

Time Series Foundation Models: From Forecasting to Reasoning

Yan Liu

Fletcher Jones Foundation Endowed Chair Professor
Thomas Lord Computer Science Department
Director, Machine Learning Center
University of Southern California

What This Tutorial Covers

Four parts: from foundations to a full temporal decision stack

PART I

Foundations of Temporal AI

Time series as partial observations of dynamical systems — and how to evaluate models beyond average error.

PART III

Building Temporal Foundation Models

Four hands-on modules: representation, generation, physical grounding, and orchestration.

PART II

Sixty Years of Temporal Modeling

Classical, deep, and pretrained approaches — and the reliability gap they leave open.

PART IV

Closing: Toward Decision Intelligence

What we solved, what remains open, and why it matters for real-world decision systems.

Foundations of Temporal AI

Everything you need before we build: data, metrics, and the core paradox

01

Temporal Data & Dynamics

Time series as incomplete observations of evolving systems.

02

Evaluation & Reliability

Beyond average error: calibration, sharpness, and decision utility.

03

The Foundation-Model Promise

Why temporal FMs need different interfaces than LLMs.

AI is Everywhere: From Daily Life to Science



OpenAI
ChatGPT

Gemini



Stable Diffusion



AlphaFold

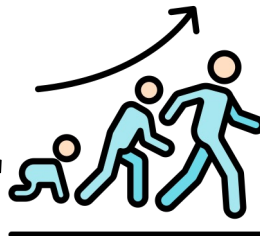


AI still Lacks a Foundation Model for Dynamic Systems



The next frontier of AI

"What is it?" → "How does it change?"



Intelligence without temporal understanding is incomplete.

~ **90%** of all data has a temporal or time-stamped.

What Is a Time Series?

DEFINITION

$$\mathbf{X}_{1:T} = (\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_T)$$

$\mathbf{X}_t \in \mathbb{R}^d$ — observations indexed by time

THREE COMMON SHAPES

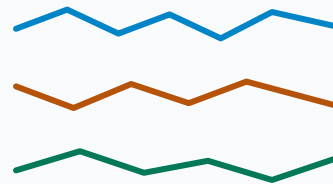
- **Univariate / Multivariate**
- **Panel / Collection of series**
- **Spatio-temporal & Event streams**

Key takeaway — a time series is a partial observation of a changing system.

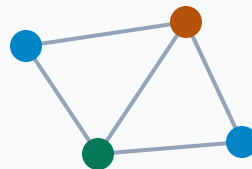
1 · Univariate curve



2 · Multi-channel



3 · Spatio-temporal graph



sensors linked on a network

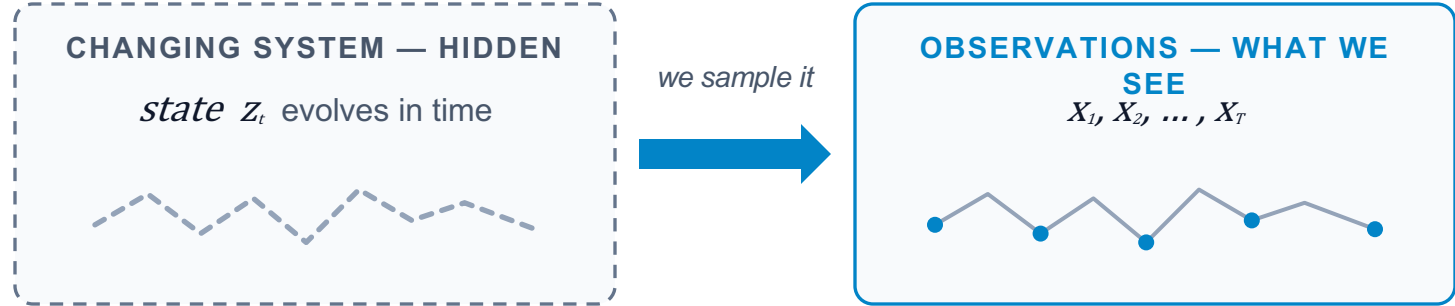
4 · Irregular events



arrivals at arbitrary times

FROM INTUITION TO NOTATION

A time series is a **partial observation** of a changing system.



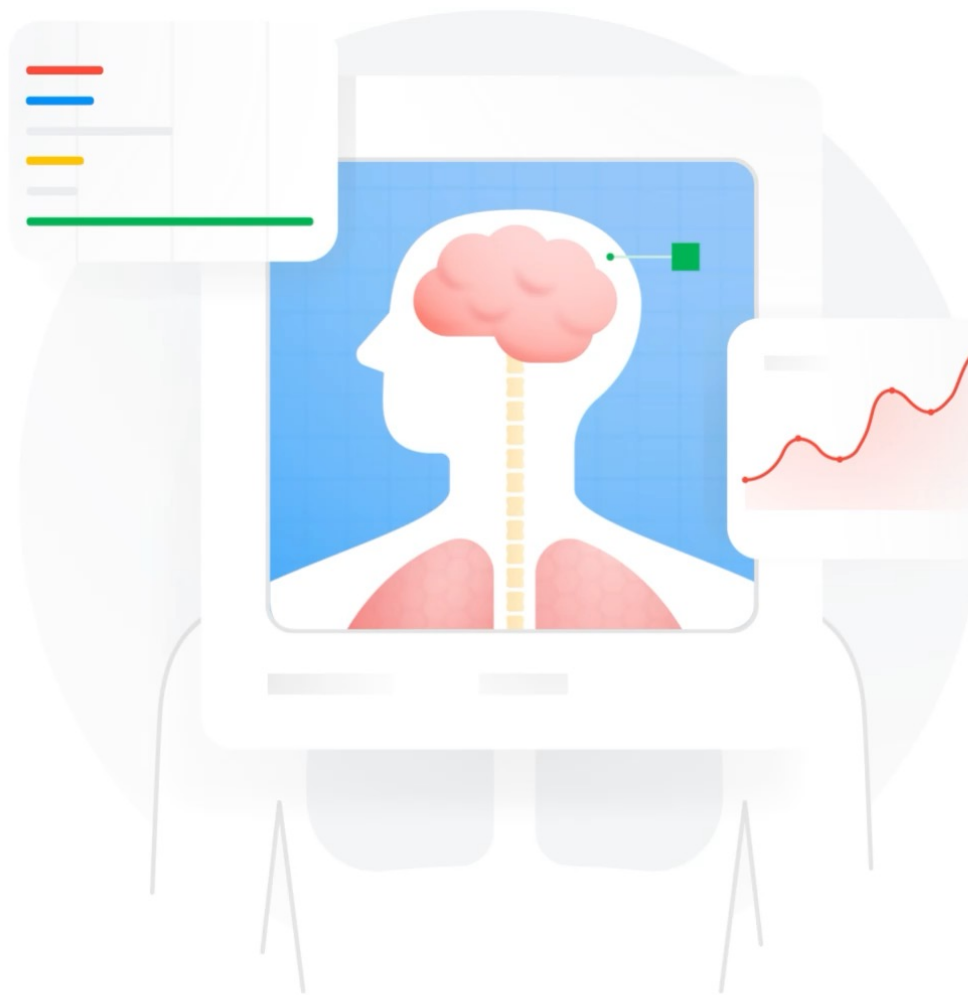
Next: just enough notation to state the tasks precisely — **forecast** · **impute** · **detect**

Why Temporal AI Matters



Human Health & Bio-Signals

Early detection of physiological shifts saves lives.



Why Temporal AI Matters



Human Health & Bio-Signals



Climate & Energy

Accurate forecasting protects billions from grid failure and extreme events.

AI & Machine Learning

How AI-powered weather forecasting can transform energy operations

August 18, 2025



Nelly Wilson

Head of AI Services for Weather and
Climate, Google Cloud

Matt Rettagliata

Head of Energy Solutions, Google
Cloud

AI-powered weather forecasting and automated agents are transforming the energy and utilities industries by providing more accurate predictions and enabling proactive operational responses.

Why Temporal AI Matters



Human Health & Bio-Signals

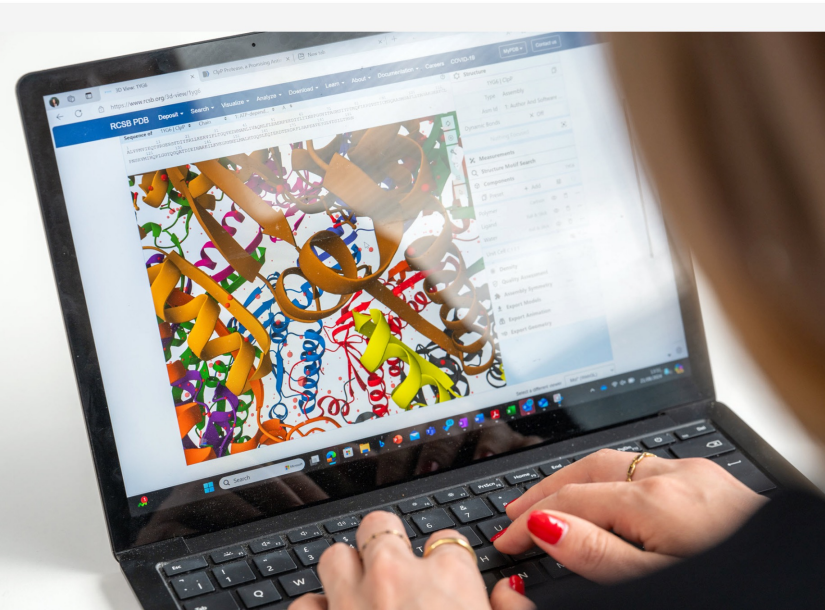


Climate & Energy



Fundamental Science Discovery

Temporal models accelerate rare event detection from years to months.



AI

October 8, 2024

From forecasting storms to designing molecules: How new AI foundation models can speed up scientific discovery

By [Catherine Bolgar](#)

Temporal AI underpins Decisions in Health, Climate, and Science



Human Health & Bio-Signals

Trillions

Economic value tied to forecasting.



Climate & Energy

Millions

Heterogeneous sensors generating temporal data.



Fundamental Science Discovery

Seconds

Decisions that must be made in real time.

Temporal intelligence is the backbone of dynamic decision-making.

Real Temporal Data Are Messy by Default

TYPE 01

Regular grid



fixed sampling interval

benchmark default

TYPE 02

Missing values



scheduled samples never arrive

needs imputation

TYPE 03

Irregular gaps



uneven spacing between samples

continuous-time models

TYPE 04

Asynchronous

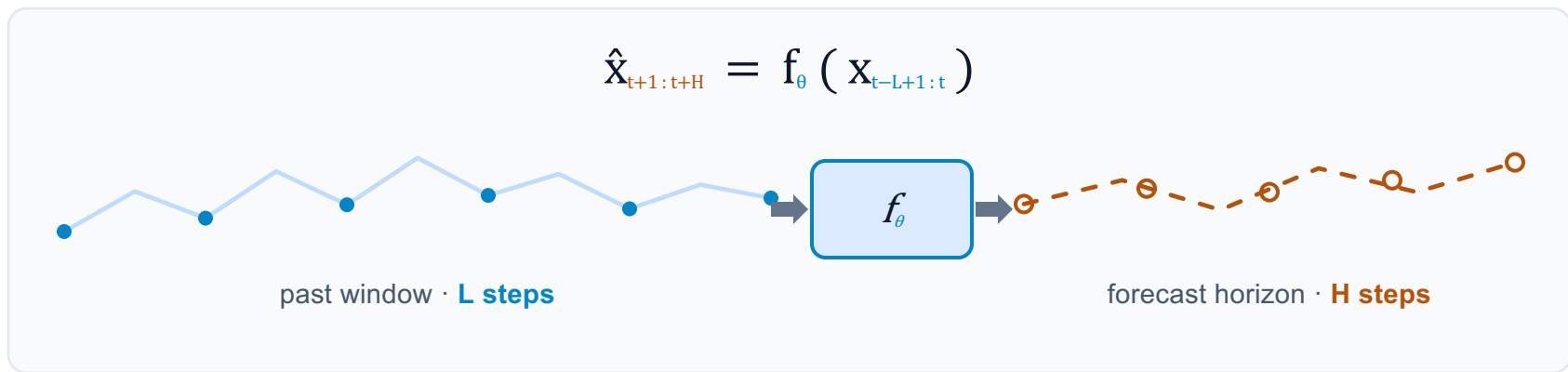


channels sampled at different times

needs alignment

Core premise — missingness, irregularity and misalignment are part of the inference problem, not a preprocessing chore.

Forecasting Setup



STRATEGY 01

One-step

Predict only the next value — the $H = 1$ special case.

STRATEGY 02

Recursive

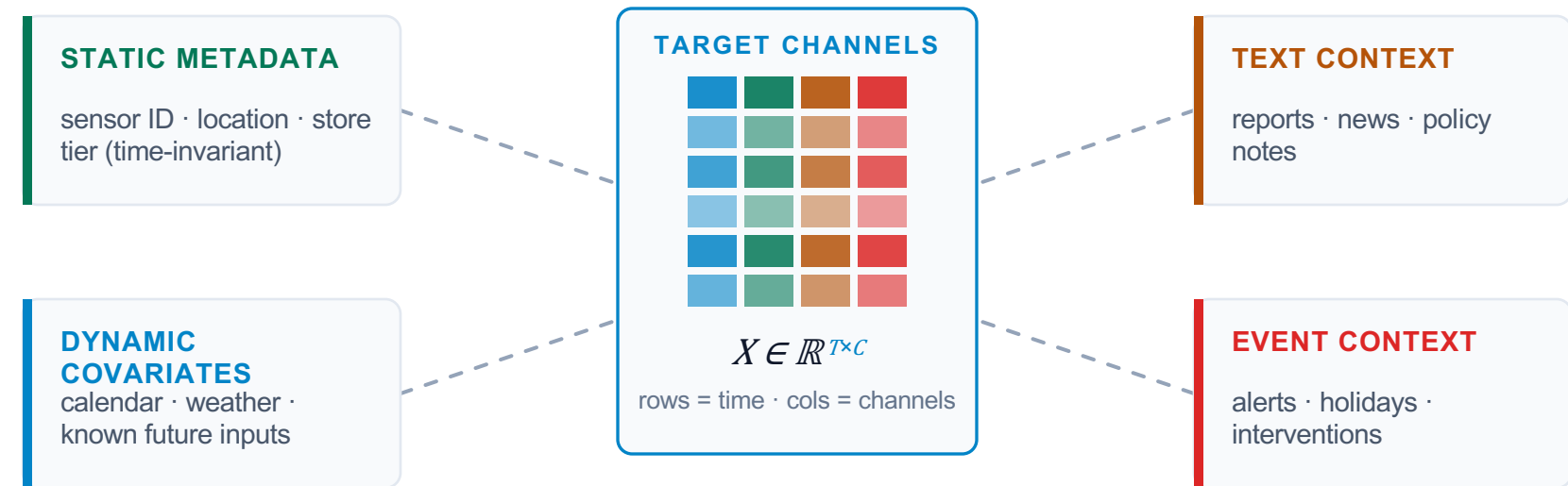
Feed each prediction back as input — errors can accumulate.

STRATEGY 03

Direct / seq-to-seq

Emit the entire H -step horizon in one forward pass.

Multivariate Series + Exogenous Context



The same numeric spike can mean failure, anomaly, or a planned change — **context disambiguates.**

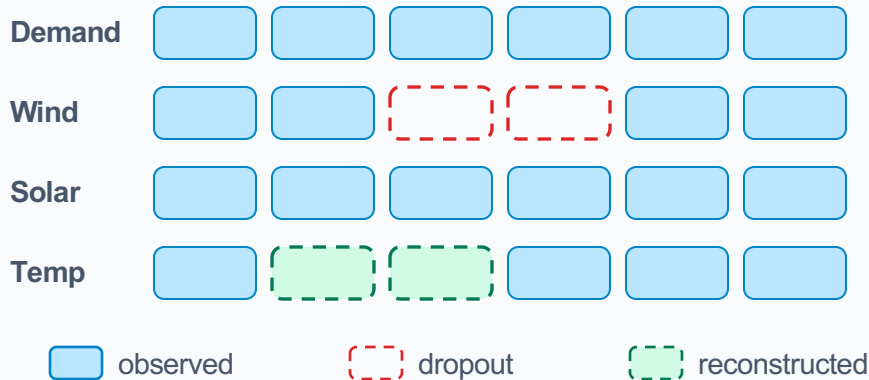
Imputation: Sensor Dropout → Reconstruction

FILL THE GAPS, CONDITIONALLY

- 1 Sensors drop out**
feeds cut out asynchronously — gaps appear mid-stream
- 2 Reconstruct conditionally**
use temporal past & future plus correlated channels
- 3 Decisions can't wait**
the system keeps operating despite unobserved segments

CHANNELS × TIME

$$\hat{\mathbf{x}}_{\text{miss}} \sim p(\mathbf{x}_{\text{miss}} | \mathbf{x}_{\text{obs}})$$



Missingness is an inference problem — reconstruct from joint dynamics, don't interpolate blindly.

Anomaly Detection: Failure or Regime Shift?

THE QUESTION

Is this segment inconsistent with the system's learned dynamics?

- **Many explanations**

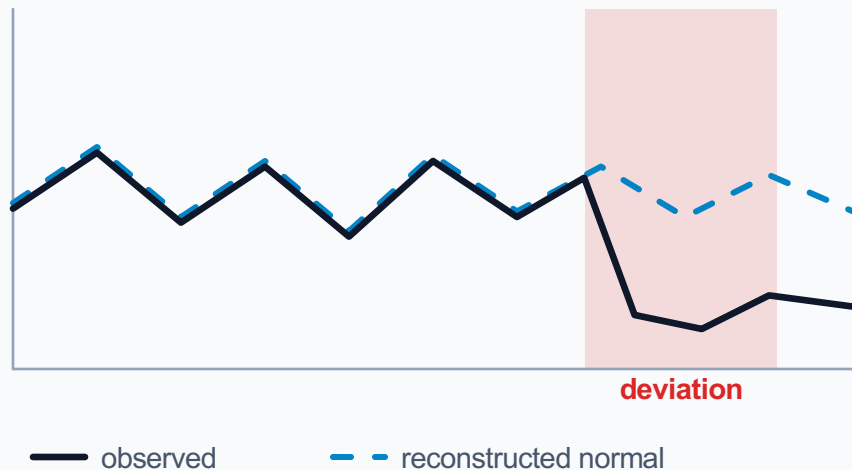
failure · weather · holiday · curtailment

- **Generative check**

reconstruct the window under normal dynamics

$$s_t = \text{distance}(x_t, \hat{x}_t) \quad s_t > \tau \Rightarrow \text{flag}$$

OBSERVED VS RECONSTRUCTED NORMAL



Large deviation flags a candidate — **context decides: failure, or a new regime.**

Naive Sequence View vs. Dynamic-System View

Naive sequence view

- **Flat 1D arrays**
evenly spaced grids of scalar values
- **Local extrapolation**
projects from window statistics alone
- **Point-error objective**
MSE/MAE collapses multimodal uncertainty

vs

Dynamic-system view

- **Partial, noisy projections**
of a latent evolving state (Kalman, 1960)
- **Native to messy sampling**
irregular, missing & asynchronous data
- **Governed by dynamics**
respects the system generating the data

Core insight — time series are incomplete observations of underlying dynamics, not just flat numbers.

Latent Dynamics & Observation Models

LATENT STATE TRANSITION

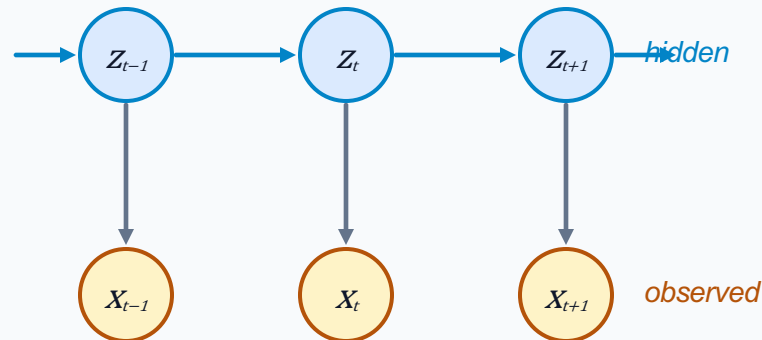
$$z_{t+1} = f(z_t) + \eta_t$$

OBSERVATION PROCESS

$$x_t = h(z_t) + \varepsilon_t$$

z hidden state · f transition · η process noise
 h measurement map · ε observation noise

GRAPHICAL MODEL — MARKOV CHAIN



Takeaway — temporal modeling is reasoning under incomplete observation of an evolving hidden system.

Evaluation is a Modeling Choice, not an Afterthought

What you measure decides which model looks "best."

Point error

- MAE / RMSE: one number
- Hides when it fails
- Blind to uncertainty

Distributional

- CRPS scores whole forecast
- Rewards honest spread
- Closer to decisions

The lesson

- Pick metric before model
- Match it to the decision
- Low average $\neq$ reliable

Deterministic Forecasting Metrics

Point Forecast Setup

A deterministic model outputs a single predicted scalar trajectory value:

$\hat{y}_t$

Point projections summarize temporal patterns as single mean or median values.

Metrics encode preferences.

Every metric choice privileges one error type over another.

Metric	What It Rewards	Primary Failure Mode
MAE	Robust absolute accuracy; median trajectory.	Ignores forecast uncertainty & variance.
MSE/RMSE	Penalizes large deviations; estimates the mean.	Extremely sensitive to localized outliers.
MAPE	Scale-invariant percentage comparison.	Highly unstable when true values near zero.
sMAPE	Symmetric relative percentage bounds.	Still problematic for tiny near-zero readings.
MASE	Cross-series comparison via seasonal baseline.	Heavily dependent on seasonal baseline choice.

Probabilistic Forecasting Metrics

Probabilistic Output Distribution Model

Observed Context ($x_{1:t}$)

Generated Multi-path Uncertainty Density: $p(y_{t+1:t+H} | x_{1:t})$

METRIC 01

Negative Log-Likelihood

How much probability density the model assigns to the actually observed outcomes.

Density Assessment (NLL)

METRIC 02

CRPS

Generalizes MAE to full predictive distributions via a strictly proper score.

Strictly Proper Score

METRIC 03

Quantile Loss

Evaluates targeted quantiles (P10 / P50 / P90) for asymmetric risk decisions.

Asymmetric Risk Lines

METRIC 04

Interval Coverage

Checks whether prediction intervals bound true outcomes at the promised rates.

Calibration Assessment

Scale and Outlier Sensitivity

Why Data Scale Matters

Magnitude Inequity

A 10-unit error means very different things for grid load (MW) vs. room temperature (°C).

Scale Bias

Scale-dependent metrics (MSE/MAE) favor high-magnitude series, skewing cross-domain rankings.

Cross-Domain Evaluation

Fair benchmarking needs scaled metrics like MASE to compare across domains.

Scale Rule: scale-dependent metrics hide weak generalization on small series.

MAE vs. MSE Outlier Sensitivity

Toy Example: Vector of Absolute Errors

Errors = [1, 1, 1, 10]

MAE (L1 - Linear)

3.25

MSE (L2 - Quadratic)

25.75

MAE scales linearly with error, giving robust, outlier-tolerant fits to the median.

MSE/RMSE scales quadratically — a single extreme outlier dominates the optimization.

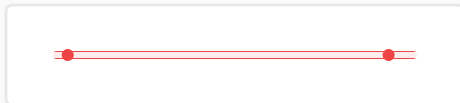
Outlier Consequence: MSE models trade normal-step accuracy to cushion rare outliers.

Calibration and Sharpness

Defect 01

Overconfident

Sharp intervals that miss the promised coverage frequency.

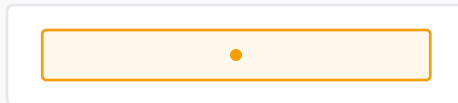


Uncalibrated & Unreliable

Defect 02

Underconfident

Intervals so wide they cover everything but guide no decision.



High Coverage & Low Sharpness

Optimal Setup

Calibrated & Sharp

Narrow intervals that still hit their promised coverage (80% covers 80%).



Reliable & Informative

Core Premise: reliable uncertainty needs both calibration and sharpness.

Metric Mismatch vs. Decision Utility

The Operational Pipeline

1. Training Objective **Loss Minimization (MSE)**



2. Evaluation Metric **Benchmark Ranking (sMAPE)**



3. Decision Utility **Real-world Consequence**

The Mismatch Gap: when objective $\neq$ metric $\neq$ utility, point metrics drive poor decisions.

Operational Decision Mismatches

ICU / Medical Diagnostics

Calibrated patient-risk estimates matter far more than raw vital-sign accuracy.

Energy Demand Planning

Underestimating peak load causes grid failure—an asymmetric loss MSE ignores.

Financial Risk Management

Assessing tail-risk probability beats optimizing flat mean trajectories.

Actionable Intelligence: evaluate predictions in the context of the decisions they drive.

Why Generalization in Temporal AI Remains Elusive



Health Care



Climate

Finance

The Solved
Era

Regular
Sample

Correlation

The
Frontier

Irregular
Sample

Physics

Why Generalization in Temporal AI Remains Elusive



Health Care



Climate



Finance

The Solved
Era

Regular
Sample

Correlation

Fixed Grid

The
Frontier

Irregular
Sample

Physics

Multi-
Resolution

Why Generalization in Temporal AI Remains Elusive



Health Care



Climate



Finance



Traffic

The Solved
Era

Regular
Sample

Correlation

Fixed Grid

Perfection

The
Frontier

Irregular
Sample

Physics

Multi-
Resolution

Miss
Values

Why Generalization in Temporal AI Remains Elusive



Health Care



Climate



Finance



Traffic



Science

The Solved
Era

Regular
Sample

Correlation

Fixed Grid

Perfection

Tokens

The
Frontier

Irregular
Sample

Physics

Multi-
Resolution

Miss
Values

Singles

The Foundation Model Promise

Pre-trained time-series systems offer opportunities to move beyond single-domain, task-specific models

- **Broad Transferability**

Reason across unseen domains, scales, and frequencies with little to no adaptation.

- **Unified Downstream Tasks**

Handle forecasting, imputation, anomaly detection, and synthesis under one interface.

- **General Prior Dynamics**

Reuse temporal priors learned from massive multi-domain corpora.

FROM PRETRAINING TO ACTIONABILITY

1. Broad Pretraining

Acquiring temporal pattern priors across large-scale heterogeneous corpora.

2. Few-Shot / Zero-Shot Adaptation

Adapt to new sequences, frequencies, and horizons with minimal local updates.

3. Grounded Decision Support

Embed physical constraints for precise, trustworthy control actions.

"Leveraging large-scale pretraining to construct generalized temporal representations."

Forecasting View: Predict Future Demand

Given the lookback window, how do we forecast load over the target horizon?



Lookback Context

Historic electricity demand, concurrent meteorological patterns, and calendar indicators.



Predictive Horizon

Extrapolating forward curves H steps into the unobserved future.



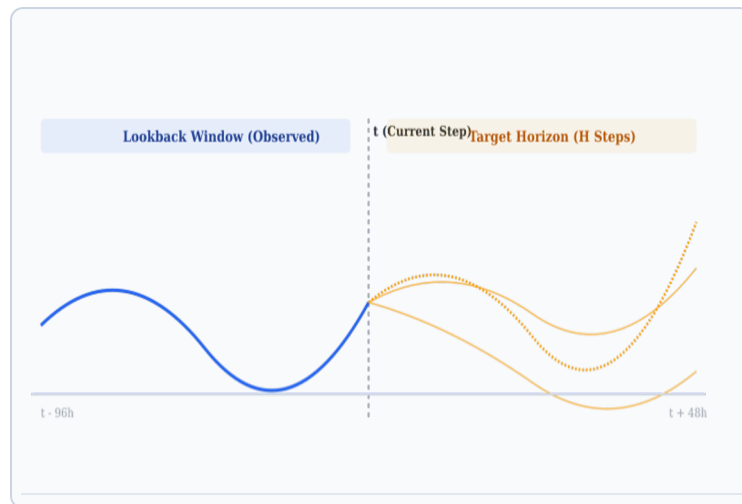
Grid Dispatch Decisions

Schedulers must evaluate multiple sampled trajectories to hedge against risk.

Main Takeaway

Operators require full trajectory distributions, not just point-wise statistical averages.

Lookback Context vs. Predictive Horizon



Stochastic Trajectory Simulation

Constraint View: Plausible but Operationally Invalid

Problem: a trajectory can look statistically plausible yet violate physical limits.



Physical Rules: generation caps, ramp rates, thermal limits, supply-demand conservation.



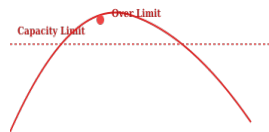
Posterior Refinement: steers the generative prior toward physical limits without retraining.

Constraint Question

Which plausible trajectories stay valid under operational limits?

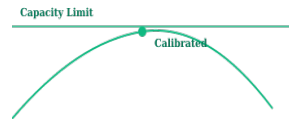
Enforcing Operational Boundaries

Unconstrained Prior



Prior forecast violates generation limits.

Posterior Guided



Guided forecast strictly respects operational rules.

Physics-Informed Posterior Guidance

Orchestration View: Which Expert Should We Trust Today?

No single model wins every regime—robust deployment needs dynamic coordination.

Classical Seasonal Model: Excels under stationary, high-frequency baseline weekdays.

Deep Autoregressive: captures nonlinear, multivariate weather interactions.

Conditional Diffusion: simulates extreme profiles and high-uncertainty storms.

Physics-Guided: handles grid-critical cases needing strict physical validity.

Agentic Auditor: detects regime shifts, resolves metric conflicts, updates trust scores.

Orchestration Question

How to select, combine, and explain experts under deployment shift?

Dynamic Expert Dispatch Matrix

Regime	Best Model	Selected Weight
Stationary Weekday	ARIMA / ETS	0.75
Extreme Temperature	TimesFM (GNN)	0.80
Storm / High Uncertainty	TimeDiT (Diffusion)	0.90
Grid-Critical Peak	PINFDiT (Physics)	1.00

Transition: with this example in mind, we enter the first layer—representation.

The core claim: TSFMs should be compositional inference systems.

THE INTELLECTUAL CLAIM OF THIS Tutorial

*"Time-series foundation models should not be single universal forecasters; they should be designed as **compositional inference systems** that expose temporal structure, generate plausible trajectories, respect domain constraints, and orchestrate evidence under deployment shift."*

1. Expose Structure

Decomposing raw observations into trend, seasonality, and residual components to align cleanly with pre-trained priors.

2. Generate Trajectories

Modeling joint probability distributions over sequence spaces, unifying diverse downstream analysis tasks.

3. Respect Constraints

Refining unconstrained statistical samples by projecting predictions onto governing physical manifolds at inference time.

4. Orchestrate Evidence

Auditing validation errors and managing expert selections dynamically to preserve high faithfulness under deployment shifts.

Sixty Years of Temporal Modeling

A reading lens on the literature: three waves, and the gap they leave open

01

Classical Forecasting

ARIMA, ETS, state-space:
strong per-series baselines with
clear assumptions.

02

Deep Temporal Models

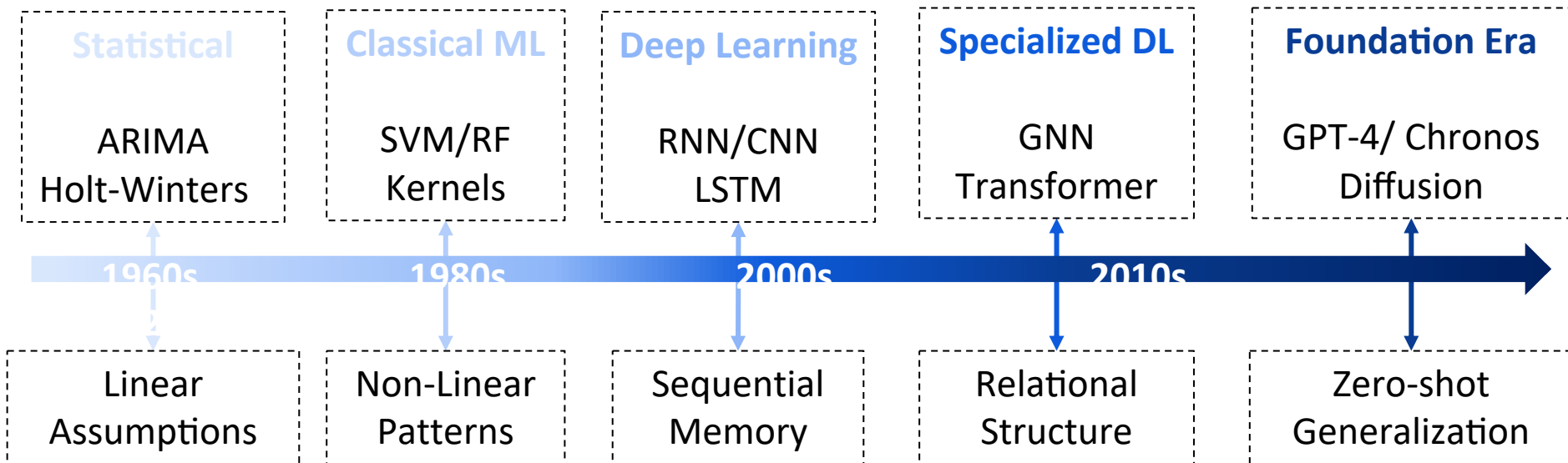
From RNNs to Transformers:
architectures as representation
interfaces.

03

Pretrained TSFMs

Transfer across domains — and
why reliability remains
unsolved.

From Statistics to Intelligence: A 60-Year Journey



The field has Progressed, but the Pieces Remain Fragmented.

Classical / Decomposition

Strong temporal structure,
weak transfer across
domains.

Deep / Transformer Models

Strong capacity, weak
general inference.

Pretrained TSFMs

Strong zero-shot
forecasting, weak
constraints and
orchestration.

Generative / Physics / Agents

Promising pieces, not yet
integrated.

This motivates our four-layer stack.

Classical Forecasting: Explicit Structure Before Scale

Autoregression ARIMA

Encodes memory via lags, differencing, and residual terms.

Lag-dependent residuals

Exponential Smoothing ETS Models

Separates level, trend, and seasonality for explicit projection.

Additive component math

Latent Systems State-Space Models

Treats data as noisy glimpses of a latent state evolving over time.

Uncertainty & filtering

Signal Deconstruction Decomposition

Surfaces trend, seasonality, and residuals as structural anchors.

STL & classical windows

Deep Temporal Models: Capacity Through Interfaces

Recurrence

Memory Interface

Recurrent hidden states encode continuous Markovian memory.

Step-by-Step Recurrence

Local Neighborhoods

Convolutional

Convolutions capture local and dilated context without sequential steps.

Parallel Dilated Fields

Global Context

Attention/Token

Tokenizes series (points, patches, channels) to learn non-local relations.

Tokenized Attention

Interconnected Systems

Relational

Graphs model multivariate links, encoding physical topology.

Spatio-Temporal Graphs

Core Premise: architecture choices are interface choices—capacity helps, but reliable inference needs structured protocols.

N-BEATS: Neural Basis Expansion Analysis for Interpretable Time Series Forecasting

Boris N. Oreshkin, Dmitri Carпов, Nicolas Chapados, and Yoshua Bengio — *ICLR*, 2020

Core Problem

How can neural nets forecast nonlinear series while keeping decomposable structure?

Key Idea / Methodology

Stacks backward/forward residuals with polynomial (trend) and harmonic (seasonal) bases.

Doubly Residual Block Stacking

The Temporal Interface

Learns trend and seasonality bases that map inputs to decodable projections.

Why It Matters for TSFMs

Neural forecasters hit SOTA precisely by adding structured temporal inductive biases.

Neural Interpretable Basis

Remaining Gap

A forecasting-only architecture—no generative adaptation or multi-task conditioning.

Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long — *NeurIPS*, 2021

Core Problem

Can Transformers model long-range periodicity without attention's cost and scatter?

Key Idea / Methodology

Embeds decomposition blocks in each layer; swaps attention for FFT-based Auto-Correlation.

FFT-based Auto-Correlation

The Temporal Interface

Makes progressive decomposition a core internal layer of the sequence model.

Why It Matters for TSFMs

Classical decomposition stays valuable even inside large Transformers.

Transformer-Embedded Decomposition

Remaining Gap

Still a task-specific forecaster—no generative multi-task, zero-shot transfer, or constraints.

Are Transformers Effective for Time Series Forecasting?

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu — AAAI, 2023

Core Problem

Are Transformers really needed for long-horizon forecasting, or can simpler models win?

Key Idea / Methodology

DLinear splits series into trend and seasonal parts, then maps each with one linear layer.

Decomposition + 1-Layer Linear Mapping

The Temporal Interface

A lean decomposition + scale-aware linear interface, no positional-encoding tricks.

Why It Matters for TSFMs

The temporal interface often matters more than backbone complexity.

Temporal Interface Dominance

Remaining Gap

Not pretrained, generative, or constraint-aware—can't join foundation-model workflows.

Graph WaveNet for Deep Spatial-Temporal Graph Modeling

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang — *IJCAI*, 2019

Core Problem

How do we model spatial-temporal links when the graph is missing or wrong?

Key Idea / Methodology

Graph WaveNet learns an adaptive graph from data, stacked with dilated causal convolutions.

Adaptive Graph Learning + WaveNet Convolutions

The Temporal Interface

A learned relational interface that infers variable coupling from observations.

Why It Matters for TSFMs

Relational structure can be discovered automatically, not hand-specified.

Learned Latent Graph Topology

Remaining Gap

Still a localized forecaster—not a general multi-task pretraining paradigm.

DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks

Valentin Salinas, Jan Gasthaus, Bernhard Schulz, David Stella, and Jan Peter — *International Journal of Forecasting*, 2020

Core Problem

Can we learn one global probabilistic model across related series instead of local fits?

Key Idea / Methodology

A global autoregressive LSTM maps history and covariates to a probabilistic likelihood.

Global Likelihood Optimization

The Temporal Interface

A recurrent memory interface with parametric output distributions (e.g., Negative Binomial).

Why It Matters for TSFMs

A defining shift from per-series statistics to globally-learned representations.

Recurrent Probabilistic Architecture

Remaining Gap

Forecasting-centric—no unified interface for missingness, synthesis, or physical constraints.

An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun — *arXiv preprint arXiv:1803.01271*, 2018

Core Problem

Are recurrent nets (LSTM/GRU) really necessary for long-range temporal memory?

Key Idea / Methodology

TCNs use causal, dilated 1D convolutions with residuals for parallel training.

Parallel Dilated Temporal Receptive Fields

The Temporal Interface

A parallel receptive-field interface that replaces recurrence with dilated projections.

Why It Matters for TSFMs

Recurrence isn't the only way to store memory—parallel architectures work too.

Dilated Causal Temporal Convolutions

Remaining Gap

A generic sequence model—no time-series tokenization, alignment, or transfer layers.

Modeling Long- and Short-Term Temporal Patterns with Deep Neural Networks

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu — *SIGIR*, 2018

Core Problem

How do we capture local high-frequency features and long-term periodicity at once?

Key Idea / Methodology

LSTNet fuses 1D convolutions, an RNN, recurrent-skip connections, and a linear AR term for stability.

Convolution-Recurrent-Autoregressive Stack

The Temporal Interface

A multi-scale interface: local conv features, recurrent memory, and a linear stability path.

Why It Matters for TSFMs

An early deep model with distinct pathways for different temporal scales.

Multi-Scale Inductive Bias

Remaining Gap

Task-specific forecaster—weak zero-shot transfer, not a general adaptor.

Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang — AAAI, 2021

Core Problem

How do we scale Transformers to very long sequences without quadratic cost?

Key Idea / Methodology

Informer adds ProbSparse attention, sequence distilling, and a generative long-horizon decoder.

ProbSparse $O(L \log L)$ Attention

The Temporal Interface

A sparse global attention interface that propagates only the most informative steps.

Why It Matters for TSFMs

Proves attention can scale efficiently across long temporal histories.

Sparse Self-Attention Mechanics

Remaining Gap

Solves attention cost only—no cross-domain tokenization, scale drift, or constraints.

A Time Series Is Worth 64 Words: Long-Term Forecasting with Transformers

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam — *ICLR*, 2023

Core Problem

How should continuous point-level series become stable tokens for Transformers?

Key Idea / Methodology

PatchTST groups neighboring steps into overlapping patches and models channels independently.

Sub-series Patch-level Tokenization

The Temporal Interface

A patch-based tokenization interface that shields tokens from point-level noise.

Why It Matters for TSFMs

Tokenization design is the key factor for adapting Transformers to time series.

Patch-wise Attention Mapping

Remaining Gap

A strong forecaster, but not a generative, multi-task, constraint-aware foundation model.

iTransformer: Inverted Transformers Are Effective for Time Series Forecasting

Yuxuan Liu, Tianping Zhang, Jianmin Wang, and Mingsheng Long — *ICLR*, 2024

Core Problem

Should attention run over time steps or over variable channels?

Key Idea / Methodology

iTransformer embeds each variable's full history as a token and attends across variables.

Inverted Channel-as-Token Dimension Embedding

The Temporal Interface

An inverted channel-wise interface where variables act as tokens for cross-series links.

Why It Matters for TSFMs

Redefining the token radically changes what the model learns.

Inter-Channel Attention Modeling

Remaining Gap

Still a discriminative forecaster—no unified generative inference or imputation.

Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu — *ICLR*, 2018

Core Problem

How do we model traffic series where channels are coupled by road networks?

Key Idea / Methodology

DCRNN runs diffusion convolutions over directed graphs with an encoder–decoder GRU.

Diffusion Convolution + Recurrent Gated Units

The Temporal Interface

A spatio-temporal graph interface fusing network topology with recurrence.

Why It Matters for TSFMs

Multivariate series are coupled systems needing relational structure.

Spatial-Temporal Graph Coupling

Remaining Gap

Relies on static predefined graphs—no zero-shot transfer across domains.

Deep models gave us capacity, but not general temporal inference.

Transformers scale to huge data – but still train one task, one dataset at a time.

Strength

- High capacity, flexible
- Strong on benchmarks

Limitation

- Task-specific training
- No unified inference

Takeaway

- Capacity is necessary
- But not general reasoning

Gap: Capacity Without General Temporal Inference

01

Capacity $\neq$ Reliability

Deep temporal models have massively expanded neural representation capacity, yet they often struggle with extreme sensitivity to raw scale-shifts and variable frequencies.

02

Supervised Task Walls

Most architectures are designed strictly as task-specific (e.g., forecasting-centric) or dataset-specific predictors, breaking down under arbitrary missingness or multi-task inference.

03

Unconstrained Projections

Deep models are fundamentally unconstrained, incapable of actively incorporating or enforcing external physical boundaries, logical invariants, or physical laws.

Critical Unmet Requirements:

✖ Unified Generative Adaptability

✖ Enforceable Invariant Constraints

✖ Zero-Shot Cross-Domain Transfer

✖ Reliable Runtime Orchestration

"The next step is temporal foundation modeling: can broad pretraining produce reusable temporal interfaces?"

Time-Series Foundation Models: The New Landscape

Defining TSFMs

A TSFM is a large neural network pretrained on diverse temporal corpora.

It transfers zero-shot across unseen domains, horizons, variables, and frequencies.

Cross-Domain Pretraining Paradigm

THE SEVEN ARCHITECTURAL INTERFACE DIRECTIONS

1. Value-Token / Quantization

Maps values to discrete tokens for language backbones (e.g., Chronos).

2. Decoder-Only Forecasting

Predicts future intervals autoregressively using causal patches (e.g., TimesFM).

3. Lag-Based Probabilistic

Leverages lag covariates for parameterizing uncertainty curves (e.g., Lag-Llama).

4. Masked Patch Encoder

Learns multi-variate correlations with mask reconstruction (e.g., MOMENT, Moirai).

5. Mixture-of-Experts (MoE)

Sparse routing scales model capacity efficiently (e.g., Time-MoE).

6. Generative / Task-Unification

Unifies forecasting, imputation, and anomaly tasks (e.g., Timer).

Core Premise: the key question isn't scale but how temporal structure is represented, tokenized, and conditioned.

Temporal Tasks as Conditional Generation

Predictive Forecasting Observe historical context, generate clean future trajectories. [Past] → [Future?]	Interpolative Imputation Observe partial scattered points, generate missing values inside. [Ob-X-Ob-X-Ob]	Diagnostic Anomaly Observe possibly corrupted inputs, infer structural coherence. [Reconstruct Diff]	Generative Synthesis Unconditioned or class-conditioned sampling of whole paths. [Noise] → [Data]	Scientific Simulation Condition on physical boundary evidence, generate valid trajectories. [Boundaries] → [PDEs]
--	--	---	--	--

Core Premise: Task identity can be represented entirely by the **conditioning mask**. Under this unified view, all temporal tasks collapse into conditional generative inference over a dynamic process.

The Current Landscape of Time-Series Foundation Models

TimeGPT

2023

Nixtla

- ✓ Strong zero-shot accuracy, simple API, minimal tuning.
- ! Proprietary service, limited visibility, no local deployment.

Chronos Series

2023–26

Amazon **Science**

- ✓ Open weights, T5-style tokenizer, strong benchmarks.
- ! Univariate bias in early versions; prompt sensitive; heavy.

TimesFM Series

2024–25

G Google

- ✓ Large decoder-only, long context, supports covariates.
- ! Initially fixed horizons; some variants closed; complex tuning.

A comprehensive overview of leading architectures and their trade-offs

Moirai Series

2024–25

☁ Salesforce

- ✓ Universal design for multivariate/covariates; quantile output.
- ! v1.0 inefficient training; mixed performance vs TiDE.

IBM TTM-R

2024–25

IBM

- ✓ Fine-tuned for demand forecasting; strong on retail data.
- ! Commerce focused (narrow domain); less available tooling.

TabPFN-TS

2024–25

Prior Labs

- ✓ Fast forecasting via TabPFN; strong in low-data regimes.
- ! Restricted context length; not for very long sequences.



WHAT THEY ACHIEVED

Strong zero-shot forecasting, open-source ecosystem growth, large-scale pretraining feasibility.



WHAT REMAINS MISSING

Probabilistic calibration, data robustness (irregularity), and scientific grounding (physical consistency).

TimeGPT-1

Azul Garza and Max Mergenthaler-Canseco — *arXiv preprint arXiv:2310.03260*, 2023

Core Problem

Can one model, pretrained on massive diverse data, forecast zero-shot?

Key Idea / Methodology

An encoder-decoder pretrained on 100B+ points, served as a zero-shot cloud API.

Billion-Scale Multi-Domain Pretraining

The Temporal Interface

A clean API mapping historical windows to zero-shot futures.

Why It Matters for TSFMs

An early commercial TSFM proving zero-shot forecasting works in practice.

Zero-Shot Global Forecasting Prior

Remaining Gap

Forecasting-only and closed-source—no multi-task generation, constraints, or orchestration.

Chronos: Learning the Language of Time Series

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, and co-authors — *arXiv preprint arXiv:2403.07815*, 2024

Core Problem

Can continuous values use the same recipes as discrete language?

Key Idea / Methodology

Chronos normalizes, quantizes values into token bins, and trains T5/GPT-style models.

Value-Token Bin Quantization

The Temporal Interface

A value-token interface turning scalars into language tokens.

Why It Matters for TSFMs

Shows language models can “read” time series with no structural change.

Language Model Value Mapping

Remaining Gap

Quantization loses resolution, misses cross-channel links, and breaks under scale mismatch.

A Decoder-Only Foundation Model for Time-Series Forecasting

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yuxin Zhou — *ICML*, 2024

Core Problem

Can a decoder-only model, trained only on time series, forecast zero-shot?

Key Idea / Methodology

TimesFM patches the series and trains a causal decoder for multi-step prediction.

Decoder-Only Patch-level Causal Attention

The Temporal Interface

A causal patch interface predicting future horizons from context.

Why It Matters for TSFMs

Open-source proof that causal pretraining yields industry-grade zero-shot forecasts.

Causal Autoregressive Decoders

Remaining Gap

Forecasting-only—a causal predictor can't unify imputation or physical invariants.

Lag-Llama: Towards Foundation Models for Probabilistic Time Series Forecasting

Kashif Rasul, Anderson de Souza, Dustin G. Reis, and co-authors — *arXiv preprint arXiv:2310.08278*, 2023

Core Problem

Can a decoder-only model output calibrated intervals, not just points?

Key Idea / Methodology

Lag-Llama feeds multi-period lag features into a Llama decoder that outputs full distributions.

Lag Covariate Input + Probabilistic Head

The Temporal Interface

A lag-based probabilistic interface generating interval densities, not flat means.

Why It Matters for TSFMs

Brings uncertainty bounds into TSFMs for risk-aware zero-shot decisions.

Causal Probabilistic Modeling

Remaining Gap

Forecasting-only—no masked conditional generation or constraint handling.

MOMENT: A Family of Open Time-Series Foundation Models

Mononito Goswami, Konrad Sadowski, Arjun Dubey, and co-authors — *ICML*, 2024

Core Problem

Can we build open, reproducible TSFMs spanning classification, representation, and forecasting?

Key Idea / Methodology

Builds the Time-Series Pile corpus and pretrains open patch encoders via masked reconstruction.

Time-Series Pile + Masked Reconstruction Encoder

The Temporal Interface

A masked-reconstruction interface learning embeddings for many tasks.

Why It Matters for TSFMs

Moves the field toward open-source, reproducible foundation-model analysis.

Open Representation Architectures

Remaining Gap

Representations still need generative, constraint, and orchestration layers on top.

Unified Training of Universal Time Series Forecasting Transformers

Gerald Woo, Chenghao Liu, Akshat Kumar, and co-authors — *ICLR*, 2024

Core Problem

Can one Transformer forecast across diverse domains, dimensions, and frequencies?

Key Idea / Methodology

Moirai pairs Any-variate Attention with any-size patches, trained on the massive LOTSA archive.

Any-variate Masked Encoder Transformer

The Temporal Interface

A universal masked-encoder interface for flexible multivariate resolutions.

Why It Matters for TSFMs

A milestone toward universal zero-shot forecasting across unseen domains.

Universal LOTSA Pretraining Archive

Remaining Gap

Tested mainly on forecasting—conditional synthesis and physical constraints stay open.

Timer: Generative Pre-trained Transformers Are Large Time Series Models

Yuxuan Liu, Haixu Wu, Fuliang Zhang, and co-authors — *ICLR*, 2024

Core Problem

Can generative pretraining unify forecasting, imputation, and anomaly detection?

Key Idea / Methodology

Timer frames every task as next-patch prediction, pretrained across domains for zero-shot multi-task use.

Autoregressive Next-Patch Token Prediction

The Temporal Interface

A sequence-generation interface unifying prediction and token imputation.

Why It Matters for TSFMs

A key step toward a unified, task-adaptive temporal foundation model.

Unified Task Sequence Modeling

Remaining Gap

Still open: probabilistic sampling, physical constraints, and deployment-scale orchestration.

Time-MoE: Billion-Scale Time Series Foundation Models with Mixture of Experts

Sheng Shi, Shuo Yu, Minshu Wang, and co-authors — *arXiv preprint arXiv:2411.02344*, 2024

Core Problem

Can TSFMs scale to billions of parameters without prohibitive cost?

Key Idea / Methodology

Time-MoE swaps dense FFNs for sparse Mixture-of-Experts, routing patches to specialized experts.

Sparse Gating routing + Specialized Experts

The Temporal Interface

A dynamic expert-routing interface mapping temporal sub-patterns to subnetworks.

Why It Matters for TSFMs

Billion-parameter TSFMs need sparse routing, not raw dense scaling.

Billion-Scale Sparse MoE Scaling

Remaining Gap

Scale alone can't guarantee physical validity, reliable generation, or expert orchestration.

Building Temporal Foundation Models

A hands-on walkthrough in four modules, each anchored by a working system

01

Representation

Align temporal structure with pretrained backbones — Module 1: TEMPO.

02

Generation

One diffusion model for forecasting, imputation, and anomalies — Module 2: TimeDiT.

03

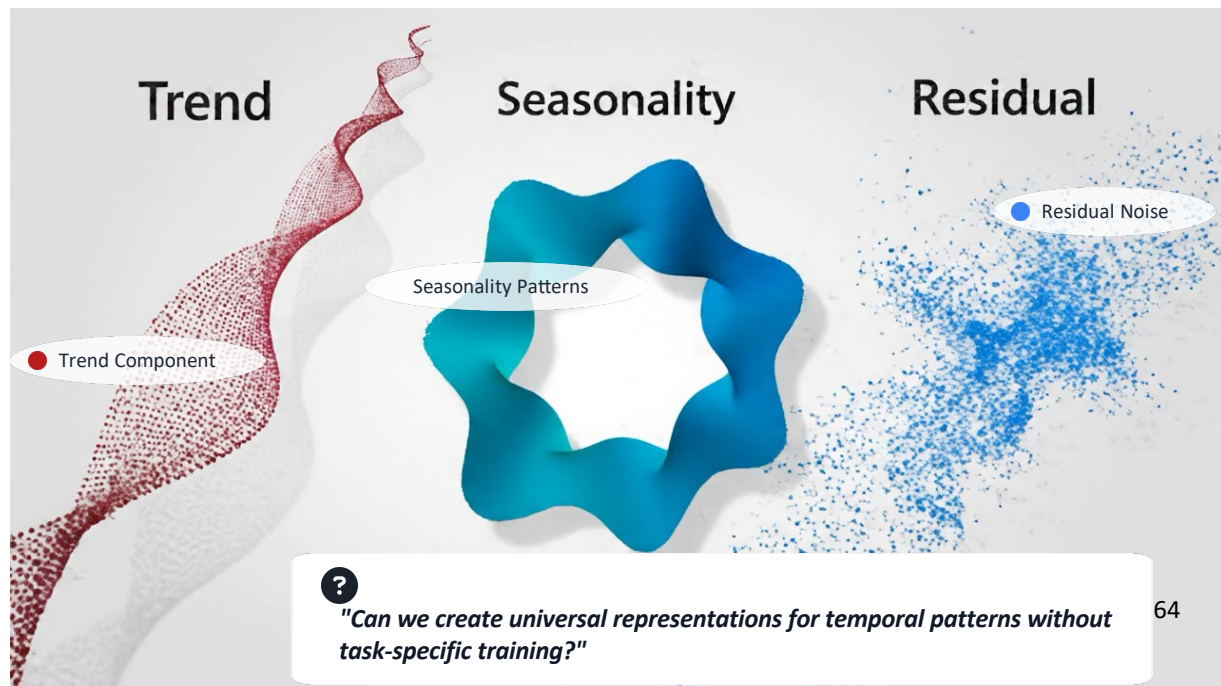
Grounding & Orchestration

Physics-valid outputs and expert routing — Modules 3–4: PINFDiT & agents.

Pillar I: Learning Universal Temporal Semantics

Q: How can a pretrained Transformer understand continuous signals?

A: Expose trend, seasonality, and residual structure before adaptation. (TEMPO)



Representation: Teaching Models to See Structure

How to align temporal signals with pretrained backbones — case study: TEMPO

01

Why Raw Numbers Fail

Discrete tokens vs. continuous signals: what LLMs miss in time series.

02

Decomposition as Prior

Expose trend, season, and residual before the backbone sees the data.

03

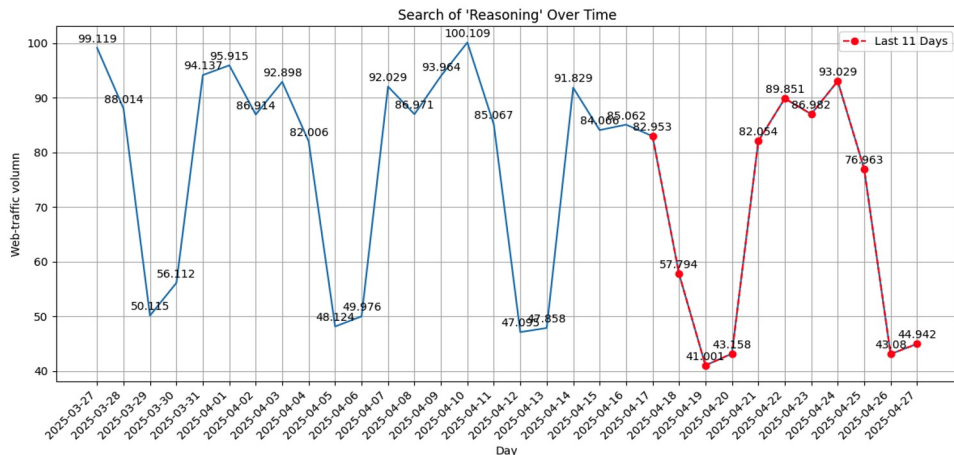
Zero-Shot Transfer

Structured inputs enable competitive transfer to unseen domains.

LLMs Do Not Internalize Temporal Structure

Empirical Evidence: Seasonality Is Not Explicitly Modeled

Ground Truth



✓ Structured periodic signal

Experiment Setup

Prompt: “Given the historical time series, predict the next 11 days.”

Model Forecast Behavior

Day	Claude	GPT-4o	DS-R1
+1	83.7	77.1	100.1
+2	82.9	76.9	85.0
+3	83.2	76.7	47.1
+4	84.1	76.6	47.8
+5	85.4	76.4	91.8
+6	87.1	76.2	84.0
...	...	...	...

Observed Structural Patterns

Claude Smooth trend continuation ↗
GPT-4o Monotonic drift upward ~
DeepSeek Rigid periodic copying ↻

❗ All models extrapolate surface statistics.

Sequence continuation ≠ Temporal understanding

LLMs predict token transitions — they do not model frequency dynamics.

Discrete Tokens vs. Continuous Signals: Why LLMs Struggle with Numbers

Language Models See Tokens — Not Signals

What LLMs Are Trained On

"XX University is a public research university..."

XX University is a
public research

- ✓ Semantic structure
- ✓ Grammar and contextual meaning
- ✓ Strong linguistic inductive bias

Language has grammar.

What Happens With Numbers?

48.123, 49.9768, 92.02112...

"48" "." "123" "," "49"
"." "9768"

- ✗ No semantic anchoring
- ✗ No syntactic grammar
- ✗ Fragmented discrete tokens

No signal awareness.
Only token prediction.

A language model has never learned the grammar of time.

Training on Numbers Does Not Solve the Structural Problem

Data exposure $\neq$ Temporal inductive bias

Controlled Pretraining Experiment



Model Selection

Pretrain open-source LLMs on same time-series data



Identical Conditions

Same datasets & forecasting tasks



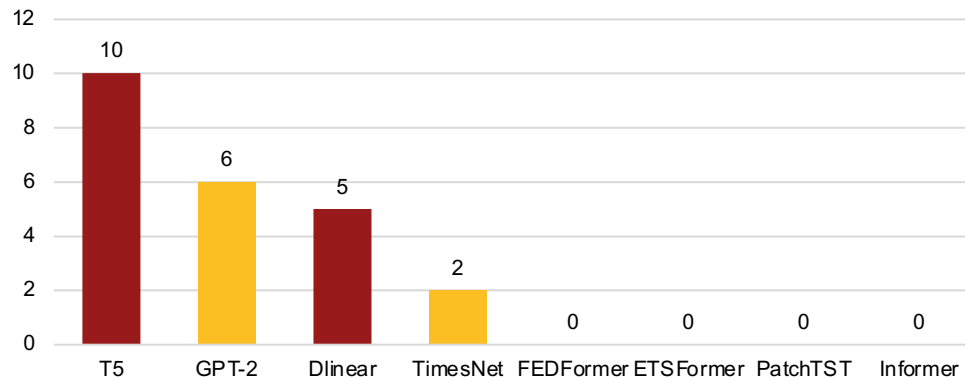
Rigorous Evaluation

21 settings (7 datasets $\times$ 3 horizons)



"If data were sufficient, results should converge."

Top-1 Wins Across 21 Settings



Observed Outcome

T5

10 wins

GPT-2

6 wins

Specialized

Competitive



Performance varies significantly despite identical data exposure.

Access to numbers is not the bottleneck

Before TEMPO, we need one classical idea: decomposition.

Any series splits into trend + seasonality + residual. TEMPO builds on exactly this.

Trend

- Slow, long-term movement
- The overall direction

Seasonality

- Repeating cycles
- Daily / weekly / yearly

Residual

- What's left over
- Noise and surprises

Why Decompose a Time Series?

The Core Algebraic Idea

A continuous time series mixes multiple temporal components:

$$y_t = \text{trend}_t + \text{seasonality}_t + \text{residual}_t$$

• Trend Component

The slow-moving, persistent directional movement representing long-term structural changes.

• Seasonality Component

Repeated, cyclical fluctuations tied to fixed calendar intervals or frequency rates.

• Residual / Remainder

The remaining transient variations, high-frequency noise, anomalies, or unmodeled physical shocks.

Why Decompose? It helps neural models avoid mixing scales, frequencies, and transient shocks, providing an interpretable representation interface.

Component Stack Visualization

1. Raw Sequence (y_t)



2. Slow Trend (T_t)



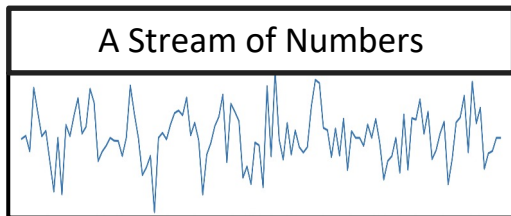
3. Periodic Seasonality (S_t)



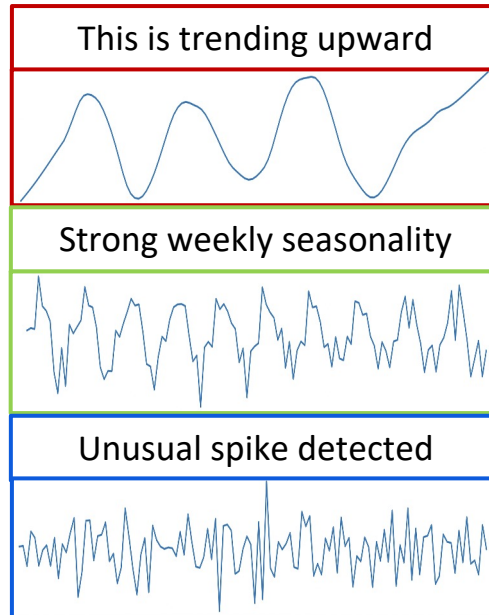
Visual Guide: Structural components describe different physical frequencies of the system, enabling the neural network to analyze slowly varying properties and high-frequency residuals independently.

Motivation: Models See Numbers, Not Patterns

Current Models See:



What We Need:



The Core Question: What if we could build a model that learns the universal grammar of time?

STL: Seasonal-Trend Decomposition Using LOESS

The LOESS Smoothing Loop Pipeline



Feature 01

Flexible Seasonality

Lets the seasonal component evolve over time, with a user-controlled rate of change.

Dynamic Period Fitting

Feature 02

Robust to Outliers

Robust fitting options damp extreme anomalies, keeping trend and seasonal waves undistorted.

Outlier-Shielded Smoothing

Feature 03

Multi-Frequency Capacity

Handles any seasonal frequency (daily, weekly, monthly) and scales to long contexts.

Scale-independent Interface

Decomposition in Neural Forecasting

ICLR 2020

N-BEATS

Learns basis expansions with interpretable trend and seasonality blocks.

Shows deep networks can preserve structural components without losing accuracy.

Learned Basis Expansions

NeurIPS 2021

Autoformer

Embeds decomposition blocks directly inside a Transformer architecture.

Uses self-correlation over components to guide attention in long-horizon forecasting.

Inner-Layer Decomposition

AAAI 2023

DLinear

Combines a simple moving average filter with **single-layer linear mapping heads**.

Out-performed un-decomposed attention networks — structural interface design can rival back-end complexity.

Linear Component Mapping

When Decomposition Fails

Failure Case 01

Drifting Seasonality

Periodic amplitude and wavelength shift over time, confusing static seasonal filters.

Dynamic Phase Shifts

Failure Case 02

Regime Shifts

A structural break alters the baseline, making historic trends obsolete.

Trend Breakdown

Failure Case 03

Residual Shocks

Residuals are not white noise; they carry abrupt physical shocks and policy events.

Critical Noise Errors

Failure Case 04

Wrong Period Bias

Wrong cyclic lengths (daily vs. weekly) heavily distort the decomposition.

Model Distortion

Core Premise: Classical decomposition is a useful structured proxy — not an oracle. Foundation models must learn to reweight and calibrate its components.

TEMPO Aligns Temporal Dynamics with Semantic Structure.



Semantic Decomposition

Reveal latent temporal structure.



Prompt Alignment

Embed structure into a shared representation space.

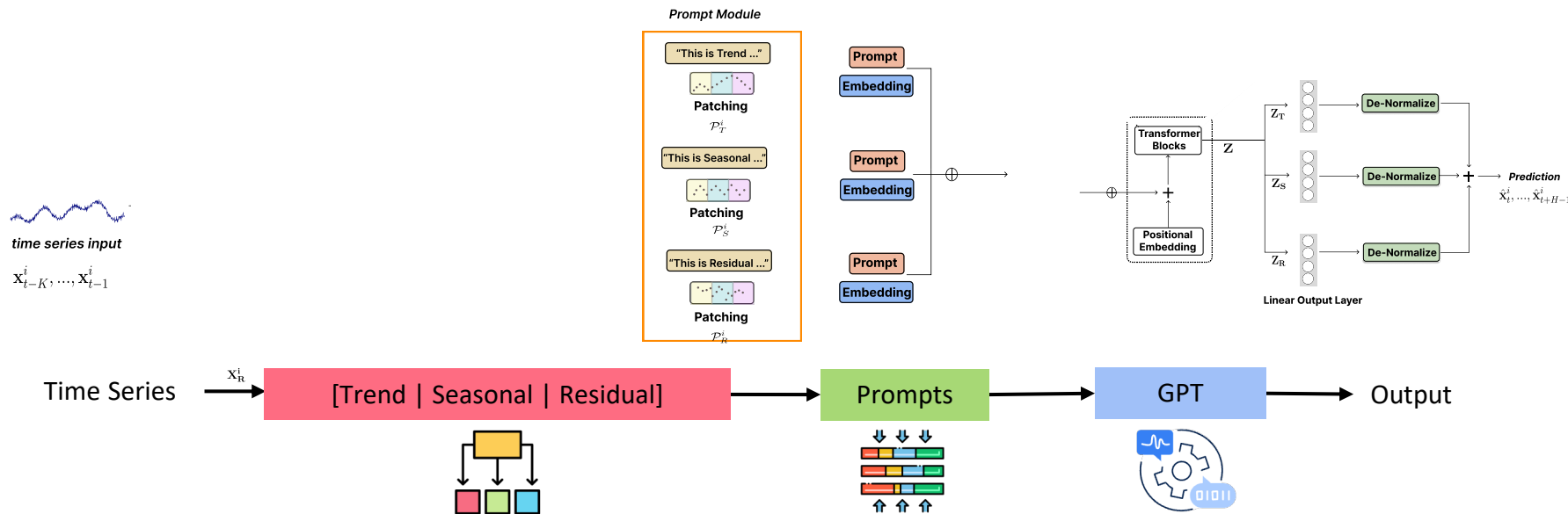


Pretrained Priors

Leverage transferable pattern knowledge.



TEMPO exposes structure before using a pretrained backbone.



Why Attention Cannot Do Decomposition Automatically

*The fundamental conflict between **linear projections** and **non-orthogonal components**.*



Theorem 3.1: Orthogonality Constraint

$$\text{Linear}(X_T + X_S) \neq \text{Linear}(X_T) \oplus \text{Linear}(X_S)$$

Standard attention mechanisms apply a single global linear projection to the input embedding.

When Trend and Seasonal components overlap in the feature space, a single projection cannot separate them into disjoint subspaces.

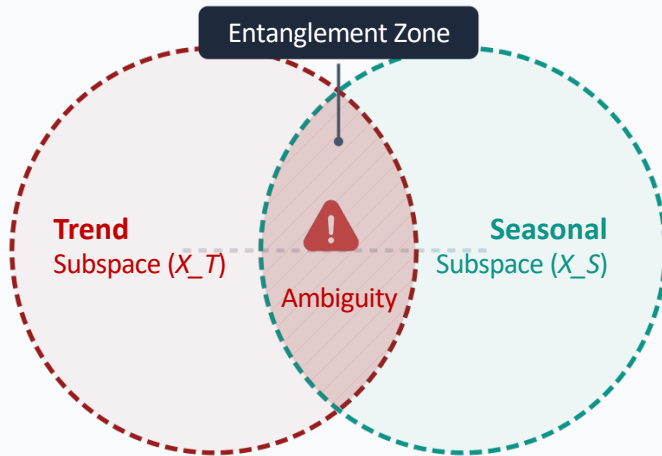
The Consequence



Information Loss

Trying to isolate one component inevitably corrupts the other within the shared intersection

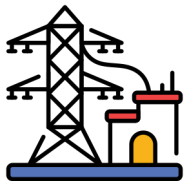
Visual Proof Overlapping Subspaces



Experiments Setting

➤ Dataset:

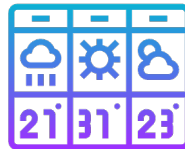
Long term forecasting (Use 336 history time steps to predict the future 96/192/336/720 time steps)



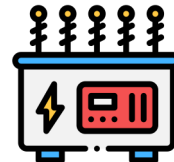
Electricity



Traffic



Weather



Electricity Transformer
Temperature

Table 1: Zero-shot of long-term forecasting results on time series benchmark datasets. We use prediction length $O \in \{96, 192, 336, 720\}$. A lower MSE indicates better performance. Hereafter, for the tables, the best results are marked in bold and the second optimal in underlined, respectively with MSE/MAE.

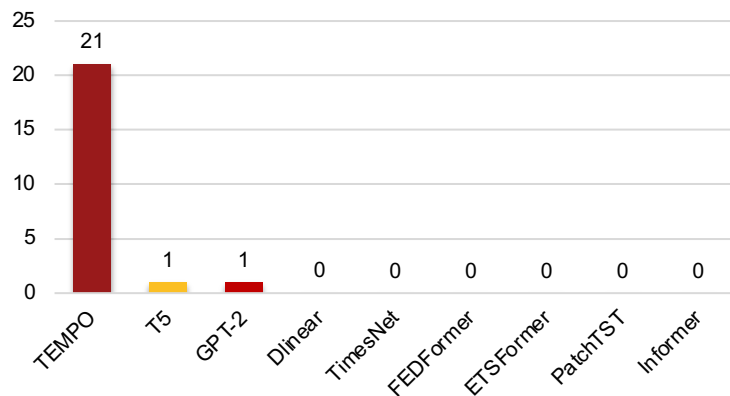
Experiments Results - ZS

- TEMPO consistently outperforms state-of-the-art (SOTA) methods in zero-shot settings across various datasets;
- TEMPO's performance on unseen datasets showcases its potential as a foundational model for time series forecasting.

Horizon	Model	ECL	Traffic	Weather	Ettm1	Ettm2	Eth1	Eth2
		MSE/MAE	MSE/MAE	MSE/MAE	MSE/MAE	MSE/MAE	MSE/MAE	MSE/MAE
96	TEMPO	0.178/0.276	0.476/0.343	0.211/0.254	0.438/0.424	0.185/0.267	0.400/0.406	0.301/0.353
	GPT2	0.193/0.288	0.522/0.380	0.226/0.274	<u>0.486/0.438</u>	0.193/0.273	0.400/0.416	0.320/0.363
	T5	<u>0.185/0.282</u>	<u>0.508/0.366</u>	<u>0.217/0.271</u>	0.529/0.464	<u>0.190/0.268</u>	0.400/0.409	0.328/0.366
	PatchTST	0.489/0.546	1.023/0.641	0.247/0.301	0.733/0.554	0.273/0.345	0.570/0.518	0.379/0.412
	Timesnet	0.293/0.369	0.585/0.401	0.247/0.295	0.518/0.470	0.202/0.290	<u>0.407/0.423</u>	<u>0.315/0.362</u>
	FEDformer	0.300/0.399	0.835/0.564	0.292/0.346	0.698/0.553	0.665/0.634	0.509/0.502	0.385/0.426
	ETSformer	0.707/0.638	1.419/0.795	0.453/0.416	1.117/0.678	0.353/0.404	0.469/0.457	0.405/0.428
	Informer	0.512/0.531	1.400/0.830	0.837/0.711	0.880/0.657	0.263/0.360	0.642/0.562	0.704/0.651
	DLinear	0.195/0.292	0.609/0.424	0.212/0.275	0.624/0.522	0.264/0.352	0.414/0.421	0.334/0.389
192	TEMPO	0.198/0.293	0.496/0.355	0.254/0.298	0.461/0.432	0.243/0.304	0.426/0.421	0.355/0.389
	GPT2	0.207/0.300	0.533/0.387	0.273/0.312	<u>0.516/0.461</u>	0.254/0.312	0.441/0.433	<u>0.381/0.402</u>
	T5	<u>0.205/0.302</u>	<u>0.524/0.374</u>	0.277/0.321	<u>0.523/0.454</u>	<u>0.246/0.306</u>	<u>0.428/0.426</u>	0.413/0.410
	PatchTST	0.465/0.535	0.992/0.633	0.277/0.324	0.739/0.563	0.299/0.355	0.580/0.528	0.387/0.417
	Timesnet	0.283/0.366	0.64/0.431	0.316/0.342	0.55/0.490	0.261/0.318	0.439/0.439	0.394/0.406
	FEDformer	0.390/0.468	0.869/0.579	0.372/0.426	0.819/0.608	0.358/0.416	0.683/0.596	0.921/0.748
	ETSformer	0.721/0.645	0.995/0.658	0.545/0.466	1.598/0.803	0.390/0.416	0.548/0.503	0.476/0.468
	Informer	0.625/0.619	0.872/0.506	0.431/0.455	1.461/0.892	0.494/0.516	0.798/0.632	0.455/0.883
	DLinear	<u>0.204/0.300</u>	<u>0.505/0.412</u>	<u>0.250/0.298</u>	<u>0.500/0.511</u>	<u>0.202/0.265</u>	<u>0.430/0.427</u>	<u>0.381/0.415</u>
336	TEMPO	0.209/0.309	0.503/0.356	0.292/0.332	0.515/0.467	0.309/0.345	0.441/0.430	0.379/0.408
	GPT2	0.231/0.324	0.566/0.421	0.441/0.379	<u>0.571/0.502</u>	<u>0.315/0.35</u>	0.449/0.440	0.394/0.416
	T5	<u>0.229/0.321</u>	<u>0.550/0.391</u>	<u>0.330/0.330</u>	0.572/0.504	<u>0.316/0.346</u>	<u>0.442/0.438</u>	0.416/0.427
	PatchTST	0.531/0.569	0.987/0.626	0.317/0.347	0.755/0.576	0.342/0.382	0.677/0.573	0.386/0.425
	Timesnet	0.733/0.633	1.609/0.864	0.359/0.372	0.638/0.532	0.38/0.392	0.555/0.503	<u>0.384/0.413</u>
	FEDformer	0.317/0.406	1.006/0.640	0.639/0.600	0.785/0.624	0.372/0.424	0.582/0.542	-/5.755
	ETSformer	0.862/0.707	0.940/0.621	0.487/0.444	1.154/0.682	0.409/0.428	0.728/0.585	0.446/0.451
	Informer	1.222/0.863	0.978/0.507	0.370/0.412	0.949/0.631	0.788/0.622	1.125/0.810	1.389/0.848
	DLinear	0.231/0.325	0.624/0.427	0.304/0.342	0.622/0.534	0.361/0.411	0.463/0.464	<u>0.471/0.482</u>
720	TEMPO	0.279/0.355	0.538/0.376	0.370/0.379	0.591/0.509	0.386/0.395	0.443/0.451	0.409/0.440
	GPT2	<u>0.262/0.347</u>	0.596/0.399	0.484/0.422	0.646/0.54	<u>0.394/0.397</u>	<u>0.445/0.454</u>	0.434/0.448
	T5	0.266/0.351	<u>0.578/0.404</u>	0.528/0.451	0.694/0.568	<u>0.394/0.397</u>	<u>0.443/0.458</u>	<u>0.425/0.440</u>
	PatchTST	0.475/0.532	1.152/0.706	0.375/0.388	0.739/0.57	0.421/0.421	0.540/0.521	0.425/0.448
	Timesnet	1.166/0.859	1.974/0.971	0.423/0.405	0.723/0.577	0.399/0.409	0.438/0.461	0.394/0.431
	FEDformer	0.423/0.48	0.965/0.652	0.409/0.425	0.816/0.614	0.455/0.462	0.688/0.618	0.427/0.452
	ETSformer	0.666/0.640	0.798/0.518	0.592/0.506	1.038/0.665	0.444/0.438	0.615/0.561	0.446/0.466
	Informer	0.881/0.778	1.532/0.800	1.133/0.842	0.779/0.616	1.075/0.725	0.836/0.687	1.330/0.866
	DLinear	0.259/0.352	0.623/0.42	<u>0.363/0.389</u>	0.639/0.559	0.515/0.490	0.467/0.481	0.639/0.559

Structured representation enables zero-shot transfer across domains

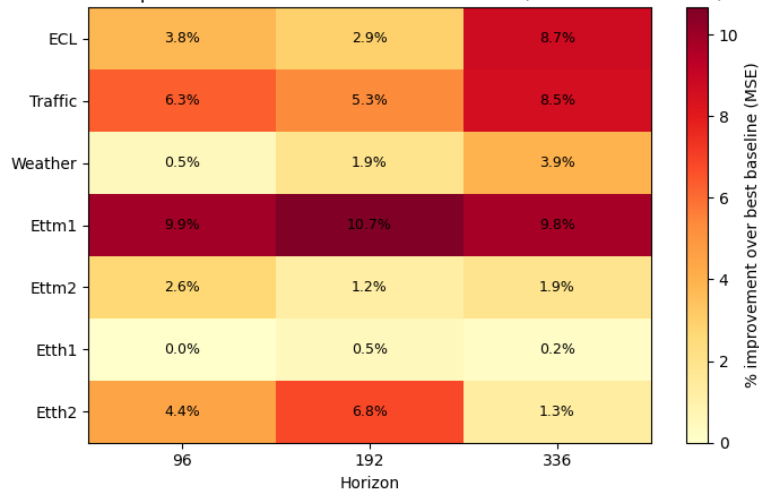
TEMPO generalizes across domains and horizons without seeing target data.



Top-1 Wins Across 21 Settings



TEMPO: % Improvement vs Best Non-TEMPO Baseline (Lower MSE is better)



The bottleneck is not scale alone — but **inductive structure**.

Experiments Results - TEMPO vs full-shot SOTAs:

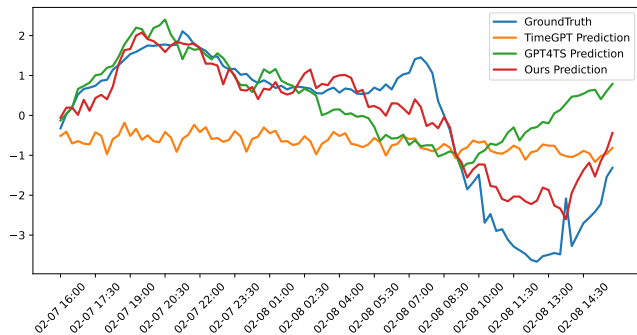
Table 7: Results of long term forecasting experiments on ETTh1 and ETTh2. The best results are marked in bold and the second optimal in underlined, respectively with MSE/MAE. Note that the TEMPO is under zero-shot setting and other models are under full-shot setting.

		TEMPO	iTransformer	TimesNet	PatchTST	Crossformer	TiDE	DLinear	SCINet	FEDformer
		MSE/MAE	MSE/MAE	MSE/MAE	MSE/MAE	MSE/MAE	MSE/MAE	MSE/MAE	MSE/MAE	MSE/MAE
ETTh1	96	0.400/0.406	0.386/0.405	<u>0.384/0.402</u>	0.414/0.419	0.423/0.448	0.479/0.464	0.386/ 0.400	0.654/0.599	0.376 /0.419
	192	<u>0.426/0.421</u>	0.441/0.436	0.436/ <u>0.429</u>	0.460/0.445	0.471/0.474	0.525/0.492	0.437/0.432	0.719/0.631	0.420 /0.448
	336	0.441/0.430	0.487/ <u>0.458</u>	0.491/0.469	0.501/0.466	0.570/0.546	0.565/0.515	0.481/0.459	0.778/0.659	<u>0.459</u> /0.465
	720	0.443/0.451	0.503/0.491	0.521/0.500	<u>0.500/0.488</u>	0.653/0.621	0.594/0.558	0.519/0.516	0.836/0.699	0.506/0.507
	Avg.	0.428/0.427	0.454/ <u>0.447</u>	0.458/0.450	0.469/0.454	0.529/0.522	0.541/0.507	0.456/0.452	0.747/0.647	<u>0.440</u> /0.460
ETTh2	96	<u>0.301/0.351</u>	0.297/0.349	0.340/0.374	0.302/ 0.348	0.745/0.584	0.400/0.440	0.333/0.387	0.707/0.621	0.358/0.397
	192	0.355/0.389	<u>0.380/0.400</u>	0.402/0.414	0.388/0.400	0.877/0.656	0.528/0.509	0.477/0.476	0.860/0.689	0.429/0.439
	336	0.379/0.408	0.428/ <u>0.432</u>	0.452/0.541	<u>0.426</u> /0.433	1.043/0.731	0.643/0.571	0.594/0.541	1.000/0.744	0.496/0.487
	720	0.409/0.440	<u>0.427/0.445</u>	0.462/0.657	0.431/0.446	1.104/0.763	0.874/0.679	0.831/0.657	1.249/0.838	0.463/0.474
	Avg.	0.361/0.398	<u>0.383/0.407</u>	0.414/0.427	0.387/0.407	0.942/0.684	0.611/0.550	0.559/0.515	0.954/0.723	0.437/0.449
		Zero-Shot	Full-Shot							

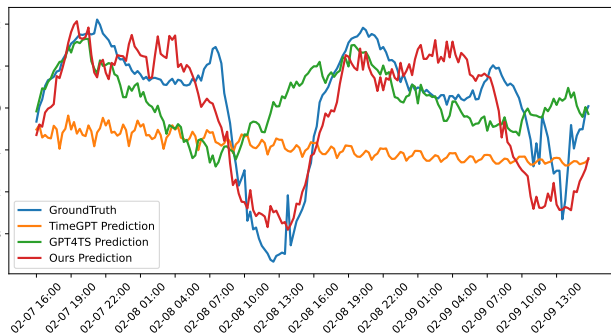
TEMPO beats specialized models trained on target data while seeing ZERO examples!

Experiments Results – Visual

- **Pattern Capture:** TEMPO identifies trends/seasonality others miss
- **Uncertainty Handling:** Graceful degradation in difficult regions
- **No Overfitting:** Smooth, realistic predictions without training

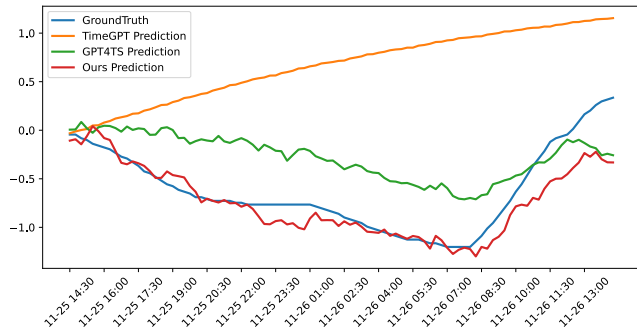


(a) prediction length $O = 96$

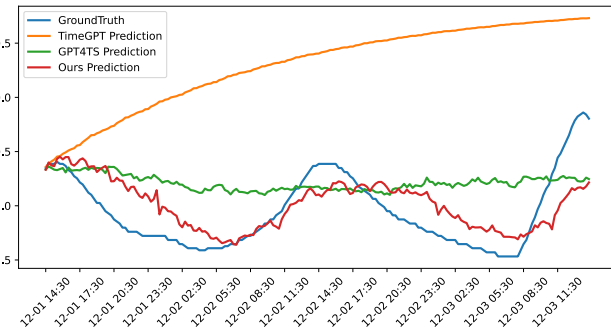


(b) prediction length $O = 192$

ETTM1



(a) prediction length $O = 96$

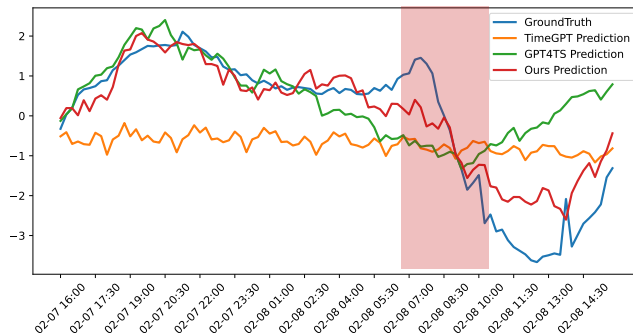


(b) prediction length $O = 192$

ETTM2

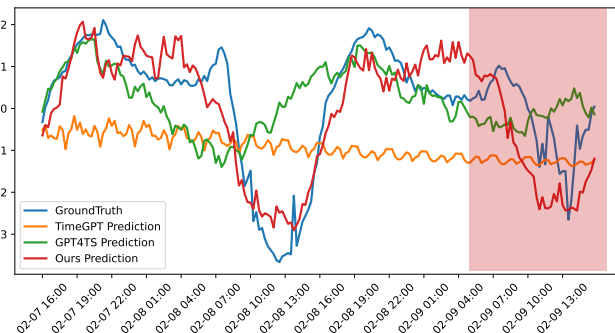
Experiments Results – Visual

- **Pattern Capture:** TEMPO identifies trends/seasonality that others miss
- **Uncertainty Handling:** Graceful degradation in difficult regions
- **No Overfitting:** Smooth, realistic predictions without training

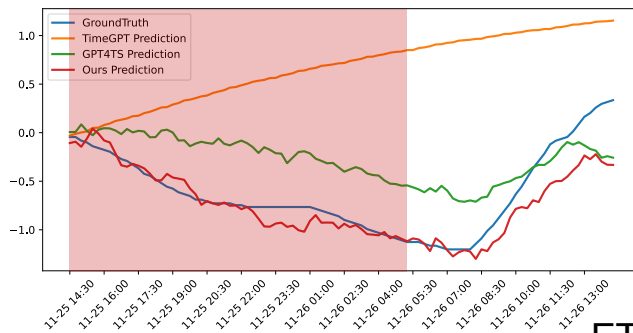


(a) prediction length $O = 96$

ETTm1

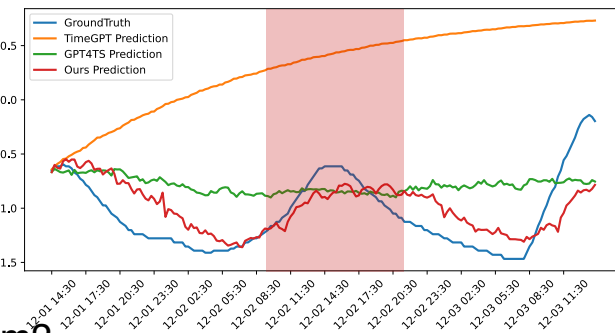


(b) prediction length $O = 192$



(a) prediction length $O = 96$

ETTm2



(b) prediction length $O = 192$

Demonstrations on New Multimodal TETS Dataset

- TEMPO's capabilities are further showcased on the newly introduced TETS (Time Series Enriched with Text) dataset
- TETS combines time series data with corresponding textual information, providing a rich multimodal forecasting challenge

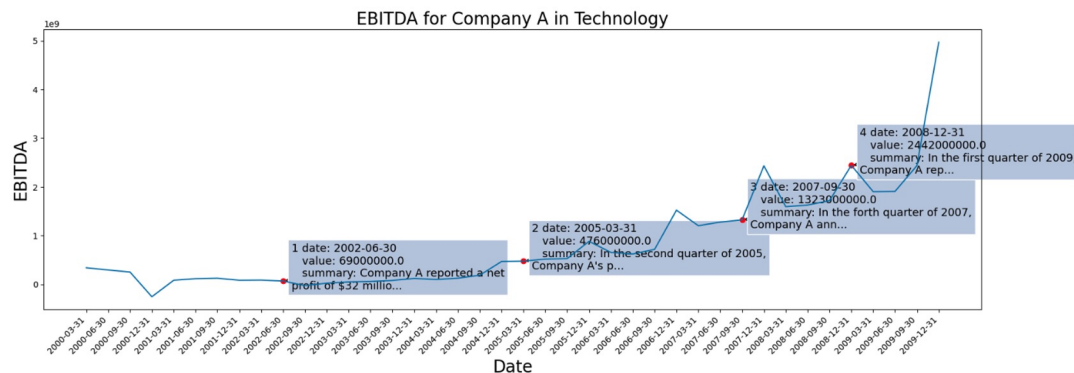


Figure 15: EBITDA for Company A with contextual information

Experiments Results - Multi-modal Input

- TEMPO demonstrates remarkable performance on datasets that include textual inputs alongside time series.
- The model's ability to integrate and utilize textual inputs highlights its versatility and potential for multi-modal time series forecasting

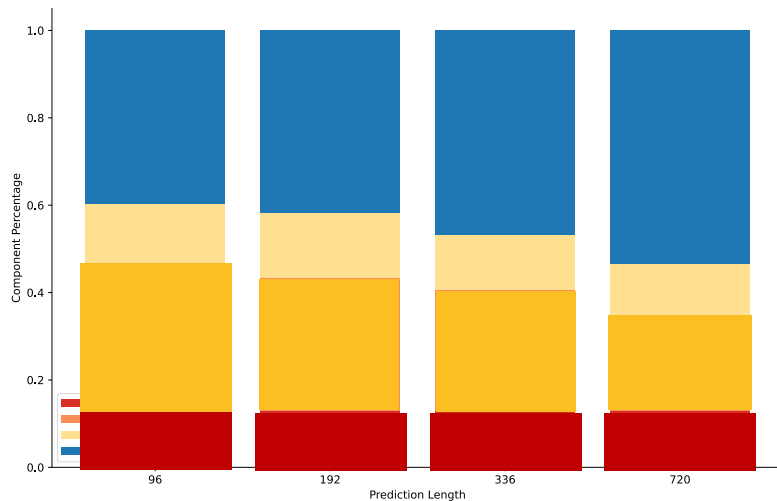
Table 2: SMAPE results of EBITDA from TETS and GDELT. The result of EBITDA includes outliers removed where SMAPE exceeds 0.8/0.9. The best results are marked in bold and the second optimal in underlined respectively with 0.8 & 0.9. (*Sectors*: CC: Consumer Cyclical; CD: Consumer Defensive; Ind: Industrials; RE: Real Estate; *Events*: 11: Disapprove; 17: Coerce; 19:Fight.)

EBITDA Dataset									
Sectors	TEMPO	LLaMA	GPT2	Bert	T5	Informer	PatchTST	Reformer	DLinear
CC	32.27/33.48	33.13/34.31	33.77/35.37	33.42/35.33	<u>32.65/33.83</u>	41.12/43.17	41.44/43.18	37.23/39.09	33.53/35.65
CD	25.9/26.25	<u>26.34/26.62</u>	26.86/27.15	27.34/28.3	<u>26.44/26.79</u>	35.65/36.08	31.6/31.98	29.93/30.36	27.01/28.04
Ind	26.7/27.42	<u>27.17/27.98</u>	27.9/28.63	27.89/28.95	27.3/28.12	34.83/35.87	33.84/34.87	30.23/31.28	27.59/28.84
RE	29.46/30.11	<u>29.63/30.48</u>	30.62/31.21	30.62/31.66	30.1/30.64	36.4/37.22	37.63/38.31	31.23/31.69	29.95/30.92
GDELT Dataset									
11	38.77	40.23	39.03	<u>38.89</u>	39.04	42.00	40.45	46.72	40.14
17	41.02	42.50	41.20	<u>41.10</u>	41.24	44.44	42.72	48.08	42.45
19	44.03	45.49	44.17	<u>44.09</u>	44.29	47.45	45.49	48.30	45.40

TEMPO Actually Uses the Injected Temporal Structure

Seasonal priors dominate the prediction across all horizons.

Trend anchors long-term global dynamics, giving a reliable backbone for far horizons.



SHAP value of ETTm1

Decomposition creates a structured inductive bias.

Practice

 DC-research / TEMPO

Q Type [\[?\]](#) to search

[Code](#) [Issues](#) [4](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

 **TEMPO** [Public](#)

[Edit Pins](#) [Unwatch](#) [2](#) [Fork](#) [14](#) [Starred](#) [100](#)

[huggingface.co/Melady/TEMPO](#) [Relaunch to update](#)

 **Hugging Face**

[Models](#) [Datasets](#) [Spaces](#) [Posts](#) [Docs](#) [Enterprise](#) [Pricing](#) [Log In](#) [Sign Up](#)

Melady / TEMPO [like](#) 8 [Follow](#) [USCMeladyLab](#) 9

[Time Series Forecasting](#) [Transformers](#) [ETDataset/ett](#) [English](#) [feature-extraction](#) [Time-series](#) [foundation-model](#)

Salesforce / GIFT-Eval [like](#) 96 [Running](#)

GIFT-Eval Time Series Forecasting

We introduce the General Time Series Forecasting Model Evaluation, GIFT-Eval, a pioneering benchmark aimed at promoting evaluation across domains spanning seven domains, 10 frequencies, multivariate inputs, and prediction lengths ranging from short to long-term forecasts.

[Overall](#) [By Domain](#) [By Frequency](#) [By Term Length](#) [By Variate Type](#)

Search

Separate multiple queries with ';'.

Select Columns to Display:

☒ MASE ☒ CRPS ☒ Rank

Model types

☒ pretra

Show Rows with

T	model	MASE	CRPS
	TiRex	0.650	0.421
	TEMPO_ensemble	0.773	0.434

[pypi.org/project/timeagi/](#)

timeagi 0.0.2

```
pip install timeagi
```

Downloads last month
930

Time Series Foundation Model - TEMPO: Prompt-based Generative Pre-trained Transformer for Time Series Forecasting

Inference Framework

Time Series Foundation Model

This model is not

The model cannot support time

Navigation

[Project description](#)

[Release history](#)

[Download files](#)

Project description

Time Series Foundation Model - TEMPO: Prompt-based Generative Pre-trained Transformer for Time Series Forecasting

[arXiv](#) [2310.04948](#) [Hugging Face](#) [Models](#) [License](#) [Apache-2.0](#)

Verified details

These details have been [verified by PyPI](#)

Project links

[Homepage](#)

[Bug Reports](#)

[Source](#)



TEMPO solves Representation Alignment, but Not Full Temporal Reliability.

Proving the Power of Semantic Priors



Scientific Contributions

- ✓ LLMs understand time series representation
- ✓ Decomposition provides semantic structure
- ✓ Zero-shot forecasting is practical



Generalization Frontier

- ✗ Task Versatility: Forecasting only
- ✗ Robustness: Fails on missing data
- ✗ Uncertainty: No probabilistic outputs

*Representation is necessary — **but not sufficient.***

Generation: One Model, Many Tasks



How to unify forecasting, imputation, and anomaly detection — case study: TimeDiT

01

Diffusion, Made Simple

A nine-step primer on denoising diffusion for continuous signals.

02

Tasks as Masking

Every temporal task is a choice of what is observed and what is hidden.

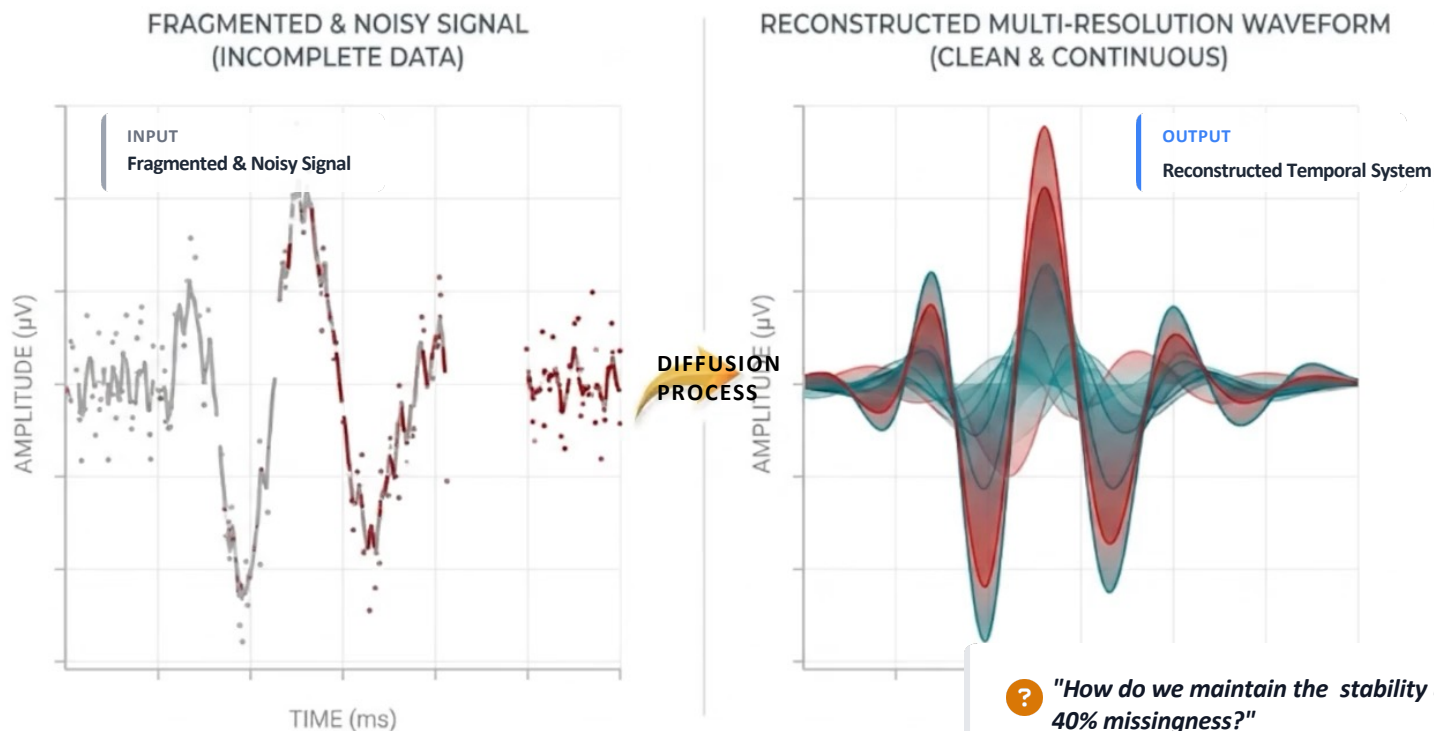
03

Uncertainty & Robustness

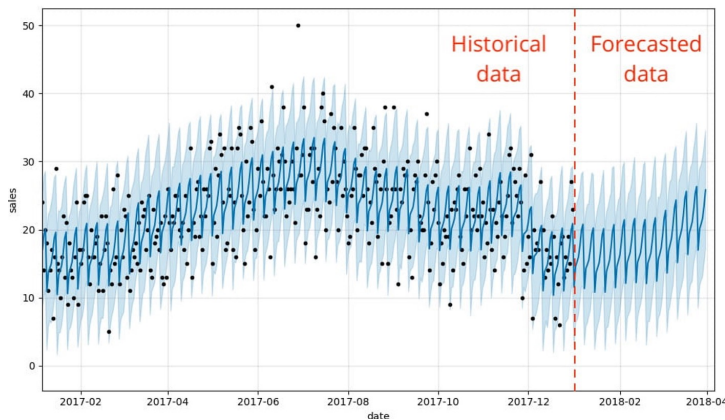
Sample-based uncertainty: knowing when the model does not know.

Pillar II: Scaling to Observational Chaos.

How do we model missingness, uncertainty, and multiple tasks together?



The Reality Gap: Benchmarks Underestimate Complexity



Data Gap

Benchmark:

- Clean,
- Regular sampling,
- Fixed resolution

Reality:

- Irregular, missing,
- Multi-resolution,
- Non-stationary

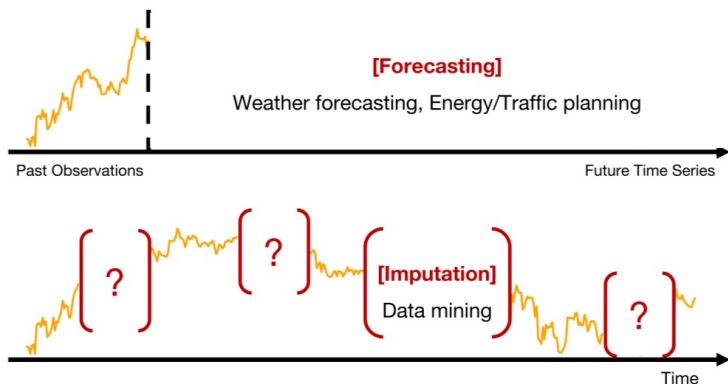
Task Gap

Benchmark:

- Single-task forecasting

Reality:

- Multi-objective



The gap is not about the architecture capacity—
it is about **assumptions about reality.**

Why Generative Modeling for Time Series?

The real question is not one future, but a distribution of futures.



Deterministic Forecasting Assumes:

- A single predicted future trajectory
- Clean and complete observations
- A fixed downstream objective



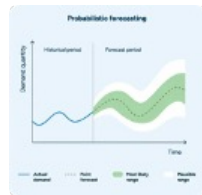
Old Question

"What will the temperature be?"



But Real-World Systems Exhibit:

- ✓ Multiple plausible futures
- ✓ Observational uncertainty & missing data
- ✓ Task diversity (forecasting, imputation, generation)



New Question

"What is the range of outcomes?"

What Is a Diffusion Model?

The Generative Paradigm

A diffusion model learns to generate data by reversing a gradual noising process — refining samples through iterative loops rather than a single step.

STEP 01: FORWARD PROCESS

Corrupt Data into Noise

Gradually adds small Gaussian noise to the clean sequence x_0 step-by-step:

$$x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_T \approx \text{noise}$$

Systematic Destruction Chain

STEP 02: REVERSE PROCESS

Learn to Denoise

A neural network learns to estimate and remove the noise, reconstructing the clean sequence step-by-step:

$$\text{noise} \rightarrow x_{\{T-1\}} \rightarrow \dots \rightarrow x_1 \rightarrow x_0$$

Iterative Reconstruction

Forward Diffusion: Corrupt Data into Noise

Progressive Noising Sequence

$t = 0$ (Clean Signal)



$t = 100$ (Slightly Noisy)



$t = T$ (Pure Noise)



The Noising Goal: The forward process is fixed and deterministic, generating input/noise pairs across timesteps for the reverse net to learn from.

Mathematical Formulation

The forward process maps any step t directly from the clean state x_0 via closed-form Gaussian multipliers:

$$x_t = \sqrt{\alpha_t} x_0 + \sqrt{1 - \alpha_t} \epsilon$$

α_t (Alpha Bar): The scheduled retention of the clean signal; decreases monotonically as t to T .

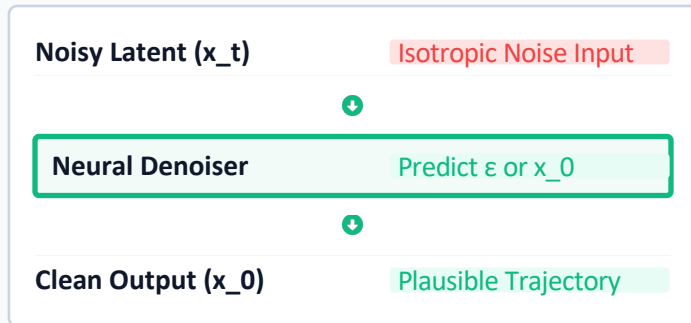
ϵ (Epsilon): Standard Gaussian noise injected into the sequence.

Step Scaling: Near $t = T$ the original signal is almost fully obliterated into random noise.

Training Context: Exposure to noisy variants at varying t teaches a generalized score-matching prior over the data manifold.

Reverse Diffusion: Learn to Denoise

The Denoising Engine



Generation Method: Generation is a progressive denoising chain from random Gaussians, guided by the learned prior.

Uncertainty Denoising Mechanics

Learning the Transition $p_{\theta}(x_{t-1} | x_t)$

The network learns a reverse Markov transition to estimate and remove signal pollution.

Prediction Target Choices

Models predict the injected noise ϵ , the clean trajectory x_0 , or velocity vectors.

Trajectory Reconstruction

The reconstructed manifold is a continuous multi-step physical trajectory, not static pixels.

Structural Takeaway: Resolving tiny fractional denoising steps lets the model capture complex probabilistic boundaries.

Conditioning: Generating Data Consistent with Evidence

The Conditional Formulation

Rather than sampling blindly, we guide trajectory generation with explicit observed evidence:

$$\text{sample } x \sim p(x \mid \text{evidence})$$

Evidence Scope: Observed histories, future indicators, channel covariates, spatial metadata, or physical constraints.

For Time Series: The conditioning signal is a partially observed temporal mask selecting target steps.

Main Takeaway: Conditioning turns generic generation into precise posterior inference under partial evidence.

Multi-Task Conditioning Scenarios

→ 1. Past Window → Future Horizon

Condition on past values to predict forward trajectories (Forecasting).

⇄ 2. Observed Marks → Missing Gaps

Condition on scattered observations to fill missing holes (Imputation).

“ 3. Semantic Text → Synthetic Series

Condition on text alerts or metadata to sample matching paths (Synthesis).

Sensing Context: Conditioning expresses any task as targeted inference, not just autoregressive extrapolation.

Masks as Temporal Conditions

The Masking Matrix

$m_t = 1$ Observed Element

Known conditional evidence — left untouched during reverse denoising.

$m_t = 0$ Target Element (Unknown)

Unobserved points the reverse diffusion model must generate.

Structural Unified View: Changing the conditional mask changes the task itself — no model-head changes needed.

Unifying Tasks via Masks

Forecasting Mask

Observed Past [1 1 1 1 1] Target Future [0 0 0 0]

Imputation Mask

Scattered Observed [1 X 1 1 X 1] Missing Gaps [X 0 X X 0 X]

Pure Synthesis Mask

No Context [None] Full Trajectory [0 0 0 0 0 0]

System Relevance: Pretrained generative models adapt to any mask on the fly, enabling multi-task zero-shot capability.

Forecasting as Conditional Generation

The Forecasting Probabilistic Condition Mapping

Observed Condition: $x_{\{1:t\}}$

Generated Target: $x_{\{t+1:t+H\}} \sim p(x_{\{t+1:t+H\}} | x_{\{1:t\}})$

Benefit 01

Plural Futures

Samples multiple plausible futures, preserving statistical variance over long horizons.

Uncertainty Preserved

Benefit 02

No Mode Collapse

Avoids regression's point-averaging trap, where multi-modal predictions collapse into flat averages.

Multi-modal Densities

Benefit 03

Scenario Simulation

Enables risk-sensitive decisions and stress-testing via multi-path scenarios.

Actionable Risk Analytics

Imputation and Anomaly Detection as Reconstruction

Temporal Task A

Probabilistic Imputation

Given scattered observations, sample the missing entries conditioned on available evidence: From scattered observations, sample the missing entries given available evidence:

$$p(\mathbf{x}_{\text{missing}} \mid \mathbf{x}_{\text{observed}})$$

Conditional diffusion rebuilds coherent values that preserve cross-series structure, not flat averages.

Non-causal Contextual Restoration

Temporal Task B

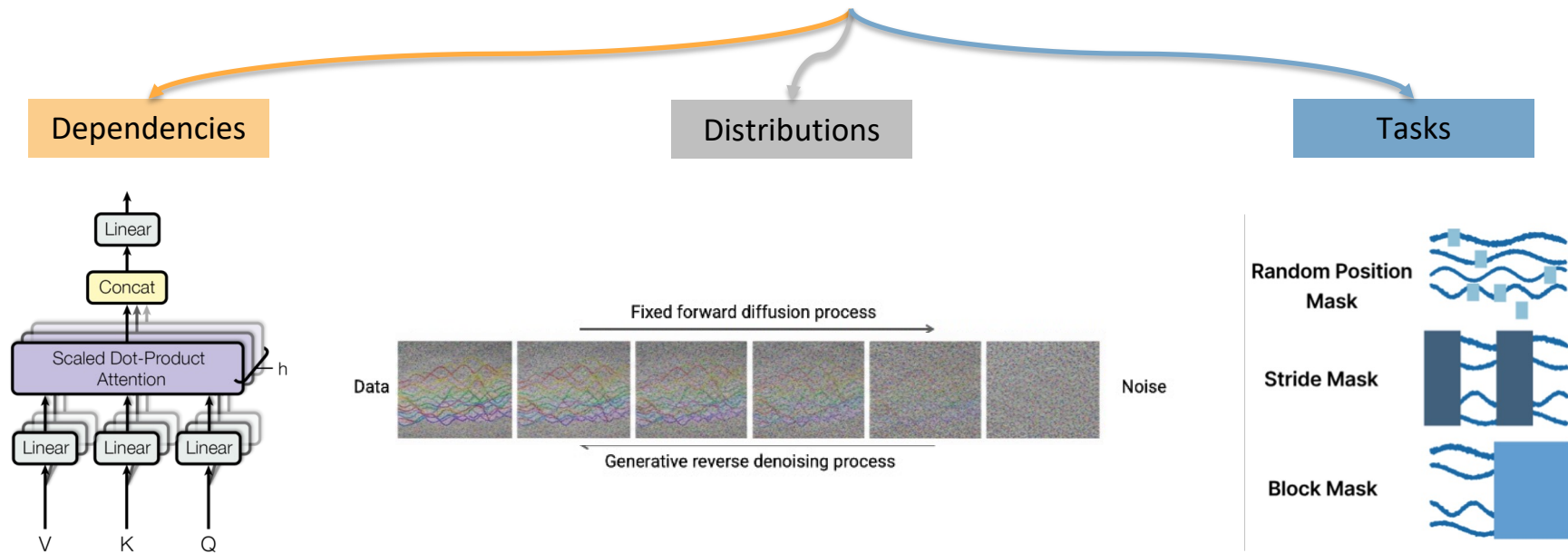
Anomaly Detection

Compare observed curves against what the generative model reconstructs as plausible:

Reconstruction Loss:	$\ \mathbf{x}_{\text{obs}} - \mathbf{x}_{\text{recon}} \ ^2$
Probability Likelihood:	$\log p(\mathbf{x}_{\text{obs}} \mid \text{context})$

Generative Paradigm: points far from the model's posterior reconstruction are flagged as anomalies.

TimeDiT Unifies Dependency, Distribution, and Task Modeling



learns temporal structure

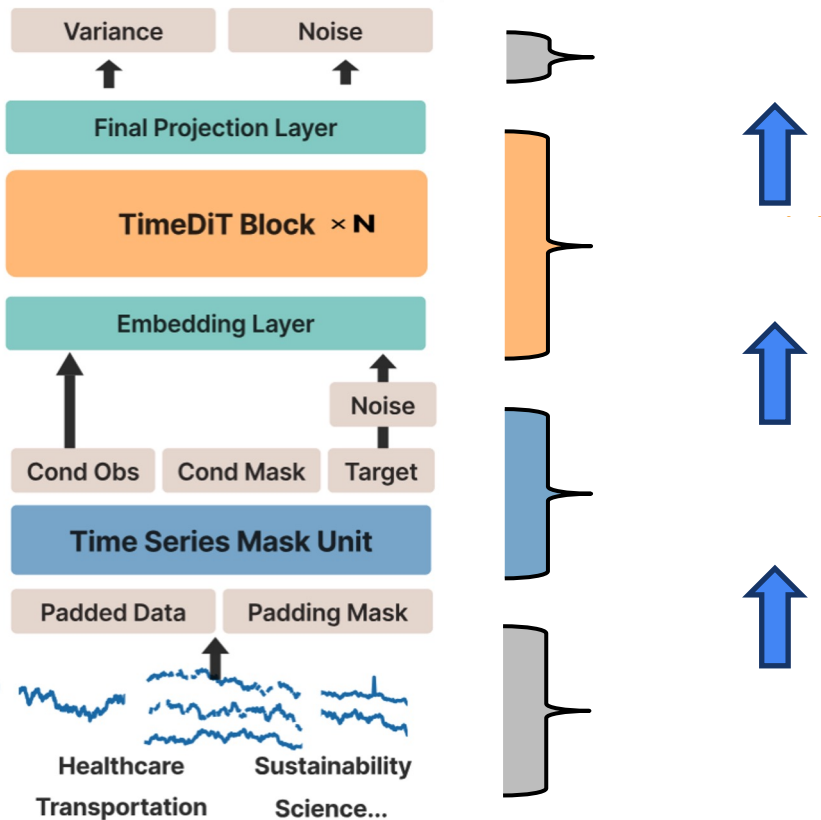
captures stochasticity

defines conditional structure

Generalization emerges when **dependencies**, **distributions**, and **tasks** are modeled in a unified framework.

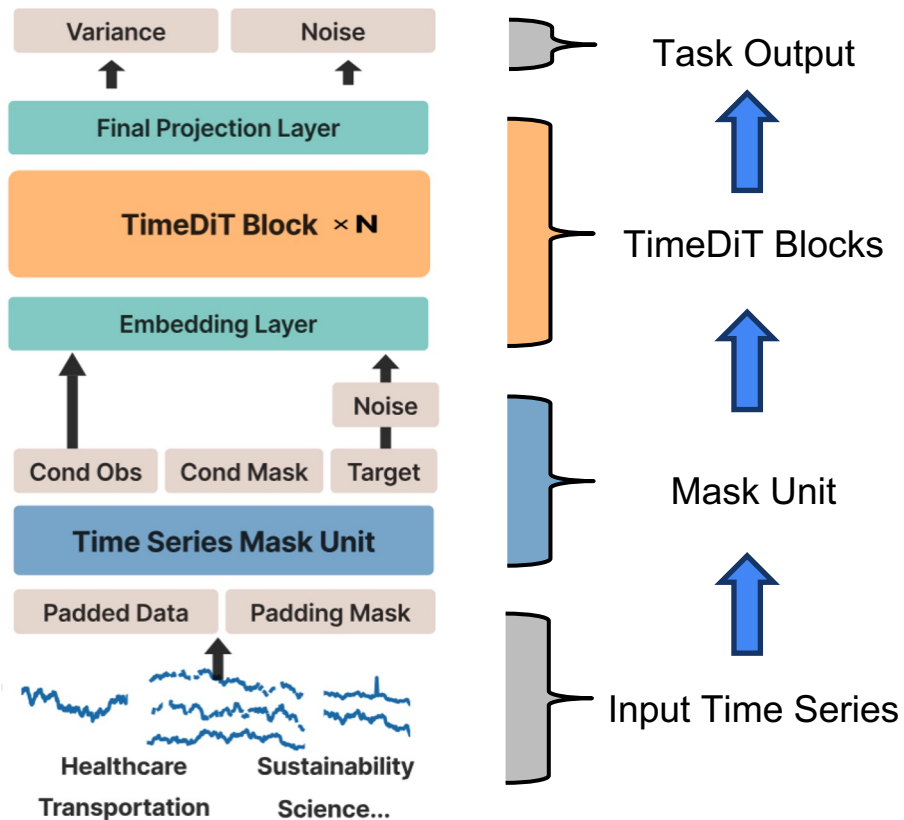
TimeDiT Architecture

Fig.: Model Architecture of TimeDiT



Model Architecture

Fig.: Model Architecture of TimeDiT



Model Architecture - Mask

Fig.: Model Architecture of TimeDiT

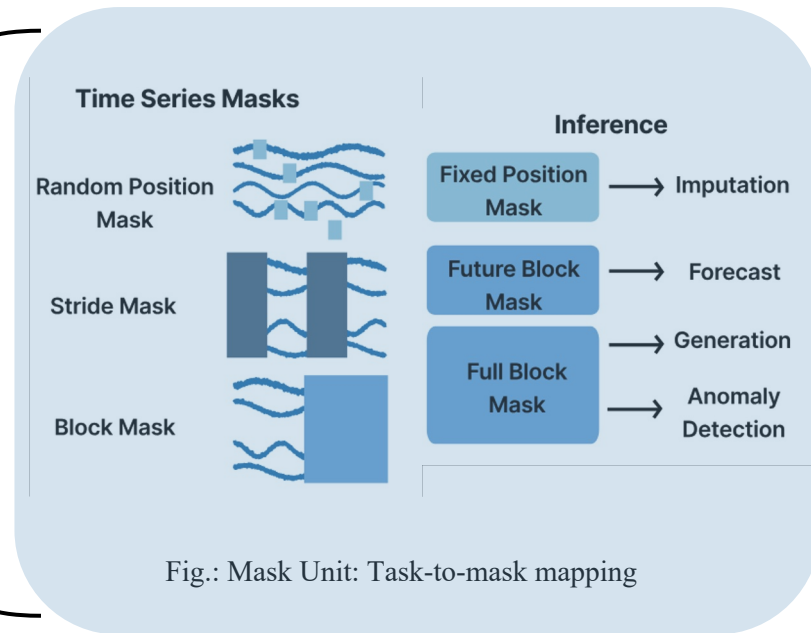
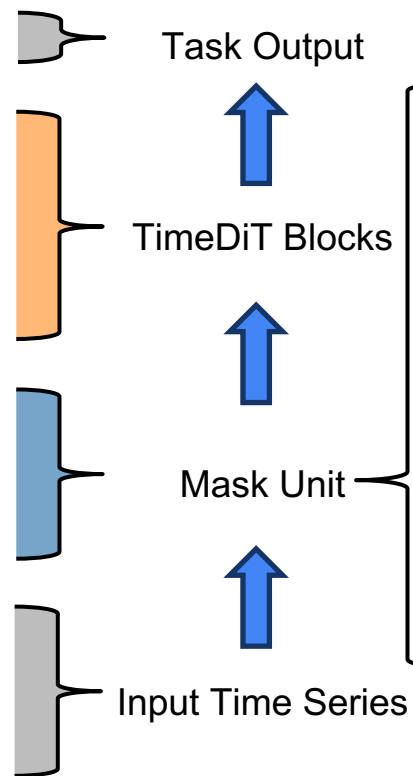
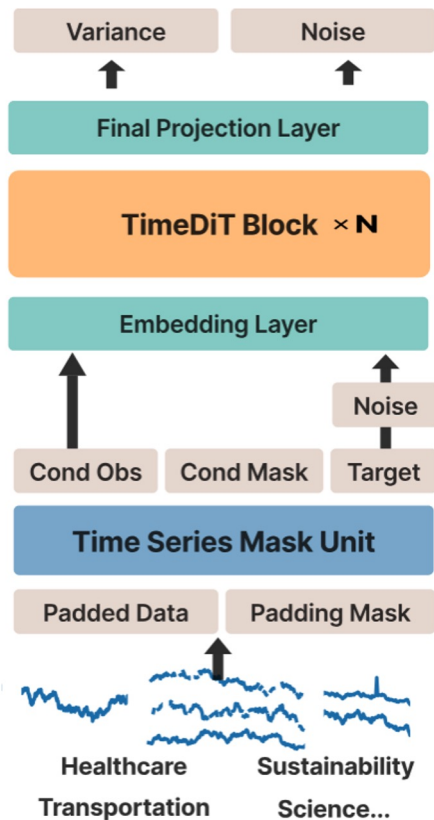
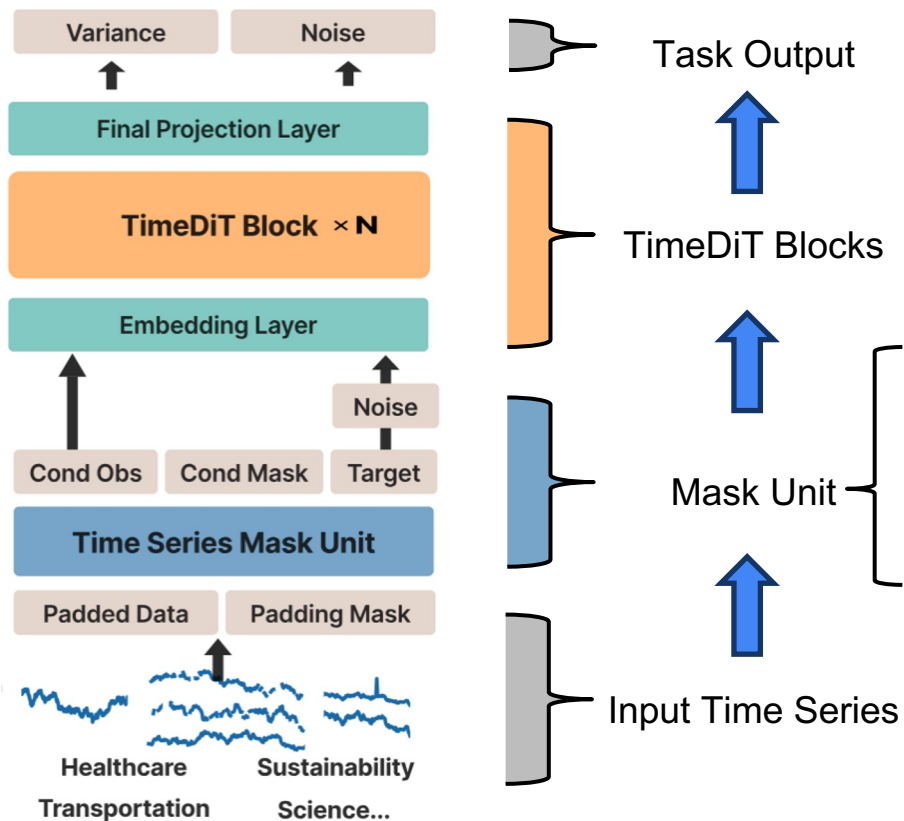


Fig.: Mask Unit: Task-to-mask mapping

Model Architecture - Mask

Fig.: Model Architecture of TimeDiT



Forecasting	mask=[1,0]	Future
Imputation	mask=[1,0,1]	Fill gaps
Generation	mask=[0,0]	New samples
Custom	mask=[?]	Your task

Model Architecture – TimeDiT Block

Fig.: Model Architecture of TimeDiT

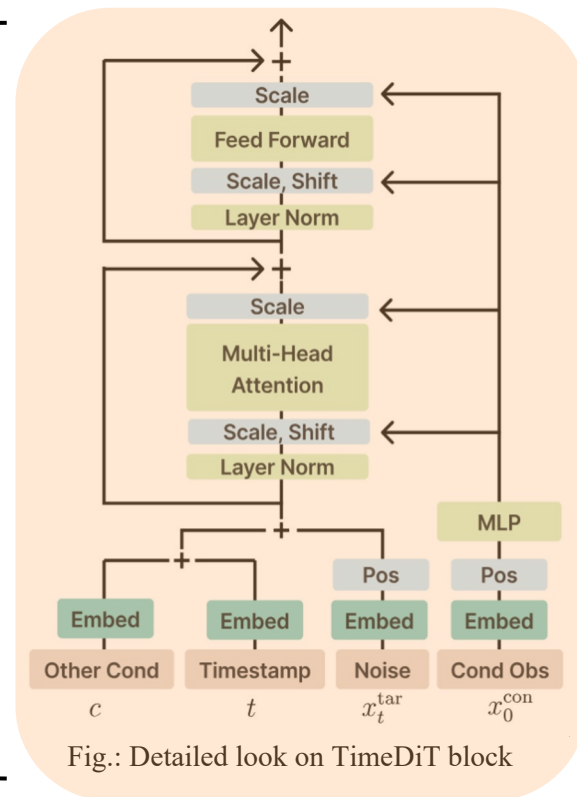
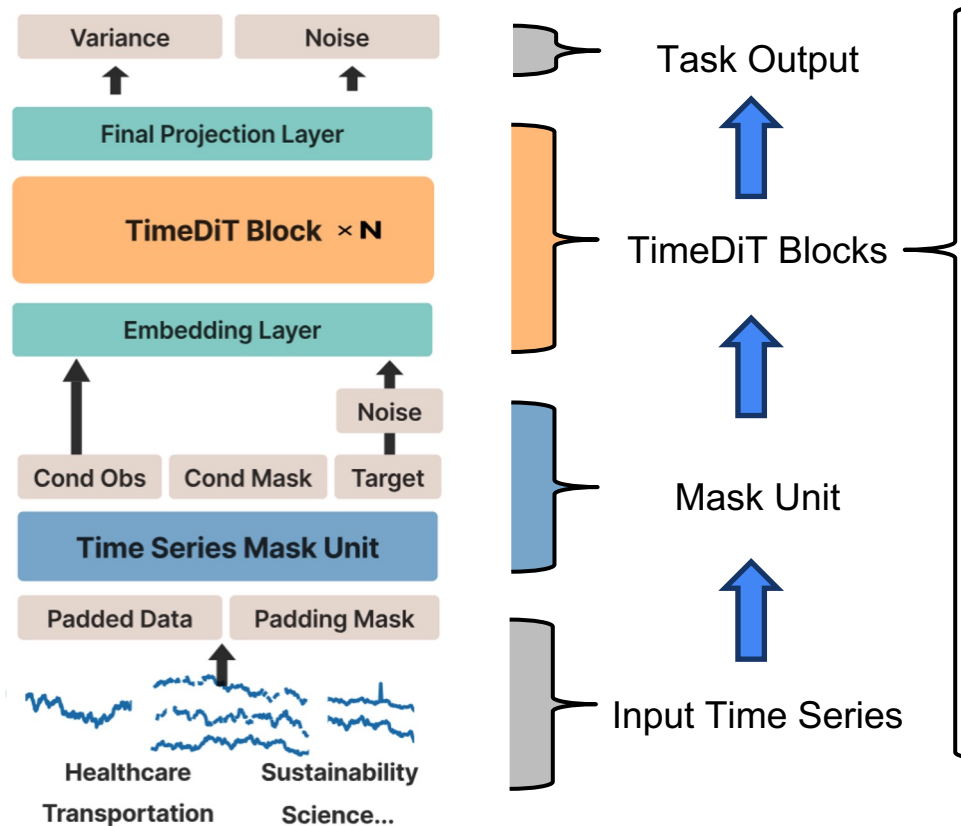


Fig.: Detailed look on TimeDiT block

Experiments

Datasets from different domain:

- Energy
- Nature
- Finance
- Cloud
- Climate
- Space

Datasets with different resolution:

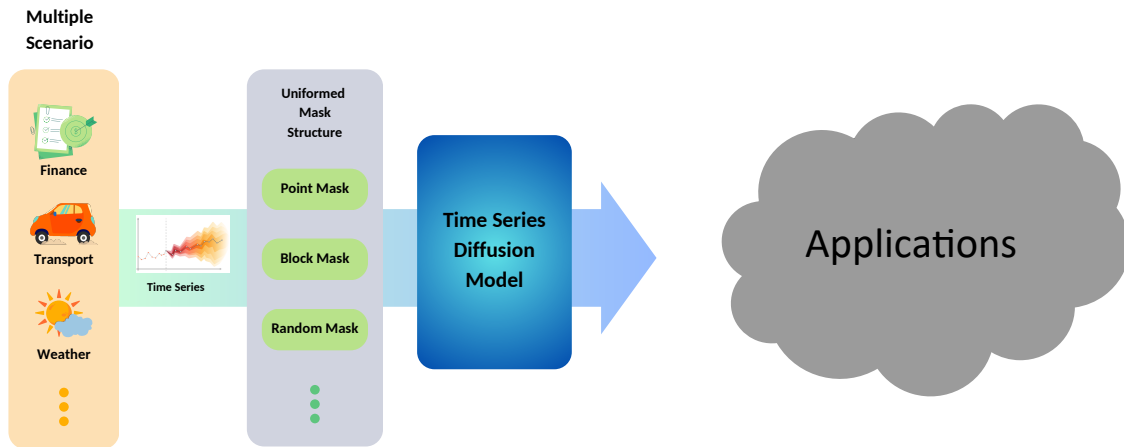
- Second
- Minute
- Hour
- Day

Dataset	Domain	Length	Dimension	Frequency
ETTh	Energy	17420	7	1 hour
ETTm	Energy	69680	7	15 min
Weather	Nature	52696	21	10 min
Electricity	Energy	26304	321	1 hour
Air Quality	Nature	9357	13	1 hour
Sine	Synthetic	10000	5	N/A
Stock	Finance	3685	6	1 day
Energy	Energy	19745	28	10 min
MSL	Space	132046	55	1 min
PSM	Cloud	220322	25	1 min
SMAP	Space	562800	25	1 min
SMD	Cloud	1416825	38	1 min
SWaT	Energy	944920	51	1 second
Requests Minute	Cloud	64800	10	1 min
Function Delay Minute	Cloud	64800	10	1 min
Platform Delay Minute	Cloud	64800	10	1 min
Memory Usage Minute	Cloud	64800	10	1 min
CPU Limit Minute	Cloud	64800	10	1 min
Memory Limit Minute	Cloud	64800	10	1 min
Instances Minute	Cloud	64800	10	1 min
Weather_2	Climate	3001	695	1 day
PEMS_SF	Traffic	4320	852	1 hour
PhysioNet(b)	Health Care	-	7	Irregular
PhysioNet(b)	Health Care	-	7	Irregular
PhysioNet(c)	Health Care	-	7	Irregular
MIMIC-III	Health Care	-	19	1 day
NASDAQ	Finance	2516	20	Multiresolution

Scaling Across Domains and Resolutions

Domain Diversity: Energy · Climate · Finance · Cloud · Space · Nature

Temporal Resolution Diversity: Second → Minute → Hour → Day



A single generative model across heterogeneous domains and time scales.

Experiments

A comparable analysis of representative general purposes time series models

Model	Parameter Size	Model Architecture	Channel Setting	Task Type	Pretrain Dataset		Data Size
Lag-LLama	-	Transformer	Univariate	Forecasting	Monash	[Godaheva et al., 2021b]	300 Million Time Points
Moriai	S: 14M B: 91M L: 311M	Transformer	Univariate	Forecasting	LOTSa	[Woo et al., 2024b]	27 Billion Time Points
TimeDiT	S: 33M B: 130M L: 460M XL: 680M	Transformer + Diffusion	Multivariate	Forecasting, Imputation, Anomaly Detection, Data Generation	Chronos	[Ansari et al., 2024]	About 5 Billion Time Points

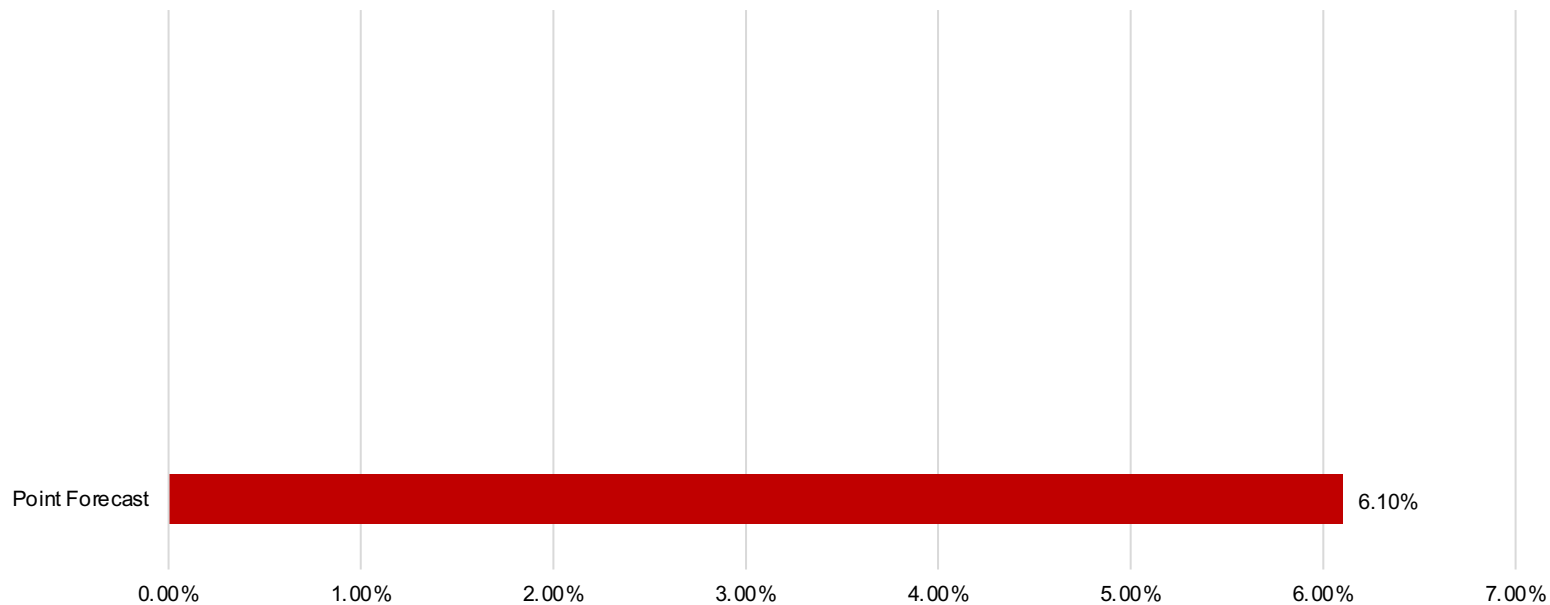
Forecasting on Zero-shot Setting: Deterministic Forecasting

Table 1: Full results of long sequence forecasting experiments evaluated on diverse datasets, where **bold** indicates the best performance and underlined indicates the second best performance. Baseline results are sourced from Woo et al. [2024a].

		Zero-shot				Full-shot							
		TimeDiT	MOIRAI _{Small}	MOIRAI _{Base}	MOIRAI _{Large}	iTransformer	TimesNet	PatchTST	Crossformer	TiDE	DLinear	SCINet	FEDformer
ETTh1 (MSE)	96	0.325	<u>0.375</u>	0.384	0.380	0.386	0.384	0.414	0.423	0.479	0.386	0.654	<u>0.376</u>
	192	0.347	<u>0.399</u>	0.425	0.440	0.441	0.436	0.460	0.471	0.525	0.437	0.719	0.420
	336	0.347	<u>0.412</u>	0.456	0.514	0.487	0.491	0.501	0.570	0.565	0.481	0.778	0.459
	720	0.404	<u>0.413</u>	0.470	0.705	0.503	0.521	0.500	0.653	0.594	0.519	0.836	0.506
	Avg.	0.356	<u>0.400</u>	0.434	0.510	0.454	0.458	0.469	0.529	0.541	0.456	0.747	0.440
ETTh2 (MSE)	96	0.257	0.281	<u>0.277</u>	0.287	0.297	0.340	0.302	0.745	0.400	0.333	0.707	0.358
	192	0.316	<u>0.340</u>	<u>0.340</u>	0.347	0.380	0.402	0.388	0.877	0.528	0.477	0.860	0.429
	336	0.341	<u>0.362</u>	0.371	0.377	0.428	0.452	0.426	1.043	0.643	0.594	1.000	0.496
	720	0.447	0.380	<u>0.394</u>	0.404	0.427	0.462	0.431	1.104	0.874	0.831	1.249	0.463
	Avg.	0.340	<u>0.341</u>	0.345	0.354	0.383	0.414	0.387	0.942	0.611	0.559	0.954	0.437
ETTh1 (MAE)	96	0.386	0.402	0.402	0.398	0.405	0.402	0.419	0.448	0.464	0.400	0.599	0.419
	192	0.390	0.419	0.429	0.434	0.436	0.429	0.445	0.474	0.492	0.432	0.631	0.448
	336	0.396	<u>0.429</u>	0.450	0.474	0.458	0.469	0.466	0.546	0.515	0.459	0.659	0.465
	720	0.440	<u>0.444</u>	0.473	0.568	0.491	0.500	0.488	0.621	0.558	0.516	0.699	0.507
	Avg.	0.403	<u>0.424</u>	0.438	0.469	0.448	0.450	0.455	0.522	0.507	0.452	0.647	0.46
ETTh2 (MAE)	96	0.331	0.334	0.327	0.325	0.349	0.374	0.348	0.584	0.440	0.387	0.621	0.397
	192	0.380	<u>0.373</u>	0.374	0.367	0.400	0.414	0.400	0.656	0.509	0.476	0.689	0.439
	336	0.418	<u>0.393</u>	0.401	0.393	0.432	0.541	0.433	0.731	0.571	0.541	0.744	0.487
	720	0.441	0.416	0.426	<u>0.421</u>	0.445	0.657	0.446	0.763	0.679	0.657	0.838	0.474
	Avg.	0.393	<u>0.379</u>	0.382	0.376	0.407	0.497	0.407	0.684	0.550	0.515	0.723	0.449

One Generative Model can Outperform Specialized Systems

Improve with the best specialized model



Cross-task knowledge transfer emerges from unified pretraining

Forecasting on Zero-shot Setting: Probabilistic Forecasting

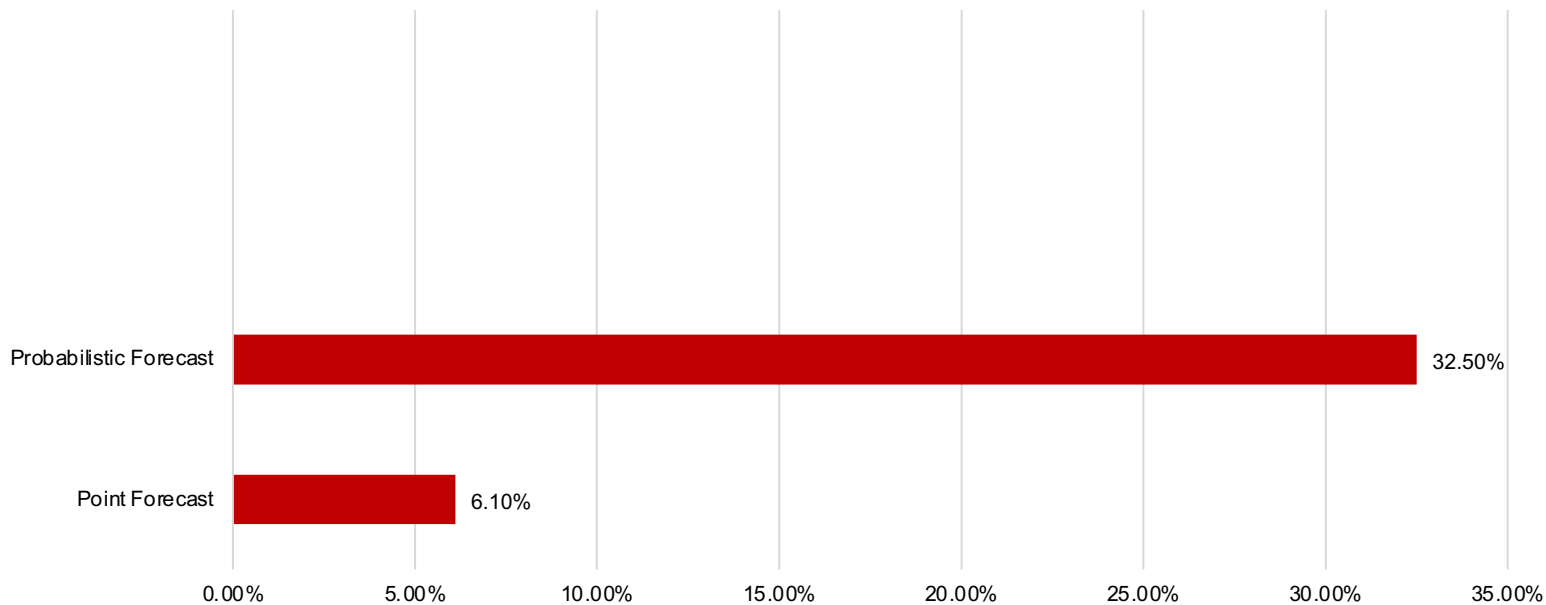
- Zero-shot TimeDiT (ZS) achieves superior performance compared to other pretrained models
- Fine-tuned TimeDiT (FS) surpasses specialized probabilistic forecasting models across multiple datasets.

Table 2: Forecasting results on CRPS_sum for both zero-shot and full-shot settings.

Setting	Dataset	TimeDiT (ZS)	TEMPO	MOIRAI _{Small}	MOIRAI _{Base}	MOIRAI _{Large}	LagLLaMA	TimeMixer	TimeLLM	Timer
Zero Shot	Solar	0.424	<u>0.581</u>	0.884	0.948	1.042	0.690	0.999	0.997	0.101
	Electricity	0.030	0.081	0.079	0.072	<u>0.039</u>	0.065	0.302	0.303	0.301
	Traffic	0.351	<u>0.147</u>	0.215	0.191	0.111	0.275	0.403	0.368	0.384
	Taxi	0.392	<u>0.400</u>	0.463	0.428	0.597	0.620	0.785	0.782	0.788
	Exchange	0.019	<u>0.030</u>	0.007	0.012	<u>0.011</u>	0.024	0.079	0.076	0.072
Setting	Dataset	TimeDiT (FS)	DLinear	PatchTST	Latent ODE	GPT4TS	TransMAF	TimeGrad	CSDI	Diffusion-TS
Full Shot	Solar	0.278	0.432	0.457	0.445	0.467	0.301	0.287	0.298	<u>0.286</u>
	Electricity	0.005	0.033	0.037	0.140	0.033	0.021	0.021	<u>0.017</u>	0.019
	Traffic	0.019	0.070	0.405	0.095	0.069	0.056	0.044	<u>0.020</u>	0.097
	Taxi	<u>0.123</u>	0.177	0.190	0.181	0.187	0.179	0.114	0.123	0.303
	Exchange	0.005	0.011	0.026	0.013	0.013	<u>0.005</u>	0.006	0.007	0.009

One Generative Model can Outperform Specialized Systems

Improve with the best specialized model



Cross-task knowledge transfer emerges from unified pretraining

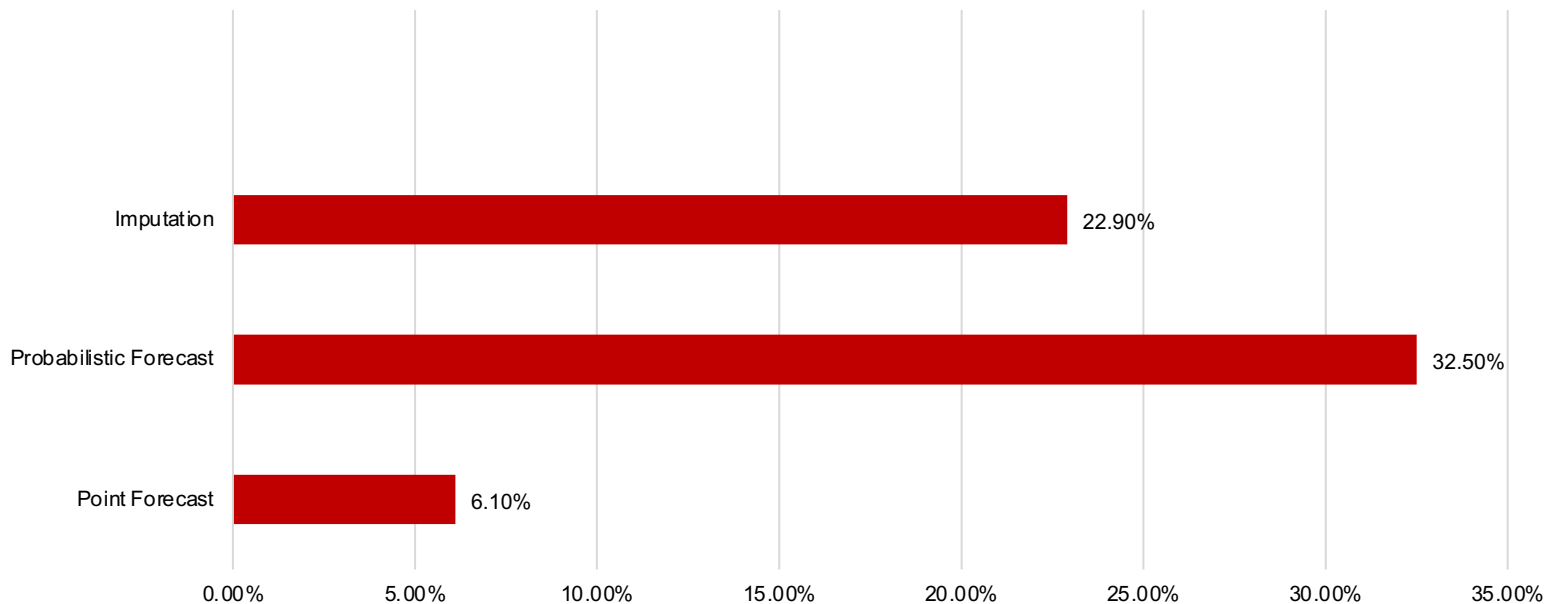
Experiments: Imputation Task

Datasets	ETTh1		ETTh2		ETTm1		ETTm2		Weather		Electricity		1st Pl Count
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	
DLinear	0.201	0.306	0.142	0.259	0.093	0.206	0.096	0.208	0.052	0.110	0.132	0.260	0
LightTS	0.284	0.373	0.119	0.250	0.104	0.218	0.046	0.151	0.055	0.117	0.131	0.262	0
ETSformer	0.202	0.329	0.367	0.436	0.120	0.253	0.208	0.327	0.076	0.171	0.214	0.339	0
FEDformer	0.117	0.246	0.163	0.279	0.062	0.177	0.101	0.215	0.099	0.203	0.130	0.259	0
Autoformer	0.103	0.214	0.055	0.156	0.051	0.150	0.029	0.105	0.031	0.057	0.101	0.225	0
PatchTST	0.115	0.224	0.065	0.163	0.047	0.140	0.029	0.102	0.034	0.055	<u>0.072</u>	<u>0.183</u>	0
TimesNet	0.078	0.187	0.049	0.146	<u>0.027</u>	0.107	<u>0.022</u>	0.088	0.030	<u>0.054</u>	0.092	0.210	1
GPT2(3)	<u>0.069</u>	<u>0.173</u>	<u>0.048</u>	<u>0.141</u>	0.028	<u>0.105</u>	0.021	<u>0.084</u>	<u>0.031</u>	0.056	0.090	0.207	1
TimeDiT	0.042	0.135	0.042	0.139	0.023	0.098	0.024	0.083	<u>0.031</u>	0.036	0.069	0.174	10

Table 4: Imputation result on 96-length multivariate time series averaged over the four mask ratios. We calculate MSE and MAE for each dataset. **Bold** indicates best result, Underline indicates the second best result.

One Generative Model can Outperform Specialized Systems

Improve with the best specialized model



ONE model beats FOUR specialized state-of-the-art models (over 35 baselines) at their own tasks

Cross-task knowledge transfer emerges from unified pretraining

Experiments: Synthetic Generation Task

Metric	Methods	Sine	Stocks	Air Quality	Energy
Discriminative Score	TimeGAN	0.1217(0.039)	0.2038(0.057)	0.3913(0.039)	0.4969 (0.000)
	TimeVAE	0.0489(0.0562)	0.1987(0.037)	0.2869(0.053)	0.4993(0.001)
	Diffusion-TS	<u>0.0099(0.003)</u>	<u>0.1869(0.0159)</u>	0.1227(0.006)	<u>0.2301(0.006)</u>
	TimeDiT	0.0086(0.004)	0.0087(0.006)	<u>0.2109(0.003)</u>	0.0053(0.002)
Predictive Score	TimeGAN	0.2797(0.015)	0.0481(0.002)	0.035(0.002)	0.3305(0.003)
	TimeVAE	0.2285(0.000)	0.0485(0.000)	0.0269(0.001)	0.2878(0.001)
	Diffusion-TS	<u>0.2262(0.000)</u>	0.042(0.000)	<u>0.022(0.002)</u>	<u>0.2506(0.000)</u>
	TimeDiT	0.1915(0.000)	<u>0.0445(0.000)</u>	0.0217(0.000)	0.2489(0.000)

Table 6: Synthetic Generation result on 24-length multivariate time series. We calculate discriminative and predictive score according to Yoon et al. [2019] and results are averaged over five runs. **Bold** indicates the best performance.

Experiments: Synthetic Generation Task

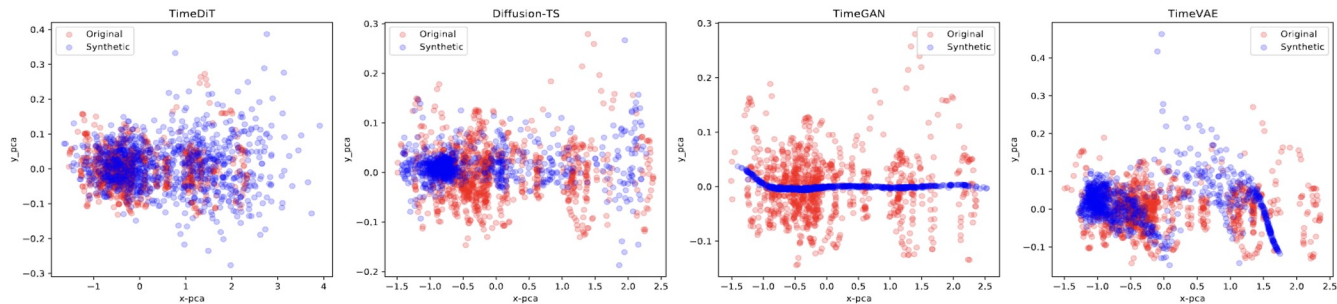


Figure 8: PCA Evaluation of Synthetic TSD from TimeDiT and other baselines on the stock dataset.

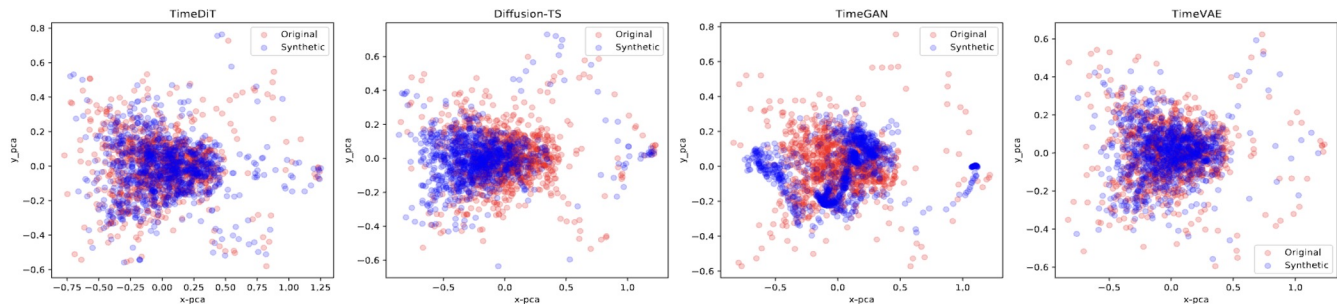
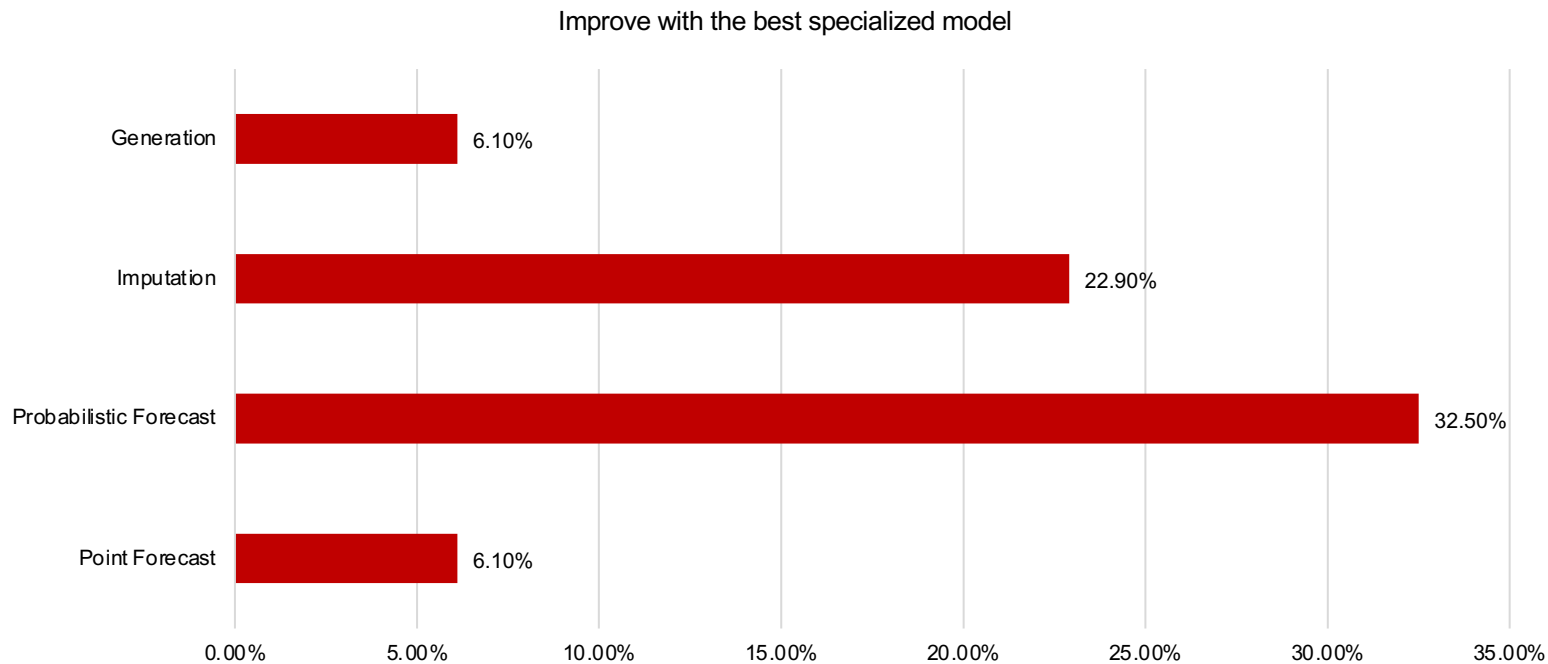


Figure 9: PCA plot for air quality dataset.

One Generative Model can Outperform Specialized Systems



ONE model beats FOUR specialized state-of-the-art models (over 35 baselines) at their own tasks

Cross-task knowledge transfer emerges from unified pretraining

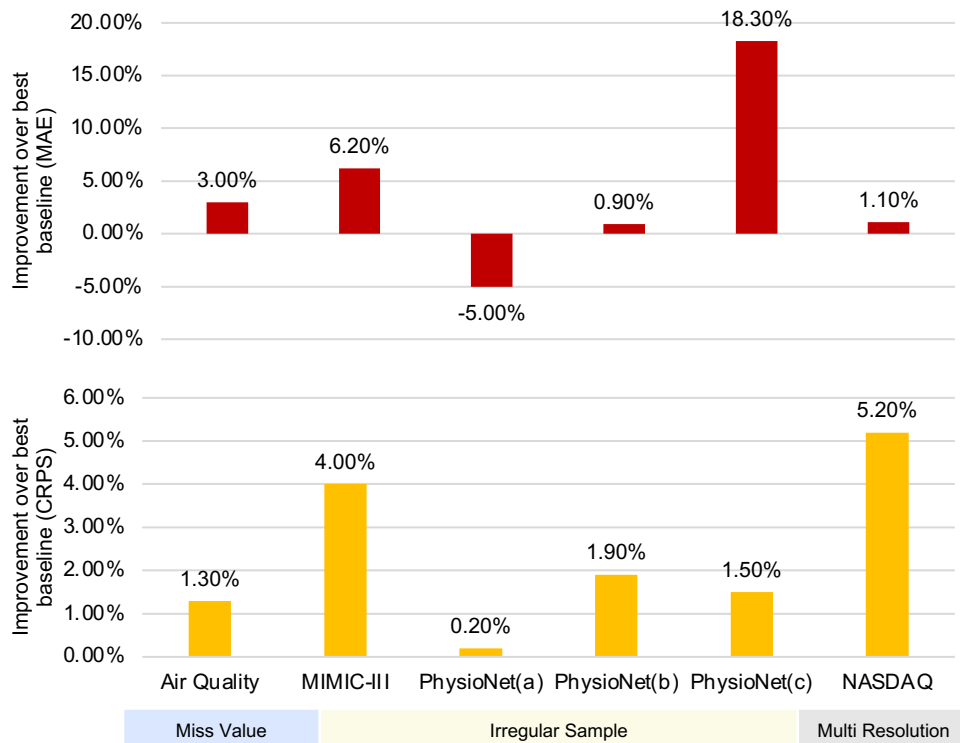
Robustness under Real-world Imperfections

Table 1: Forecasting results on practical scenarios with both deterministic metric (MAE/MSE) for *accuracy* evaluation and probabilistic metric (CRPS/CRPS_sum) for *uncertainty quantification*. **Bold** indicates best result, Underline indicates the second best result.

	Air Quality	MIMIC-III	PhysioNet(a)	PhysioNet(b)	PhysioNet(c)	NASDAQ
	MAE/MSE	MAE/MSE	MAE/MSE	MAE/MSE	MAE/MSE	MAE/MSE
DLinear	0.683/0.685	0.786/1.000	0.686/0.758	0.733/0.922	0.715/0.813	2.715/8.137
Neural ODE	0.678/0.679	0.784/0.999	0.685/0.756	0.732/0.918	0.713/0.811	3.227/11.155
Neural CDE	0.683/0.685	0.787/1.002	0.688/0.754	0.733/0.921	0.713/0.814	3.319/11.816
PatchTST	0.685/0.683	0.778/0.987	0.699/0.780	0.733/0.932	0.714/0.802	3.182/10.635
GPT2(3)	0.696/0.701	0.750/0.921	0.697/0.772	0.734/0.921	0.713/0.817	3.176/10.873
CSDI	0.539/0.554	0.551/0.681	0.548/0.548	0.665/0.792	0.665/0.695	<u>0.524/0.388</u>
DiffTS	<u>0.521/0.538</u>	0.677/0.908	0.610/0.742	0.701/0.880	0.678/0.872	1.951/9.515
TimeDiT	0.457/0.354	0.517/0.534	<u>0.577/0.620</u>	0.659/0.766	0.543/0.561	0.516/0.418
	CRPS/_sum	CRPS/_sum	CRPS/_sum	CRPS/_sum	CRPS/_sum	CRPS
DLinear	0.662/0.544	0.770/0.748	0.764/0.812	0.794/0.793	0.767/0.797	0.342
Neural ODE	0.657/0.529	0.769/0.733	0.763/0.806	0.792/0.789	0.765/0.793	0.426
Neural CDE	0.659/0.551	0.771/0.754	0.763/0.799	0.792/0.786	0.765/0.791	0.439
PatchTST	0.664/0.564	0.771/0.721	0.769/0.812	0.791/0.775	0.766/0.777	0.410
GPT2(3)	0.666/0.584	0.751/0.690	0.767/0.809	0.795/0.798	0.770/0.768	0.419
CSDI	<u>0.598/0.620</u>	0.504/0.798	<u>0.620/0.641</u>	0.725/0.787	0.669/0.748	0.096
DiffTS	0.649/0.719	<u>0.633/0.676</u>	0.628/0.668	<u>0.720/0.724</u>	<u>0.679/0.719</u>	0.283
TimeDiT	0.554/0.522	<u>0.599/0.649</u>	0.616/0.640	0.708/0.710	0.668/0.708	0.091

- **Missing Values (Climate)**
 - Incomplete conditioning handled natively
- **Irregular Sampling (Healthcare)**
 - Masking decouples modeling from sampling artifacts.
- **Multi-Resolution (Finance)**
 - Shared backbone captures cross-scale dynamics.

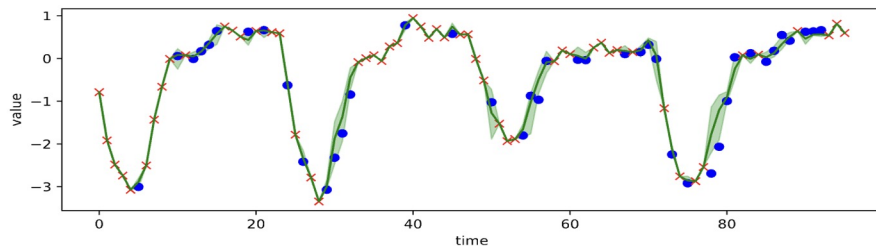
Robustness under Real-world Imperfections



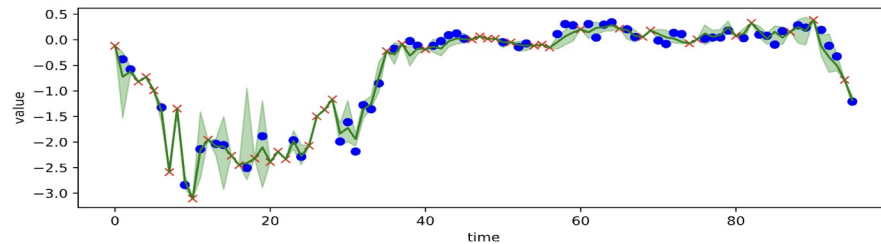
- **Missing Values (Climate)**
 - Incomplete conditioning handled natively
- **Irregular Sampling (Healthcare)**
 - Masking decouples modeling from sampling artifacts.
- **Multi-Resolution (Finance)**
 - Shared backbone captures cross-scale dynamics.

Robustness emerges from **modeling distributions**,
not from task-specific tuning.

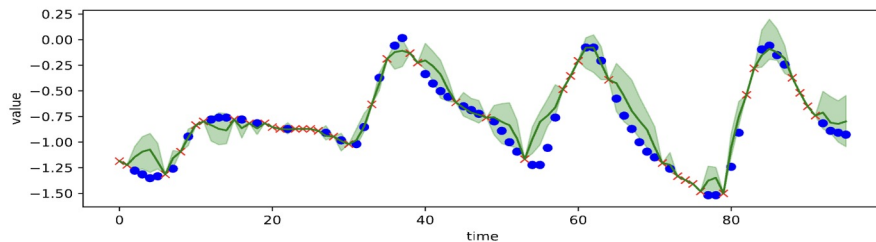
TimeDiT Knows when It Does Not Know



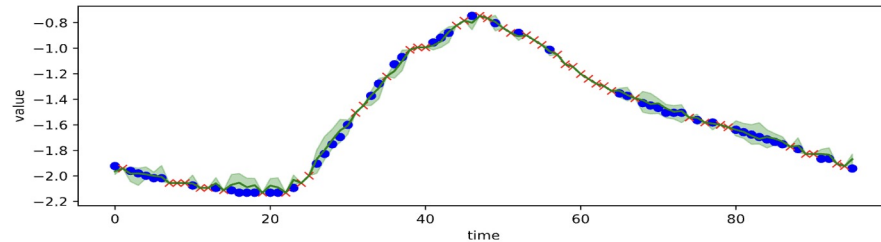
(a) Imputation results on ETTh1 dataset.



(c) Imputation results on ETTm1 dataset.



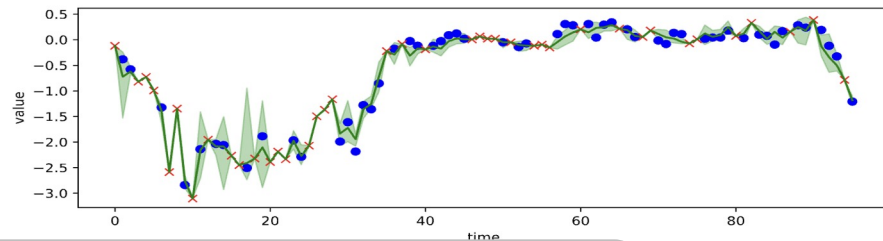
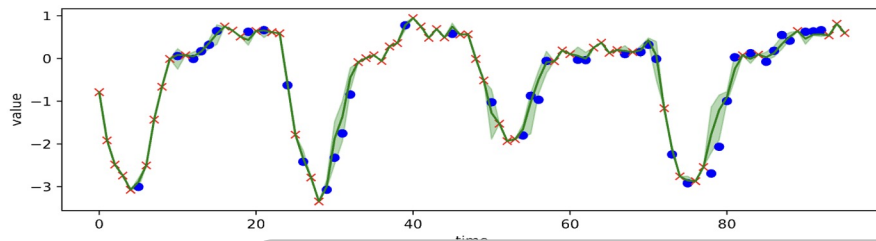
(b) Imputation results on ETTh2 dataset.



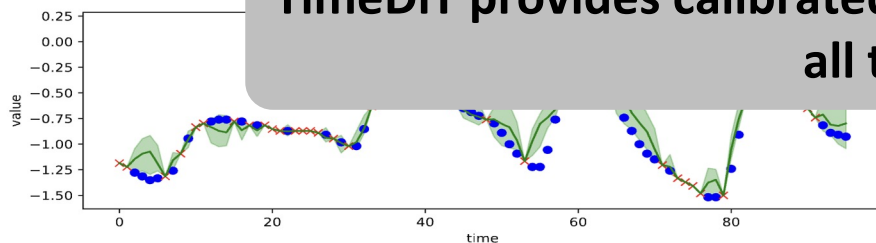
(d) Imputation results on ETTm2 dataset.

Visualization of imputation task on ETT datasets. This figure illustrates TimeDiT's performance, with red $\times$'s marking observed values, blue dots showing ground truth points for interpolation, a green line representing TimeDiT's mean of interpolation, and green shading indicating its estimated uncertainty intervals.

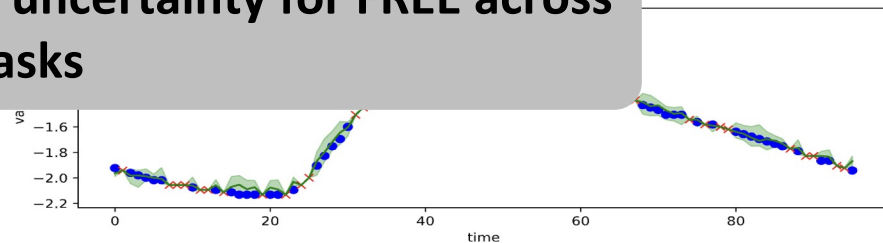
Uncertainty Quantification - Knowing When We Don't Know



TimeDiT provides calibrated uncertainty for FREE across all tasks



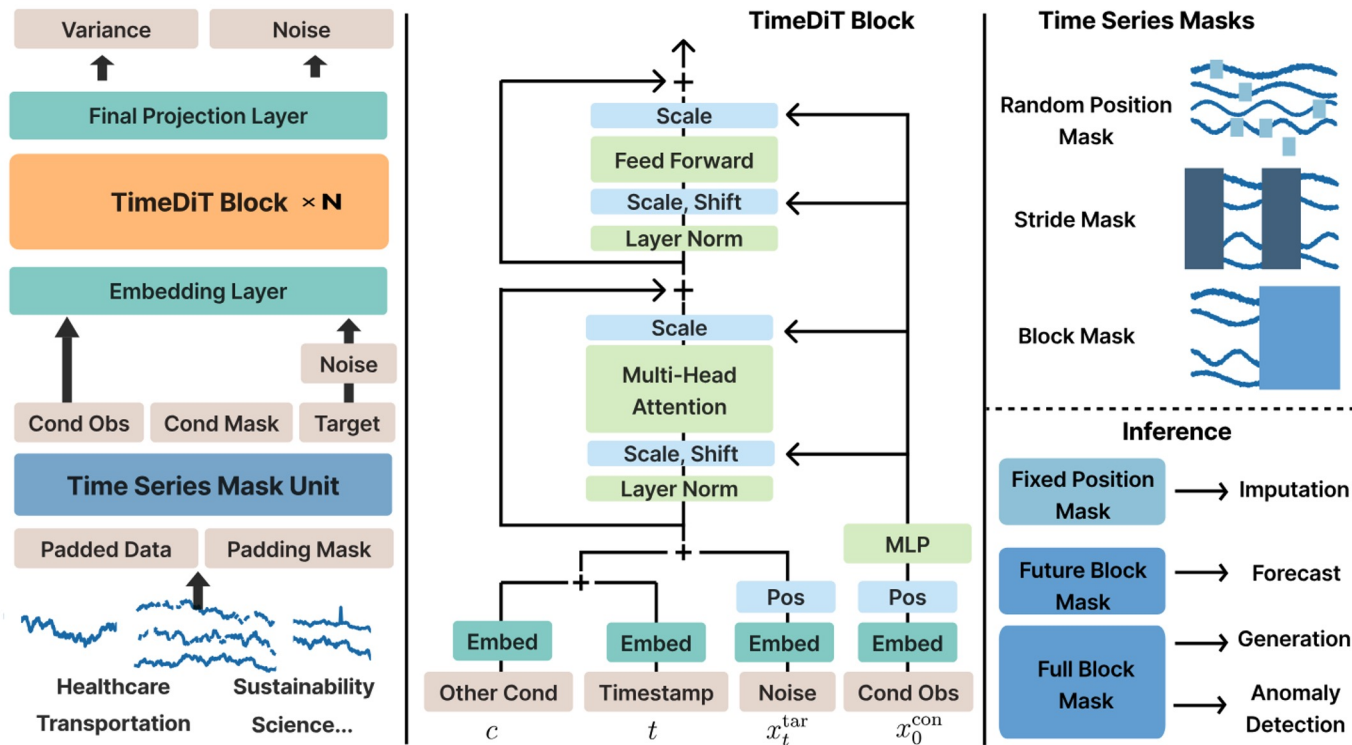
(b) Imputation results on ETTh2 dataset.



(d) Imputation results on ETTm2 dataset.

Visualization of imputation task on ETT datasets. This figure illustrates TimeDiT's performance, with red 'x's marking observed values, blue dots showing ground truth points for interpolation, a green line representing TimeDiT's mean of interpolation, and green shading indicating its estimated uncertainty intervals.

Recap: TimeDiT Architecture



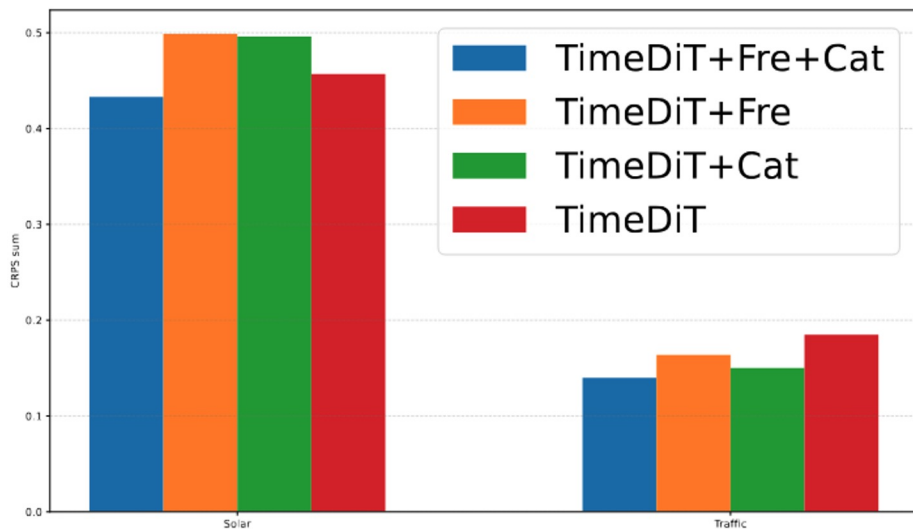
Left: TimeDiT model framework with diverse multivariate time series from different domains with multi-resolution or missing values;

Middle: TimeDiT block;

Right:

- Top: Illustration of Time Series Mask Unit;
- Bottom: Mask examples for downstream tasks

Experiments: Multimodal TimeDiT



TimeDiT + Fre: TimeDiT with frequency constraint
TimeDiT + Cat: TimeDiT with category constraint
TimeDiT + Fre + Cat: TimeDiT with previous two constraints

Utilizing both types of information significantly boosts zero-shot performance

Ablation Study

- Mask Strategy
 - Three masks (TimeDiT)
 - w/o Random Mask
 - w/o Stride Mask
 - w/o Block Mask
- Attention mechanism
 - Temporal Attention (TimeDiT)
 - Dual attention
 - Channel-wise attention
- Input Token
 - Direct Token (TimeDiT)
 - Patch Token
- Conditional Way
 - Adaptive Normalization (TimeDiT)
 - Additive Condition

Table 8: Ablation study on zero-shot forecasting

Dataset	TimeDiT	w/o Random M.	w/o Stride M.	w/o Block M.	w/o Phys
Solar	0.424	0.465	0.469	0.862	0.445
Electricity	0.030	0.035	0.037	0.101	0.033
Dataset	Dual-attention	Channel-wise	Patch Token	Additive	Cross-attention
Solar	0.467	0.461	0.874	0.677	0.711
Electricity	0.037	0.039	0.145	0.079	0.077

From Generalization to Physical AI

Bridging Representation and Generalization



Generative Modeling

- ✓ Task-Agnostic Versatility
- ✓ Distributional Resilience
- ✓ Probabilistic Calibration



Scientific Blindness

- ✗ Mechanistic Disconnect
- ✗ Constraint Violation
- ✗ Interpretability Bottleneck



Generalization $\neq$ ***Physical Validity***

Physical Grounding: From Generalization to Physical Validity

How to make generated trajectories obey physics — case study: PINFDiT

01

Physics, Intuitively

A six-step primer on the constraints scientific data imposes.

02

Energy-Guided Sampling

Inject physics as an energy term at inference time — no retraining needed.

03

Long-Horizon Stability

Grounding improves accuracy and physical consistency at climate scale.

Physics does not mean memorizing every equation.

Physics enters as a constraint that nudges samples toward what's physically possible.

Not this

- Memorize every PDE
- Re-derive the math live

But this

- A residual = how much a law is broken
- Lower residual = more valid

The intuition

- Steer generation toward valid states
- Like lane-assist for the model

Pillar III: From Pattern Matching to Physical Grounding

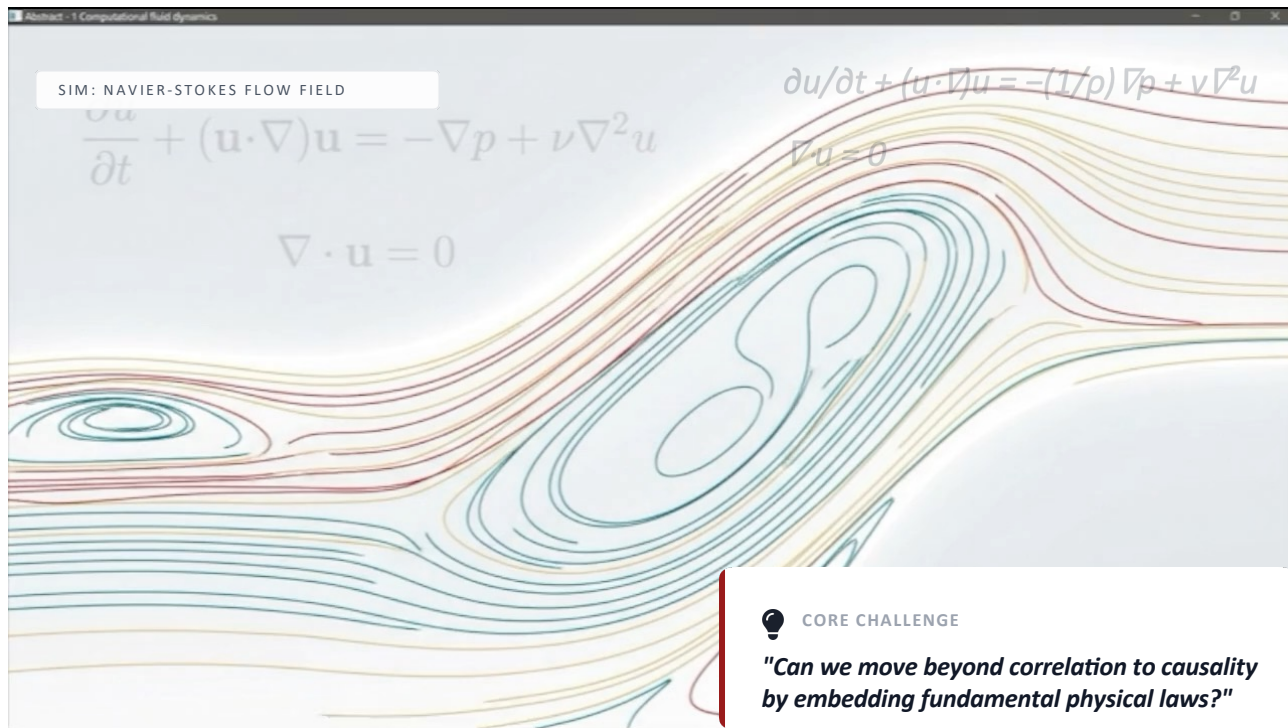
*Can foundation models obey physical laws without retraining? **Yes** — with guided correction during sampling.*



PATTERN MATCHING



PHYSICAL REASONING



What Does Physical Consistency Mean?

The Grounding Concept

Physical consistency: outputs obey the known mathematical laws of the physical world.

Conservation Laws & PDEs

Mass, energy, and momentum balances described by PDEs (e.g., fluid dynamics).

Boundary & Operational Limits

Inviolable physical bounds, grid capacity maximums, and mechanical ramping limits.

Physical Validity vs Point Plausibility

A. Statistically Plausible but Physically Invalid

The curve fits history yet breaks mass conservation at a closed node.

B. Constraint-Consistent & Valid

Satisfies statistical patterns and physical conservation simultaneously.

Deep Grounding: pretrained nets learn trends but miss differential physics—explicit constraints are needed.

Key Takeaway: a statistically plausible trajectory can still break physical invariants.

A Simple Conservation-Law Example

Flow Conservation at a Node

At a junction with inflow $f_{in}(t)$, outflow $f_{out}(t)$, and storage $\Delta s(t)$:

$$f_{in}(t) - f_{out}(t) = \Delta s(t)$$

The Multi-channel Leak: low per-channel MSE can still break the conservation balance at runtime.

Main Takeaway: physical validity lives in cross-channel relationships, not point errors.

Physical Flow Conservation Node



Structural Coupling: coupling channels as equations keeps generation on valid physical manifolds.

Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear PDEs

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis — *Journal of Computational Physics*, 2019

Core Problem

Can neural networks solve forward and inverse PDE systems data-driven and mesh-free?

Key Idea / Methodology

PINNs embed governing PDEs into the loss via automatic differentiation.

Data-Driven Loss + PDE Residual penalty

The Temporal Interface

A loss-regularized physics-residual interface combining statistical fits with mathematical constraints: A physics-residual loss combining data fit with math constraints:

$$L_{\text{total}} = L_{\text{data}} + \lambda ||F(u_{\theta})||^2$$

Why It Matters for TSFMs

The primary reference for differentiable constraints—equations act as training labels.

Loss-Regularized Physics Integration

Remaining Gap

Per-system training-time constraints make PINNs rigid, costly, and non-zero-shot.

Neural Operators

Infinite-Dimensional Mapping

Unlike vector nets, Neural Operators map between function spaces.

The Solution Operator

They learn the solution operator mapping initial coefficients to whole continuous fields: They learn the operator mapping initial conditions to full fields:

$$\mathbf{a}(\mathbf{x}) \rightarrow \mathbf{G}_{\theta} \rightarrow \mathbf{u}(\mathbf{x}, t)$$

Key Benefit: mesh-independent—evaluate at any resolution without retraining.

Operator Architecture Advantages

Resolution Independence

Generalizes across different grid sizes and discretizations.

Spectral Convolutions

Fourier Neural Operators (FNO) parameterize layers in Fourier frequency space.

PDE Surrogates

Fast, robust neural approximations of fluid and weather systems.

Thesis Limitation: operators need massive simulation labels and no generative interface.

Training-Time Constraints vs. Inference-Time Guidance

Methodology A

Training-Time Constraints

Enforces physical equations inside the objective function during training.

- ⊘ Requires system-specific equations from day one
- ⊘ Restricts the model's cross-domain flexibility
- ⊘ Requires complete model retraining if rules change

Inductive Bias: Locks parameters to one predefined set of equations.

Methodology B

Inference-Time Guidance

Starts from a pretrained prior and evaluates constraint violations during sampling.

- ✔ Retains full zero-shot, cross-domain prior power
- ✔ Guides noisy estimates toward physical limits
- ✔ Adapts to new physical laws without retraining

Posterior Refinement: aligns samples with physics without hurting representation.

Posterior Refinement Intuition

Guided Probability Flow

Plausible Prior Sample

From Temporal Prior



Constraint Evaluator

Energy Score $E(x)$



Ground-Truth Valid Output

Refined Trajectory

The Bayesian Bridge: refinement is posterior optimization (prior $\times$ constraint likelihood), keeping plausibility.

Refinement Process Steps

Evaluate Constraint Energy $E(x)$

Each reverse step adds a physics-violation energy gradient.

Gradient-Guided Step Adjustments

Slide the noisy representation toward lower-energy regions, minimizing violations.

Preserve Plausibility Boundaries

The generative prior stops collapse into non-physical states.

Inference Advantage: guidance turns blind generation into bounded physical inference.



Scientific Data adds hard constraints and extreme scarcity.

RESOLVED "STANDARD" CHALLENGES



Statistical Robustness

Handling missingness and irregular intervals via diffusion.



High-dimensional Scaling

Modeling complex correlations in massive multivariate streams.

THE "DEEP SCIENCE" BARRIER



Hard Constraints

Conservation laws and symmetries must be strictly obeyed.



Extreme Data Scarcity

High-cost experiments mean learning from few examples.

THE REALITY GAP of

Scientific Data

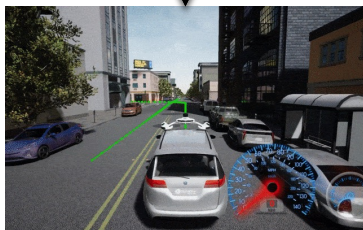
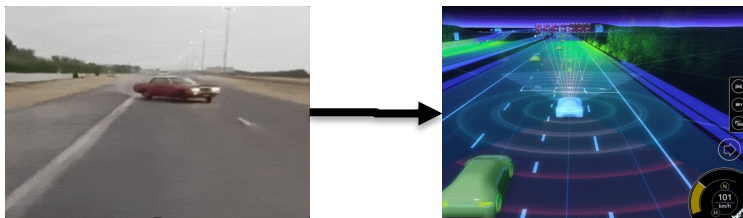
=

Structure+Scarcity

Grounding enforces **validity**.

The Core Idea: Incorporate Physics Knowledge into FMs

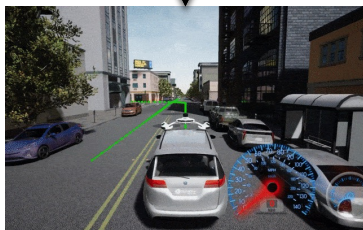
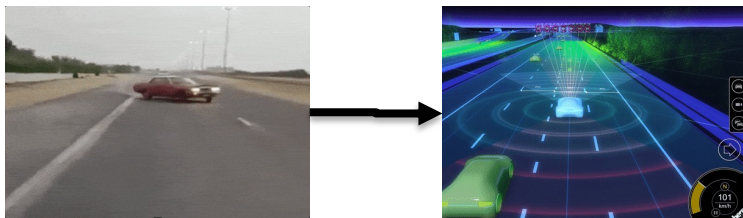
"Model's potential output:" "Physics constraint:"



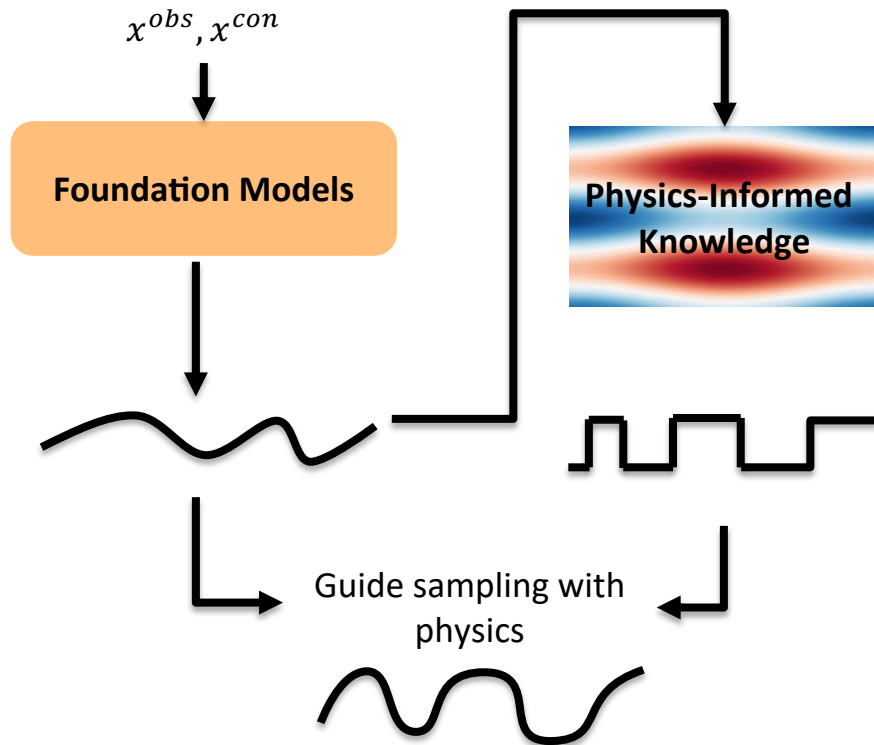
Ideal way to go

Physics Guided Foundation Models

"Model's potential output:" "Physics constraint:"



Ideal way to go



Physics as an Energy Term during Inference

Proving that inference-time correction is the optimal solution for physical consistency.

Physics as Energy

$$\mathcal{K}(x; \mathcal{F}) = \|\mathcal{F}(x, \nabla x, \nabla^2 x, \dots)\|^2$$

Physics Residual (Energy)

- > $\mathcal{F}$ represents physical operators (e.g., Navier-Stokes, Heat Eq).
- > $\mathcal{K}$ measures physical violation.

✓ **Low Energy = High Physical Adherence**

The Optimal Goal (Theorem 3.1)

The "Ideal way to go" (the optimal distribution q) is a Boltzmann Energy Distribution:

Optimal Distribution

$$q(x) \propto \exp(\underbrace{-\mathcal{K}(x)}_{\text{Physics}} + \underbrace{\alpha \log p_{\theta}(x)}_{\text{Data Prior}})$$

Term 1: $\mathcal{K}(x; \mathcal{F})$

The "Lane Constraint" (Physics Residual): Minimizing this keeps the car on the road.

Term 2: $\log p_{\theta}(x)$

The "Engine" (Data Prior): Ensures the car still looks and move like a correct car.



No Retraining: Physics is injected at inference time, allowing a single model to adapt to various physical systems.



PINFDiT steers diffusion samples toward the physical manifold.

Treating physical laws as a differentiable **energy functional** rather than hard-coded constraints.

The Steering Mechanism (Algorithm 1)

Stochastic Gradient Langevin Dynamics

$$x_{k+1} = x_k + \eta \nabla [-\mathcal{K}(x_k; \mathcal{F}) + \alpha \log p_\theta(x_k)] + \text{noise}$$



Physics Corre

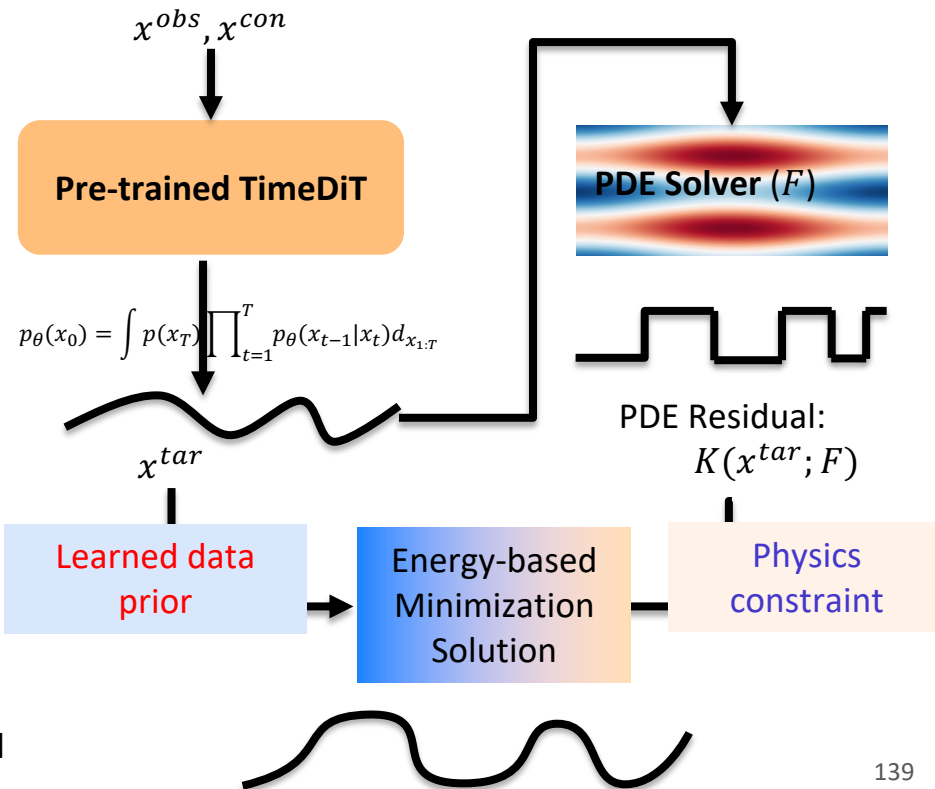
Maximizes adherence to
fundamental laws (PDEs)

Result:
Physically Consistent
Generation

Ensures samples remain on
the realistic data manifold

Prior

- 💡 Model-Agnostic: Work as a “plug-and-play” corrector.
- 💡 Guaranteed Convergence: Theorem 3.2
- 💡 Zero Retraining: No need to rebuild the foundation model



Comprehensive Validation: From Theory to Turbulence

- Three PDEs: covering major physics types
 - Zero-shot testing: No training on target physics
 - Baseline comparison: 20+ state-of-the-art models
 - CFD case study: Real computational fluid dynamics

Physics Type	Governing PDE System	Key Challenge	Real-World Application
Wave Dynamics	Advection Equation	Shock formation	Modeling weather fronts
Heat Transfer	Burgers' Equation	Turbulence	Simulating engine cooling
Fluid Flow	Navier-Stokes Equations	Chaotic behavior	Aerodynamics and design

Table 1: Overview of Physical Systems Under Study

Main Results - First Foundation Model with Physics Guarantees

Integrating physics-informed priors into the diffusion model sampling process further improves performance.

Model	Advection			Burgers			Navier-Stokes		
	RMSE	MAE	CRPS	RMSE	MAE	CRPS	RMSE	MAE	CRPS
TimesFM	0.0602	0.0428	0.0640	0.0278	0.0164	0.0822	0.0100	0.0076	0.0065
Moirai-Small	0.1033	0.0738	0.1165	0.1432	0.1193	0.5057	0.0368	0.0250	0.0243
Moirai-Base	0.0669	0.0481	0.0731	0.0972	0.0824	0.3701	0.0147	0.0107	0.0097
Moirai-Large	0.0666	0.0481	0.0721	0.0365	0.0254	0.1227	0.0145	0.0097	0.0089
Moirai-MoE	0.0463	0.0300	0.0458	0.0275	0.0164	0.0827	0.0273	0.0199	0.0162
Chronos-T5-B	0.0414	0.0258	0.0390	0.0202	0.0100	0.0474	0.0081	0.0055	0.0047
Chronos-T5-S	0.0334	0.0218	0.0330	0.0424	0.0304	0.1366	0.0121	0.0080	0.0066
Chronos-Bolt-T	0.0919	0.0691	0.1056	0.0776	0.0703	0.3232	0.0465	0.0366	0.0298
Chronos-bolt-B	0.1009	0.0706	0.1029	0.0205	0.0131	0.0634	0.0365	0.0285	0.0228
Chronos-bolt-S	0.0678	0.0534	0.0776	0.0445	0.0351	0.1585	0.0378	0.0329	0.0261
DLinear	0.3674	0.2943	0.4248	0.0287	0.0216	0.0975	0.9321	0.9101	0.6977
PatchTST	0.3767	0.3026	0.4410	0.0271	0.0208	0.0953	0.8972	0.8861	0.6928
CSDI	0.0118	0.0154	0.0176	0.0167	0.0076	0.0178	0.0094	0.0060	0.0055
DiffusionTS	0.3393	0.2484	0.3540	0.0361	0.0263	0.1363	0.8267	0.6192	0.5063
NeuralODE	0.3667	0.2968	0.4268	0.0284	0.0214	0.0966	0.9314	0.9045	0.6965
NeuralCDE	0.3690	0.3021	0.4314	0.0292	0.0221	0.1001	0.9318	0.9051	0.6972
SNPE	0.0529	0.0422	0.0667	0.0472	0.0373	0.1903	0.1281	0.1134	0.0912
LFBC	0.0352	0.0280	0.0512	0.0382	0.0294	0.1615	0.0739	0.0576	0.0491
TSDiff	0.5128	0.5612	1.0000	0.1949	0.1670	0.9999	0.9212	0.9077	0.9999
PINFDiT_DDPM	0.0052	0.0040	0.0061	0.0136	0.0036	0.0163	0.0039	0.0031	0.0028
PINFDiT	0.0039	0.0032	0.0050	0.0133	0.0035	0.0159	0.0037	0.0030	0.0027

Table 2: Comparison of model performance across different physical systems using both deterministic metrics for **accuracy** and probabilistic metrics (CRPS) for **uncertainty** quantification evaluation.

Grounding Improves Both Accuracy and Physical Consistency

Demonstrating superior performance and physical consistency across diverse scientific domains.

Constraint Violation



Purely data-driven models frequently generate physically implausible states

Solution Space Restriction



Physics-guided sampling restricts the generation manifold to valid physical states

Dual Improvement

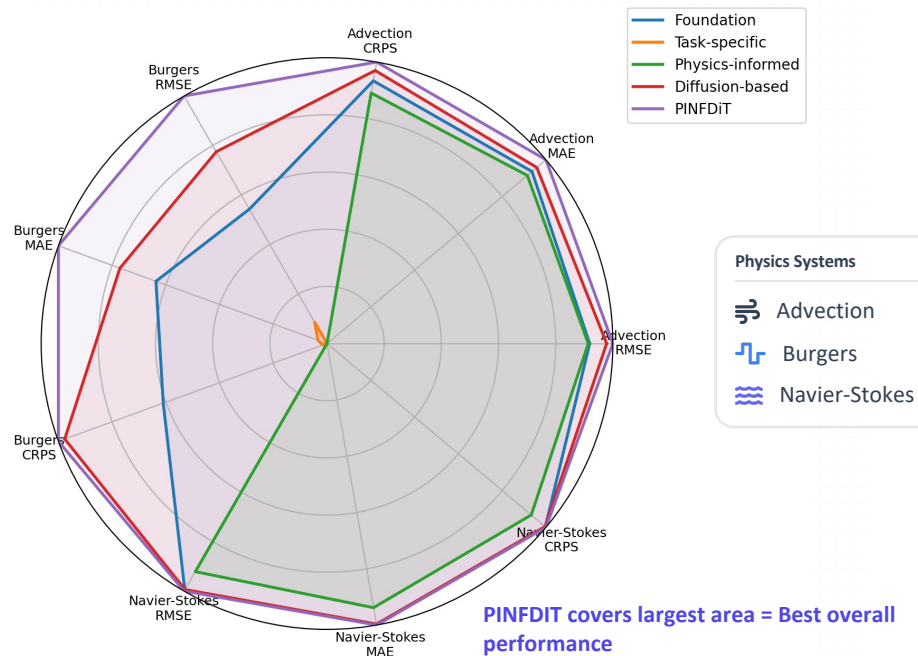


Constrained generation improves both **accuracy** and **calibration**

Paradigm Comparison

Normalized Metrics (Higher = Better)

PINFDIT (Ours)



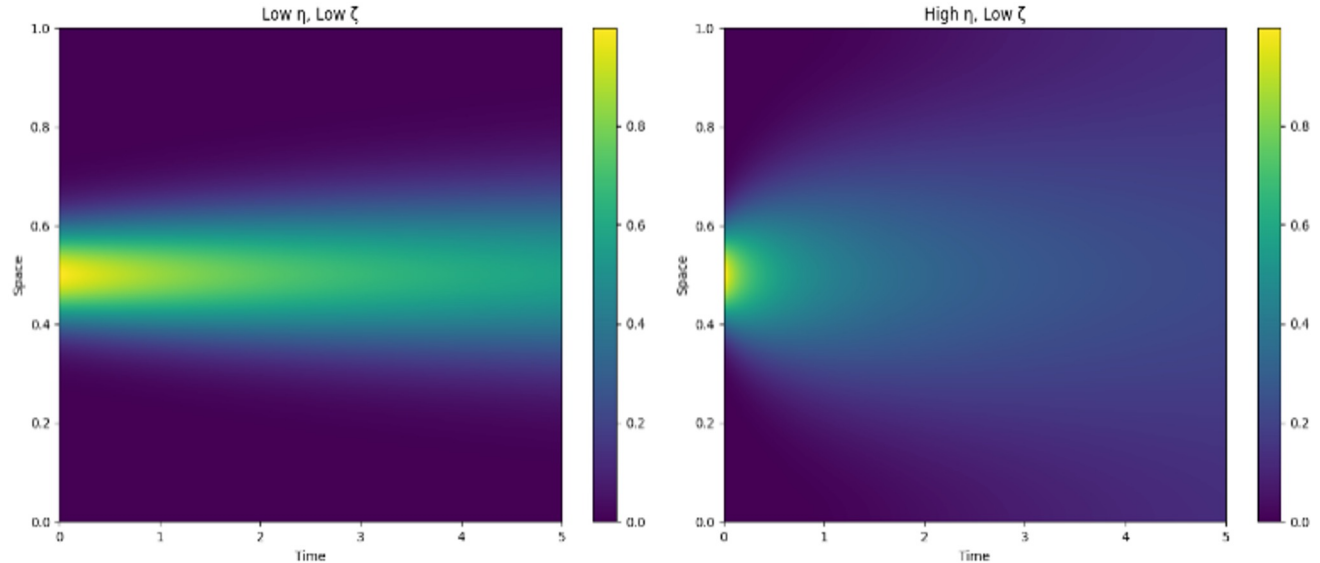
Metrics: RMSE (Error), MAE (Error), CRPS (Probabilistic Score)

CFD Case Study - Real-World Impact

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} - \eta \frac{\partial^2 u}{\partial x^2} = \sin(\zeta x)$$

η (eta), viscosity (smoothness)
↑ η = More diffusion, smoother solutions
↓ η = Less diffusion, sharper gradients

ζ (zeta): forcing frequency (complexity)
↑ ζ = Higher frequency oscillations, smaller structures
↓ ζ = Lower frequency oscillations, larger structures



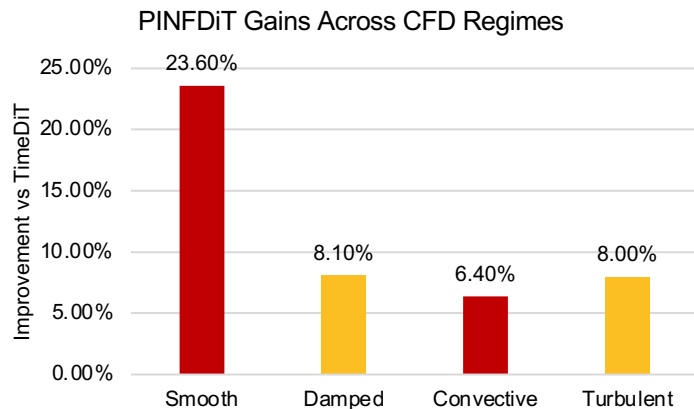
Physics FM Stabilizes Long-horizon Scientific Forecasting

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} - \eta \frac{\partial^2 u}{\partial x^2} = \sin(\zeta x)$$

Physical Regimes

- Viscosity (η) controls smoothness
- Forcing frequency (ζ) controls dynamical complexity

	Low ζ	High ζ
Low η	Smooth	Convective
High η	Damped	Turbulent



Grounding:

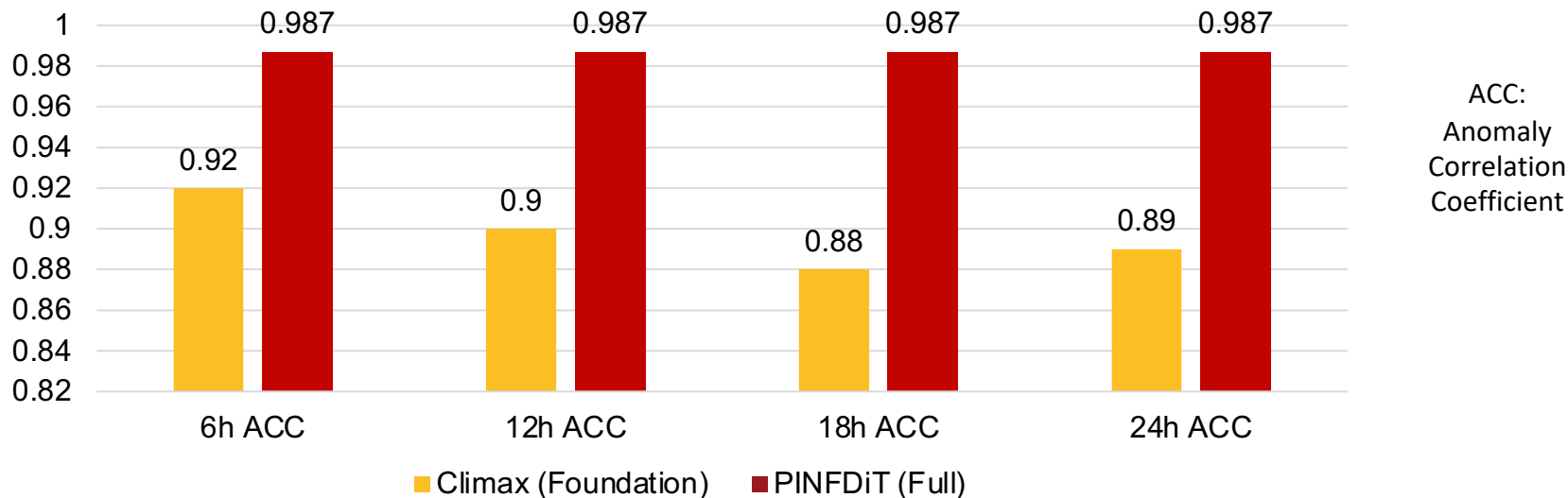
- Enforces physical validity
- Prevents drift in complex regimes
- Calibrates uncertainty

PINFDiT for Climate Intelligence

- Data-driven forecasting



First-principle atmospheric physics

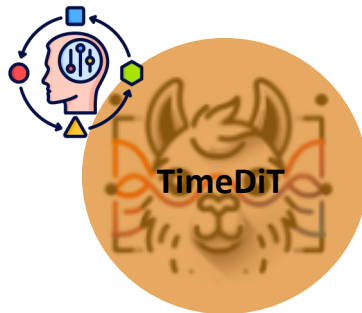


PINFDiT maintains a near-perfect **0.987 ACC** consistently across all lead times, whereas baseline models degrade as the horizon expands.

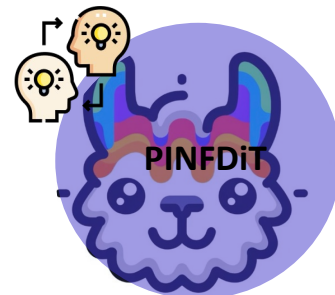
Consolidating the Temporal Foundation



Can we create universal representations for temporal patterns? **Yes!** — through representation alignment.



Can one model handle all tasks despite data imperfections? **Yes!** — via unified generative modeling.



Can we inject domain knowledge without retraining? **Yes!** — via unified generative modeling.



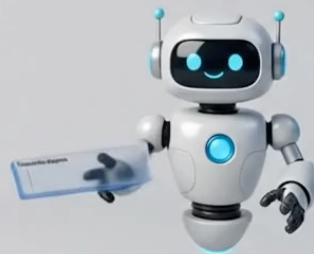
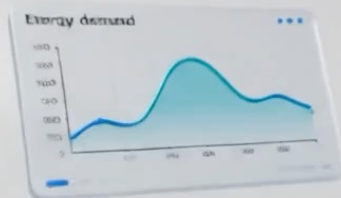
Can we build an agentic layer that evaluates, selects, and coordinates these foundation models to execute reliable actions?

The Reasoning Paradigm: Orchestrating for Decisions

TSOrchestra — Reasoning & Orchestration

Time Series Agent Orchestra

Agents coordinating foundation models
under changing time-series regimes



CORE CHALLENGE

Strong performance alone does not guarantee
correct decisions

Orchestration: No Single Model Wins

How to route, combine, and audit temporal experts — case study: agentic orchestration

01

Agent Systems 101

A step-by-step primer on agentic pipelines for temporal decision-making.

02

LLMs as Auditors

Don't ask LLMs to forecast numbers — ask them to audit evidence.

03

Routing That Works

SHAP-based expert selection, evaluated on the GIFT-Eval benchmark.

What Is an LLM Agent?

An LLM agent runs a closed loop: perceive, reason, act, and learn from feedback.

🧠 **Reasoning Core:** the LLM plans and emits structured tool commands.

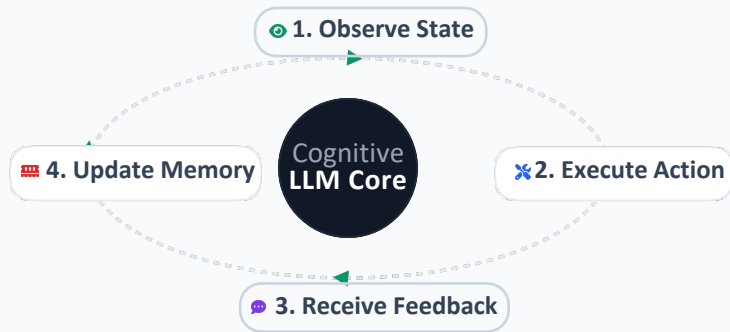
📚 **State & Memory:** tracks history, errors, and environment updates across steps.

🔄 **Dynamic Policy Loop:** Continuously repeats the cycle:
Observe → Plan → Act → Feedback.

Main Takeaway

An agent is not just a static prompt; it is a feedback-driven decision system.

The Agent-Environment Interaction Cycle



*Yao et al. "ReAct." 2023.
Wang et al. "Voyager." 2023.*

Agent Harness: LLM + Tools + Memory + Environment

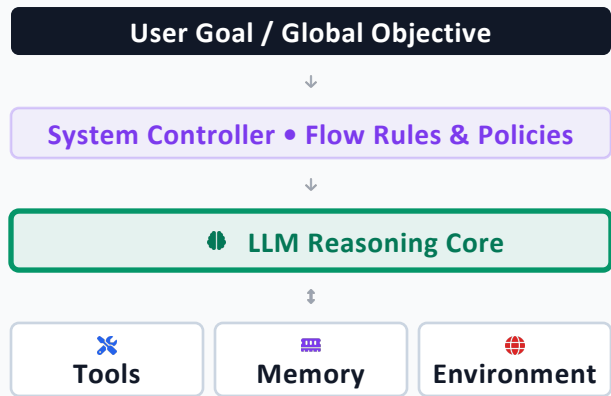
A practical agent needs a robust harness—the infrastructure that wraps the LLM.

- 1. Model Core:** The language model that performs logical reasoning and generation.
- 2. Executable Tools:** APIs, statistical models, simulators, and computation engines.
- 3. Multi-Tier Memory:** Session state, long-term logs, and past error patterns.
- 4. Dynamic Environment:** External state that evolves with agent actions.
- 5. System Controller:** Manages retries, tool routing, and failure recovery.

Main Takeaway

The harness determines the boundaries of what an agent can execute.

Agent Harness Layered Stack



Schick et al. "Toolformer." 2023.

Yao et al. "ReAct." 2023.

ReAct: Reasoning and Acting in One Loop

Problem: Reasoning alone cannot touch the world; acting without reasoning is blind.

The ReAct Paradigm

Interleaves Thought and Action step-by-step for interpretability and resilience.

- ✓ **Supports multi-step tasks:** Decomposes complex goals into iterative steps.
- ✓ **External feedback-driven:** Real-time observations steer subsequent reasoning paths.
- ✓ **Explicit tool integration:** Transforms API execution directly into logical premises.

Trace Simulation: Execution Loop

THOUGHT 1

Check validation errors for the target horizon.

| ACTION 1

`query_validation_metrics(horizon="96")`

OBSERVATION 1

A: MAE 0.23 (short) · B: MAE 0.88 (long)

THOUGHT 2

Expert A wins short-horizon → 80% weight to A.

| ACTION 2

`update_ensemble_weights(A=0.80, B=0.20)`

Yao, Shunyu et al. "ReAct." 2023.

Toolformer: Learning When to Use Tools

Problem: LLMs are fluent but weak at arithmetic, lookup, and specialized computation.

- ▶ **Self-Supervised Decision:** models learn to emit API-call tokens within normal text.
- ▶ **Delegation Principle:** offload heavy computation to external tools.

Main Takeaway

Agent skill hinges on knowing when to call specialized tools.

Inline API Decoupling Mechanics

Input Stream • Auto-triggered Execution

A validation error triggers an inline API call that returns a concrete value.

[API: compute_mae(pred, true)]

0.142

1. Inline Call Detection 2. Sandbox Compute 3. Context Injection

Recognizes the specific API tag

Executes code or queries models

Merges results back into output

Schick, Timo et al. "Toolformer." 2023.

Reflexion: Self-Feedback Without Weight Updates

Problem: multi-step agents loop and fail, but constant fine-tuning is too costly.

1 Verbal Reinforcement Learning

Translates raw failures into linguistic autopsies of what went wrong.

2 Episodic Memory Accumulation

Stores reflections in a context buffer to avoid repeating past errors.

Main Takeaway

Agents become reliable by converting runtime failures into reusable self-feedback.

Verbal Feedback Loop Architecture

▶ 1. Action Attempt Agent output rollout

❗ 2. Failure Signal Observed Constraint / Metric Violation

💡 3. Self-Reflection "Why did I fail?"

📖 4. Short-Term Memory Storage Conditioning future attempts

Shinn et al. "Reflexion." 2023.

Voyager: Lifelong Skill Acquisition

Voyager agents accumulate reusable long-horizon skills with no parameter updates.

- ✓ **Automatic Curriculum:** proposes progressively harder goals from current mastery.
- ✓ **Skill Library:** Stores synthesized code modules as persistent, reusable assets.
- ✓ **Iterative Prompting:** refines scripts live from compiler and error feedback.

Main Takeaway

Long-horizon agents require memory, skill reuse, and continuous task adaptation.

Lifelong Skill Generation Loop

1. Automatic Curriculum

Sets progressive exploration goals

</> 2. Action Script Generation

Compiles and tests execution code

3. Skill Library Integration

Query / Store

Saves proven scripts as executable APIs

Scaling Agents Requires Scaling Environments

Static corpora can't teach interactive reasoning—agents need live environments.

- **Interactive Experiential Data:** rollouts expose agents to edge cases and unexpected failures.
- **Validation streams as simulators:** forecast logs, grid state machines, and drift benchmarks become environments.

Main Takeaway

Agent intelligence is shaped by the environments and feedback loops it experiences.

Generation-Execution-Feedback (GEF) Paradigm

1. TASK GENERATION

Synthesizes adaptive scenarios to test boundaries



2. AGENT EXECUTION

Agent executes multiround plans in the environment



3. ENVIRONMENT FEEDBACK

Real-time execution outputs, compilers, metrics



4. CONTINUAL LEARNING

Extracts experience to condition future reasoning logs



Huang et al. "Scaling Environments for LLM Agents." 2025.

Agent Evaluation: From Static Accuracy to Interactive Success

NLP benchmarks grade text; agents must be graded on state transformation and execution process.

 **Task Success Rate**
Binary or continuous verification of final objective completion.

 **Planning Efficiency**
Measures shortest path execution and avoiding reasoning loops.

 **Error Recovery**
Ability to adapt strategy dynamically when compiler or tool errors strike.

 **Audit Trail**
Verifiability and mathematical alignment of reasoning steps.

Main Takeaway

Agents must be evaluated by process and outcome, not only final text quality.

Static NLP vs. Interactive Agent Benchmark

Dimension	Static Benchmark	Agent Benchmark
Interaction	Single-turn prompt	Multi-round feedback loop
Environment	None (Isolated execution)	Dynamic state changes
Tool Use	None	APIs, compilers, optimizers
Failure Recovery	One-shot success/failure	Dynamic self-correction

Liu et al. "AgentBench." 2023.
Zhou et al. "WebArena." 2024.

From General Agents to Temporal Agents

TSOrchestra bridges general agent-interaction models and numerical time-series algorithms.

- 🔗 **Isomorphism of Concepts:** observation, action, tools, memory map onto forecasting-stack operators.
- ⚡ **Action Boundary Shift:** temporal agents tune weights and constraints from numerical evidence.

Main Takeaway

TSOrchestra is a temporal agent acting over model evidence, not raw space.

Theoretical Analogy Mapping

General Agent System	Temporal Orchestration System (TSOrchestra)
Observation	Validation errors, rolling loss metrics, regime flags
Tools	Decomposition modules, TSFMs, classic statistics APIs
Memory	Historical validation matrices, past out-of-bounds error patterns
Actions	Select expert models, adjust ensemble weights, call auditors
Feedback	Target loss reduction, alignment accuracy, explanation faithfulness

Yao et al. "ReAct." 2023. •
Bates & Granger. 1969. •
Wolpert. 1992.

Bridge to TSOchestra: Agentic Orchestration

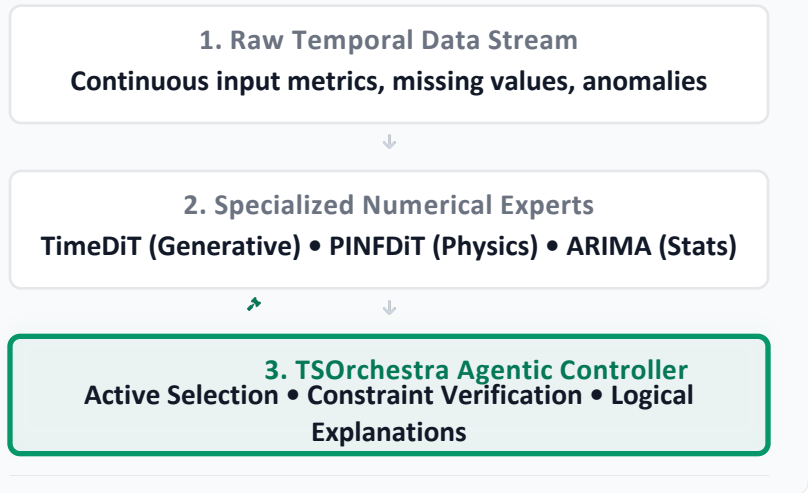
Agent systems theory forms the final metacognitive control layer of the temporal stack.

- ✓ **Decision Loop:** Moves from static prompting to multi-step execution over feedback.
- ✓ **Harness Power:** Delegates intensive forecasting computations to specialized models.
- ✓ **Systemic Reflection:** Stores diagnostic failure trails as persistent episodic memory.
- ✓ **Evaluated Success:** Assesses forecasting metrics alongside execution safety and logic.

TSOrchestra Architecture Paradigm

The language agent never forecasts raw numbers; it orchestrates and audits numerical evidence.

Orchestrated Forecasting Infrastructure



Shinn et al. "Reflexion." 2023.
• Wang et al. "Voyager." 2023.
• Liu et al. "AgentBench." 2023.

Motivation: The Gap in Temporal AI

LLMs reason, but hard to forecast

- Language pretraining lacks temporal bias and numerical precision.

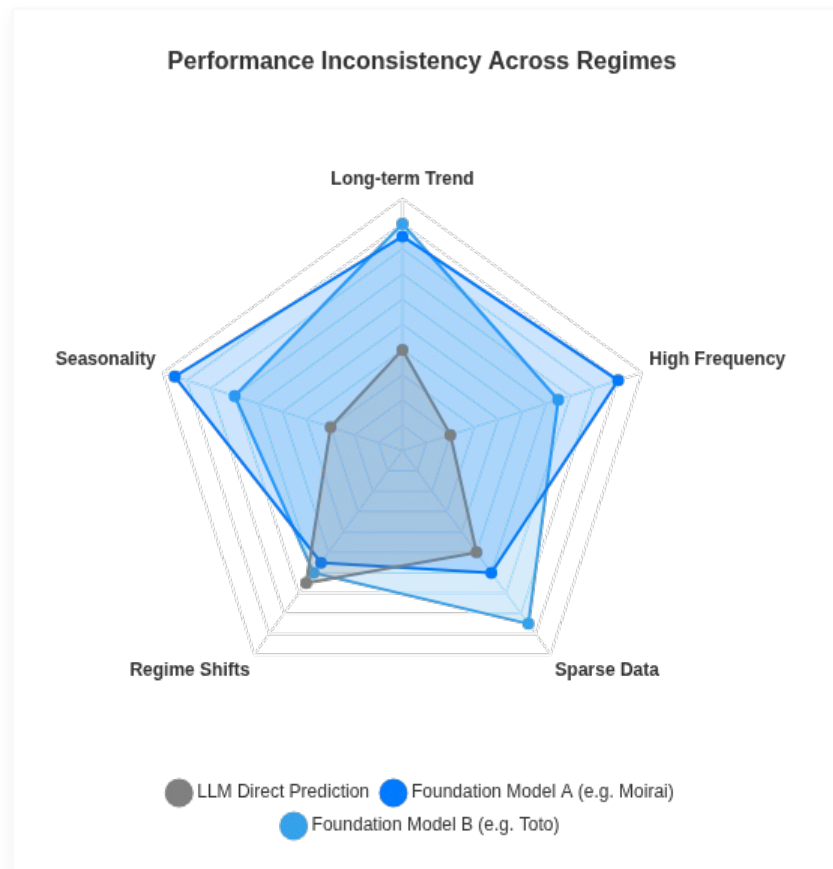
No single foundation model wins

- Moirai-2, Sundial, Toto each win some regimes—none dominates.

Real-world regimes keep shifting

- Non-stationarity moves the best model over time; static picks fail.

Goal: orchestrate models with interpretable, forward-looking control.



No Single Temporal Model Wins across All Regimes

Performance depends on domain, horizon, frequency, metric, missingness, regime shift, and constraints.

⚡ Short Horizon Success

A lightweight linear model can win on short horizon electricity demand...

⚠ Long Horizon Failure

...but fail on long-horizon traffic, irregular health logs, or rare financial shocks.

Main Takeaway

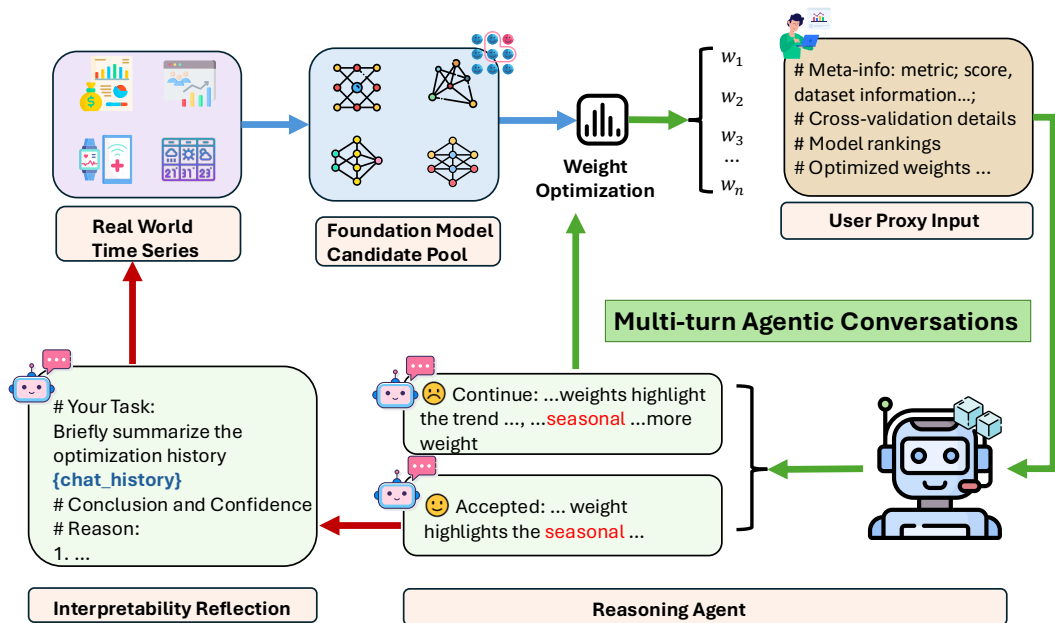
Temporal heterogeneity makes model complementarity inevitable.

Model Reliability Matrix Across Regimes

Domain	Short Horizon	Long Horizon	Regime Shifts
Electricity (Power)	DLinear	PatchTST	Chronos
Traffic Flows	PatchTST	Moirai	ARIMA-g
Healthcare Logs	TimesFM	LAG-Llama	Moirai

Makridakis et al. "The M4 Competition." 2018.
Aksu et al. "GIFT-Eval." 2024.

Core Concept: Intelligent Judge



Metacognitive Control

- Acts as a **controller**, not a predictor
- Reads forecasts, CV metrics, domain priors → **strategic weight updates**



Iterative Refinement

- **Multi-turn reasoning** tests hypotheses (“Is this shift spurious?”)
- Balances specialists & anticipates regime shifts — with **causal explanations** before final weights

Post-Training: R1-Style Finetuning

Numerical optimization finds a solution. Reasoning decides whether the solution is safe.



Phase 1: Protocol SFT

Objective

Learn decision protocols & boundary calibration from expert trajectories + synthetic edge cases.

Next-Token Loss

$$L_{\text{SFT}} = -\sum \log p_{\theta}(y_t | y_{<t}, x)$$

Phase 2: GRPO (RL)

Objective

Actor-only RL refines the reasoning policy; KL regularization keeps it stable.

Composite Reward

$$r = \text{clip}(r_{\text{base}} + \alpha \cdot r_{\text{conf}} + \beta \cdot r_{\text{faith}}, 0, 1)$$

Phase 3: Inference Loop

Objective

Multi-turn reasoning steers the SLSQP optimizer with interpretable outputs.

JSON Output

```
{ "confidence": 0.85, "continue": false, "explanation": "..."
```

Innovation: *SHAP-based*

Faithfulness Rewards

Ground the agent's reasoning in temporal physics — not linguistic plausibility — using causal interventions.

ALIGNMENT MECHANISM



Agent Explanation

"Trend is rising..."



SHAP Values

Physical Contribution

Pearson Correlation Reward

$$r_{\text{faith}} = \text{PCC}(\text{CE}, \text{EE})$$

⚠ The Hallucination Trap

- LLMs invent **plausible but false** rationales — e.g. “captures seasonality” when it tracks noise
- Reasoning becomes **post-hoc rationalization**, not a real driver — unreliable in regime shifts

⚖ Causal Faithfulness (SHAP)

- Measure **Causal Effects (CE)** of Trend / Seasonality / Residual via SHAP counterfactuals
- Reward **$r_{\text{faith}} = \text{corr}(\text{Explanation}, \text{true CE})$** — forces the words to match the numbers

Experimental Results: GIFT-Eval Benchmark

Validated on **23 datasets / 97 settings** — orchestrates Moirai-2, Sundial, Toto & TabPFN-TS via SLSQP.

State-of-the-Art Ranking

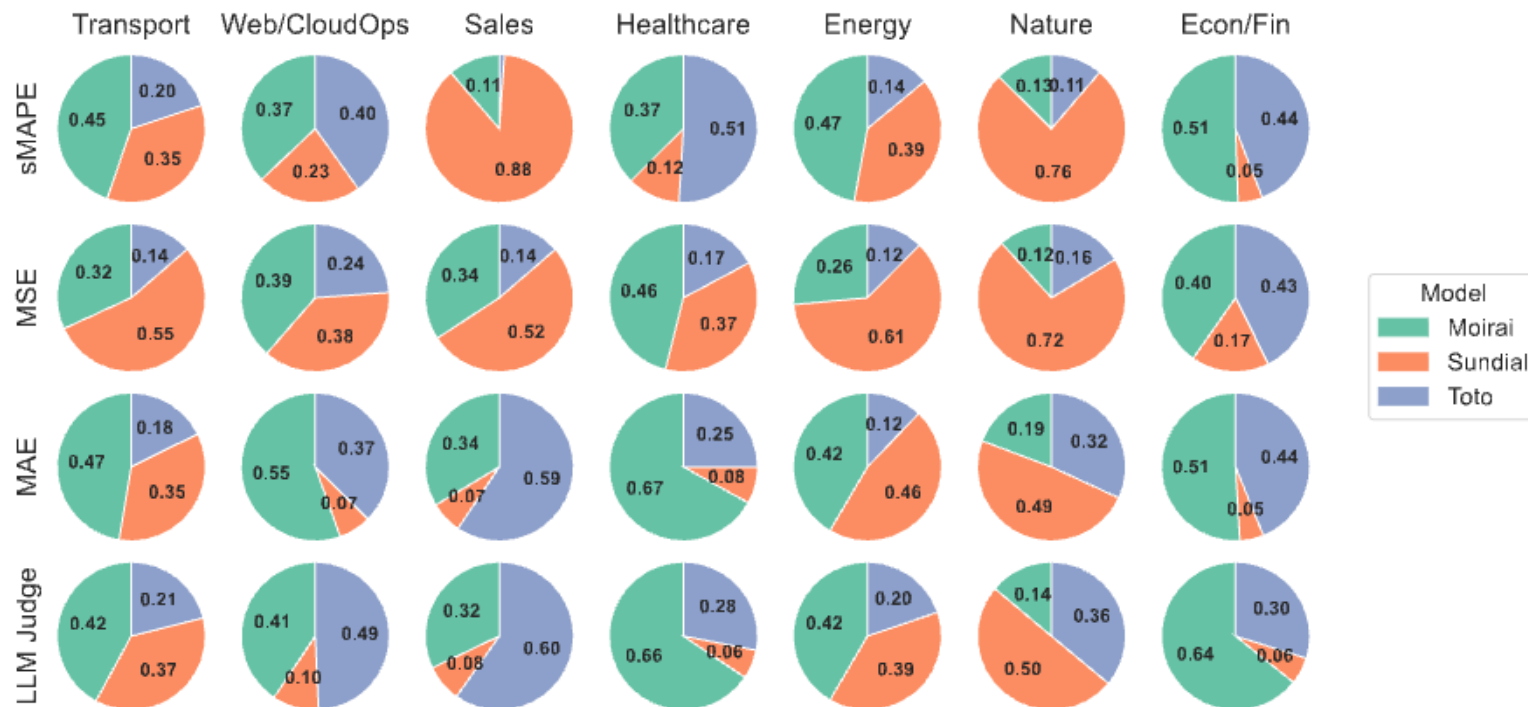
Efficiency at Scale

Candidate Pool Diversity

T	model	Organization	Test Leak.	Replication Code	MASE	MASE_Rank	CRPS	CRPS_Rank
●	TSOrchestra (code)	Melady Lab @ USC	No	No	0.677	8.505	0.468	7.948
●	DeOSAlphaTimeGPTPredict	vencortex*	No	No	0.682	8.629	0.466	8.021
●	MoiraiAgent	Salesforce AI Res	No	No	0.679	9.495	0.474	11.825
●	Samay	Kairosity	No	No	0.698	11.794	0.477	10.134
●	Synapse	Google Cloud AI	No	No	0.699	12.216	0.479	10.33
●	TimeCopilot (code)		No	Yes	0.705	12.33	0.483	11.392
●	Migas-1.0	Syntheyfy	No	No	0.697	12.588	0.598	38.454
●	Kairos-1.0	ContinualIST	No	No	0.702	12.99	0.482	11.856
●	Chronos-2 (code)	AWS	No	Yes	0.698	13.392	0.485	13.742
●	TimesFM-2.5 (code)	Google Research	No	Yes	0.705	15.979	0.49	15.567
●	TiReX (code)	NXAI	No	Yes	0.716	17.01	0.488	13.969
●	Xihe-ultra	Ant	No	No	0.701	18.278	0.488	17.629

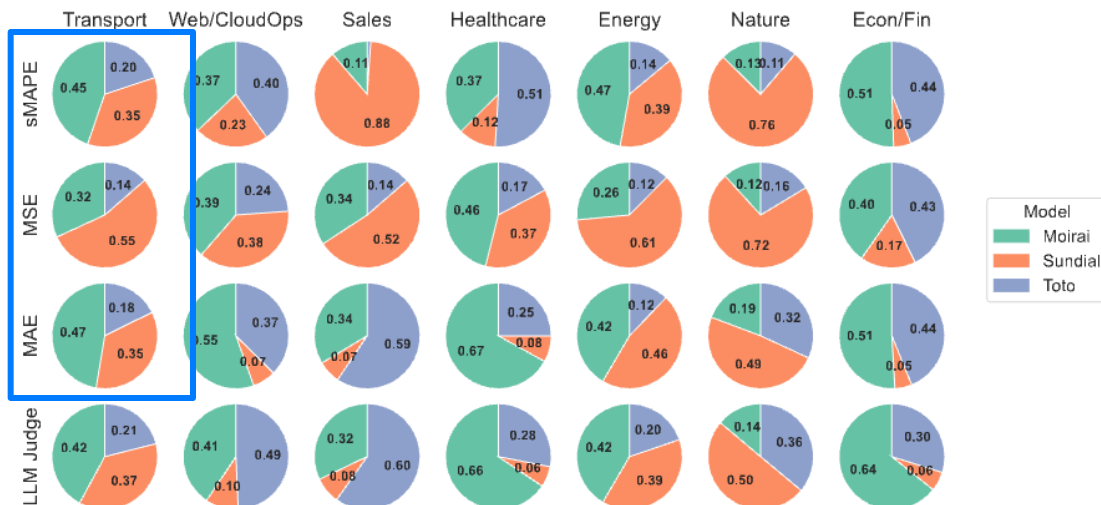
Analysis: The Illusion of a “Best” Single Model

Why dynamic agentic orchestration outperforms static statistical metrics.



Analysis: The Illusion of a “Best” Single Model

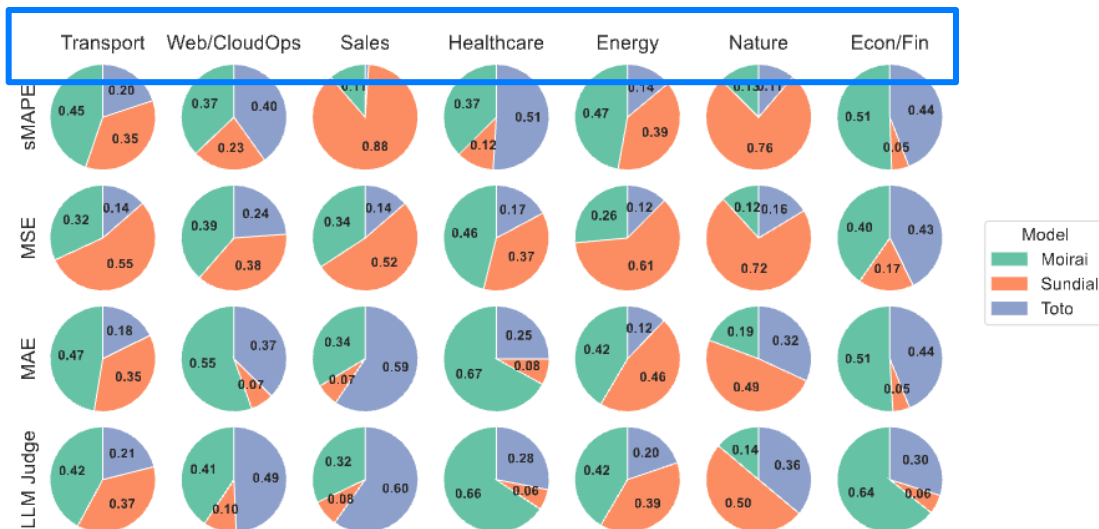
Why dynamic agentic orchestration outperforms static statistical metrics.



- **Metric Contradiction** — metrics disagree: winning on MSE $\neq$ winning on MAE in the same domain.

Analysis: The Illusion of a “Best” Single Model

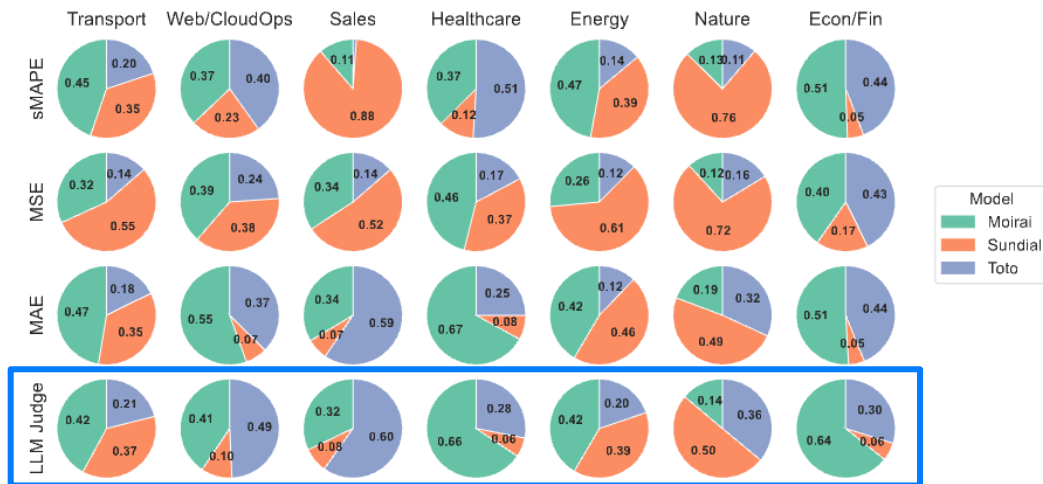
Why dynamic agentic orchestration outperforms static statistical metrics.



- **Metric Contradiction** — metrics disagree: winning on MSE $\neq$ winning on MAE in the same domain (see Sales).
- **Domain Fragmentation** — no model wins everywhere; results split across Healthcare, Energy & Finance.

Analysis: The Illusion of a “Best” Single Model

Why dynamic agentic orchestration outperforms static statistical metrics.



- **Metric Contradiction** — metrics disagree: winning on MSE $\neq$ winning on MAE in the same domain (see Sales).
- **Domain Fragmentation** — no model wins everywhere; results split across Healthcare, Energy & Finance.
- **The LLM Judge** — instead of a winner-takes-all pick on one flawed metric, the agent is a meta-evaluator.
- **Dynamic Orchestration** — shifts ensemble weights per regime for robust decisions across all environments.

Conclusion: Static metrics can't pick one model — we must dynamically orchestrate a portfolio of specialists.

Closing: From Temporal Foundation Models to Decision Intelligence

Stepping back: what the whole stack means for systems that evolve

01

The System View

One stack, four interfaces:
representation, generation,
grounding, orchestration.

02

Solved — and Open

The four gaps we closed, and the
research challenges that remain.

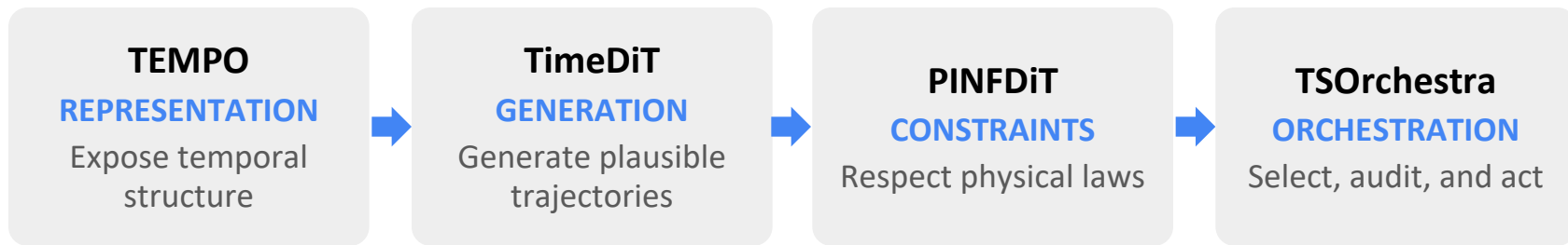
03

The Takeaway

In real systems, the output is not a
prediction — it is a decision under
uncertainty.

One Stack, Four Interfaces

Everything we built today, in one picture — from data to decision



“Temporal intelligence emerges not from one model, but from the coordination of representation, generation, grounding, and orchestration.”

What We Solved

Each layer closes a specific gap in the temporal stack

Structure Alignment

TEMPO: decomposition aligns continuous signals with pretrained backbones.

Unified Conditional Generation

TimeDiT: forecasting, imputation, and synthesis become one masked generation problem.

Physical Grounding

PINFDiT: domain constraints injected at inference time — no retraining.

Decision Orchestration

TSOrchestra: heterogeneous experts coordinated under deployment shift.

“We move from forecasting accuracy to reliable temporal inference.”

CLOSING · OPEN PROBLEMS

Open Research Problems

This stack opens a research direction — it does not close one

Evaluation — measuring reliability beyond average error

Causality — separating correlation, intervention, and mechanism

Multimodal Context — combining series with text, images, logs, graphs, and events

Continual Adaptation — updating systems safely when regimes change

Human-in-the-Loop — letting experts inspect, override, and trust model outputs

“Reliability is not a solved metric; it is an ongoing system property.”

Why This Matters for Real-World Decisions

Back to where we started: health, climate, energy, and science

Energy Grids

Forecast demand, respect capacity, dispatch robustly.

Healthcare

Impute missing vitals, quantify uncertainty, detect deterioration.

Climate & Science

Generate physically consistent long-horizon trajectories.

Finance & Operations

Adapt to regime shift and choose which expert to trust.

“In real systems, the final output is not a prediction. It is a decision under uncertainty.”

Thank You — Questions?

