	0.212	0.300
Exchange	96	**0.081**	**0.205**	0.102	0.212	0.093	0.223	0.094	0.246	0.097	0.232	0.097	0.277	0.093	0.215	0.113	0.269	0.116	0.274	0.103	0.216
	192	**0.146**	**0.283**	0.184	0.303	0.154	0.287	0.173	0.297	0.166	0.286	0.169	0.297	0.167	0.308	0.166	0.312	0.168	0.297	0.185	0.316
	336	**0.263**	**0.384**	0.285	0.400	0.271	0.391	0.276	0.406	0.273	0.407	0.272	0.407	0.292	0.421	0.298	0.399	0.285	0.414	0.298	0.411
	720	**0.585**	**0.616**	0.772	0.697	0.764	0.694	0.853	0.697	0.784	0.702	0.765	0.705	0.786	0.718	0.768	0.712	0.769	0.695	0.795	0.719
	Avg	**0.269**	**0.372**	0.336	0.403	0.321	0.399	0.349	0.412	0.330	0.407	0.326	0.422	0.335	0.416	0.336	0.423	0.335	0.420	0.345	0.416
Traffic	96	0.559	0.376	0.431	0.310	0.461	0.296	0.437	0.271	0.468	0.306	0.430	0.294	**0.395**	**0.268**	0.427	0.272	0.651	0.391	0.650	0.396
	192	0.591	0.383	0.462	0.291	0.463	0.297	0.447	**0.272**	0.467	0.294	0.452	0.307	**0.417**	0.276	0.454	0.289	0.602	0.363	0.598	0.370
	336	0.614	0.397	0.515	0.303	0.478	0.304	0.457	**0.281**	0.480	0.299	0.470	0.316	**0.433**	0.283	0.450	0.282	0.609	0.366	0.605	0.373
	720	0.641	0.401	0.670	0.347	0.521	0.331	0.507	**0.281**	0.522	0.314	0.498	0.323	**0.467**	0.302	0.584	0.301	0.647	0.385	0.645	0.394
	Avg	0.601	0.389	0.520	0.313	0.481	0.307	0.462	**0.276**	0.484	0.303	0.463	0.310	**0.428**	0.282	0.479	0.286	0.627	0.376	0.625	0.383
1st Count		**18**	**18**	0	1	2	3	3	4	2	3	1	0	7	4	0	0	0	0	0	0

Table 7 shows that PhyTM achieves competitive or superior forecasting performance across most datasets and prediction horizons. Notably, PhyTM achieves a 23.4% MSE reduction on Exchange at $H = 720$ compared with the best baseline. Consistent improvements are observed on ETTh2 and Weather across multiple horizons, while particularly strong gains on Exchange and ETTm2 suggest that the introduced physical priors are especially effective for datasets with pronounced

dynamical structures. The more moderate performance on Traffic and Electricity is mainly associated with their higher dynamical complexity, as shown in Section 4.4 (*Dynamical-Complexity Analysis*). These datasets require a larger number of modes for accurate characterization, and increasing K progressively narrows the performance gap. Nevertheless, all results in Table 7 use the same setting of K=8 physical modes to ensure a fair and unified comparison across datasets.

E Full Short-Term Forecasting Results

To assess whether the physics-informed approach generalizes beyond long-horizon settings, PhyTM is further evaluated on the M4 competition benchmark, covering Yearly, Quarterly, Monthly, and Others granularities with granularity-specific forecasting settings. Comprehensive results are summarized in Table 2.

Table 8: Short-term forecasting results on the M4 dataset across different temporal granularities, together with the overall results. **Red**: best, Blue: second best.

Dataset	Metric	PhyTM	AURORA	AutoHFormer	TimeDRL	CycleNet	FilterNet	iTransformer	PatchTST	FITS	DLinear
Yearly	SMAPE	13.331	13.225	13.210	13.419	13.479	**13.128**	13.924	13.178	13.943	16.967
	MASE	**2.872**	3.736	3.274	3.631	3.958	3.274	3.235	3.221	3.433	4.285
	OWA	**0.769**	0.875	0.816	0.867	0.910	0.814	0.833	0.808	0.859	1.058
Quarterly	SMAPE	**10.181**	10.885	11.138	11.483	11.676	10.693	10.757	10.381	11.058	12.151
	MASE	**1.211**	1.281	1.335	1.397	1.447	1.303	1.284	1.234	1.354	1.523
	OWA	**0.904**	0.961	0.993	1.031	1.058	0.961	0.957	0.921	0.996	1.107
Monthly	SMAPE	13.577	13.517	13.269	13.323	14.091	13.397	13.797	**12.961**	13.257	13.515
	MASE	**0.936**	1.006	0.996	1.032	1.091	0.962	1.084	0.971	0.969	1.039
	OWA	0.911	0.942	0.928	0.947	1.002	0.917	0.988	**0.906**	0.915	0.957
Others	SMAPE	**4.062**	5.818	5.442	4.985	4.424	5.155	5.569	4.953	4.593	6.710
	MASE	**1.237**	1.694	1.647	1.875	1.575	1.347	3.941	3.349	2.990	4.953
	OWA	**0.631**	0.893	0.844	0.829	0.723	0.767	1.208	1.051	0.957	1.487
Weighted Average	SMAPE	12.230	12.433	12.353	12.487	12.887	12.274	12.685	**11.991**	12.454	13.641
	MASE	**1.462**	1.734	1.634	1.760	1.860	1.595	1.770	1.671	1.729	2.097
	OWA	**0.863**	0.929	0.914	0.943	0.980	0.896	0.956	0.894	0.924	1.043
1st Count		**11**	0	0	0	0	1	0	3	0	0

Table 8 shows that PhyTM achieves competitive or superior forecasting performance across different temporal granularities and evaluation metrics. In particular, on the Others subset, PhyTM reduces OWA by 12.7% and MASE by 8.2% compared with the best-performing baselines. In terms of the overall results, PhyTM achieves the best MASE and OWA, reducing MASE by 8.3% over the best baseline and demonstrating strong generalization across short-term and mixed-frequency forecasting settings.

F Detailed Ablation Studies

To systematically investigate the contributions of the key designs in PhyTM, detailed ablation studies are conducted from two complementary perspectives: physics-informed components and tri-modal structure. The former examines the contributions of DMD-based physical information extraction, Phys-Transformer, and Phys-Mamba to incorporating physical dynamics into temporal modeling, while the latter evaluates the roles of the textual, visual, and numerical paths in exploiting complementary multimodal representations. Together, these studies provide a component- and structure-level analysis of the effectiveness of PhyTM.

F.1 Ablation on Physics Components

PhyTM introduces physical priors through three components: DMD-based physical information extraction, Phys-Transformer, and Phys-Mamba. To evaluate their contributions, four variants are considered in Table 9: ① removes all physics components; ② replaces Phys-Mamba with standard Mamba while retaining DMD and Phys-Transformer; ③ replaces Phys-Transformer with a standard Transformer while retaining DMD and Phys-Mamba; and ④ is the complete PhyTM framework.

Table 9: Ablation study of physics components in PhyTM. Results are reported at prediction horizon $H = 96$. **Red**: best.

Case	Physics Components			ETTh1		ETTh2		ETTm1		ETTm2		Weather		Exchange	
	DMD	Phys-Trans.	Phys-Mamba	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
①	✗	✗	✗	0.452	0.450	0.309	0.357	0.368	0.389	0.212	0.293	0.179	0.202	0.091	0.211
②	✓	✓	✗	0.449	0.447	0.294	0.339	0.349	0.376	0.186	0.253	0.169	0.181	0.089	0.209
③	✓	✗	✓	0.376	0.391	0.301	0.349	**0.315**	0.357	0.176	0.252	0.178	0.181	0.083	0.206
④	✓	✓	✓	**0.368**	**0.390**	**0.293**	**0.331**	0.317	**0.352**	**0.171**	**0.249**	**0.163**	**0.178**	**0.081**	**0.205**

Table 9 evaluates the contribution of the physics-informed components. Compared with ④, ① causes substantial degradation across all datasets, with MSE increasing by up to 24.0% on ETTm2, highlighting the importance of physics-informed modeling. Comparing ② and ③ further shows that Phys-Mamba and Phys-Transformer produce distinct performance changes across datasets, demonstrating their complementary roles in modeling physical dynamics. Such dataset-dependent effects suggest that these components capture different temporal dynamics rather than providing a uniform regularization effect. Overall, ④ achieves the best or competitive performance across all datasets, validating the synergy of the physics-informed components.

F.2 Ablation on Tri-Modal Structure

To evaluate the contribution of each modality path, all physics components are retained while one modality is removed at a time. Specifically, the textual path captures semantic temporal dependencies, the visual path extracts structural patterns from GAF representations, and the numerical path models raw temporal dynamics. As shown in Table 10, ①, ②, and ③ remove the textual, visual, and numerical paths, respectively, while ④ denotes the complete PhyTM framework.

Table 10: Ablation study of modality paths in PhyTM. Results are reported at prediction horizon $H = 96$. **Red**: best.

Case	Modality Paths			ETTh1		ETTh2		ETTm1		ETTm2		Weather		Exchange	
	Text	Image	Num	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
①	✗	✓	✓	0.420	0.429	0.313	0.356	0.331	0.364	0.197	0.267	0.203	0.223	0.081	0.209
②	✓	✗	✓	0.418	0.423	0.326	0.371	0.416	0.415	0.189	0.268	0.181	0.193	0.084	0.213
③	✓	✓	✗	0.415	0.427	0.348	0.383	0.325	0.353	0.202	0.286	0.185	0.198	**0.080**	0.208
④	✓	✓	✓	**0.368**	**0.390**	**0.293**	**0.331**	**0.317**	**0.352**	**0.171**	**0.249**	**0.163**	**0.178**	0.081	**0.205**

Table 10 evaluates the contribution of each modality. Compared with ④, the incomplete variants generally exhibit degraded performance across datasets, confirming the complementary contributions of the three modalities. Among them, ② causes particularly pronounced degradation on ETTm1, highlighting the importance of visual structural patterns. In contrast, ① notably affects Weather, ETTm2, and ETTh1, while ③ generally degrades performance on most datasets. Overall, ④ achieves the strongest performance across nearly all evaluation settings, validating the effectiveness of tri-modal modeling.

G Compatibility Analysis

To evaluate the compatibility of the proposed physics-informed components with different forecasting architectures, we construct physics-enhanced variants of DLinear, iTransformer, and PatchTST, denoted as +Phys. For DLinear, we augment the original forecasting architecture with an additional module that incorporates DMD-derived dynamical information. For iTransformer and PatchTST, we incorporate the dynamical priors into their attention mechanisms, with adaptations to their respective token representations.

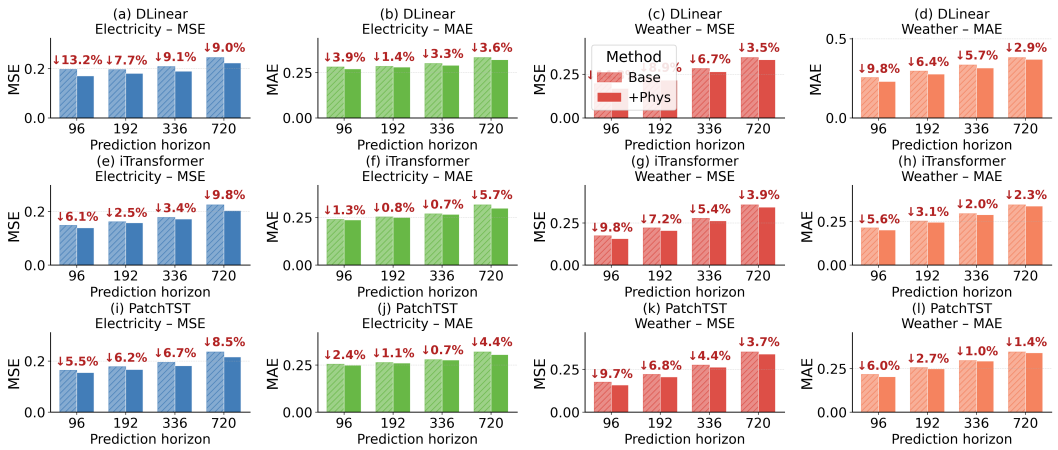


Figure 7: Performance of integrating PhyTM's physics-informed attention into different backbones on Electricity and Weather datasets across prediction horizons {96, 192, 336, 720}.

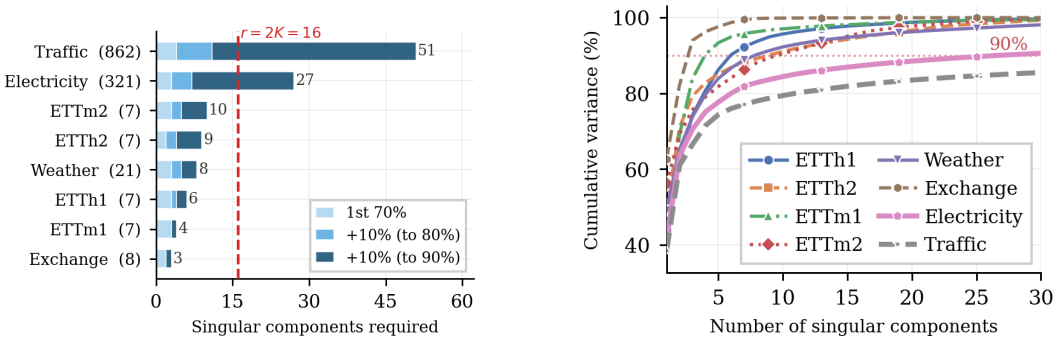
As shown in Fig. 7, incorporating the proposed physics-informed module consistently improves forecasting performance across all backbones. The largest gain is achieved by DLinear, with an MSE reduction of up to 14.3% on Weather, while Transformer-based models also benefit from the additional physical guidance. Notably, the substantial improvement on the lightweight linear backbone indicates that the benefit of dynamical priors does not depend on backbone expressiveness. These results further demonstrate that the proposed module is architecture-agnostic and can complement both simple and advanced forecasting models.

H Dynamical Property Analysis

To further investigate the physical dynamics captured by DMD, we analyze their properties from two complementary perspectives: intrinsic and extrinsic properties. The intrinsic analysis characterizes the complexity and spectral concentration of the extracted dynamics, while the extrinsic analysis examines their reconstruction fidelity and its relationship with forecasting performance.

H.1 Intrinsic Property Analysis

To characterize the intrinsic properties of the extracted dynamics, we analyze the effective rank and cumulative explained variance across datasets. The effective rank measures the number of dominant components required to explain 90% of the total variance, while the cumulative explained variance reflects how rapidly information is concentrated in these components. Together, the two measures characterize the complexity and compactness of the underlying dynamical structure.



(a) Effective rank at 90% variance.

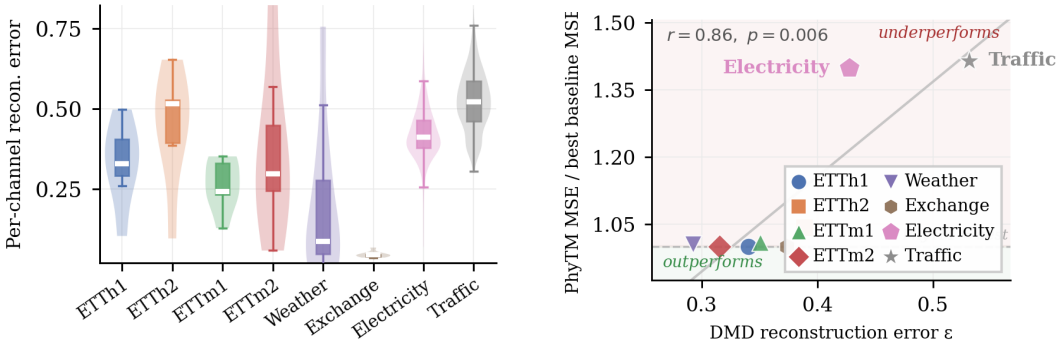
(b) Cumulative explained variance.

Figure 8: Spectral complexity measured by effective rank and explained variance.

As shown in Fig. 8, the ETT datasets require relatively few dominant components to explain most of the variance and exhibit rapid variance saturation, indicating compact dynamical structures. In contrast, Electricity and Traffic have higher effective ranks and slower variance accumulation, suggesting richer and more heterogeneous dynamics distributed across a larger number of components. These results reveal clear differences in intrinsic dynamical complexity across datasets, with more complex datasets requiring higher-dimensional representations to preserve their dominant temporal structures.

H.2 Extrinsic Property Analysis

To characterize the extrinsic properties of the extracted dynamics, we evaluate DMD reconstruction error and its relationship with relative forecasting performance. Reconstruction error measures how faithfully the extracted DMD components recover the original temporal dynamics, while relative performance measures the forecasting error of PhyTM relative to the strongest baseline. Their relationship further indicates whether the quality of the extracted dynamical representation is associated with the forecasting behavior of PhyTM.



(a) DMD reconstruction error.

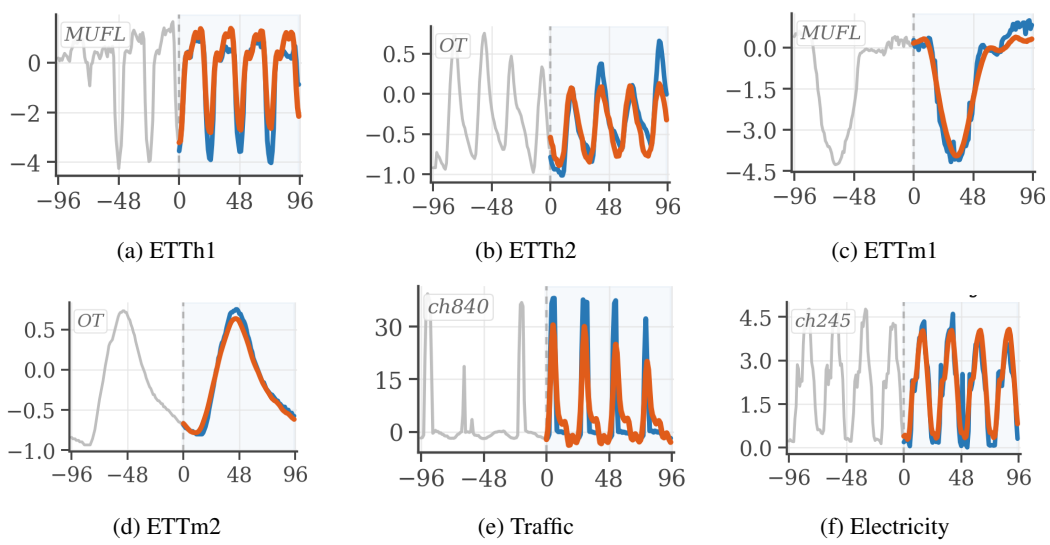
(b) Reconstruction error versus relative performance.

Figure 9: DMD reconstruction quality and its association with forecasting performance.

As shown in Fig. 9, Electricity and Traffic exhibit relatively larger reconstruction errors, indicating that their richer temporal dynamics are more difficult to represent using a limited set of DMD components. Across datasets, reconstruction error is positively correlated with relative forecasting error ($r = 0.86$, $p = 0.006$), showing that datasets with lower reconstruction errors tend to exhibit stronger relative forecasting performance. This association suggests that the forecasting benefit of PhyTM is closely related to the fidelity of the DMD-derived dynamical representation, providing complementary evidence for the relevance of the extracted physical dynamics to downstream forecasting.

I Prediction Visualization

To qualitatively evaluate the forecasting behavior of PhyTM, we conduct visual comparisons between the predicted series and the ground truth under diverse temporal patterns. The analysis examines the model's ability to capture periodic structure, long-term trends, and irregular fluctuations, jointly reflecting both forecasting accuracy and robustness under different temporal dynamics.



(a) ETTh1 (b) ETTh2 (c) ETTm1

(d) ETTm2 (e) Traffic (f) Electricity

Figure 10: Qualitative forecasting visualization of PhyTM across multiple datasets with forecasting horizon $H = 96$.

As shown in Fig. 10, PhyTM produces accurate forecasts across datasets with diverse temporal characteristics. On strongly periodic datasets such as ETTh1, ETTm1, and Electricity, the predictions closely match both the phase and amplitude of the ground truth. On ETTh2 and ETTm2, PhyTM accurately captures the overall trends and turning points despite smoother temporal variations. Even on Traffic, which contains sharp and irregular spikes, the predicted series still preserves the overall waveform and peak locations, demonstrating that the learned DMD modes generalize beyond strictly periodic dynamics.

J Further Related Work

Time series forecasting connects sequence modeling, representation learning, and scientific machine learning. We review forecasting architectures, physics-informed modeling, and multimodal temporal representations, emphasizing how their inductive biases relate to PhyTM.

J.1 Time Series Forecasting Architectures

Early studies on time series forecasting primarily rely on recurrent architectures to capture temporal dependencies through sequential state updates (Cho et al., 2014). Graph-based extensions incorporate relationships among correlated variables into this temporal modeling process (Li et al., 2018; Bai et al., 2020). These architectures provide a natural representation of sequential observations, but their recurrent computation limits parallelization and makes information propagation across long sequences challenging. Temporal convolutional networks offer an alternative by expanding receptive fields through convolutions while supporting parallel computation (Bai et al., 2018; Luo & Wang, 2024; Dudukcu et al., 2023). Extending this local modeling perspective requires representations that connect short-term variations with broader temporal structure. Local and global context modeling, together with structured temporal representations, supports this connection at different scales (Wang et al., 2023; Oreshkin et al., 2020; Challu et al., 2023). This perspective also motivates representing recurring patterns across temporal resolutions (Wu et al., 2023; Wang et al., 2024; 2025b) and learning representations that preserve seasonal-trend structure and dependencies between temporal segments (Ma et al., 2024; Zhang et al., 2024; Woo et al., 2022; Dong et al., 2024). These developments enrich the temporal information available for forecasting, while the need to relate distant observations motivates more flexible mechanisms for modeling long-range dependencies.

Transformer-based architectures provide such mechanisms through self-attention over temporal and cross-variable relationships (Vaswani et al., 2017; Zhao et al., 2026; Zhou et al., 2021; Zhang & Yan, 2023). Patching, decomposition, and multiscale processing improve the organization of long

sequences (Nie et al., 2023; Wu et al., 2021; Chen et al., 2024a), while spectral representations and non-stationarity-aware modeling account for periodic structure and changing statistics (Zhou et al., 2022; Wang et al., 2025a; Kim et al., 2022; Han et al., 2024; Liu et al., 2025a). Efficiency-oriented architectures reduce the cost of forecasting over these sequences (Lai et al., 2023; Lin et al., 2026). Large-scale temporal pre-training supports transferable forecasting representations across diverse series (Woo et al., 2024; Goswami et al., 2024; Das et al., 2024; Liu et al., 2024c). Long-context modeling and sparse expert architectures extend the context and capacity available to these forecasting models (Liu et al., 2025b; Shi et al., 2025). Recent state-space formulations provide another route to long-sequence modeling through compact latent state updates (Rangapuram et al., 2018; Gu et al., 2022b; Gu & Dao, 2024), supported by structured parameterizations, parallel computation, and connections to attention (Smith et al., 2023; Gu et al., 2022a; Orvieto et al., 2023; Dao & Gu, 2024). Complementary generative formulations model distributions over temporal trajectories for probabilistic forecasting and conditional time series generation (Rasul et al., 2021; Yuan & Qiao, 2024). These advances improve the modeling of temporal dependencies, but temporal order and learned state transitions do not explicitly specify the physical evolution underlying observations. In particular, predictive training alone does not require attention or state updates to reflect the same growth, decay, and oscillation patterns. PhyTM incorporates DMD-derived dynamical priors into both components to provide this additional structure.

J.2 Physics-Guided Time Series Modeling Approaches

Integrating physical priors into temporal modeling provides a way to relate learned predictions to the dynamics of an observed system (Karniadakis et al., 2021). Physics-informed learning incorporates differential constraints and temporal dependencies into the approximation of physical trajectories (Raissi et al., 2019; Zhao et al., 2024). Related formulations learn solution operators or identify equations from observations (Li et al., 2021; Luo et al., 2023). The use of explicit equation-based constraints depends on access to suitable physical knowledge, which is often incomplete in generic time series forecasting. This motivates extracting dynamical information from observations themselves. Continuous-time models represent temporal evolution through learned latent dynamics (Chen et al., 2018; Morrill et al., 2021), while learning suitable observables allows nonlinear evolution to be approximated in a linear representation space (Takeishi et al., 2017). DMD provides a related spectral description by estimating linear dynamics from observed sequences and identifying modes associated with growth, decay, and oscillation. Such descriptions offer physical structure without requiring a fully specified governing equation.

Data-derived dynamical representations have been incorporated into forecasting to address non-stationarity and improve interpretability (Liu et al., 2023; Chen et al., 2024b). Memory-based modeling and domain-specific physical guidance further connect temporal predictions with the evolution of physical systems (Buitrago Ruiz et al., 2025; Luo et al., 2025). The usefulness of these representations depends on the assumptions and observation windows used to estimate the dynamics. An approximately linear representation need not remain equally valid across different conditions or prediction horizons. Beyond extracting a suitable representation, its integration into the forecasting architecture is also important: using dynamical features as auxiliary inputs does not directly constrain how attention weights or latent state transitions process temporal information. PhyTM addresses this integration problem by estimating local spectral features through DMD and using them as shared priors within a tri-modal Transformer-Mamba framework. These priors guide textual attention and visual state evolution, linking the forecasting operations to the same estimated dynamical structure.

J.3 Multimodal Temporal Representation Learning

Real-world temporal forecasting can involve numerical measurements, textual context, and visual observations that provide complementary information about the same evolving process (Chacon Maldonado et al., 2025; Liu et al., 2024a). Textual representations describe temporal characteristics such as trend direction, fluctuations, and changes in operating conditions, connecting numerical variations with interpretable semantic information (Xue & Salim, 2023; Jin et al., 2024; Gruver et al., 2023). The advantage of the textual path lies in organizing these characteristics into descriptions that can be considered together with contextual information, allowing forecasting to draw on both observed values and their semantic interpretation. Visual representations provide a different advantage by arranging temporal observations into image-like structures, where recurring shapes, local variations,

and relationships between distant observations can be represented as spatial patterns (Chen et al., 2025). This organization makes structural relationships within a sequence accessible to visual feature extraction and state-space modeling (Zhu et al., 2024; Liu et al., 2024d). The textual path therefore emphasizes the semantic interpretation of temporal changes, while the visual path emphasizes the structural organization of temporal patterns. Alignment and fusion mechanisms combine these complementary perspectives with numerical observations to support forecasting (Zhou et al., 2025; Li et al., 2024; Wang et al., 2026).

The value of these representations ultimately depends on whether their semantic and structural information contributes to accurate temporal prediction (Tan et al., 2024). When textual and visual views describe the same numerical series, their complementarity also raises a consistency requirement: both views should remain connected to the dynamics of the underlying process. Statistical feature alignment can associate representations across modalities, but does not explicitly require their temporal processing to follow a common evolution. Attention and state transitions may therefore capture predictive relationships without being guided by the same dynamical structure. PhyTM addresses this issue by introducing DMD-derived priors as a shared physical basis for its tri-modal forecasting framework. The extracted spectral information describes local growth, decay, and oscillation patterns and guides both textual and visual processing. Specifically, the textual path incorporates these priors into dual-stream attention, while the visual path uses them to structure state-space evolution. The numerical path preserves the original temporal observations. This design connects semantic and structural representations through a common dynamical basis, incorporating physical information directly into the mechanisms used for cross-modal forecasting.