From Guesswork to Guarantee: Towards Faithful Multimedia Web Forecasting with TimeSieve

Songning Lai*
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
songninglai@hkust-gz.edu.cn

Ninghui Feng*
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
ninghuifeng@hkust-gz.edu.cn

Jiechao Gao*
Stanford University
Center for SDGC, CA, USA
jiechao@stanford.edu

Hao Wang
Carnegie Mellon University
Pittsburgh, PA, USA
haow2@alumni.cmu.edu

Haochen Sui
University of Michigan
Ann Arbor, USA
hcsui@umich.edu

Xin Zou
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
xinzou@hkust-gz.edu.cn

Jiayu Yang
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
xinzou@hkust-gz.edu.cn

Wenshuo Chen
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
xinzou@hkust-gz.edu.cn

Lijie Hu
MBZUAI
Abu Dhabi, AE
lijie.hu@kaust.edu.sa

Hang Zhao
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
hangzhao@hkust-gz.edu.cn

Xuming Hu
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
xuminghu@hkust-gz.edu.cn

Yutao Yue†
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
Jiangsu Industrial Technology Research Institute
Jiangsu, China
yutaoyue@hkust-gz.edu.cn

Abstract

The domain of time series forecasting has gained significant attention due to its critical applications in multimedia-rich web traffic (including video streaming workloads and dynamic content delivery) and cross-platform advertisement click predictions, which are essential for web operations planning. While models like TimeSieve have demonstrated strong capabilities in predicting web visitation metrics, they suffer from critical unfaithfulness issues, including sensitivity to random seeds, input noise, layer noise, and parametric perturbations. To address these limitations, we propose **Faithful TimeSieve (FTS)**, an enhanced framework designed to improve prediction reliability and robustness. Our approach systematically detects and mitigates unfaithfulness in TimeSieve, significantly enhancing its stability and consistency. Experimental results demonstrate that FTS substantially improves the model's faithfulness, setting a new standard for temporal forecasting methods. This advancement not only increases TimeSieve's reliability but also contributes to more robust temporal modeling, particularly crucial for web traffic forecasting where prediction accuracy directly impacts operational decisions. Our work thus represents a significant step toward more dependable time series predictions in web-related applications.

*These three authors contributed equally to this research.
†Corresponding author.

MM '25, Dublin, Ireland.

ACM ISBN 979-8-4007-2035-2/2025/10
https://doi.org/10.1145/3746027.3754963

CCS Concepts

• **Computing methodologies → Regularization**; • **Information systems → Data mining**; **Multimedia information systems**.

Keywords

Media Data Prediction, Time Series Forecasting, Web Traffic Prediction, Unfaithfulness Mitigation, Model Stability

ACM Reference Format:
Songning Lai, Ninghui Feng, Jiechao Gao, Hao Wang, Haochen Sui, Xin Zou, Jiayu Yang, Wenshuo Chen, Lijie Hu, Hang Zhao, Xuming Hu, and Yutao Yue. 2025. From Guesswork to Guarantee: Towards Faithful Multimedia Web

Forecasting with TimeSieve. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25), October 27–31, 2025, Dublin, Ireland.* ACM, New York, NY, USA, 17 pages. https://doi.org/10.1145/3746027.3754963

1 Introduction

Time series forecasting has become indispensable in modern multimedia ecosystems, particularly for analyzing web traffic patterns that involve video streaming behaviors, real-time content updates, and heterogeneous user interactions [9, 11]. These predictions are crucial for optimizing multimedia content delivery networks, where accurate predictions of website visitation patterns are crucial for effective resource allocation and business strategy [4, 8, 10, 12]. The rapid evolution of time series models, including advanced architectures like TimeSieve [5], has significantly improved forecasting capabilities. However, while these models excel in predictive performance, there is a growing need to address their robustness, stability, and faithfulness, aspects that are often overlooked but critical for real-world applications like web traffic prediction. Unstable models can lead to unreliable forecasts, potentially causing mismanagement of resources and missed opportunities.

TimeSieve, with its unique integration of wavelet, transforms, and information bottleneck theory, has demonstrated superior performance compared to contemporaries like LightTS [1] and Autoformer [30]. However, our investigation revealed that TimeSieve's stability and faithfulness are vulnerable to perturbations, with significant variations across different training runs and sensitivity to input noise. This inconsistency can undermine the trustworthiness of its predictions, especially in the context of web traffic forecasting, where even minor fluctuations can have significant consequences.

Our study reveals TimeSieve's vulnerability to perturbations, with substantial performance variations across different training runs due to random seed changes, reaching up to 50% differences. Furthermore, introducing minor input perturbations led to a performance decline of around 30%, highlighting the model's sensitivity to noise and questioning its faithfulness. To illustrate this issue, Figure 1 compares the performance of TimeSieve (TS) and Faithful TimeSieve (FTS) trained with ten distinct random seeds.

Motivated by these findings, we chose TimeSieve due to its simplicity and effectiveness, providing a suitable platform to investigate faithfulness issues. We aim to define a **Faithful TimeSieve (FTS)** and develop a framework that can be generalized to other time series models. Our focus on TimeSieve allows us to establish a foundation for understanding faithfulness in complex models, with our team actively working on extending this concept to a broader framework. Our key contributions are as follows:

(1) **Comprehensive Faithfulness Assessment:** We perform an in-depth analysis of TimeSieve, identifying the challenges and factors that affect its faithfulness, and providing valuable insights into the model's limitations.

(2) **Definition of Faithful TimeSieve:** We propose a rigorous definition for FTS, encompassing three critical attributes that ensure the model's robustness and stability. This definition serves as a guideline for evaluating and enhancing TimeSieve's faithfulness.

(3) **Multimedia-aware Robustness Framework:** We develop a novel framework incorporating content-adaptive stabilization strategies to address faithfulness issues while maintaining predictive performance. This framework is designed to improve TimeSieve's fidelity by adhering to the defined FTS attributes. In particular, FTS can be easily transferred to all time series models, showing strong scalability.

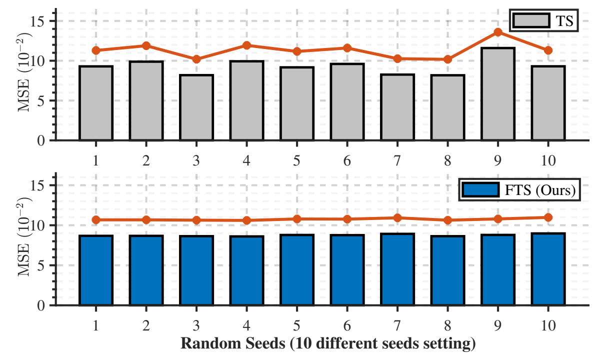

Figure 1: Ten different random seeds are selected to train TimeSieve (TS) and Faithful TimeSieve (FTS) respectively.

(4) **Theoretical and Experimental Validation:** We provide mathematical proofs and experimental evidence to demonstrate the effectiveness of our framework in improving TimeSieve's faithfulness. These results validate the practical utility and theoretical soundness of our approach. FTS implements SOTA on multiple datasets and shows strong robustness.

2 Related Work

Time Series Forecasting. Recent years, significant advancements have been made in the field of time series forecasting, leading to the development of several effective forecasting models. Many of these innovative models leverage Multi-Layer Perceptrons (MLPs) to harness their powerful capabilities. Examples of such models include the MSD-Mixer [38], DLinear [34], and FreTs [31]. These MLP-based models utilize sophisticated data manipulations and learning strategies to improve their forecasting performance. Meanwhile, Transformer have also gained considerable prominence in time series forecasting. Models such as PatchTST [24] and the memory-efficient Informer [40] have proven to be highly effective in capturing temporal dependencies through advanced data transformation techniques. These Transformer-based models excel at extracting meaningful patterns from time series data by leveraging their ability to model long-range dependencies and capture intricate relationships. Recent work - TimeSieve, Feng et al.[5] have introduced an innovative approach that integrates wavelet transform with contemporary machine learning methodologies and information bottleneck theory for time series forecasting. The experimental evaluation of TimeSieve demonstrates its effectiveness compared to existing models. By evaluating the model on diverse time series datasets, TimeSieve presents significant improvements in forecasting accuracy and robustness. These results showcase the potential of integrating wavelet transform and information bottleneck theory with contemporary machine learning techniques, opening up new avenues for advancing time series forecasting research.

Faithful Time Series. The challenge of achieving faithful time series prediction, which is crucial for numerous applications, has been a longstanding issue [6]. Previous research has tackled this problem by proposing more faithful conventional prediction methods [16, 17, 19, 21, 41] based on auto-regression, aiming to mitigate the impact of noise. Another line of work focuses on the training data [28] and explores techniques such as data perturbation to alleviate bias [32]. In recent years, deep learning has gained attention for time series prediction. However, deep learning models are often sensitive and prone to overfitting, particularly in the presence of outliers. Recent studies propose various improvements in model training to address these challenges. Approaches such as teacher forcing [26], specialized loss functions [13], model ensembles [14], as well as data denoising [23], data augmentation [7], data decomposition [36], and data compensation [39] have been explored to enhance model generalization and train models with more reliable datasets. Among these efforts, Cheng et al. [3] proposed DARF, which leverages adversarial learning to capture correlations across multiple time series and reduce data bias. Zhang et al. [37] introduced localized stochastic sensitivity (LSS) minimization for Recurrent Neural Networks (RNNs) to reduce output sensitivities. TimeX [25] presents a time series model that enhances model faithfulness by introducing model behavior consistency. Yu et al. [33] propose a decomposition-based Transformer to improve the predictability of Transformers. These studies demonstrate the significant potential of learning-based models in robust time series prediction. However, they lack a unified framework and do not explicitly focus on the definition of "faithfulness" itself, which poses challenges in evaluation and interpretation. To address these limitations, our work aims to provide a comprehensive and unified framework for achieving faithful time series prediction. We recognize the importance of visual attention on the definition of faithfulness, enabling a more thorough understanding and evaluation of the models. By integrating key insights from previous works and leveraging advanced techniques, our proposed framework aims to enhance the faithfulness of time series prediction in a principled and interpretable manner.

3 Method

3.1 Preliminaries: TimeSieve [5]

TimeSieve is a novel time series forecasting model that combines the strengths of wavelet transform and the information bottleneck theory to enhance prediction performance.

Wavelet Decomposition and Reconstruction. The Wavelet Decomposition Block (WDB) decomposes the input time series $x(t) \in \mathbb{R}^{T\times C}$ into π_a and π_d using wavelet transform, where the approximation coefficients π_a representing the low-frequency trends, and the detail coefficients π_d representing the high-frequency details:

$$\pi_a = \int_{-\infty}^{\infty} x(t)\phi(t)dt\ ,\quad \pi_d = \int_{-\infty}^{\infty} x(t)\psi(t)dt\ , \tag{1}$$

where $\phi(t)$ and $\psi(t)$ are the scaling function and wavelet function, respectively, defined as follows:

$$\phi(t+1) = \sum_{k=1}^{N} a_k\phi(2t-k)\ ,\quad \psi(t+1) = \sum_{k=1}^{M} b_k\phi(2t-k)\ , \tag{2}$$

Here, a_k and b_k are the filter coefficients for the scaling and wavelet functions, and k is the index of the coefficients.

This decomposition allows for the extraction of both trends and details from the data. The WRB then reconstructs the time series from the processed coefficients and the reconstructed time series is passed through a simple MLP network for the final prediction:

$$y = MLP(\sum \hat{\pi_a}\phi(t) + \sum \hat{\pi_d}\psi(t))\ , \tag{3}$$

where $\hat{\pi_a}$ and $\hat{\pi_d}$ denote the filtered approximation and detail coefficients respectively.

Information Filtering and Compression. Information Filtering and Compression Block (IFCB) employs the IB principle to filter noise and retain essential information. The objective is to minimize the mutual information $I(\pi_i; z)$ while maximizing $I(z; \hat{\pi_i})$, where π_i represents the input coefficients, $\hat{\pi_i}$ is the filtered coefficient, and z is the intermediate hidden layer and use $\hat{\pi_i}$, where $i \in \{a, d\}$. The optimization problem is formulated as:

$$\min\{I(\pi_i; z) - \beta \cdot I(z; \hat{\pi_i})\}\ , \tag{4}$$

where β is the trade-off parameter. The IFCB uses a DNN with a Gaussian distribution for $p(z|i)$ and a decoder function $f(z; \theta_d)$ to predict $\hat{\pi_i}$. The loss function for training combines the original prediction loss and the IB loss:

$$\mathcal{L}_{\text{IB}} = D_{KL}[\mathcal{N}(\mu_z, \Sigma_z)\ ||\ \mathcal{N}(0, I)] + D_{KL}[p(z)\ ||\ p(z|i)]\ , \tag{5}$$

$$\mathcal{L} = \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{IB}}\ , \tag{6}$$

where $\mathcal{L}_{\text{reg}}$ is the regression loss, and $\mathcal{L}_{\text{IB}}$ is the IB loss.

The contemporary SOTA model TimeSieve, has made significant strides in time series forecasting by effectively harnessing wavelet transform for multi-scale feature extraction and the IB method for noise reduction, but they are not without limitations. Despite their improved predictive performance, this model exhibits a certain fragility when confronted with external disturbances or unforeseen perturbations in the data and parameters. This lack of robustness and stability can lead to suboptimal performance, particularly in dynamic environments where data characteristics may shift with random seeds and other perturbations.

3.2 Our Research Paradigm Formulation

To ensure systematic investigation, we establish a rigorous research paradigm comprising five interconnected phases: 1 **Problem Identification** - systematically identify and formally define the core research challenge through empirical observation and domain-specific constraint analysis; 2 **Solution Formulation** - develop theoretically-grounded resolution strategies with measurable success criteria; 3 **Framework Construction** - establish a unified architecture integrating theoretical principles and operational mechanisms; 4 **Analytical Verification** - conduct rigorous mathematical derivation and analytical proof of the framework's solution capability; 5 **Empirical Validation** - implement experimental protocols to test practical feasibility. This phased approach ensures methodological integrity through bidirectional verification, where theoretical predictions inform experimental design while empirical results refine theoretical assumptions, forming a closed-loop validation system that preserves original research intentions while enhancing analytical depth.

3.3 Faithfulness Issues in TimeSieve

1 The focus of our study lies in addressing the faithfulness challenges specific to TimeSieve, one of the state-of-the-art time series forecasting models. While TimeSieve excels in predictive accuracy, its sensitivity to random seeds, input perturbations, and parameter perturbations reveals a lack of faithfulness that can limit its reliability in practical applications, such as web traffic prediction. By concentrating on TimeSieve, we aim to develop a targeted solution

that enhances its faithfulness, with the ultimate goal of providing a case study for improving the trustworthiness of similar models within the time series domain.

3.4 The Concept of "faithful TimeSieve forecasting"

2 In the context of our work, *faithful TimeSieve forecasting* refers to the ability of TimeSieve to consistently and accurately capture the temporal dynamics of a time series while maintaining robustness under various conditions. This notion emphasizes the model's resilience to input perturbations, like fluctuations in web traffic, and parameter perturbations, including those introduced by random seeds. By ensuring faithfulness in TimeSieve, we aim to stabilize its predictions over time, making its explanations reliable.

In web traffic prediction, a faithful TimeSieve would adapt to minor variations in user behavior and consistently extract meaningful patterns from the data, regardless of the initial conditions. Our approach highlights the importance of generating stable predictions within the TimeSieve framework, even in the presence of changes in input data or model parameters. This stability is crucial for maintaining trust in the model's forecasts, as it guarantees the consistency of its understanding of temporal patterns.

By focusing on enhancing the faithfulness of TimeSieve, we aim to demonstrate a method that can be adapted and applied to other time series models, contributing to the development of more reliable forecasting tools. This targeted approach not only improves the accuracy of predictions for TimeSieve but also sets a precedent for future research on enhancing faithfulness in the broader time series forecasting landscape.

To address the issues of faithfulness in TimeSieve, a precise definition is required: what constitutes a **Faithful TimeSieve (FTS)**? We propose that a FTS encompasses three key attributes:

(i) **Similarity in IB Space (Sib).** This attribute pertains to preserving essential information between the original time series and its filtered representation in the IB space. The filtering process involves applying the TimeSieve model to the original coefficients, resulting in filtered coefficients denoted as $\hat{\pi}_i$. To satisfy this attribute, the overlap between the original coefficients π_i and the disturbed coefficients $\hat{\pi}_i$, denoted as $D_1(\hat{\pi}_i, \pi_i)$, must exceed a predefined threshold β. This ensures that the filtering process retains the crucial information necessary for accurate predictions.

(ii) **Consistency in Prediction Space (Cps).** This attribute focuses on the quality of predictions made using the original weights ω compared to those made using the fine-tuned weights $\tilde{\boldsymbol{\omega}}$. The forecasts, denoted as $y(x(t), \omega)$ and $y(x(t), \tilde{\boldsymbol{\omega}})$, respectively, are based on the original and fine-tuned outputs. The attribute requires that the difference between these two forecasts, measured by a suitable distance metric D donated as D_2, is bounded by a constant α_1. In other words, the predictive accuracy of the TimeSieve model should not significantly degrade when using the fine-tuned weights instead of the original weights.

(iii) **Stability in Noise Perturbations (Snp).** This attribute addresses the robustness of the TimeSieve model's predictions of perturbations. It ensures that the model's forecasts remain stable when perturbed within a certain range. Specifically, the difference between the forecast $y(x(t), \tilde{\boldsymbol{\omega}})$ based on the original outputs and the forecast $y(x(t)+\delta, \tilde{\boldsymbol{\omega}})$ when the input is perturbed by a small amount δ, should be bounded by a constant α_2. This attribute guarantees that the TimeSieve model's predictions exhibit limited sensitivity to perturbations in the input or the model's internal state, thus maintaining stability.

3.5 Defination for FTS

Definition 3.1 (3 Faithful TimeSieve). A TimeSieve model is considered $(\boldsymbol{\alpha_1}, \boldsymbol{\alpha_2}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{R_1}, \boldsymbol{R_2})$-Faithful if it satisfies the following three attributes for any input $x(t)$ and we want to get $\tilde{\boldsymbol{\omega}}$ which is a Faithful TimeSieve's weights:

- **(i) Similarity in IB Space (Sib):** $D_1(\hat{\pi_a}(x(t)), \hat{\pi_a}(x(t)+\boldsymbol{\delta})) \le \beta$ for all $\|\boldsymbol{\delta}\| \le R_1$ and $D_1(\hat{\pi_d}(x(t)), \hat{\pi_d}(x(t)+\boldsymbol{\delta})) \le \beta$ for all $\|\boldsymbol{\delta}\| \le R_1$, where D_1 represents some probability distance or divergence, $\|\cdot\|$ denotes a norm and $R_1 \ge 0$ for all $\|\boldsymbol{\delta}\| \le R_1$.
- **(ii) Consistency in Prediction Space (Cps):** $D_2(y(x(t), \tilde{\boldsymbol{\omega}}), y(x(t), \boldsymbol{\omega})) \le \alpha_1$ for some $\alpha_1 \ge 0$, where D_2 is some probability distance or divergence, $\boldsymbol{\omega}$ is the weights of the original TimeSieve and $\tilde{\boldsymbol{\omega}}$ is the weights of the fine-tuned model.
- **(iii) Stability in Noise Perturbations (Snp):** $D_3(y(x(t), \tilde{\boldsymbol{\omega}}), y(x(t)+\boldsymbol{\delta}, \tilde{\boldsymbol{\omega}})) \le \alpha_2$ for all $\|\boldsymbol{\delta}\| \le R_2$, where D_3 is probability distance or divergence, $\|\cdot\|$ is a norm and $R_2 \ge 0$,

A TimeSieve model, characterized by its unwavering fidelity, is distinguished by its sustained predictive accuracy and stability under perturbations in input data or intrinsic model dynamics.

Forecasting Consistency in TimeSieve. The similarity between forecasts from original and fine-tuned coefficients is quantified by α_1, measuring the distance D between $y(x(t), \tilde{\boldsymbol{\omega}})$ and $y(x(t), \boldsymbol{\omega})$. An ideal scenario is $\alpha_1 = 0$, indicating identical forecasts. Our goal is to minimize α_1 for high forecast consistency.

Stability in TimeSieve. The stability criterion is defined by R_2 and α_2, with R_2 being the robustness radius and α_2 being the stability level. A model is highly stable if $R_2 = \infty$ and $\alpha_2 = 0$, indicating immunity to perturbations. Practically, we seek large R and small α_2 for robustness. In essence, Definition 3.1 offers a holistic framework for FTS, encompassing forecast similarity, stability, and accuracy in time series forecasting.

Definition 3.2 (μ-Rényi divergence). Given two probability distributions P and Q, and $\alpha \in (1, \infty)$, the μ-Rényi divergence $D_\mu(P||Q)$ is defined as $D_\mu(P||Q) = \frac{1}{\mu-1} \log \mathbb{E}_{x\sim Q}(\frac{P(x)}{Q(x)})^\mu$.

Rényi divergence [27] is a generalization of the traditional Kullback-Leibler divergence that can be controlled with a parameter μ. It measures the difference between probability distributions. When $\mu \to 1$, Rényi divergence tends towards Kullback-Leibler divergence, and when $\mu = 0$ or $\mu = \infty$, Rényi divergence becomes the minimum and maximum divergence, respectively.

The next two theorems will state properties in IB Space, namely Similarity in IB Space. Theorem 3.3 begins with a rough estimate of the similarity from which we can estimate an upper bound on the fluctuations in IB Space for any IB Space and any input. Theorem 3.4 derives a lower bound on the fluctuations in IB Space in the condition is satisfied, it can be shown that our model is stable.

Theorem 3.3 (4 Upper bound for sib). *If function $f(\cdot)$ is a decoder function, which is considered under $(\boldsymbol{\alpha_1}, \boldsymbol{\alpha_2}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{R_1}, \boldsymbol{R_2}, \|\cdot\|)$-Faithful TimeSieve, then if*

$$\beta \le \|\hat{\pi}_i(x(t)) - \hat{\pi}_i(x(t)+\boldsymbol{\delta})\|,$$

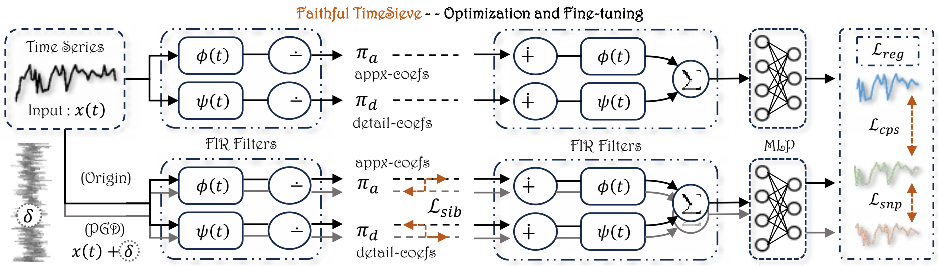

Figure 2: Framework of our proposed Faithful TimeSieve (FTS).

where $i \in [a, d]$, we have for all $x(t)' \triangleq x(t) + \boldsymbol{\delta}$ such that where $\|x(t) - x(t)'\| \leq R_1$,

$$\|f(z, \hat{\pi}_i; \theta_d) - f(z, \hat{\pi}_i{}'; \theta_d)\| \leq c\delta \,,$$

where $\hat{\pi}_i{}' \triangleq \hat{\pi}_i(x(t) + \boldsymbol{\delta})$ and c is a constant. The proof of Theorem 3.3 shows in Appendix B.

Theorem 3.4 (Lower bound for sib). *If function $f(\cdot)$ is a decoder function, which is considered under $(\boldsymbol{\alpha_1}, \boldsymbol{\alpha_2}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{R_1}, \boldsymbol{R_2}, \|\cdot\|)$-Faithful TimeSieve and IB Space expanded by measurements D_μ, $\|\cdot\|$. Then if $\delta \sim \mathcal{N}\left(0, \sigma^2\right)$, for any $\mu \in (1, \infty)$ and input $x(t)$ we have*

$$\sigma^2 \geq \max\left\{\delta, \frac{\mu R_1^2}{2\beta}\right\}.$$

The proof of Theorem 3.4 is shown in Appendix C. Theorem 3.4 indicates that similarity exists in IB Space for input $x(t)$ when σ^2 is large enough.

Equivalently, based on Theorem 3.3 and 3.4 we can also get the bound for CPS and SNP, we give these two properties without proof in the Appendix.

Theorem 3.5 (Consistency in prediction space). *If function $y(\cdot)$ is the function of the model, which is considered under $(\boldsymbol{\alpha_1}, \boldsymbol{\alpha_2}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{R_1}, \boldsymbol{R_2}, \|\cdot\|)$–Faithful TimeSieve , we have*

$$D_2(y(x(t), \tilde{\boldsymbol{\omega}}) - y(x(t), \boldsymbol{\omega}) \leq c\delta = \alpha_1 \,,$$

where c is a constant and D_2 could be a norm $\|\cdot\|$.

Definition 3.6 (Stable Diameter). In FTS, if R_1 and α_1 are bounded, the **Stable Diameter** R_2 is defined as

$$R_2 = \sup_{R \in \mathcal{R}} R \leq R_1, s.t. D_2(y(x(t), \tilde{\boldsymbol{\omega}}) - y(x(t), \boldsymbol{\omega}) \leq \alpha_1 \,,$$

where $\mathcal{R}$ is the set of all potential diameters.

Our FTS framework ensures SNP under the stable diameter as defined in Definition 3.6, supported by Theorem 3.7.

Theorem 3.7 (Stability in noise perturbations). *If function $y(\cdot)$ is the function of the FTS model, which is considered under $(\boldsymbol{\alpha_1}, \boldsymbol{\alpha_2}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{R_1}, \boldsymbol{R_2}, \|\cdot\|)$-Faithful TimeSieve expanded by measurements D_μ, $\|\cdot\|$. Then for all R in set $\mathcal{R}$, we have*

$$D_3(y(x(t), \tilde{\boldsymbol{\omega}}), y(x(t) + \boldsymbol{\delta}, \tilde{\boldsymbol{\omega}})) \leq \alpha_2, s.t. \|\boldsymbol{\delta}\| \leq R_2 \,,$$

where D_3 could be a norm $\|\cdot\|$, and $\alpha_2 \leq \alpha_1$, which is decided by stable diameter.

3.6 Faithful TimeSieve Framework

We have already presented a rigorous definition of FTS. To construct FTS, we formulate a minimax optimization problem incorporating three conditions as outlined in Definition 3.1. The definition enables us to establish an initial optimization problem, from which we derive the following comprehensive objective function (the framework is shown in Figure 2):

$$\begin{aligned}
&\max_{\|\boldsymbol{\delta}\| \leq R} \lambda_1(\beta - D_1(\hat{\pi}_a(x(t)), \hat{\pi}_a(x(t) + \boldsymbol{\delta}))) \\
&+ \max_{\|\boldsymbol{\delta}\| \leq R} \lambda_1(\beta - D_1(\hat{\pi}_d(x(t)), \hat{\pi}_d(x(t) + \boldsymbol{\delta}))) \\
&+ \min_{\tilde{\omega}} \mathbb{E}_x[\lambda_2(D_2(y(x(t), \tilde{\omega}), y(x(t), \omega)) - \alpha_1)] \\
&+ \max_{\|\boldsymbol{\delta}\| \leq R} \lambda_3(D_3(y(x(t), \tilde{\omega}), y(x(t) + \boldsymbol{\delta}, \tilde{\omega})) - \alpha_2).
\end{aligned} \tag{7}$$

The min-max optimization problem discussed involves hyperparameters λ_i, where $i \in [3]$.

Inspired by the Projected Gradient Descent (PGD) methodology proposed by Madry et al. [22], the optimization process involves iterative updates to $\boldsymbol{\delta}$. At the p-th iteration for updating the current noise $\boldsymbol{\delta}^*_{p-1}$, we perform the following steps:

$$\begin{aligned}
&\boldsymbol{\delta_p} = \boldsymbol{\delta^*_{p-1}} + \frac{\gamma_p}{|A_{p-1}|} \sum_{x \in A_{p-1}} \nabla_{\boldsymbol{\delta}^*_{p-1}} \\
&[D_1(\hat{\pi}_a(x(t)), \hat{\pi}_a(x(t) + \boldsymbol{\delta^*_{p-1}})) + D_2(y(x(t), \tilde{\boldsymbol{\omega}}), y(x(t), \boldsymbol{\omega})) \\
&+ D_3(y(x(t), \tilde{\boldsymbol{\omega}}), y(x(t) + \boldsymbol{\delta^*_{p-1}}, \tilde{\boldsymbol{\omega}}))] \,,
\end{aligned} \tag{8}$$

where $\boldsymbol{\delta^*_p} = \arg\min_{||\boldsymbol{\delta}|| \leq R} ||\boldsymbol{\delta} - \boldsymbol{\delta_p}||$ and A_{p-1} denotes a batch of samples, γ_p is the step size parameter for PGD, and R is the norm bound for the perturbation.

Once $\boldsymbol{\delta_P}$ is obtained after P iterations, we update $\tilde{\omega}^{t-1}$ to $\tilde{\omega}^t$ using batched gradients.

Finally, we have the following objective function:

$$\begin{aligned}
\min_{\tilde{\omega}} \mathbb{E}_x[\lambda_1 & \underbrace{(D_1(\hat{\pi}_i(x(t)), \hat{\pi}_i(x(t) + \boldsymbol{\delta})))}_{\mathcal{L}_{sib}, i \in [a,d]} \\
&+ \lambda_2 \underbrace{D_2(y(x(t), \tilde{\boldsymbol{\omega}}), y(x(t), \boldsymbol{\omega}))}_{\mathcal{L}_{cps}} \\
&+ \lambda_3 \underbrace{D_3(y(x(t), \tilde{\boldsymbol{\omega}}), y(x(t) + \boldsymbol{\delta}, \tilde{\boldsymbol{\omega}}))}_{\mathcal{L}_{snp}}].
\end{aligned} \tag{9}$$

We incorporate the three mentioned losses as auxiliary attention stability losses into the original TS model loss $\mathcal{L} = \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{IB}}$ for fine-tuning. Eventually, we obtain:

$$\mathcal{L} = \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{IB}} + \lambda_1 \cdot \mathcal{L}_{sib} + \lambda_2 \cdot \mathcal{L}_{cps} + \lambda_3 \cdot \mathcal{L}_{snp} \,, \tag{10}$$

where λ_1, λ_2 and λ_3 are regularizations of each loss function. The pseudocode for the faithful TimeSieve framework – Algorithm 1 is shown in Appendix E.

4 Experiments and Results

4.1 Settings

5 **Datasets.** We conduct extensive experiments to evaluate the efficacy of FTS. When it comes to multivariate forecasting, we employ a range of real-world benchmarks, such as ETT [40] and

Table 1: Forecasting results with different forecast lengths H ∈ {48, 96, 144, 192} . We set the lookback length T = 2H. Red indicates the best result, while Blue refers to the second best.

Models		FTS		TS		Koopa		PatchTST		TSMixer		DLinear		NSTformer		LightTS		Autoformer	
Metric		MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
wiki	48	0.305	0.467	0.323	0.490	0.314	0.496	0.312	0.495	0.318	0.490	0.350	0.517	0.307	0.508	0.313	0.494	0.315	0.522
	96	0.262	0.430	0.266	0.433	0.283	0.451	0.280	0.445	0.277	0.435	0.286	0.462	0.303	0.633	0.287	0.460	0.348	0.550
	144	0.268	0.445	0.270	0.447	0.284	0.459	0.280	0.451	0.320	0.496	0.287	0.458	0.309	0.521	0.291	0.471	0.360	0.616
	192	0.273	0.447	0.276	0.452	0.294	0.467	0.289	0.461	0.370	0.644	0.289	0.463	0.339	0.605	0.301	0.490	0.373	0.629
ETTh1	48	0.360	0.340	0.361	0.341	0.385	0.364	0.375	0.342	0.432	0.407	0.372	0.342	0.465	0.614	0.406	0.404	0.432	0.678
	96	0.383	0.376	0.384	0.376	0.411	0.406	0.395	0.377	0.473	0.466	0.395	0.380	0.498	0.653	0.431	0.435	0.496	0.578
	144	0.396	0.391	0.397	0.393	0.426	0.424	0.412	0.394	0.528	0.537	0.401	0.394	0.536	0.602	0.442	0.453	0.521	0.761
	192	0.406	0.402	0.408	0.402	0.434	0.430	0.437	0.416	0.592	0.642	0.416	0.408	0.543	0.684	0.457	0.471	0.568	0.598
Exchange	48	0.139	0.042	0.140	0.043	0.149	0.046	0.145	0.048	0.149	0.046	0.145	0.046	0.187	0.073	0.159	0.067	0.205	0.124
	96	0.196	0.084	0.197	0.086	0.211	0.092	0.204	0.090	0.211	0.092	0.223	0.089	0.294	0.159	0.247	0.168	0.778	0.409
	144	0.242	0.123	0.243	0.124	0.265	0.141	0.265	0.138	0.265	0.141	0.256	0.133	0.375	0.292	0.272	0.310	0.680	0.671
	192	0.287	0.170	0.292	0.179	0.329	0.212	0.298	0.181	0.329	0.212	0.301	0.182	0.464	0.494	0.354	0.403	0.979	0.544

Exchange [15]. Additionally, we utilize the Wiki pageviews dataset [29], which provides per-article and per-project view counts for all Wikimedia Foundation projects since May 2015. This dataset has been meticulously filtered to exclude as many spiders and bots as possible, ensuring that the data reflects genuine user interest and behavior. For our analysis, we specifically focus on the top ten hot word search rankings from 2015 to 2016, allowing us to delve into trends in user engagement and interest over time. We further validate FTS on other datasets in Appendix I.2 and observe similar improvements. Notably, rather than employing a fixed lookback window length, we adopt the strategy where the lookback window length T is set to twice the forecast window length (i.e., $T = 2H$). This methodology is grounded in the accessibility of historical data in practical scenarios, where leveraging a larger volume of observed data can significantly improve the model's performance, particularly for longer forecasts.

Random seeds. In the context of our random seeds experiments, we deliberately varied the initial random seed values to assess the impact on model training and subsequent performance. Specifically, we opted for a set of ten distinct seeds: 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, and 2030, and use 2021 as the based random seed. By utilizing these different seeds, we aimed to investigate the sensitivity of the trained models to variations in the random initialization process.

Input perturbations and intermediate layer perturbations. Our framework demonstrated certain faithfulness against input perturbations and intermediate layer perturbations. To prove this, we designed and conducted comparative experiments. In these experiments, we introduced a type of noise more suited to time series forecasting tasks. This noise is generated through sequence decomposition [30], and we selected the residual part as the noise source [38]. By introducing this noise, we are able to simulate disturbances commonly encountered in real-world time series tasks, thereby assessing the model's capability to handle specific disruptions in time series data. To further prove the model's robustness, we opted to directly add Gaussian noise to the parameters of the model's critical layer (IFCB layer). Specifically, we target the input of the model and the parameters of the IFCB layer and then embed noise directly into these inputs ($x(t)' = x(t) + \mathcal{N}(0, \sigma)$, with a perturbation of a certain radius σ) and parameters ($\theta'_{IFCB} = \theta_{IFCB} + \mathcal{N}(0, \sigma)$). This approach allows for a more direct assessment of the model's performance in the face of key parameter perturbations. More introductions can be seen in Appendix J and the reasons for this experimental setup and why adding noises in this way are given in Appendix D.

Setup. The hardware utilized for our experiments comprised an NVIDIA GeForce RTX 4090 GPU and an Intel(R) Xeon(R) E5-2686 v4 CPU, ensuring the computational efficiency required for intensive model training and evaluation. We trained our models over 10 epochs with a batch size of 32 and a learning rate of 0.0001 to optimize convergence without overfitting.

4.2 Performance under No Perturbations

In Table 1, we compare the forecasting performance of the proposed Faithful TimeSieve (FTS) model against several baseline models, including TimeSieve(TS) [5], Koopa [18], PatchTST [24], TSMixer [2], DLinear [35], Non-stationary Transformers (NSTformer) [20], LightTS [1], and Autoformer [30]. The comparison spans different datasets (ETTh1, Exchange, Wiki) and various prediction horizons ($H \in \{48, 96, 144, 192\}$). The best results are highlighted in red, while the second-best are marked in blue. The data in the table clearly demonstrate that FTS consistently outperforms existing TimeSieve models and other baseline methods in most scenarios, even without the addition of any perturbations. This validates FTS's ability to effectively optimize model performance, delivering more accurate and stable predictions without introducing extra disturbances.

Table 2: For 96-step forecasting on the Exchange dataset, FTS consistently achieves lower MSE and is less sensitive to random seeds than TS. We introduce a "Preference (%)" metric to quantify this combined advantage in performance and stability, using the 2021 seed as a baseline.

Random seed	TS	FTS	Preference(%)
2021*	0.0929	0.0868	NA
2022	0.0989	0.0867	**99.61%**
2023	0.0819	0.0864	**96.09%**
2024	0.0993	0.0860	**87.95%**
2025	0.0917	0.0878	**4.49%**
2026	0.0960	0.0876	**69.27%**
2027	0.0826	0.0892	**74.16%**
2028	0.0818	0.0863	**95.81%**
2029	0.1160	0.0879	**94.69%**
2030	0.0897	0.0898	**1.53%**

Table 3: Experimental results demonstrating the robustness of the proposed model under various perturbation conditions. The table compares the MAE and MSE of the model under different perturbation scenarios: No Perturbation (NP), No Perturbation with Optimization (NPO), Input Perturbation (IP), Input Perturbation with Optimization (IPO), Intermediate Layer Perturbation (ILP), and Intermediate Layer Perturbation with Optimization (ILPO).

Conditions		NP		NPO		IP		IPO		ILP		ILPO	
	Metric	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
Wiki	48	0.323	0.490	0.305 $\downarrow_{-0.018}$	0.467 $\downarrow_{-0.023}$	0.433	0.608	0.367 $\downarrow_{-0.066}$	0.531 $\downarrow_{-0.077}$	0.453	0.590	0.399 $\downarrow_{-0.054}$	0.551 $\downarrow_{-0.039}$
	96	0.266	0.433	0.262 $\downarrow_{-0.004}$	0.430 $\downarrow_{-0.003}$	0.405	0.575	0.321 $\downarrow_{-0.084}$	0.482 $\downarrow_{-0.093}$	0.367	0.681	0.329 $\downarrow_{-0.038}$	0.613 $\downarrow_{-0.068}$
	144	0.270	0.447	0.268 $\downarrow_{-0.002}$	0.445 $\downarrow_{-0.002}$	0.409	0.600	0.326 $\downarrow_{-0.083}$	0.542 $\downarrow_{-0.058}$	0.375	0.967	0.305 $\downarrow_{-0.070}$	0.723 $\downarrow_{-0.244}$
	192	0.276	0.452	0.273 $\downarrow_{-0.003}$	0.447 $\downarrow_{-0.005}$	0.408	0.580	0.327 $\downarrow_{-0.081}$	0.493 $\downarrow_{-0.087}$	0.404	1.548	0.366 $\downarrow_{-0.038}$	1.165 $\downarrow_{-0.383}$
ETTh1	48	0.361	0.341	0.360 $\downarrow_{-0.001}$	0.340 $\downarrow_{-0.001}$	0.386	0.375	0.376 $\downarrow_{-0.010}$	0.361 $\downarrow_{-0.014}$	0.437	0.456	0.392 $\downarrow_{-0.045}$	0.392 $\downarrow_{-0.064}$
	96	0.384	0.377	0.383 $\downarrow_{-0.001}$	0.376 $\downarrow_{-0.001}$	0.411	0.424	0.404 $\downarrow_{-0.007}$	0.408 $\downarrow_{-0.016}$	0.415	0.422	0.401 $\downarrow_{-0.014}$	0.397 $\downarrow_{-0.025}$
	144	0.397	0.393	0.396 $\downarrow_{-0.001}$	0.391 $\downarrow_{-0.002}$	0.422	0.437	0.413 $\downarrow_{-0.009}$	0.422 $\downarrow_{-0.015}$	0.483	0.520	0.447 $\downarrow_{-0.036}$	0.472 $\downarrow_{-0.048}$
	192	0.408	0.404	0.406 $\downarrow_{-0.002}$	0.402 $\downarrow_{-0.002}$	0.431	0.445	0.420 $\downarrow_{-0.011}$	0.426 $\downarrow_{-0.019}$	0.443	0.451	0.428 $\downarrow_{-0.015}$	0.430 $\downarrow_{-0.021}$
Exchange	48	0.140	0.043	0.139 $\downarrow_{-0.001}$	0.042 $\downarrow_{-0.001}$	0.220	0.102	0.186 $\downarrow_{-0.034}$	0.073 $\downarrow_{-0.029}$	0.160	0.045	0.148 $\downarrow_{-0.012}$	0.044 $\downarrow_{-0.001}$
	96	0.197	0.086	0.196 $\downarrow_{-0.001}$	0.084 $\downarrow_{-0.002}$	0.265	0.131	0.237 $\downarrow_{-0.028}$	0.104 $\downarrow_{-0.027}$	0.221	0.102	0.202 $\downarrow_{-0.019}$	0.092 $\downarrow_{-0.010}$
	144	0.243	0.124	0.242 $\downarrow_{-0.001}$	0.123 $\downarrow_{-0.001}$	0.312	0.175	0.271 $\downarrow_{-0.041}$	0.141 $\downarrow_{-0.034}$	0.292	0.164	0.253 $\downarrow_{-0.039}$	0.149 $\downarrow_{-0.015}$
	192	0.292	0.179	0.287 $\downarrow_{-0.005}$	0.170 $\downarrow_{-0.009}$	0.345	0.239	0.307 $\downarrow_{-0.038}$	0.178 $\downarrow_{-0.061}$	0.331	0.205	0.304 $\downarrow_{-0.027}$	0.190 $\downarrow_{-0.015}$

In the Wiki pageviews dataset, FTS demonstrates exceptional forecasting capabilities. For all prediction horizons, FTS leads in both MAE and MSE, highlighting its strength in capturing trending topics and short-term fluctuations. For instance, in the 48-step prediction, FTS achieves an MAE of 0.305 and an MSE of 0.467, outperforming all other baseline models. This indicates that FTS possesses strong generalization and stability when dealing with network data characterized by complex dynamic features.

In the ETTh1 and Exchange datasets, FTS exhibits outstanding performance across all prediction horizons. Notably, for longer prediction horizons (144 and 192), FTS achieves superior MAE and MSE metrics compared to other models, underscoring its advantage in capturing long-term dependencies and seasonal patterns.

Even with only faithful optimization, the performance on the original dataset without added noise is still superior to that of the unoptimized model. We are surprised to find that our framework not only did not perform worse in the original case without additional perturbations but instead managed to achieve higher performance (**SOTA**). The explanation for our preliminary analysis is that the data for timing problems inherently has perturbations, and our method effectively improves the generalization of the model, making it perform better on the original task. However, in reality, the contribution of our framework is more focused on the trustworthiness level of the model, and the following sections will emphasize the robustness and trustworthiness of the framework in the face of various perturbations.

4.3 Faithfulness under Different Random Seeds

To demonstrate the stability of our model's performance across various random seeds, we conducted a comparative analysis of the performances of FTS and TS under multiple seed settings, with detailed results presented in Table 2 in Exchange dataset. We specifically chose the performance metrics from the 2021 seed's model as our base model, which allowed us to quantitatively assess how FTS diminishes the variability induced by different seeds compared to TS. This analysis revealed that FTS consistently reduced the influence of seed variability on performance metrics, with a significant reduction of up to 99.61%. Details on the specific calculation metric - preference for this reduction can be found in the Appendix L.

We selected the performance under the random seed 2021 as our base model and calculated the reduction in seed variability by FTS compared to TS. Figures 1 clearly illustrate that FTS maintains more consistent performance than TS.

4.4 Faithfulness under Input and Parameter Perturbations

To demonstrate the effectiveness of the proposed framework, we evaluated its faithfulness under different perturbation conditions in this experiment, as shown in Table 3. It should be further emphasized that the presence of "O" represents FTS model and the absence of "O" represents TS-baseline, so as to show that the introduction of Faithful optimization framework can effectively improve the robustness of the model. The table lists the model's performance under no perturbation, input perturbation, and intermediate layer perturbation, both before and after optimization. The experimental results are measured using two metrics: MAE and MSE. We conducted experiments on three datasets, with prediction lengths of 48, 96, 144, and 192 steps.

No perturbation and no perturbation with optimization. Under no perturbation conditions (NP), the baseline performance of the model is shown. By comparing the results of no perturbation with optimization (NPO), we can see that the optimized model shows slight improvement in most cases. This is consistent with our previous conclusion that time series data inherently contains perturbations, and our method effectively enhances the model's generalization ability, thereby improving its performance on the original task. For example, in the Wiki dataset, the MAE for 48 steps prediction decreased from 0.323 to 0.305 and MSE decreased from 0.490 to 0.467. To further demonstrate the **scalability** of the FTS framework to alternative timing frameworks, we have integrated the core mechanisms of FTS into the PatchTST model. The

Table 4: This table presents the results of the loss function ablation study, which evaluates the robustness of the proposed framework by combining different loss functions. The performance metrics used are MAE and MSE, measured across four prediction lengths: 48, 96, 144, and 192 steps. The datasets used are ETTh1 and Exchange. Red indicates the best result, while Blue refers to the second best.

	Loss	$\mathcal{L}_{total}$		$\mathcal{L}_{no\_snp}$		$\mathcal{L}_{no\_cps}$		$\mathcal{L}_{no\_sib}$		$\mathcal{L}_{no\_(cps+sib)}$		$\mathcal{L}_{no\_(snp+sib)}$		$\mathcal{L}_{no\_(snp+cps)}$		$\mathcal{L}_{no\_(snp+cps+sib)}$	
	Metric	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
ETTh1	48	0.376	0.361	0.383	0.370	0.381	0.363	0.378	0.361	0.381	0.365	0.383	0.370	0.384	0.372	0.382	0.371
	96	0.404	0.408	0.409	0.419	0.405	0.410	0.404	0.408	0.404	0.410	0.410	0.419	0.411	0.421	0.410	0.420
	144	0.413	0.422	0.423	0.431	0.413	0.423	0.417	0.422	0.414	0.422	0.423	0.431	0.423	0.430	0.420	0.429
	192	0.420	0.426	0.425	0.433	0.428	0.431	0.423	0.426	0.425	0.430	0.428	0.433	0.430	0.433	0.427	0.429
Exchange	48	0.186	0.073	0.195	0.078	0.190	0.084	0.191	0.074	0.190	0.085	0.195	0.078	0.197	0.083	0.195	0.080
	96	0.237	0.104	0.238	0.112	0.242	0.126	0.235	0.106	0.242	0.126	0.238	0.112	0.244	0.139	0.241	0.137
	144	0.271	0.141	0.277	0.150	0.281	0.177	0.270	0.139	0.281	0.178	0.278	0.150	0.280	0.192	0.280	0.188
	192	0.307	0.178	0.309	0.191	0.317	0.236	0.305	0.180	0.318	0.236	0.310	0.188	0.314	0.254	0.314	0.254

empirical findings from this adaptation are detailed in Appendix G. This appendix not only demonstrates the powerful expansibility of our framework by extending the FTS framework experiments to PatchTST, but also sketches a Faithful Time Series (FTS) definition for all time series models. Moreover, with the objective of highlighting the robustness of FTS against disturbances, we conducted a series of comparative anti-jamming experiments beyond those involving TimeSieve. These additional experiments include several advanced baseline models to comprehensively assess the disturbance rejection capabilities of FTS. The results from these investigations are presented in Appendix H.

Input perturbation and input perturbation with optimization. Under input perturbation conditions (IP), the model's performance declines to some extent, indicating that input noise indeed affects the model's prediction accuracy. In the Wiki dataset, the MAE for 48 steps prediction increased from 0.323 to 0.433, while in the ETTh1 dataset, the MAE for 48 steps prediction increased from 0.361 to 0.386. However, through input perturbation with optimization (IPO), the model's performance significantly improved. For instance, in the Wiki pageviews dataset, the MAE for 48 steps prediction decreased from 0.433 to 0.367, a 60.0% reduction in perturbation. This demonstrates that PGD attacks can effectively identify the weak parts of the input data, and through optimization, these weak parts are strengthened, thereby enhancing the model's robustness and faithfulness. Details on the experiments with different types of noise for input perturbations, as well as the PGD parameter settings for input perturbations, can be found in Appendix I.1 and I.4.

Intermediate layer perturbation and intermediate layer perturbation with optimization. Under intermediate layer perturbation conditions (ILP), the impact on the model's performance is more significant, especially in the MSE metric, indicating that perturbations in the intermediate layer parameters have a greater effect on the model's predictions. For example, in the Wiki dataset, the MAE for 144 steps prediction increased from 0.323 to 0.453, while in the ETTh1 dataset, the MAE for 48 steps prediction increased from 0.361 to 0.437. However, through intermediate layer perturbation with optimization (ILPO), the model's performance also improved. In the Wiki pageviews dataset, the 144 steps prediction MSE dropped from 0.967 to 0.723, a 46.9% reduction. This shows that PGD attacks can optimize model weaknesses under intermediate layer perturbation, enhancing robustness. Further experiments on larger perturbations are detailed in Appendix I.3. In particular, we also do experiments on the computational efficiency of the framework, and the results are presented in Appendix F. FTS incurs moderate training overhead (e.g., TS training: 35.91s vs. 4.81s/epoch for H=192) while preserving inference speed (13.40s vs. 13.13s baseline). This trade-off is critical for domains like energy/finance, where minor accuracy/robustness gains prevent catastrophic failures. Overhead stems from GPU-parallelizable adversarial steps (15-step PGD, 72% utilization), outweighed by reliability gains.

4.5 Loss Function Ablation Study

Our analysis of Table 4 reveals three critical findings. The full constraint combination $\mathcal{L}_{total}$ achieves optimal performance across both datasets, with ETTh1 showing consistent MAE superiority (0.376-0.382) and Exchange attaining the lowest 192-step MSE (0.178). Component removal induces distinct degradation patterns: eliminating $\mathcal{L}_{snp}$ increases Exchange's 48-step MAE from 0.186 to 0.195, while omitting $\mathcal{L}_{cps}$ worsens ETTh1's 192-step MAE from 0.420 to 0.428. The $\mathcal{L}_{sib}$ constraint demonstrates dataset-specific impacts – its removal marginally improves Exchange's 96-step MAE (0.235 vs 0.237) but degrades ETTh1's 144-step prediction (0.417 vs 0.413). Multi-component ablations amplify errors, particularly for volatile series where $\mathcal{L}_{no\_(snp+cps)}$ increases Exchange's 192-step MSE by 42.7% (0.178→0.254). Detailed experimental settings for the loss function parameters can be found in Appendix I.5.

5 Conclusions

We address the sensitivity of the TimeSieve (TS) forecasting framework to random seeds and input perturbations—a critical limitation in volatile multimedia applications like adaptive video streaming. We propose a methodology to rigorously define and enhance model fidelity, effectively mitigating these instabilities while maintaining predictive accuracy on heterogeneous, real-time multimedia workloads. By improving model robustness, our work enables more trustworthy decision-making for resource allocation and content delivery optimization. This underscores the necessity of fidelity in temporal models deployed in complex multimedia ecosystems.

References

[1] David Campos, Miao Zhang, Bin Yang, Tung Kieu, Chenjuan Guo, and Christian S Jensen. 2023. LightTS: Lightweight time series classification with adaptive ensemble distillation. *Proceedings of the ACM on Management of Data* 1, 2 (2023), 1–27.

[2] Si-An Chen, Chun-Liang Li, Nate Yoder, Sercan O Arik, and Tomas Pfister. 2023. Tsmixer: An all-mlp architecture for time series forecasting. *arXiv preprint arXiv:2303.06053* (2023).

[3] Yunyao Cheng, Peng Chen, Chenjuan Guo, Kai Zhao, Qingsong Wen, Bin Yang, and Christian S Jensen. 2023. Weakly guided adaptation for robust time series forecasting. *Proceedings of the VLDB Endowment* 17, 4 (2023), 766–779.

[4] Rahul Dey and Fathi M. Salem. 2017. Gate-variants of Gated Recurrent Unit (GRU) neural networks. In *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*. 1597–1600. doi:10.1109/MWSCAS.2017.8053243

[5] Ninghui Feng, Songning Lai, Fobao Zhou, Zhenxiao Yin, and Hang Zhao. 2024. TimeSieve: Extracting Temporal Dynamics through Information Bottlenecks. arXiv:2406.05036 [cs.LG] https://arxiv.org/abs/2406.05036

[6] Jurgen Franke. 1984. On the robust prediction and interpolation of time series in the presence of correlated noise. *Journal of Time Series Analysis* 5, 4 (1984), 227–244.

[7] Jingkun Gao, Xiaomin Song, Qingsong Wen, Pichao Wang, Liang Sun, and Huan Xu. 2020. Robusttad: Robust time series anomaly detection via decomposition and convolutional neural networks. *arXiv preprint arXiv:2002.09545* (2020).

[8] Klaus Greff, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. 2016. LSTM: A search space odyssey. *IEEE transactions on neural networks and learning systems* 28, 10 (2016), 2222–2232.

[9] Renxiang Guan, Zihao Li, Wenxuan Tu, Jun Wang, Yue Liu, Xianju Li, Chang Tang, and Ruyi Feng. 2024. Contrastive Multiview Subspace Clustering of Hyperspectral Images Based on Graph Convolutional Networks. *IEEE Transactions on Geoscience and Remote Sensing* 62 (2024), 1–14.

[10] Renxiang Guan, Tianrui Liu, Wenxuan Tu, Chang Tang, Wenhan Luo, and Xinwang Liu. 2025. Sampling Enhanced Contrastive Multi-View Remote Sensing Data Clustering with Long-Short Range Information Mining. *IEEE Transactions on Knowledge and Data Engineering* (2025), 1–15.

[11] Renxiang Guan, Wenxuan Tu, Zihao Li, Hao Yu, Dayu Hu, Yuzeng Chen, Chang Tang, Qiangqiang Yuan, and Xinwang Liu. 2024. Spatial-Spectral Graph Contrastive Clustering with Hard Sample Mining for Hyperspectral Images. *IEEE Transactions on Geoscience and Remote Sensing* (2024), 1–16.

[12] Renxiang Guan, Wenxuan Tu, Siwei Wang, Jiyuan Liu, Dayu Hu, Chang Tang, Yu Feng, Junhong Li, Baili Xiao, and Xinwang Liu. 2025. Structure-Adaptive Multi-View Graph Clustering for Remote Sensing Data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 16933–16941.

[13] Tian Guo, Zhao Xu, Xin Yao, Haifeng Chen, Karl Aberer, and Koichi Funaya. 2016. Robust online time series prediction with recurrent neural networks. In *2016 IEEE international conference on data science and advanced analytics (DSAA)*. Ieee, 816–825.

[14] Sascha Krstanovic and Heiko Paulheim. 2017. Ensembles of recurrent neural networks for robust time series forecasting. In *Artificial Intelligence XXXIV: 37th SGAI International Conference on Artificial Intelligence, AI 2017, Cambridge, UK, December 12-14, 2017, Proceedings 37*. Springer, 34–46.

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

[16] Yitong Li, Kai Wu, and Jing Liu. 2023. Self-paced ARIMA for robust time series prediction. *Knowledge-Based Systems* 269 (2023), 110489.

[17] Sibei Liu, Yuanzhe Zhang, Xiang Li, Yunbo Liu, Chengwei Feng, and Hao Yang. 2025. Gated Multimodal Graph Learning for Personalized Recommendation. *INNO-PRESS: Journal of Emerging Applied AI* 1, 1 (2025).

[18] Yong Liu, Chenyu Li, Jianmin Wang, and Mingsheng Long. 2024. Koopa: Learning non-stationary time series dynamics with koopman predictors. *Advances in Neural Information Processing Systems* 36 (2024).

[19] Yunbo Liu, Xukui Qin, Yifan Gao, Xiang Li, and Chengwei Feng. 2025. SE-Transformer: A Hybrid Attention-Based Architecture for Robust Human Activity Recognition. *INNO-PRESS: Journal of Emerging Applied AI* 1, 1 (2025).

[20] Yong Liu, Haixu Wu, Jianmin Wang, and Mingsheng Long. 2023. Non-stationary Transformers: Exploring the Stationarity in Time Series Forecasting. arXiv:2205.14415 [cs.LG]

[21] Zongdong Liu and Jing Liu. 2020. A robust time series prediction method based on empirical mode decomposition and high-order fuzzy cognitive maps. *Knowledge-Based Systems* 203 (2020), 106105.

[22] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations*.

[23] Wenjuan Mei, Zhen Liu, and Yuanzhang Su. 2021. MRPM: multistep robust prediction machine for degradation time series projection. In *2021 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. IEEE, 1–7.

[24] Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. 2023. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. arXiv:2211.14730 [cs.LG]

[25] Owen Queen, Tom Hartvigsen, Teddy Koker, Huan He, Theodoros Tsiligkaridis, and Marinka Zitnik. 2024. Encoding time-series explanations through self-supervised model behavior consistency. *Advances in Neural Information Processing Systems* 36 (2024).

[26] Matteo Sangiorgio and Fabio Dercole. 2020. Robustness of LSTM neural networks for multi-step forecasting of chaotic time series. *Chaos, Solitons & Fractals* 139 (2020), 110045.

[27] Tim Van Erven and Peter Harremos. 2014. Rényi divergence and Kullback-Leibler divergence. *IEEE Transactions on Information Theory* 60, 7 (2014), 3797–3820.

[28] Qingsong Wen, Kai He, Liang Sun, Yingying Zhang, Min Ke, and Huan Xu. 2021. RobustPeriod: Robust time-frequency mining for multiple periodicity detection. In *Proceedings of the 2021 international conference on management of data*. 2328–2337.

[29] Wikimedia Foundation. n.d.. Terms of Use. https://foundation.wikimedia.org/wiki/Policy:Terms_of_Use [Online; accessed 15-October-2024].

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

[31] Kun Yi, Qi Zhang, Wei Fan, Shoujin Wang, Pengyang Wang, Hui He, Defu Lian, Ning An, Longbing Cao, and Zhendong Niu. 2023. Frequency-domain MLPs are More Effective Learners in Time Series Forecasting. arXiv:2311.06184 [cs.LG]

[32] TaeHo Yoon, Youngsuk Park, Ernest K Ryu, and Yuyang Wang. 2022. Robust probabilistic time series forecasting. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1336–1358.

[33] Yang Yu, Ruizhe Ma, and Zongmin Ma. 2024. Robformer: A robust decomposition transformer for long-term time series forecasting. *Pattern Recognition* (2024), 110552.

[34] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. 2022. Are Transformers Effective for Time Series Forecasting? arXiv:2205.13504 [cs.AI]

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

[36] Anguo Zhang, Wei Zhu, and Juanyu Li. 2018. Spiking echo state convolutional neural network for robust time series classification. *IEEE Access* 7 (2018), 4927–4935.

[37] Xueli Zhang, Cankun Zhong, Jianjun Zhang, Ting Wang, and Wing WY Ng. 2023. Robust recurrent neural networks for time series forecasting. *Neurocomputing* 526 (2023), 143–157.

[38] Shuhan Zhong, Sizhe Song, Weipeng Zhuo, Guanyao Li, Yang Liu, and S. H. Gary Chan. 2024. A Multi-Scale Decomposition MLP-Mixer for Time Series Analysis. arXiv:2310.11959 [cs.LG]

[39] Guancheng Zhou. 2020. Robust time series prediction with missing data based on deep convolutional neural networks. In *2020 International Conference on Computer Communication and Network Security (CCNS)*. IEEE, 178–183.

[40] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. arXiv:2012.07436 [cs.LG]

[41] Zhiqiang Zhou, Linxiao Yang, Qingsong Wen, and Liang Sun. 2024. RobustTSVar: A Robust Time Series Variance Estimation Algorithm. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 161–165.