Understanding and Mitigating Structural Forgetting in Fine-Tuned Time Series Foundation Models

Zeyu Shi, Yirong Xue, Xin Xue, Ziming Wang, Jintao Wu, Shiqi Gao, Haoyi Zhou*
Beihang University, Beijing 100191, China
haoyi@buaa.edu.cn

Abstract

While fine-tuning is the standard recipe for adapting Time Series Foundation Models (TSFMs) to downstream tasks, we reveal that it triggers a critical yet overlooked problem: structural forgetting. Distinct from widely studied point-wise error degradation, structural forgetting represents a fundamentally more destructive phenomenon: the erasure of universal structural patterns acquired during pre-training. Unlike point-wise errors, this structural collapse renders models incapable of supporting basic structure-dependent decisions (e.g., trend-based trading), a critical vulnerability that existing forgetting mitigation strategies fail to address. To bridge this gap, we first introduce a diagnostic framework equipped with a novel Temporal-Frequency Attribution analysis, mechanistically revealing that fine-tuning unnecessarily overwrites a sparse set of pattern-critical parameters. We then propose NeST, a post-hoc framework that precisely restores these components while compensating surrounding weights to preserve task adaptation. Extensive experiments across 3 TSFMs and 9 datasets demonstrate that NeST recovers an average of 85% of structural awareness without sacrificing task performance, offering a new paradigm for structurally-preserved TSFM adaptation.

1 Introduction

Time Series Foundation Models (TSFMs) [1–3] are designed with the explicit goal of learning the universal structural patterns that govern temporal data. By pre-training on massive corpora, these models internalize general patterns such as trends, distributional shifts, and spectral structures that transfer across domains [4, 5]. This structural awareness is what distinguishes a foundation model from trivial baselines such as moving averages, and forms the basis of its generalizability for downstream tasks [6].

In practice, however, TSFMs are not always deployed in their pre-trained form: domain-specific fine-tuning remains a standard recipe for specialized tasks [7–9]. Yet such adaptation is well known to incur *catastrophic forgetting* [10]. Prior studies of forgetting in TSFMs have been largely restricted to a surface-level evaluation of point-wise performance degradation (e.g., increased MSE or MAE), overlooking the deeper

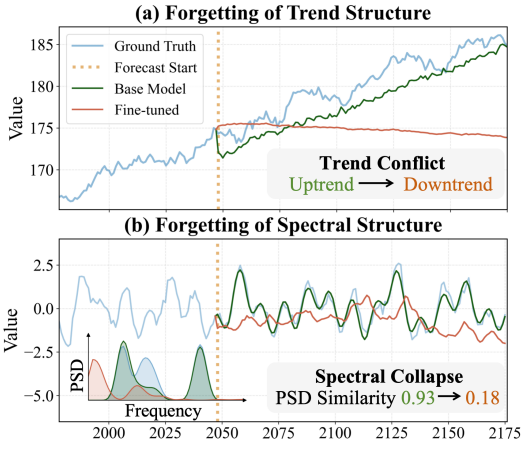


Figure 1: Structural forgetting illustrated. Fine-tuning TimesFM 2.0 on Gold and evaluated on sequences with structural patterns: the base model retains trend and spectral structure; the fine-tuned model loses both.

*Corresponding Authors

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

erosion of structural representations learned during pre-training [11–13]. In this paper, we identify a fundamentally more severe degradation, which we term *structural forgetting: after fine-tuning, TSFMs systematically lose the universal structural patterns, such as trends, shifts, and spectral characteristics acquired during pre-training.* Taking trend and spectral patterns as examples (Figure 1), when TimesFM 2.0 is fine-tuned on a downstream task and evaluated on structured sequences, the base model successfully preserves the inherent uptrend and high Power Spectral Density (PSD) similarity. In contrast, the fine-tuned model exhibits conflicting trend predictions and a complete spectral collapse. This degradation poses a far greater threat than point-wise error—for in many real decisions, structural signals matter more than absolute values: traders make buy/sell decisions primarily on directional trend [14], and grids schedule power dispatch according to diurnal periodicity [15]. A model with degraded point-wise accuracy but intact structural awareness still supports such decisions. One that has undergone structural forgetting does not.

Despite the severity of structural forgetting, existing mitigation strategies, including Rehearsal [10], LoRA [16], EWC [17], and LwF [18], remain structurally unaware interventions. They fail to investigate the underlying mechanisms of structural forgetting and lack the precision to target and resolve it. To address this critical gap, we build a synthetic diagnostic framework to systematically quantify structural forgetting, confirming its occurrence across diverse TSFMs and downstream tasks. Inspired by mechanistic interpretability in large language models [19], we then investigate the mechanism of structural forgetting at the neuron level. Through a Temporal-Frequency Attribution (TFA) analysis, we localize a sparse set of *pattern-critical neurons* responsible for structural awareness in TSFMs, and reveal that fine-tuning gradients of downstream tasks systematically conflict with these neurons. However, we further demonstrate that this conflict is entirely unnecessary: the remaining parameters alone are sufficient to absorb downstream adaptation.

Building on these findings, we propose **NeST** (**Ne**uron-level **S**tructural res**T**oration), a post-hoc, targeted framework that precisely reverts pattern-critical neurons to their pre-trained states while compensating the surrounding parameters to retain the task performance, recovering an average of 85% of structural awareness across 3 TSFMs and 9 downstream tasks without sacrificing downstream gains. We summarize our main contributions as follows:

- **Quantifying structural forgetting.** As far as we know, we are the first to characterize structural forgetting in fine-tuned TSFMs, and we introduce a synthetic diagnostic framework to measure it.
- **A neuron-level explanation of structural forgetting.** We provide a mechanistic explanation for this forgetting via a novel Temporal-Frequency Attribution analysis, revealing that task-specific gradient updates systematically and unnecessarily conflict with localized, pattern-critical neurons.
- **A targeted mitigation framework.** We propose NeST, a post-hoc framework that precisely mitigates structural forgetting. Extensive experiments demonstrate that NeST significantly recovers structural awareness without sacrificing task performance.

2 Empirical Evidence of Structural Forgetting

In this section, we introduce a synthetic diagnostic framework to assess structural awareness retained by TSFMs after fine-tuning (§2.1), and show that fine-tuning causes structural forgetting while improving downstream task performance (§2.2).

2.1 Quantifying Structural Forgetting via Synthetic Diagnostic Evaluation

Standard evaluations for assessing forgetting and degradation in forecasting previous tasks rely on point-wise metrics (e.g., MSE, MAE) that fail to capture structural integrity [11–13]. To quantify structural forgetting after fine-tuning, we propose a synthetic diagnostic framework that, motivated by previous works [4–6], probes seven typical structural patterns across three dimensions. These three dimensions are not exhaustive but representative, selected to cover prevalent structural behaviors across real-world time series domains as a tractable diagnostic probe [4].

1. **Distributional Shifts** (*Level Shift*, *Variance Shift*): probing whether the model retains sensitivity to changes in mean and volatility, i.e., the regime transitions in sensor [20] and biomedical data [21]. We assess *Level Shift* via Mean Reversion Bias (MRB$\downarrow$), and *Variance Shift* via Volatility Tracking Ratio (VTR $\rightarrow 1$).
2. **Stochastic and Structured Dynamics** (*AR(1) Process*, *Stochastic Trend*, *Deterministic Trend*): probing whether the model preserves temporal autocorrelation, stationarity distinctions, and deter-

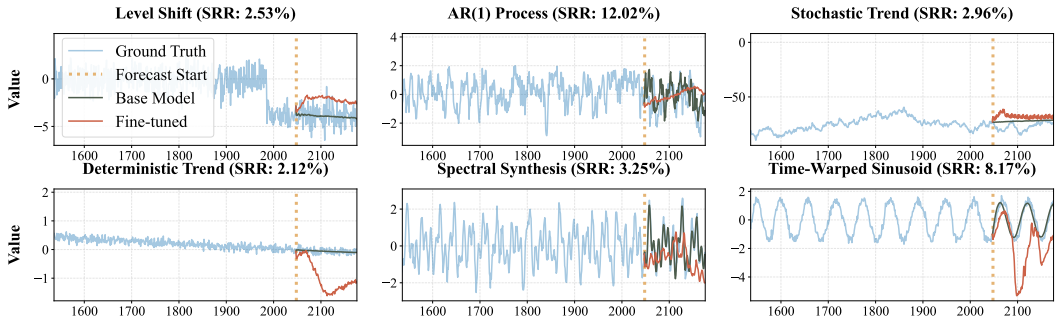


Figure 2: Structural forgetting in TimesFM 2.0 after fine-tuning on the Gold dataset across pattern types. *Variance Shift* is excluded as TSFMs universally lack high-frequency prediction capacity prior to fine-tuning, rendering forgetting analysis uninformative for this pattern type.

ministic structural sensitivity that are required in hydrological [22] and climate time series [23]. We compute ACF-L2 distance (↓) for *AR(1) Process*, the Ornstein–Uhlenbeck coefficient $\hat{\lambda}$ ($\to 0$) for *Stochastic Trend*, and relative slope error (↓) for *Deterministic Trend*.

3. **Stationary and Non-Stationary Periodicity** (*Spectral Synthesis*, *Time-Warped Sinusoid*): probing frequency-domain awareness and sensitivity to time-varying instantaneous frequency, capabilities essential for periodic and oscillatory signal domains [24, 25]. We employ PSD Similarity (↑) for *Spectral Synthesis* and DTW distance (↓) for *Time-Warped Sinusoid*.

As described above, each pattern is thus associated with a dedicated metric m; complete details are provided in Appendix B. Given such a pattern-specific metric m, we then define the *Structural Retention Rate (SRR)* to unify measurements of heterogeneous directionality:

$$\text{Structural Retention Rate}(m) = \begin{cases} m_{\text{post}}/m_{\text{pre}} & \text{if } m \uparrow \text{ (higher is better)} \\ m_{\text{pre}}/m_{\text{post}} & \text{if } m \downarrow \text{ (lower is better)} \\ |m_{\text{pre}}| \,/\, |m_{\text{post}}| & \text{if } m \to 0 \text{ is preferred} \\ |1 - m_{\text{pre}}| \,/\, |1 - m_{\text{post}}| & \text{if } m \to 1 \text{ is preferred} \end{cases} \tag{1}$$

where m_{pre} and m_{post} are metric values before and after fine-tuning. By construction, $\text{Retention} = 1$ indicates perfect structure preservation, <1 forgetting, and >1 incidental improvement.

2.2 Fine-Tuning Improves Task Performance but Degrades Structural Awareness

With the above evaluation framework, we now empirically investigate whether fine-tuning on downstream tasks leads to structural forgetting in TSFMs.

Visual Evidence of Structural Collapse. We visualize representative predictions of TimesFM 2.0 before and after fine-tuning on the Gold dataset across six pattern types. **Notably, *Variance Shift* is excluded from all subsequent evaluations**, because virtually all assessed TSFMs lack high-frequency prediction capacity even before fine-tuning, rendering forgetting analysis uninformative for this pattern type (Appendix B.3). Nevertheless, we note that this pattern remains a potentially valuable diagnostic for future TSFMs with stronger high-frequency modeling capacity.

As shown in Figure 2, fine-tuning induces structural degradation across evaluated patterns. The model loses adaptation under *Level Shift*, autocorrelation structure under *AR(1) Process*, and trend extrapolation capacity under *Deterministic Trend*. Most strikingly, on *Stochastic Trend* sequences, the fine-tuned model imposes a spurious stationary prior, and on *Spectral Synthesis*, frequency structure undergoes severe degradation. Similar degradation is observed across other TSFMs (Appendix D.1).

Quantitative Results. Table 1 reports ΔTask Gain and per-category Structural Retention Rate for TimesFM 2.0 and Timer-XL. Despite task-performance gains, both models exhibit substantial structural forgetting: TimesFM 2.0 achieves a mean Retention of only 33.14%, with Timer-XL similarly reaching only 40.15%. Additional results across more models and datasets are in Appendix D.2.

All empirical evidence reveals a fundamental tension: while fine-tuning yields downstream task gains, it universally induces structural forgetting. This macroscopic degradation naturally motivates a microscopic investigation. To understand how fine-tuning systematically destroys structural awareness,

Table 1: Structural forgetting quantification for TimesFM 2.0 and Timer-XL. Pattern abbreviations (e.g., LS, AR) are defined in the note below. Values indicate Retention Rate <35%.

Model & Dataset		Δ Task Gain↑ (%)		Structural Retention Rate↑ (%)						
		MSE	MAE	LS	AR	ST	DT	SS	TW	**AVG**
TimesFM	Crude Oil	+7.11	+2.90	19.68	46.52	17.69	23.72	30.93	37.39	29.32
	Exchange Rate	+17.35	+8.11	13.80	43.46	38.42	23.07	22.90	43.17	30.80
	Gold	+18.80	+9.28	16.46	54.73	18.12	14.59	36.31	53.85	32.34
	MSFT	+12.78	+6.85	21.25	38.56	11.22	17.93	26.68	40.61	26.04
	SPX500	+8.99	+5.89	28.21	52.33	25.93	25.41	74.32	77.14	47.22
Timer-XL	Crude Oil	+4.87	+2.13	29.95	45.81	22.37	19.83	64.24	41.95	37.36
	Exchange Rate	+11.23	+5.67	36.18	79.91	41.05	28.74	71.17	64.93	53.66
	Gold	+13.45	+6.82	23.17	53.08	19.63	16.41	37.07	29.72	29.85
	MSFT	+8.31	+4.56	28.98	59.62	24.91	21.05	38.10	51.17	37.34
	SPX500	+5.76	+3.92	21.57	58.30	33.47	23.89	56.69	61.25	42.53

Note: **LS**: Level Shift, **AR**: AR(1) Process, **ST**: Stochastic Trend, **DT**: Deterministic Trend, **SS**: Spectral Synthesis, **TW**: Time-Warped Sinusoid.

Section 3 shifts our focus to the microscopic weight level, exploring the mechanistic origins of this forgetting through neuron-level attribution.

3 Understanding Structural Forgetting via Neuron-Level Analysis

We investigate why fine-tuning causes structural forgetting. Using Temporal-Frequency Attribution (§3.1), we show that fine-tuning gradients unnecessarily conflict with pattern-critical neurons (§3.2.1), as the remaining parameters suffice for downstream learning (§3.2.2).

3.1 Identifying Pattern Neurons via Gradient-Based Attribution

To identify neurons critical for preserving temporal and spectral structure, we quantify structural importance via gradient-based attribution, which requires a loss objective sensitive to structural fidelity over point-wise synchronization.

Temporal-Frequency Attribution Basis (TFA-Loss). A purely point-wise loss (MSE) surfaces neurons governing value accuracy while remaining blind to those sustaining temporal periodicity and spectral distribution. We therefore define a **Temporal-Frequency Joint Loss** (TFA-Loss) coupling a temporal term $\mathcal{L}_{\text{tmp}}$ and a frequency term $\mathcal{L}_{\text{freq}}$, operating on the amplitude spectrum of the DFT:

$$\mathcal{L}_{\text{TFA}} = \underbrace{\|y - \hat{y}\|_2^2}_{\mathcal{L}_{\text{tmp}}} + \lambda \cdot \underbrace{\big\| |F(y)| - |F(\hat{y})| \big\|_2^2}_{\mathcal{L}_{\text{freq}}}, \tag{2}$$

where $F(\cdot)$ is the DFT and $|\cdot|$ extracts the amplitude spectrum. Amplitude spectra rather than full complex DFT outputs are used deliberately: by Parseval's theorem [26], the complex frequency term reduces to $\mathcal{L}_{\text{tmp}}$, whereas the amplitude modulus introduces a phase-invariant spectral density constraint orthogonal to point-wise accuracy. TFA's validity is proved in Appendix A, Proof A.3.

Per-Neuron Structural Importance. Building on $\mathcal{L}_{\text{TFA}}$, we quantify the per-neuron structural importance by how much a parameter perturbation affects the $\mathcal{L}_{\text{TFA}}$. At the pre-trained checkpoint, $\nabla_\theta \mathcal{L}_{\text{TFA}} \approx 0$; by a second-order Taylor expansion, the expected loss increase under isotropic perturbation is proportional to $\text{Tr}(H)$ (Appendix A, Proof A.1). We approximate the diagonal of H via squared gradients following the empirical Fisher approximation [17, 27] (Appendix A, Proof A.2). Given a base model's layer $W_j \in \mathbb{R}^{d' \times d}$, the **TFA-Fisher** importance score of neuron k in layer j is:

$$s_{j,k}^{\text{TFA}} = \mathbb{E}_{x \sim D_{\text{pattern}}} \left[\left(\frac{\partial \mathcal{L}_{\text{TFA}}}{\partial W_{j,k}} \right)^2 \right], \tag{3}$$

evaluated on the synthetic pattern dataset D_{pattern} from §2. Per layer, the top $\lfloor d' \times \rho_{\text{neuron}} \rfloor$ neurons ranked by $s_{j,k}^{\text{TFA}}$ form the candidate set A_j^{neuron}.

Cross-Layer Displacement Prioritization. Since fine-tuning updates parameters non-uniformly across layers, we further measure the normalized displacement of structural candidates per layer:

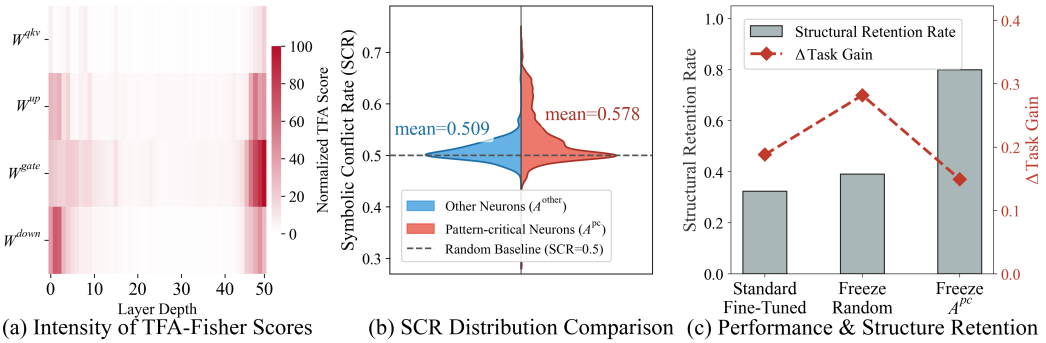
$$d_j = \frac{\|(W_j - W_j') \odot M_j\|_2}{\|W_j \odot M_j\|_2}, \tag{4}$$

where W'_j denotes the weights of layer j after fine-tuning and M_j is the binary mask derived from A_j^{neuron}. The top $\lfloor L \times \rho_{\text{layer}} \rfloor$ layers ranked by d_j form the candidate layer set A^{layer}.

Pattern-Critical Neuron Set. Intersecting the two criteria yields neurons that are structurally sensitive and reside in layers most affected by fine-tuning, which subsequently serves as our primary analytical lens to demystify the microscopic origins of structural forgetting:

$$\mathcal{A}^{pc} = \{(j,k) \mid j \in A^{\text{layer}} \text{ and } k \in A_j^{\text{neuron}}\}. \tag{5}$$

3.2 How Does Fine-Tuning Destroy Structural Awareness?



(a) Intensity of TFA-Fisher Scores (b) SCR Distribution Comparison (c) Performance & Structure Retention

Figure 3: Analysis of why fine-tuning destroys structural awareness. (a) Layer-wise Intensity of TFA-Fisher scores. (b) Sign Conflict Ratio distributions comparing pattern-critical neurons and non-pattern neurons. (c) Downstream performance and Structural Retention Rate under three conditions.

With the pattern-critical neuron set $\mathcal{A}^{pc}$ established, two targeted inquiries emerge: how fine-tuning affects these critical neurons (§3.2.1), and whether such targeting is a necessary consequence (§3.2.2). Following experiments use TimesFM 2.0 fine-tuned on Gold as the subject; similar phenomena are also verified across more models (Appendix E.2).

3.2.1 The Gradient Conflict: Task Updates Conflict with Pattern Neurons

As shown in Figure 3(a), neurons with high TFA-Fisher importance concentrate predominantly in shallow and deep MLP layers (W^{up}, W^{gate}, W^{down}), with sparse representation in intermediate layers. To characterize how fine-tuning affects these neurons, we define the Sign Conflict Ratio (SCR) to quantify per-neuron gradient conflict between the task objective $\mathcal{L}_{\text{task}}$ and the structural preservation objective $\mathcal{L}_{\text{TFA}}$:

$$\text{SCR}(\mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} \mathbb{I}(\text{sgn}(\nabla_i \mathcal{L}_{\text{task}}) \neq \text{sgn}(\nabla_i \mathcal{L}_{\text{TFA}})), \tag{6}$$

where values above 0.5 indicate that task gradients actively oppose structural preservation. We compute $\text{SCR}(\mathcal{A}_j^{pc})$ and $\text{SCR}(\mathcal{A}_j^{\text{other}})$ at checkpoints (steps 50, 100, 150, 200), averaged across layers and three independent runs. We focus on early-phase checkpoints, as prolonged fine-tuning erases structural information from $\mathcal{A}^{pc}$, causing SCR to revert toward 0.5 via structural erasure, rendering late-phase measurements uninformative. Full SCR trajectories are provided in Appendix E.1.

As shown in Figure 3(b), the SCR distribution over $\mathcal{A}^{pc}$ is substantially shifted above 0.5 during early fine-tuning, while $\mathcal{A}^{\text{other}}$ clusters near 0.5 with no systematic directional bias. This asymmetry confirms that task gradients selectively and systematically conflict with pattern-critical neurons, constituting a targeted disruption of structural awareness. See Appendix E.2 for more models' results.

3.2.2 The Unnecessary Destruction: Pattern Neurons Are Not Required for Adaptation

The conflict above raises a critical question: is disruption of pattern-critical neurons necessary for downstream adaptation, or an avoidable artifact? We test this empirically under three conditions: (1) *Standard Fine-tuning* (all parameters updated); (2) *Freeze* $\mathcal{A}^{pc}$ (only $\mathcal{A}^{\text{other}}$ optimized); and (3) *Freeze Random* (an equal-sized randomly selected set frozen). Details are provided in Appendix C.5.

Figure 3(c) shows that *Freeze* $\mathcal{A}^{pc}$ achieves a downstream MSE Gain of 14.9%, only 3.9 pp below standard fine-tuning (18.8%), while Structural Retention Rate rises from 32.3% to 79.9%, demonstrating that $\mathcal{A}^{\text{other}}$ alone retains sufficient adaptation capacity and that structural forgetting is not a necessary consequence of fine-tuning. The *Freeze Random* baseline fails to prevent structural forgetting, confirming that the benefit is specific to pattern-critical neuron identification rather than generic parameter regularization. Nevertheless, the residual 3.9 pp performance gap motivates a more targeted intervention that preserves structural awareness while fully recovering task performance. See Appendix E.2 for more results across models.

These findings confirm that structural forgetting is an artifact of unconstrained optimization rather than an inherent cost of adaptation: task gradients selectively conflict with pattern-critical neurons, yet $\mathcal{A}^{\text{other}}$ alone retains sufficient capacity for downstream learning. However, the residual 3.9 pp performance gap indicates that freezing is too blunt an instrument. This motivates a more targeted intervention without sacrificing task performance, which we develop in the following section.

4 Mitigating Structural Forgetting via Neuron Restoration

4.1 NeST: Neuron-Level Structural Restoration

The findings in §3 motivate a direct corrective action: since structural awareness is concentrated in $\mathcal{A}^{pc}$ and its disruption is unnecessary for downstream adaptation, we can restore these neurons post-hoc without retraining. We formalize this as **NeST** (**Ne**uron-level **S**tructural res**T**oration), a post-hoc correction applied to an already fine-tuned TSFM.

Mask-Guided Structural Reversion. Let $\mathbf{W}^{ft}$ and $\mathbf{W}^{pre}$ denote a given layer of the fine-tuned and pre-trained models. We define a binary structural mask $\mathbf{M}$, where elements set to 1 indicate parameters belonging to $\mathcal{A}^{pc}$. Pattern-critical parameters are reverted to their pre-trained values:

$$\mathbf{W}^{res} = \mathbf{W}^{ft} \odot (\mathbf{1} - \mathbf{M}) + \mathbf{W}^{pre} \odot \mathbf{M}. \tag{7}$$

Task Compensation. Since task neurons have co-adapted to the fine-tuned activations of $\mathcal{A}^{pc}$, naively reverting these parameters introduces an activation discrepancy on task inputs. Drawing on the Optimal Brain Surgeon (OBS) framework [28], we derive a minimum-norm compensation to neutralize this discrepancy. Let $\mathbf{W}_m$ represent a parameter at index m:

$$\mathbf{C}_m = -\frac{\mathbf{W}^{ft}_m - \mathbf{W}^{pre}_m}{[\mathbf{H}^{-1}]_{mm}} \mathbf{H}^{-1}_{:,m}, \tag{8}$$

where $\mathbf{H} = \nabla^2_{\mathbf{W}^{ft}} \|\mathbf{W}^{res}\mathbf{X}_{task} - \mathbf{W}^{ft}\mathbf{X}_{task}\|_2^2$ is the Hessian on $\mathcal{D}_{task}$ (proof in Appendix A, Proof A.4). The compensation is applied to non-critical parameters, yielding the final NeST weights:

$$\mathbf{W}^{NeST} = \mathbf{W}^{res} + (\mathbf{1} - \mathbf{M}) \odot \sum\nolimits_{m \in \mathcal{A}^{pc}} \mathbf{C}_m. \tag{9}$$

NeST thus perfectly restores structural awareness within $\mathcal{A}^{pc}$ while absorbing the task performance offset into the remaining parameters.

4.2 Experimental Setup

Models and Fine-Tuning Datasets. We evaluate NeST on three TSFMs: TimesFM 2.0 [1], Timer-XL [29], and $\text{Chronos}_{\text{bolt-base}}$ [2]. Fine-tuning datasets include nine downstream tasks, including Gold, SPX500, MSFT, Natural Gas, Crude Oil, Exchange Rate, Apple [30], and ETTh1, ETTm1 [31].

Baselines. We compare against six methods spanning complementary forgetting-mitigation strategies: (1) *Vanilla SFT*, standard fine-tuning without structural protection, serving as the lower bound; (2) *LoRA* [16], which constrains updates to low-rank subspaces; (3) *Rehearsal* [10], which mixes synthetic structural samples into fine-tuning data; (4) *EWC* [17], which penalizes updates to parameters important for prior tasks; (5) *LwF* [18], which applies soft distillation from the pre-trained model; and (6) *Model Merging* [32], which linearly interpolates between pre-trained and fine-tuned weights.

Implementation Details. Calibration sets are fixed at $|\mathcal{D}_{pattern}| = |\mathcal{D}_{task}| = 128$ samples, with $\rho_{neuron} = 0.5$, $\rho_{layer} = 0.3$ and $\lambda = 0.5$. Baseline results are reported at the configuration achieving the best structural retention/task trade-off over a grid search. All experiments are repeated three times; the mean results are reported. More experimental setup details are provided in Appendix C.

Table 2: Comparison of downstream task performance (Δ Task Gain) and Structural Retention Rate on TimesFM 2.0 across datasets. **Best results** are in bold and second-best are underlined.

Dataset & Method		Δ Task Gain↑ (%)		Structural Retention Rate↑ (%)						
		MSE	MAE	LS	AR	ST	DT	SS	TW	**AVG**
Gold	Vanilla SFT	+18.80	+9.28	16.46	54.73	18.12	14.59	36.31	53.85	32.34
	LoRA	+9.33	+3.66	27.88	60.83	17.03	80.34	82.84	86.82	59.29
	Rehearsal	+12.96	+5.28	70.40	57.78	**76.82**	16.40	63.12	77.98	60.42
	EWC	+11.71	+8.12	19.41	58.38	68.24	21.45	71.71	66.76	50.99
	LwF	+17.45	+8.41	20.74	54.06	38.49	38.79	78.11	80.86	51.84
	Model Merging	+6.25	+7.52	24.69	57.67	51.95	30.23	73.46	71.85	51.64
	NeST (Ours)	**+18.23**	**+8.72**	**79.53**	**94.62**	43.47	**93.96**	**97.20**	**97.57**	**84.39**
MSFT	Vanilla SFT	+12.78	+6.85	21.25	38.56	11.22	17.93	26.68	40.61	26.04
	LoRA	+4.16	+6.07	63.47	80.41	9.83	87.63	92.34	91.56	70.87
	Rehearsal	+6.30	+7.17	61.54	62.32	**64.66**	23.82	59.98	65.11	56.24
	EWC	+9.75	**+10.75**	58.01	46.62	16.35	89.78	86.04	58.55	59.23
	LwF	+2.40	+7.01	67.52	52.80	5.60	89.42	93.41	90.17	66.49
	Model Merging	-0.60	-0.31	55.83	32.75	19.30	80.63	78.79	34.14	50.24
	NeST (Ours)	**+11.14**	+5.67	**68.13**	**85.45**	54.57	**91.89**	**96.28**	**97.82**	**82.36**
SPX500	Vanilla SFT	+8.99	+5.89	28.21	52.33	25.93	25.41	74.32	77.14	47.22
	LoRA	+7.02	+4.50	83.93	83.40	23.16	99.33	99.07	94.23	80.52
	Rehearsal	+2.26	+0.89	61.02	37.61	71.16	46.81	63.93	15.24	49.30
	EWC	+2.58	+3.15	66.98	26.30	68.43	82.24	89.53	39.34	62.14
	LwF	-1.67	-0.32	82.16	52.52	16.53	96.67	96.26	89.62	72.29
	Model Merging	+4.05	+1.89	77.06	75.57	36.97	97.78	98.36	95.12	80.14
	NeST (Ours)	**+7.82**	**+5.12**	**85.45**	**91.67**	**71.67**	**100.10**	**99.13**	**99.40**	**91.23**

Note: **LS**: Level Shift, **AR**: AR(1) Process, **ST**: Stochastic Trend, **DT**: Deterministic Trend, **SS**: Spectral Synthesis, **TW**: Time-Warped Sinusoid.

4.3 Main Results

Table 2 reports Δ Task Gain and Structural Retention Rate across six diagnostic categories.

NeST achieves the highest average structural retention across all datasets. NeST attains average Structural Retention Rates of 84.39%, 82.36%, and 91.23% on Gold, MSFT, and SPX500, respectively, with substantial margins over the strongest baselines in each case. This demonstrates that precise, neuron-level restoration of structural awareness achieves more comprehensive pattern preservation than global regularization or parameter-efficient approaches.

NeST maintains downstream task performance at near-Vanilla SFT levels. Unlike structurally-aware baselines that sacrifice task gain for retention, NeST sustains competitive Δ Task Gain across all datasets (+18.23%, +11.14%, +7.82% in MSE), closely tracking Vanilla SFT (+18.80%, +12.78%, +8.99%). This confirms that the task compensation (Eq. (9)) successfully absorbs the task-side cost of pattern neuron restoration, leveraging the redundant capacity in task neurons established in §3.2.2.

Stochastic Trend (ST) remains the hardest pattern to retain across all methods. ST retention is consistently lower than other categories (e.g., 43.47% on Gold for NeST), reflecting the diffuse, non-stationary nature of stochastic autocorrelation structure. See Appendix F for more results.

4.4 Ablation Study

Hyperparameter Sensitivity. Figure 4 and Figure 5 show sensitivity to three hyperparameter groups. The TFA-Loss coefficient λ consistently improves retention, with gains primarily driven by better recovery of Stationary and Non-Stationary Periodicity patterns; performance saturates at $\lambda \in [0.5, 1]$, and we adopt λ=0.5 as the default. Task gain and structural retention correlate positively with $|\mathcal{D}_{task}|$ and $|\mathcal{D}_{pattern}|$ respectively, yet both saturate within 128–256 samples, indicating that a lightweight set suffices. Increasing $(\rho_{neuron}, \rho_{layer})$ monotonically improves structural retention, plateauing near $(0.5, 0.3)$ and covering only $\sim$15% of total parameters.

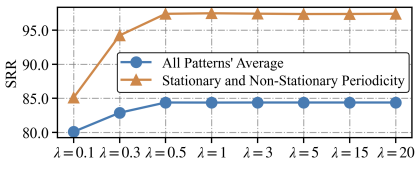
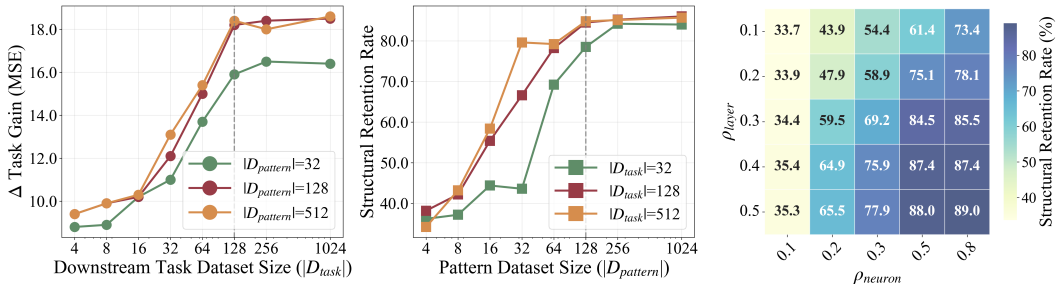


Figure 4: Effect of the TFA-Loss coefficient λ on Structural Retention Rate.

Component Effectiveness and Efficiency. Table 3 ablates Reversion (Rev.) and Compensation (Comp.) stage of NeST, where *Ours* denotes the original method for the respective stage. Random



(a) Dataset Size Study (b) Select Ratio Study

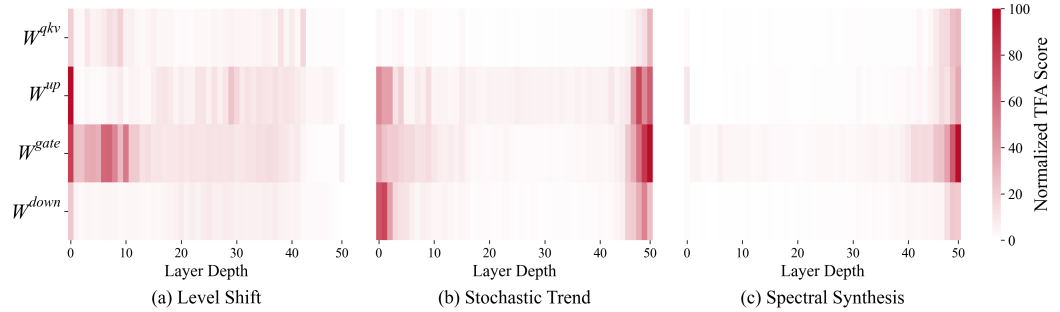
Figure 5: Hyperparameter sensitivity analysis of NeST.

neuron selection (*Random+Ours*) collapses retention to 47%, confirming that precise pattern-critical localization is foundational. Removing $\mathcal{L}_{\text{freq}}$ (*w/o freq+Ours*) recovers most performance but incurs concentrated drops on periodic categories (SS and TW), validating its role in capturing periodic structure. Ablating OBS compensation (*Ours+w/o Com*) maintains high structural retention yet degrades Task Gain, confirming its necessity for preserving fine-tuning co-adaptations. NeST completes the process within ≈100s with high efficiency; efficiency analysis is in Appendix G.1.

Table 3: Ablation study of NeST components and efficiency on Gold and MSFT datasets.

Ablation Stage		**Gold**								
		Δ Task Gain↑ (%)		**Structural Retention Rate↑ (%)**						**Time (s)**
Rev.	Comp.	MSE	MAE	LS	AR	ST	DT	SS	TW	AVG	
Random	Ours	+17.02	+7.64	23.23	55.74	38.36	28.45	68.67	68.23	47.11	92.20
w/o freq	Ours	+18.20	+8.37	77.89	93.93	43.56	95.79	80.38	81.45	78.83	98.44
Ours	w/o Com	+11.70	+4.53	**83.09**	93.15	**49.83**	**97.93**	97.06	97.38	**86.24**	28.13
Random	w/o Com	+11.34	+4.13	21.21	56.35	35.35	23.89	57.70	64.59	43.18	13.66
Ours	**Ours**	**+18.23**	**+8.72**	79.53	**94.62**	43.47	93.96	**97.20**	**97.57**	84.39	99.76
Ablation Stage		**MSFT**								
		Δ Task Gain↑ (%)		**Structural Retention Rate↑ (%)**						**Time (s)**
Rev.	Comp.	MSE	MAE	LS	AR	ST	DT	SS	TW	AVG	
Random	Ours	+11.03	+5.58	35.38	48.17	19.01	26.21	66.61	69.75	44.19	90.98
w/o freq	Ours	**+11.27**	**+5.77**	68.75	86.05	54.60	91.12	86.40	82.94	78.31	100.43
Ours	w/o Com	+7.38	+4.83	**74.28**	**87.83**	**60.22**	**92.80**	93.52	96.91	**84.26**	30.74
Random	w/o Com	+7.41	+4.77	33.59	43.79	14.65	20.44	51.18	61.51	37.53	12.95
Ours	**Ours**	+11.14	+5.67	68.13	85.45	54.57	91.89	**96.28**	**97.82**	82.36	100.84

Pattern-Specific TFA-Fisher Importance Distribution. As shown in Figure 6, TFA-Fisher importance distributions vary markedly across diagnostic patterns: Level Shift concentrates in early layers, Stochastic Trend spans both early and terminal blocks, and Spectral Synthesis localizes in late layers. This heterogeneity suggests potential for future pattern-specialized neuron identification designs.



(a) Level Shift (b) Stochastic Trend (c) Spectral Synthesis

Figure 6: Layer-wise Intensity of TFA-Fisher importance scores per diagnostic category.

5 Case Study — Persistent False Alarms from Structural Forgetting

Prior evaluations use a synthetic framework; we now validate on a real-world structurally sensitive task. We fine-tune TimesFM 2.0 on *Gold*, *MSFT*, and *SPX500* and evaluate on the NASA SMAP satellite telemetry dataset [20], where the dominant anomaly pattern type is a *Level Shift*—an abrupt, sustained change in signal mean. Anomaly scores are point-wise absolute prediction residuals; metrics are AUC, False Alarm Rate (FAR), and Post-anomaly False Alarm Rate (PFAR), which measures the fraction of timesteps still flagged after a true anomaly ends.

As shown in Table 4, fine-tuning substantially inflates FAR and PFAR across all three settings. This is a direct consequence of structural forgetting: having lost Level Shift awareness, the fine-tuned model fails to re-anchor after an anomaly ends, producing a persistent alarm tail beyond the true anomaly boundary (visualized in Appendix H.2); AUC collapsing toward chance further confirms a breakdown in discriminability. NeST successfully restores AUC, FAR, and PFAR to levels comparable to the base model, confirming that NeST translates to effectiveness in real-world structurally sensitive tasks.

Datasets & Models		AUC ↑	FAR ↓	PFAR ↓
	Base Model	61.02	10.87	21.32
Gold	Fine-tuned	48.22 ↓12.80	13.25 ↑2.38	25.28 ↑3.96
Gold	**NeST (Ours)**	**59.35** ↓1.67	**10.64** ↓0.23	**19.06** ↓2.26
MSFT	Fine-tuned	45.71 ↓15.31	13.90 ↑3.03	24.40 ↑3.08
MSFT	**NeST (Ours)**	**59.82** ↓1.20	**10.68** ↓0.19	**19.43** ↓1.89
SPX	Fine-tuned	53.10 ↓7.92	14.73 ↑3.86	25.96 ↑4.64
SPX	**NeST (Ours)**	**59.54** ↓1.48	**10.94** ↑0.07	**20.90** ↓0.42

Table 4: Anomaly detection performance on NASA SMAP across three fine-tuning sources. Lower FAR and PFAR indicate fewer false alarms.

6 Related Works

Catastrophic Forgetting in Time Series Foundation Models. TSFMs derive their generalization from pre-trained structural awareness [1, 2, 6, 33], yet downstream fine-tuning is well known to incur catastrophic forgetting [10]. Recent studies characterize this for temporal models [10–13, 34], but invariably diagnose it through *point-wise* errors (MSE, MAE, NMSE), leaving the persistence of pre-trained structural representations under adaptation unexamined. Mitigation methods inherit the same blind spot: rehearsal [10], regularization (EWC [17], LwF [18]), low-rank adapters (LoRA [16, 35]), and Model Merging [32] all optimize aggregate predictive error and are blind to the loss of trend, periodicity, or spectral patterns. We close this gap by treating structural preservation, the very capability that distinguishes a foundation model, as a first-class objective for TSFM fine-tuning.

Neuron-Level Interpretability for Foundation Models. Mechanistic interpretability has shown that capabilities in foundation models concentrate in sparse, identifiable neurons [36–39]. In LLMs, this lens has localized neurons for factual knowledge [19], multilingual competence [40], safety [41], and confidence [42]. For time series, Wang et al. [43] take a first step by isolating low-frequency neurons in urban forecasters, a narrow precedent limited to a single pattern. We extend this lens to TSFMs via a unified Temporal-Frequency Attribution analysis, localizing the pattern-critical neurons that govern structural awareness, and exposing how fine-tuning systematically erodes them.

7 Conclusion

In this paper, we systematically investigate *structural forgetting* during the fine-tuning of Time Series Foundation Models (TSFMs), a degradation phenomenon more severe than point-wise error. This forgetting erodes the universal temporal patterns learned during pre-training, causing the forecasted signals to completely lose their informative structure. To address this, we first propose a synthetic diagnostic framework to quantify structural forgetting. Second, we provide a neuron-level mechanistic explanation for this phenomenon, revealing that task-specific gradients unnecessarily conflict with a sparse, highly localized set of pattern-critical neurons. Finally, to resolve this tension, we introduce NeST, a post-hoc and targeted framework that restores these critical neurons while optimally compensating the surrounding parameters. Extensive empirical evaluations demonstrate that NeST effectively recovers structural awareness without sacrificing downstream task gains, and we further validate its practical efficacy in a real-world deployment scenario. We hope this work establishes structural preservation as a first-class objective in future TSFM adaptation.

8 Acknowledgements

This work was supported by the grants from the Natural Science Foundation of China (62572035). This work was also supported by Beijing Natural Science Foundation (7252299). Thanks for the computing infrastructure provided by Beijing Advanced Innovation Center for Big Data and Brain Computing. The corresponding author is Haoyi Zhou.

References

[1] A. Das, W. Kong, R. Sen, and Y. Zhou, "A decoder-only foundation model for time-series forecasting," in *Forty-first international conference on machine learning*, 2024.

[2] A. F. Ansari, L. Stella, C. Turkmen, X. Zhang, P. Mercado, H. Shen, O. Shchur, S. S. Rangapuram, S. P. Arango, S. Kapoor *et al.*, "Chronos: Learning the language of time series," *arXiv preprint arXiv:2403.07815*, 2024.

[3] G. Woo, C. Liu, A. Kumar, C. Xiong, S. Savarese, and D. Sahoo, "Unified training of universal time series forecasting transformers," *arXiv preprint arXiv:2402.02592*, 2024.

[4] A. Pandey, A. Neog, and G. Jajoo, "On the internal semantics of time-series foundation models," *arXiv preprint arXiv:2511.15324*, 2025.

[5] Y. Zou, Z. Wang, D. Klabjan, and H. Liu, "Investigating hallucinations of time series foundation models through signal subspace analysis," in *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[6] M. Goswami, K. Szafer, A. Choudhry, Y. Cai, S. Li, and A. Dubrawski, "Moment: A family of open time-series foundation models," *arXiv preprint arXiv:2402.03885*, 2024.

[7] Y. Fan, T. Song, C. Feng, K. Song, C. Liu, and D. Jiang, "Fine-tuning pre-trained large time series models for prediction of wind turbine scada data," *arXiv preprint arXiv:2412.00403*, 2024.

[8] Y. Fan, T. Song, C. Feng, C. Liu, and D. Jiang, "Wind power prediction using foundation large time series models enhanced by time series prompt in exogenous and tuning forms," *Applied Energy*, vol. 400, p. 126535, 2025.

[9] W. Sun, C. Wu, C. Xie, X. Wang, Y. Guo, Y. Tang, Y. Zhang, K. Li, G. Du, Z. Yang *et al.*, "Fine-tuning enables state of health estimation for lithium-ion batteries via a time series foundation model," *Energy*, vol. 318, p. 134177, 2025.

[10] Q. Besnard and N. Ragot, "Continual learning for time series forecasting: a first survey," *Engineering Proceedings*, vol. 68, no. 1, p. 49, 2024.

[11] N. Karaouli, D. Coquenet, E. Fromont, M. Mermillod, and M. Reyboz, "Are time series foundation models susceptible to catastrophic forgetting?" *arXiv preprint arXiv:2510.00809*, 2025.

[12] L. Zhao and Y. Shen, "Proactive model adaptation against concept drift for online time series forecasting," in *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, 2025, pp. 2020–2031.

[13] S. Khanal, S. Tirupathi, G. Zizzo, A. Rawat, and T. B. Pedersen, "Domain adaptation for time series transformers using one-step fine-tuning," *arXiv preprint arXiv:2401.06524*, 2024.

[14] G. Leitch and J. E. Tanner, "Economic forecast evaluation: profits versus the conventional error measures," *The American Economic Review*, pp. 580–590, 1991.

[15] T. Hong and S. Fan, "Probabilistic electric load forecasting: A tutorial review," *International Journal of Forecasting*, vol. 32, no. 3, pp. 914–938, 2016.

[16] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen *et al.*, "Lora: Low-rank adaptation of large language models." *Iclr*, vol. 1, no. 2, p. 3, 2022.

[17] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[18] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.

[19] D. Dai, L. Dong, Y. Hao, Z. Sui, B. Chang, and F. Wei, "Knowledge neurons in pretrained transformers," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2022, pp. 8493–8502.

[20] D. Entekhabi, E. G. Njoku, P. E. O'neill, K. H. Kellogg, W. T. Crow, W. N. Edelstein, J. K. Entin, S. D. Goodman, T. J. Jackson, J. Johnson *et al.*, "The soil moisture active passive (smap) mission," *Proceedings of the IEEE*, vol. 98, no. 5, pp. 704–716, 2010.

[21] C. Inclan and G. C. Tiao, "Use of cumulative sums of squares for retrospective detection of changes of variance," *Journal of the American Statistical Association*, vol. 89, no. 427, pp. 913–923, 1994.

[22] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[23] M. Mudelsee, "Climate time series analysis," *Atmospheric and*, vol. 397, 2010.

[24] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "Timesnet: Temporal 2d-variation modeling for general time series analysis," *arXiv preprint arXiv:2210.02186*, 2022.

[25] A. Shakeel, T. Tanaka, and K. Kitajo, "Time-series prediction of the oscillatory phase of eeg signals using the least mean square algorithm-based ar model," *Applied Sciences*, vol. 10, no. 10, p. 3616, 2020.

[26] M. Hassanzadeh and B. Shahrrava, "Linear version of parseval's theorem," *IEEE Access*, vol. 10, pp. 27 230–27 241, 2022.

[27] J. Martens and R. Grosse, "Optimizing neural networks with kronecker-factored approximate curvature," in *International conference on machine learning*. PMLR, 2015, pp. 2408–2417.

[28] B. Hassibi, D. G. Stork, and G. J. Wolff, "Optimal brain surgeon and general network pruning," in *IEEE international conference on neural networks*. IEEE, 1993, pp. 293–299.

[29] Y. Liu, G. Qin, X. Huang, J. Wang, and M. Long, "Timer-xl: Long-context transformers for unified time series forecasting," *arXiv preprint arXiv:2410.04803*, 2024.

[30] M. K. Islam, A. Karmacharya, T. Sue, and J. Fox, "Large language models for financial aid in financial time-series forecasting," *arXiv preprint arXiv:2410.19025*, 2024.

[31] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 12, 2021, pp. 11 106–11 115.

[32] E. Yang, L. Shen, G. Guo, X. Wang, X. Cao, J. Zhang, and D. Tao, "Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities, 2024," *URL https://arxiv. org/abs/2408.07666*, vol. 2408, no. 3, 2024.

[33] R. An, H. Qu, W. Fan, X. Shang, and Q. Li, "Dema: dual-path delay-aware mamba for efficient multivariate time series analysis," *Frontiers of Computer Science*, vol. 20, no. 11, p. 2011630, 2026.

[34] Y. Hu, Y. Li, P. Liu, Y. Zhu, N. Li, T. Dai, S. Xia, D. Cheng, and C. Jiang, "A comprehensive and practical benchmark for financial time series forecasting," *Frontiers of Computer Science*, vol. 20, no. 10, p. 2010629, 2026.

[35] D. Gupta, A. Bhatti, S. Parmar, C. Dan, Y. Liu, B. Shen, and S. Lee, "Low-rank adaptation of time series foundational models for out-of-domain modality forecasting," in *Proceedings of the 26th International Conference on Multimodal Interaction*, 2024, pp. 382–386.

[36] H. Zhang, Z. Zhang, M. Wang, Z. Su, Y. Wang, Q. Wang, S. Yuan, E. Nie, X. Duan, F. Han *et al.*, "Locate, steer, and improve: A practical survey of actionable mechanistic interpretability in large language models," *arXiv preprint arXiv:2601.14004*, 2026.

[37] Z. Shi, Z. Wang, T. Chen, S. Gao, H. Zhou, Q. Sun, and J. Li, "Fine-tuned llms know they don't know: A parameter-efficient approach to recovering honesty," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, no. 39, 2026, pp. 32 983–32 991.

[38] J. A. Hassan, M. Abdullah, and D. Aljeaid, "Integrating explainable ai with generative models for iot intrusion detection systems: a systematic review," *Frontiers in Computer Science*, vol. 8, p. 1840175, 2026.

[39] J. R. Kirk, R. E. Wray, and J. E. Laird, "Stars: extending interactive task learning with large language models," *Frontiers in Computer Science*, vol. 8, p. 1848830, 2026.

[40] Y. Zhao, W. Zhang, G. Chen, K. Kawaguchi, and L. Bing, "How do large language models handle multilingualism?" *Advances in Neural Information Processing Systems*, vol. 37, pp. 15 296–15 319, 2024.

[41] Y. Zhao, W. Zhang, Y. Xie, A. Goyal, K. Kawaguchi, and M. Shieh, "Understanding and enhancing safety mechanisms of llms via safety-specific neuron," in *The Thirteenth International Conference on Learning Representations*, 2025.

[42] A. Stolfo, B. Wu, W. Gurnee, Y. Belinkov, X. Song, M. Sachan, and N. Nanda, "Confidence regulation neurons in language models," *Advances in Neural Information Processing Systems*, vol. 37, pp. 125 019–125 049, 2024.

[43] C. Wang, Y. Zhao, S. Cai, and G. Tan, "Investigating pattern neurons in urban time series forecasting," in *The Thirteenth International Conference on Learning Representations*, 2025.

Technical Appendices and Supplementary Material

This supplementary material is organized as follows. Appendix A provides the theoretical derivations underlying the TFA-Fisher importance score and the OBS-based task compensation. Appendix B details the synthetic diagnostic framework, including pattern generation, structural metrics, and the rationale for excluding *Variance Shift*. Appendix C describes datasets, fine-tuning protocols, baselines, evaluation, and compute resources. Appendices D–G report extended empirical evidence, mechanistic analyses, and additional results of NeST. Appendix H supplements the NASA SMAP case study, and Appendix I discusses limitations and broader impacts.

A Theoretical Proofs and Derivations

A.1 Proof 1: Loss Change under Isotropic Perturbation and the Diagonal Hessian

Let θ_0 denote the pre-trained checkpoint. Since the TFA-Loss is evaluated on synthetic patterns the base model already represents well, we have $\nabla_\theta \mathcal{L}_{\mathrm{TFA}}(\theta_0) \approx \mathbf{0}$. For an isotropic perturbation $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$, a second-order Taylor expansion of Eq. (2) yields

$$\mathcal{L}_{\mathrm{TFA}}(\theta_0 + \boldsymbol{\epsilon}) = \mathcal{L}_{\mathrm{TFA}}(\theta_0) + \boldsymbol{\epsilon}^\top \nabla_\theta \mathcal{L}_{\mathrm{TFA}}(\theta_0) + \tfrac{1}{2}\, \boldsymbol{\epsilon}^\top \mathbf{H}\, \boldsymbol{\epsilon} + \mathcal{O}(\|\boldsymbol{\epsilon}\|^3), \tag{10}$$

where $\mathbf{H} = \nabla_\theta^2 \mathcal{L}_{\mathrm{TFA}}(\theta_0)$. Taking expectations and using $\mathbb{E}[\boldsymbol{\epsilon}] = \mathbf{0}$ and $\mathbb{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^\top] = \sigma^2 \mathbf{I}$, the linear term vanishes and the cubic remainder is dominated, giving

$$\mathbb{E}[\Delta \mathcal{L}_{\mathrm{TFA}}] \;=\; \tfrac{\sigma^2}{2}\, \mathrm{Tr}(\mathbf{H}) \;=\; \tfrac{\sigma^2}{2} \sum_i [\mathbf{H}]_{ii}. \tag{11}$$

The expected loss change therefore decomposes additively over parameters, with $[\mathbf{H}]_{ii}$ quantifying the per-parameter contribution under isotropic perturbation. This justifies adopting the diagonal of $\mathbf{H}$ as a per-parameter structural-importance signal. The argument requires only twice differentiability of $\mathcal{L}_{\mathrm{TFA}}$ and is independent of any probabilistic interpretation of the loss.

A.2 Proof 2: Squared-Gradient Approximation of the Hessian Diagonal for $\mathcal{L}_{\mathrm{TFA}}$

Since $\mathcal{L}_{\mathrm{TFA}}$ in Eq. (2) is not a negative log-likelihood, we derive the squared-gradient approximation of $[\mathbf{H}]_{ii}$ directly from its residual structure rather than via a probabilistic argument. Define the composite residual mapping

$$G(\theta) = \begin{bmatrix} \hat{y} \\ \sqrt{\lambda}\, |F(\hat{y})| \end{bmatrix}, \qquad G^\star = \begin{bmatrix} y \\ \sqrt{\lambda}\, |F(y)| \end{bmatrix}, \qquad r = G^\star - G(\theta), \tag{12}$$

so that $\mathcal{L}_{\mathrm{TFA}} = \|r\|_2^2$ with Jacobian $J = \partial G / \partial \theta$. A standard Gauss–Newton decomposition gives

$$\mathbf{H} \;=\; J^\top J \;-\; \sum_k r_k\, \nabla_\theta^2 G_k. \tag{13}$$

(i) Well-definedness of J. $F(\cdot)$ is linear, so the temporal Jacobian is immediate. For the modulus in $\mathcal{L}_{\mathrm{freq}}$, the chain rule yields

$$\frac{\partial\, |F(\hat{y})|_k}{\partial \theta_i} \;=\; \mathrm{Re}\!\left(\frac{F(\hat{y})_k}{|F(\hat{y})_k|} \cdot \frac{\partial F(\hat{y})_k}{\partial \theta_i} \right), \tag{14}$$

which is well-defined except on the measure-zero set $\{|F(\hat{y})_k| = 0\}$. Hence J exists almost everywhere on D_{pattern}.

(ii) Gauss–Newton reduction at θ_0. At the pre-trained checkpoint, f_{θ_0} already represents the general-purpose pre-training distribution that subsumes D_{pattern}, so the time- and frequency-domain residuals satisfy $r \approx \mathbf{0}$. The second term of Eq. (13) therefore vanishes in expectation, leaving

$$\mathbf{H} \;\approx\; J^\top J, \qquad [J^\top J]_{ii} \;=\; \sum_k \left(\frac{\partial G_k}{\partial \theta_i} \right)^2. \tag{15}$$

(iii) Connection to squared gradients. The per-sample gradient is $g_i = \partial \mathcal{L}_{\mathrm{TFA}}/\partial\theta_i = -2\sum_k r_k\, \partial G_k/\partial\theta_i$, so that

$$\mathbb{E}_{x\sim D_{\mathrm{pattern}}}[g_i^2] \;=\; 4\,\mathbb{E}\left[\left(\textstyle\sum_k r_k\, \partial G_k/\partial\theta_i\right)^2\right]. \tag{16}$$

Treating residuals at θ_0 as zero-mean with approximately isotropic covariance $\mathbb{E}[rr^\top] \approx \sigma_r^2\mathbf{I}$ (a standard near-optimum assumption for converged checkpoints), this simplifies to

$$\mathbb{E}[g_i^2] \;\approx\; 4\sigma_r^2 \sum_k \left(\frac{\partial G_k}{\partial\theta_i}\right)^2 \;=\; 4\sigma_r^2\,[J^\top J]_{ii} \;\propto\; [\mathbf{H}]_{ii}. \tag{17}$$

Hence $\mathbb{E}[g_i^2]$ matches $[\mathbf{H}]_{ii}$ up to a parameter-independent constant, so the ranking induced by squared gradients coincides with that induced by the Hessian diagonal. Setting $\mathcal{L}=\mathcal{L}_{\mathrm{TFA}}$ and $D=D_{\mathrm{pattern}}$ recovers Eq. (3), in line with the empirical Fisher form [27, 17] but derived directly from the residual structure of $\mathcal{L}_{\mathrm{TFA}}$ without invoking a probabilistic interpretation of the loss.

A.3 Proof 3 (Hypothesis): Structural-Sensitivity Justification of TFA-Fisher Attribution

Proofs A.1 and A.2 establish that $s_{j,k}^{\mathrm{TFA}}$ approximates the diagonal of the TFA-Loss Hessian and therefore quantifies the contribution of neuron (j,k) to a loss increase under isotropic perturbation. They do *not*, however, address the more fundamental motivating question of our attribution: do the neurons selected by s^{TFA} correspond to the parameters that materially sustain the *temporal-spectral structure* of the model's outputs, rather than merely contributing to point-wise accuracy or to incidental directions of the loss landscape?

We provide a partial answer by relating $s_{j,k}^{\mathrm{TFA}}$ to a notion of structural sensitivity defined directly on output space. The argument is intentionally formulated as a justifying hypothesis rather than a strict equivalence theorem: it relies on a first-order perturbation regime and on the residual (quasi-)isotropy already adopted in Proof A.2. Its purpose is to clarify *why* the empirical identification reported in §3.2.1–§3.2.2 is principled, not to replace that empirical evidence.

Step 1: Output-space structural distance and pattern criticality. We define an output-space distance that mirrors the form of $\mathcal{L}_{\mathrm{TFA}}$ but is evaluated between two model outputs:

$$\mathcal{D}_{\mathrm{struct}}(\hat{y},\hat{y}') \;=\; \|\hat{y}-\hat{y}'\|_2^2 \;+\; \lambda\big\|\,|F(\hat{y})| - |F(\hat{y}')|\,\big\|_2^2. \tag{18}$$

$\mathcal{D}_{\mathrm{struct}}$ jointly penalizes deviation in the temporal trajectory and in the (phase-invariant) amplitude spectrum, and is therefore sensitive to structural rather than purely point-wise discrepancy. We then call neuron (j,k) *pattern-critical* to the extent that a small isotropic perturbation $\delta \sim \mathcal{N}(0,\sigma^2)$ applied to $W_{j,k}$ alone induces a large expected structural distortion of the output:

$$\mathrm{PC}_{j,k} \;:=\; \mathbb{E}_{x\sim D_{\mathrm{pattern}}}\,\mathbb{E}_\delta\big[\mathcal{D}_{\mathrm{struct}}\big(\hat{y}(\theta_0),\; \hat{y}(\theta_0+\delta\, e_{j,k})\big)\big]\,. \tag{19}$$

Equation (19) formalizes "pattern-critical" from the output side: the larger $\mathrm{PC}_{j,k}$, the more strongly this single neuron is responsible for the joint temporal-spectral fingerprint of the network's predictions. Crucially, $\mathrm{PC}_{j,k}$ is defined independently of $s_{j,k}^{\mathrm{TFA}}$.

Step 2: First-order structural sensitivity. Let

$$u_{j,k} \;=\; \left.\frac{\partial \hat{y}}{\partial W_{j,k}}\right|_{\theta_0}, \qquad v_{j,k} \;=\; \left.\frac{\partial |F(\hat{y})|}{\partial W_{j,k}}\right|_{\theta_0}, \tag{20}$$

both well-defined almost everywhere on D_{pattern} by the argument in Proof A.2(i). A first-order Taylor expansion in δ gives

$$\hat{y}(\theta_0+\delta\, e_{j,k}) - \hat{y}(\theta_0) \;=\; \delta\, u_{j,k} + \mathcal{O}(\delta^2), \tag{21}$$

$$\big|F(\hat{y}(\theta_0+\delta\, e_{j,k}))\big| - \big|F(\hat{y}(\theta_0))\big| \;=\; \delta\, v_{j,k} + \mathcal{O}(\delta^2). \tag{22}$$

Substituting these into Eq. (18) and taking expectation over δ (using $\mathbb{E}[\delta]=0$, $\mathbb{E}[\delta^2]=\sigma^2$, and discarding the $\mathcal{O}(\delta^3)$ remainder) yields

$$\mathrm{PC}_{j,k} \;\approx\; \sigma^2 \cdot S_{j,k}, \qquad S_{j,k} \;:=\; \mathbb{E}_x\big[\|u_{j,k}\|_2^2 + \lambda\,\|v_{j,k}\|_2^2\big]\,. \tag{23}$$

$S_{j,k}$ is therefore (up to the global scale σ^2) a clean parameter-side functional that captures how strongly neuron (j,k) shapes both the temporal and the spectral branch of the output.

Step 3: Connecting s^{TFA} and S via the residual structure. From the residual decomposition introduced in Proof A.2,

$$g_{j,k} \;:=\; \frac{\partial \mathcal{L}_{\mathrm{TFA}}}{\partial W_{j,k}} \;=\; -2\, r_t^\top u_{j,k} \;-\; 2\lambda\, r_a^\top v_{j,k}, \tag{24}$$

with $r_t = y - \hat{y}$ and $r_a = |F(y)| - |F(\hat{y})|$. Stacking the joint residual and Jacobian as

$$\tilde{r} \;=\; \begin{bmatrix} r_t \\ \sqrt{\lambda}\, r_a \end{bmatrix}, \qquad \tilde{j}_{j,k} \;=\; \begin{bmatrix} u_{j,k} \\ \sqrt{\lambda}\, v_{j,k} \end{bmatrix}, \tag{25}$$

we obtain $g_{j,k} = -2\,\tilde{r}^\top \tilde{j}_{j,k}$ and hence

$$s_{j,k}^{\mathrm{TFA}} \;=\; \mathbb{E}\big[g_{j,k}^2\big] \;=\; 4\,\tilde{j}_{j,k}^\top\, \mathbb{E}\big[\tilde{r}\,\tilde{r}^\top\big]\, \tilde{j}_{j,k}. \tag{26}$$

Adopting the same residual quasi-isotropy assumption as Proof A.2, $\mathbb{E}\big[\tilde{r}\,\tilde{r}^\top\big] \;\approx\; \sigma_r^2\,\mathbf{I}$, Eq. (26) collapses to

$$s_{j,k}^{\mathrm{TFA}} \;\approx\; 4\sigma_r^2\,\big(\|u_{j,k}\|_2^2 + \lambda\,\|v_{j,k}\|_2^2\big) \;=\; 4\sigma_r^2 \cdot S_{j,k}. \tag{27}$$

Step 4: Equivalence of TFA-Fisher and structural sensitivity (under the stated assumptions). Combining Eqs. (23) and (27),

$$s_{j,k}^{\mathrm{TFA}} \;\approx\; \frac{4\sigma_r^2}{\sigma^2}\,\mathrm{PC}_{j,k}. \tag{28}$$

The proportionality constant $4\sigma_r^2/\sigma^2$ is independent of (j,k). Hence, under joint residual isotropy, ranking neurons by s^{TFA} coincides with ranking them by the structural sensitivity PC of Eq. (19), and the candidate set A_j^{neuron} in §3.1 is precisely the top-$\lfloor d'\rho_{\mathrm{neuron}}\rfloor$ neurons whose single-neuron perturbation most disrupts the temporal-spectral structure of the output. This is the content we informally summarize as "TFA-Fisher selects pattern-critical neurons."

Robustness under block-isotropic residuals. The joint isotropy assumption can be relaxed to block isotropy,

$$\mathbb{E}[r_t r_t^\top] = \sigma_t^2\,\mathbf{I}, \quad \mathbb{E}[r_a r_a^\top] = \sigma_a^2\,\mathbf{I}, \quad \mathbb{E}[r_t r_a^\top] = \mathbf{0}, \tag{29}$$

which subsumes the previous case as $\sigma_t^2 = \lambda\sigma_a^2$. Substituting in Eq. (26),

$$s_{j,k}^{\mathrm{TFA}} \;\approx\; 4\sigma_t^2\,\mathbb{E}\|u_{j,k}\|_2^2 \;+\; 4\lambda^2\sigma_a^2\,\mathbb{E}\|v_{j,k}\|_2^2. \tag{30}$$

Both Eq. (30) and $S_{j,k}$ in Eq. (23) are positive linear combinations of $\mathbb{E}\|u_{j,k}\|_2^2$ and $\mathbb{E}\|v_{j,k}\|_2^2$. Setting $c_{\min} = \min\{\sigma_t^2,\ \lambda\sigma_a^2\}$ and $c_{\max} = \max\{\sigma_t^2,\ \lambda\sigma_a^2\}$, the weight-comparison identity for non-negative linear combinations yields, for every neuron (j,k),

$$4\,c_{\min}\,S_{j,k} \;\le\; s_{j,k}^{\mathrm{TFA}} \;\le\; 4\,c_{\max}\,S_{j,k}. \tag{31}$$

Consequently, for any pair of neurons,

$$\frac{s_{j,k}^{\mathrm{TFA}}}{s_{j',k'}^{\mathrm{TFA}}} \;\in\; \left[\,\frac{c_{\min}}{c_{\max}}\cdot\frac{S_{j,k}}{S_{j',k'}},\ \frac{c_{\max}}{c_{\min}}\cdot\frac{S_{j,k}}{S_{j',k'}}\,\right]. \tag{32}$$

The TFA-Fisher ranking and the structural-sensitivity ranking therefore agree up to a bounded multiplicative distortion governed solely by the relative scale of temporal and spectral residuals. Choosing λ to balance the two loss terms ($\lambda \approx \sigma_t^2/\sigma_a^2$) makes $c_{\min} = c_{\max}$ and recovers exact proportionality, in line with the "balanced TFA-Loss" configuration used throughout our experiments.

Scope and limitations. We deliberately frame the result above as a hypothesis rather than a complete proof. It rests on three modeling choices that should be made explicit. (i) The structural distance $\mathcal{D}_{\mathrm{struct}}$ in Eq. (18) is constructed to mirror $\mathcal{L}_{\mathrm{TFA}}$, so the equivalence in Eq. (28) partly reflects an intentional alignment between the scoring criterion and the target metric; alternative structural distances (e.g. phase-aware or higher-order spectral norms) would yield correspondingly different criticality notions. (ii) The first-order Taylor expansion in δ is valid only in the small-perturbation regime, and may underestimate the criticality of neurons that engage in switching-like or saturating behavior whose effect on output structure is intrinsically nonlinear. (iii) The residual (quasi-)isotropy assumption inherited from Proof A.2 is plausible at a converged pre-trained checkpoint on D_{pattern} but cannot be strictly verified in practice. The empirical adequacy of s^{TFA} for identifying pattern-critical neurons is therefore supported jointly by the analytical motivation above and by the controlled experiments in §3.2.1–§3.2.2.

A.4 Proof 4: Optimal-Brain-Surgeon Compensation Derivation

Reverting $\mathcal{A}^{pc}$ via Eq. (7) introduces a task discrepancy

$$d_{\text{task}} = \left\| \mathbf{W}^{res} \mathbf{X}_{\text{task}} - \mathbf{W}^{ft} \mathbf{X}_{\text{task}} \right\|_2^2, \tag{33}$$

which, if left uncorrected, degrades downstream performance. We seek a compensation vector $\mathbf{c}$ applied to the non-critical parameters that minimizes d_{task} under a quadratic (local) approximation.

Let $\mathbf{H} = \nabla^2_{\mathbf{W}^{ft}} d_{\text{task}}$ be the layer-wise Hessian evaluated on $\mathcal{D}_{\text{task}}$. For a single pattern-critical parameter at index m, the reversion amount is $\delta_m = \mathbf{W}^{pre}_m - \mathbf{W}^{ft}_m$. Following the Optimal Brain Surgeon framework [28], we seek the minimum-norm compensation that exactly neutralizes the effect of this single-parameter change:

$$\min_{\mathbf{c}} \ \tfrac{1}{2}\, \mathbf{c}^\top \mathbf{H}\, \mathbf{c} \quad \text{s.t.} \quad \mathbf{e}_m^\top \mathbf{c} = \delta_m. \tag{34}$$

The Lagrangian $\mathcal{L} = \frac{1}{2}\mathbf{c}^\top \mathbf{H}\mathbf{c} + \lambda(\delta_m - \mathbf{e}_m^\top \mathbf{c})$ yields the stationarity condition $\mathbf{H}\mathbf{c} = \lambda \mathbf{e}_m$, so $\mathbf{c} = \lambda \mathbf{H}^{-1}\mathbf{e}_m = \lambda \mathbf{H}^{-1}_{:,m}$. Substituting into the constraint:

$$\mathbf{e}_m^\top \mathbf{c} = \lambda\, [\mathbf{H}^{-1}]_{mm} = \delta_m \implies \lambda = \frac{\delta_m}{[\mathbf{H}^{-1}]_{mm}}. \tag{35}$$

Replacing $\delta_m = \mathbf{W}^{pre}_m - \mathbf{W}^{ft}_m$ and denoting the compensation for parameter m as $\mathbf{C}_m$,

$$\mathbf{C}_m = \frac{\delta_m}{[\mathbf{H}^{-1}]_{mm}}\, \mathbf{H}^{-1}_{:,m} = -\frac{\mathbf{W}^{ft}_m - \mathbf{W}^{pre}_m}{[\mathbf{H}^{-1}]_{mm}}\, \mathbf{H}^{-1}_{:,m}, \tag{36}$$

which is exactly Eq. (8). The total compensation $\sum_{m \in \mathcal{A}^{pc}} \mathbf{C}_m$ is projected onto non-critical parameters via $(\mathbf{1} - \mathbf{M}) \odot \mathbf{C}$ in Eq. (9), ensuring the reverted entries of $\mathcal{A}^{pc}$ remain pinned to $\mathbf{W}^{pre}$.

B Synthetic Diagnostic Framework

B.1 Generation Procedures of the Seven Structural Patterns

Sequence lengths are set according to each model's official recommended inference configuration rather than a single fixed length. For each of the seven patterns we generate two non-overlapping splits. The *test set* contains $N = 1{,}000$ independently sampled sequences per pattern on which all structural metrics are computed. The *calibration set* $\mathcal{D}_{\text{pattern}}$ referenced in the main text as the structural input to TFA-Fisher scoring contains an additional $N_{\text{cal}} = 500$ sequences per pattern generated under identical distributions but disjoint random seeds, preventing any data leakage from calibration to evaluation.

D1 — AR(1) Process.

$$x_t = \phi\, x_{t-1} + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, \sigma^2), \tag{37}$$

with $|\phi| > 0.5$ (strong autocorrelation regime) and x_0 initialized at the stationary distribution. This pattern captures the serial memory structure pervasive in financial returns, climate anomalies, and neurophysiological signals. The autoregressive coefficient ϕ and noise scale σ are randomized across sequences to ensure coverage of diverse memory strengths. *Normalization*: per-series z-score.

D2 — Stochastic Trend with Drift.

$$x_t = x_{t-1} + \mu + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, \sigma^2), \tag{38}$$

with small drift μ and moderate noise σ, initialized at $x_0 = 0$. This pattern tests whether a model can extrapolate unbounded cumulative displacement without spuriously mean-reverting, a hallmark of asset price dynamics and macroeconomic trend series. No normalization is applied, as the non-stationary scale is the diagnostic signal.

D3 — Deterministic Trend.

$$x_t = \beta\, t + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, \nu^2), \tag{39}$$

with slope β randomized in sign and magnitude, and additive noise ν varied across sequences. Deterministic linear trends are representative of long-run processes in population growth, climate projections, and demand forecasting. The metric tests precise trend-slope extrapolation, which differs from the unit-root behavior in D2. *Normalization*: per-series z-score.

D4 — Level Shift.

$$x_t = \eta_t + \Delta \cdot \mathbf{1}[t \geq \tau], \quad \eta_t \sim \mathcal{N}(0, \nu^2), \tag{40}$$

with shift time τ positioned near the end of the context window and shift magnitude $\Delta \geq 3\nu$ (jump exceeding three pre-shift standard deviations) drawn with random sign. This pattern captures abrupt distributional regime changes common in sensor fault detection, policy interventions, and anomaly scenarios. Positioning τ near the context boundary ensures the model has limited exposure to the post-shift regime, sustaining non-trivial adaptation pressure. *Normalization*: none.

D5 — Variance Shift.

$$x_t \sim \begin{cases} \mathcal{N}(0, \sigma_1^2) & t < \tau, \\ \mathcal{N}(0, \sigma_2^2) & t \geq \tau, \end{cases} \tag{41}$$

with post-shift volatility amplified by $2\times$–$5\times$ relative to the pre-shift level ($\sigma_2 \in [2\sigma_1, 5\sigma_1]$). Abrupt volatility amplification is characteristic of financial stress events, industrial fault onset, and meteorological extremes. This pattern is excluded from the main retention analysis for the reasons detailed in §B.3. *Normalization*: none.

D6 — Spectral Synthesis.

$$x_t = \sum_{j=1}^{k} a_j \sin(2\pi f_j t + \phi_j) + \varepsilon_t, \quad 0 < f_j < 0.5, \tag{42}$$

with $k \in \{1, \ldots, 5\}$ components whose frequencies f_j are constrained to be well-separated (minimum separation 0.01) to avoid spectral leakage. Amplitudes a_j and phases ϕ_j are randomized independently. Multi-component periodic signals arise in machinery vibration analysis, power-grid harmonics, and atmospheric science. The PSD-similarity metric measures whether the model reproduces the correct energy distribution across frequency bands. *Normalization*: per-series z-score.

D7 — Time-Warped Sinusoid. A base sinusoid $b_s = \sin(2\pi f s + \phi)$ is warped by a monotone time map $u(t)$ obtained by accumulating i.i.d. $\mathrm{Gamma}(\alpha, 1)$ increments and re-interpolated onto the regular grid:

$$x_t = \mathrm{interp}(u,\, b,\, t) + \varepsilon_t. \tag{43}$$

The warp strength is governed by α: smaller values yield heavier-tailed inter-sample gaps (stronger instantaneous-frequency modulation). This pattern models non-stationary periodicity encountered in biological rhythm analysis (ECG, EEG), speech processing, and mechanical systems subject to load variation. DTW distance is used as the metric because it is tolerant of the resulting phase and frequency modulation. *Normalization*: per-series z-score.

B.2 Pattern-Specific Structural Metrics

We define one structural metric per pattern, chosen to capture the defining property. Table 5 lists all seven metrics.

Table 5: Pattern-specific structural metrics and their SRR directionality.

Pattern	Metric	Definition	SRR dir.
AR(1)	ACF-L2	$\left\|\widehat{\mathrm{ACF}}(\hat{y}) - \widehat{\mathrm{ACF}}(y)\right\|_2$ (lags 1–50)	$\downarrow$
Stochastic Trend	OU coeff. $\hat{\lambda}$	OLS slope of $\Delta\hat{y}_t = \lambda(\mu - \hat{y}_{t-1}) + \epsilon_t$	$\to 0$
Det. Trend	Slope error	$\lvert\hat{\beta} - \beta\rvert \,/\, \lvert\beta\rvert$	$\downarrow$
Level Shift	MRB	$\lvert\bar{\hat{y}}_{post} - \mu_{new}\rvert \,/\, \lvert\Delta\mu\rvert$	$\downarrow$
Spectral	PSD sim.	Pearson corr. $r(P_y,\, P_{\hat{y}})$	$\uparrow$
Time-Warped	DTW	Dynamic time-warping distance	$\downarrow$
Variance Shift	VTR	$\mathrm{Std}(\hat{y}_{post})/\mathrm{Std}(y_{pre}) \;\div\; \sigma_{new}/\sigma_{old}$	$\to 1$

ACF-L2 (AR(1)). Let $\hat{\rho}_k$ and ρ_k be the sample ACF at lag k of the predicted and ground-truth segments respectively.

$$\text{ACF-L2} = \sqrt{\sum_{k=1}^{50} (\hat{\rho}_k - \rho_k)^2}. \tag{44}$$

A larger value indicates deviation from the true autocorrelation structure, i.e., degraded "memory" representation.

OU Mean-Reversion Coefficient (Stochastic Trend). For predictions $\hat{y}_{T+1}, \ldots, \hat{y}_{T+H}$, let μ be the mean of the observed context. We fit via OLS:

$$\Delta\hat{y}_t = \lambda(\mu - \hat{y}_{t-1}) + \epsilon_t, \tag{45}$$

and record $\hat{\lambda}$. Under a true Stochastic Trend the future increment is independent of the current level, so $\hat{\lambda}^* = 0$; a positive $\hat{\lambda}$ reveals a spurious mean-reversion bias induced by fine-tuning.

Relative Slope Error (Deterministic Trend). Let $\hat{\beta}$ denote the OLS slope of the predicted segment. The metric $|\hat{\beta} - \beta|/|\beta|$ quantifies fractional degradation in trend-extrapolation accuracy.

Mean Recovery Bias — MRB (Level Shift).

$$\mathrm{MRB} = \frac{|\bar{\hat{y}}_{post} - \mu_{new}|}{|\Delta\mu|}, \tag{46}$$

where $\bar{\hat{y}}_{post}$ is the mean of predictions over the post-shift horizon, μ_{new} the true post-shift level, and $\Delta\mu$ the jump magnitude. $\mathrm{MRB} = 0$ corresponds to perfect level recovery; a rising post-fine-tuning MRB indicates the model predicts a value between the old and new level, i.e., fails to adapt.

PSD Similarity (Spectral). Let $P_y(f)$ and $P_{\hat{y}}(f)$ be the power spectral densities of the ground-truth and predicted segments computed via FFT:

$$P(f) = \frac{1}{H}\left|\sum_{h=1}^{H} y_{T+h}\, e^{-i2\pi fh/H}\right|^2. \tag{47}$$

The metric is the Pearson correlation $r(P_y, P_{\hat{y}})$, which is insensitive to phase offset and focuses on frequency-domain energy matching.

DTW Distance (Time-Warped Sinusoid). Dynamic time warping allows non-linear temporal alignment, making it tolerant of the phase and instantaneous-frequency modulation introduced by the warp map. A smaller DTW value indicates better structural tracking.

VTR (Variance Shift).

$$\mathrm{VTR} = \frac{\mathrm{Std}(\hat{y}_{post}) \,/\, \mathrm{Std}(y_{pre})}{\sigma_{new} \,/\, \sigma_{old}}, \tag{48}$$

where $\mathrm{Std}(\hat{y}_{post})$ is the standard deviation of predicted values and $\sigma_{new}/\sigma_{old}$ is the ground-truth volatility change ratio. The ideal value is 1; $\mathrm{VTR} < 1$ indicates under-prediction of the volatility increase.

B.3 Why *Variance Shift* Is Excluded: High-Frequency Capacity Audit

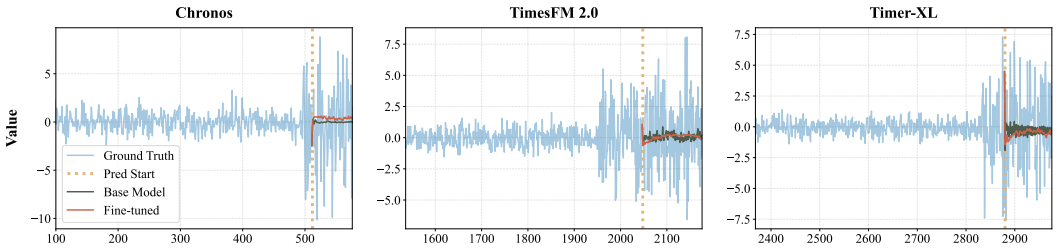


Figure 7: Volatility Tracking Ratio (VTR) of all evaluated TSFMs on *Variance Shift* sequences *before* any fine-tuning. All models exhibit a very low Volatility Tracking Ratio, confirming a fundamental absence of high-frequency volatility prediction capacity that is independent of fine-tuning.

Variance Shift tests whether a model can track an abrupt amplification of signal volatility, a capability that requires generating higher-frequency fluctuations after the regime transition rather than adjusting the mean level or spectral composition. As shown in Figure 7, all evaluated TSFMs exhibit a very

low Volatility Tracking Ratio (VTR) **even prior to any fine-tuning**, meaning the predicted post-shift volatility cannot recover. This is a fundamental architectural limitation of current TSFMs, not an artifact of downstream adaptation. We therefore exclude *Variance Shift* from all retention analyses, only this pattern fails the criterion.

Nevertheless, we emphasize that *Variance Shift* represents a structurally important and practically prevalent pattern. Volatility regime transitions arise across a broad range of high-stakes domains, including financial markets, industrial sensor streams, biomedical signals such as EEG and ECG and meteorological records. We therefore retain *Variance Shift* as a forward-looking benchmark within the diagnostic suite, anticipating that TSFMs with stronger high-frequency modeling capacity will benefit from this evaluation.

C Experimental Setup

C.1 Fine-Tuning Datasets

We evaluate on nine downstream datasets spanning financial, energy, and industrial electricity domains, summarized in Table 6. The financial datasets (Gold, SPX500, MSFT, Apple, Exchange Rate) cover commodity prices, equity indices, and foreign exchange. Crude Oil and Natural Gas represent energy commodity markets. ETTh1 and ETTm1 are electricity transformer temperature datasets sampled at hourly and 15-minute resolution, respectively, drawn from the widely used ETT benchmark.

Table 6: Summary of the nine fine-tuning datasets.

Dataset	**Domain**	**Frequency**
Gold	Commodity (Precious Metals)	Daily
SPX500	Equity Index	Daily
MSFT	Equity (Technology)	Daily
Apple	Equity (Technology)	Daily
Crude Oil	Commodity (Energy)	Daily
Natural Gas	Commodity (Energy)	Daily
Exchange Rate	Foreign Exchange / Macro	Daily
ETTh1	Electricity / Industrial	Hourly
ETTm1	Electricity / Industrial	15-Min

Each dataset is sliced into non-overlapping windows using the context and prediction horizon lengths prescribed by each model's official inference configuration. The resulting windows are partitioned chronologically into train, validation, and test sets in an 8:1:1 ratio, preserving temporal order throughout. The validation split serves a dual role: it is used for early stopping during fine-tuning and, crucially, constitutes $\mathcal{D}_{\text{task}}$ for the OBS Task Compensation step described in the main text. Using the validation set rather than any portion of the test set ensures complete separation between the calibration inputs of NeST and the held-out evaluation data, preventing data leakage.

C.2 Fine-Tuning Settings

All models are fine-tuned in a full-parameter regime, no layers are frozen and no parameter-efficient adapters are applied, so that the fine-tuned checkpoint reflects the kind of unconstrained adaptation most commonly encountered in practice. We perform a joint grid search over batch size $\in \{32, 64, 128, 256\}$ and learning rate $\in \{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}, 10^{-8}, 10^{-9}\}$ using the AdamW optimizer with default momentum parameters. All configurations are trained for a fixed budget of 10 epochs. The checkpoint achieving the lowest MSE on the validation split is selected as the basis for all subsequent experiments, ensuring that the model under examination represents the practically optimal fine-tuned state rather than an arbitrarily chosen training snapshot. This protocol mirrors the standard deployment workflow in which practitioners fine-tune to convergence and ship the best-performing checkpoint, making the observed structural forgetting and its remediation directly relevant to real-world scenarios.

C.3 Evaluation Settings

Inference configurations follow each model's officially recommended settings. Specifically, we adopt context and horizon lengths of (2048, 128) for TimesFM 2.0, (2880, 96) for Timer-XL, and (512, 64) for Chronos-bolt-base, consistent with the respective technical reports and model cards.

For datasets whose total length is insufficient to support the full context window under the above configurations, including Gold, Apple, and similarly short financial series, we apply a reduced setting of (96, 24) uniformly across all models. This adjustment preserves a valid and non-overlapping train/validation/test partition while keeping the evaluation protocol identical across models on the same dataset. All structural metrics and task performance numbers reported in both the main paper and the appendix are computed under these per-dataset context and horizon assignments.

C.4 Baseline Implementations

We compare NeST against five baselines spanning training-time intervention and post-hoc intervention paradigms.

Rehearsal. Structural synthetic data $\mathcal{D}_{\text{pattern}}$ is replayed alongside the downstream task data during fine-tuning. Because the two data sources differ in context length and sampling frequency, they are processed in separate batches with alternating gradient accumulation rather than being mixed within a single batch, which would distort temporal semantics. The replay weight $\lambda \in \{0.1, 0.5, 1.0, 2.0\}$ controlling the relative contribution of the synthetic loss is tuned per dataset.

EWC. The Fisher information matrix is computed once on $\mathcal{D}_{\text{pattern}}$ at the pre-trained weights $\mathbf{W}^{pre}$, approximated by the squared gradient of the MSE loss. During fine-tuning an elastic penalty $\frac{\lambda_{\text{ewc}}}{2}\sum_i F_i(\theta_i - \theta_i^*)^2$ is added to the task objective. The regularization strength $\lambda_{\text{ewc}} \in \{0.1, 1, 10, 100\}$ is selected by grid search.

LwF. Before fine-tuning begins, the pre-trained model generates soft prediction targets $\tilde{y} = f_{\theta^*}(x)$ for all inputs in $\mathcal{D}_{\text{task}}$ and caches them. During fine-tuning a knowledge distillation loss $\|f_\theta(x) - \tilde{y}\|_2^2$ is added with weight $\lambda_{\text{lwf}} \in \{0.1, 1, 10\}$ alongside the standard MSE task loss.

LoRA. Low-rank adaptation matrices are inserted into the linear projection layers of each TSFM. The pre-trained weights are frozen and only the rank-decomposition factors are updated. The rank $r \in \{4, 8, 16, 64\}$ is tuned per model, along with the scaling factor $\alpha_{\text{lora}} \in \{8, 16, 32\}$. All other training settings follow the full-parameter protocol in Appendix C.2.

Model Merging. The pre-trained weights $\mathbf{W}^{pre}$ and the full-parameter fine-tuned weights $\mathbf{W}^{ft}$ are combined by linear interpolation $(1-\alpha)\mathbf{W}^{pre} + \alpha\mathbf{W}^{ft}$. The mixing coefficient α is swept over $\{0.1, 0.2, \ldots, 0.9\}$, and the Pareto-optimal point jointly maximizing structural retention and task gain is reported.

Reported results. For all baselines, the result recorded in the main paper and in the appendix tables is the outcome of the above hyperparameter grid searches, selecting the configuration that achieves the best joint trade-off between Structural Retention Rate and downstream task gain as measured on the validation set. This ensures a fair comparison where every baseline is presented at its strongest attainable setting.

C.5 Freeze $\mathcal{A}^{pc}$ Experiment Details

The ablation in Section 3 compares three fine-tuning arms: Vanilla SFT, Freeze $\mathcal{A}^{pc}$, and Freeze Random. All three arms share the same optimizer, learning rate schedule, and training budget as described in Appendix C.2; the only difference is which parameters receive gradient updates.

For Freeze $\mathcal{A}^{pc}$, after the pattern-critical neuron set $\mathcal{A}^{pc}$ is identified via TFA-Fisher importance and layer displacement (as described in the main text), the gradients of all parameters belonging to $\mathcal{A}^{pc}$ are zeroed out at every training step before the optimizer update. This ensures the pre-trained values of these neurons are preserved exactly throughout fine-tuning, without any modification to the optimizer state or the loss function. For Freeze Random, we randomly sample a set of neurons whose total parameter count exactly matches that of $\mathcal{A}^{pc}$, and apply the same gradient-zeroing procedure to this randomly selected subset. The random selection is stratified by layer to match the layer-wise distribution of $\mathcal{A}^{pc}$, ensuring the comparison is controlled for both size and depth placement.

We note that Freeze Random occasionally yields a modest improvement in downstream task gain relative to Vanilla SFT, which may appear counterintuitive. We do not treat this as the focus of the ablation, but we observe it is not anomalous: freezing an arbitrary subset of parameters reduces the effective number of trainable parameters, which acts as an implicit regularizer and can alleviate overfitting on small downstream datasets. This effect is well-documented in the transfer learning literature and does not undermine the central conclusion of the ablation, namely that the structural benefit of Freeze $\mathcal{A}^{pc}$ far exceeds that of Freeze Random despite identical parameter counts, confirming that the identity of the frozen neurons matters far more than their quantity.

C.6 Hardware and Compute Budget

All experiments are conducted on a server equipped with 1 NVIDIA A100 GPU (40 GB HBM2), dual Intel Xeon Platinum 8358 CPUs, and 512 GB of system RAM. The software environment uses PyTorch 2.1 with CUDA 12.1. Each fine-tuning run across the hyperparameter grid for a single model and dataset completes within approximately 1 hour on a single GPU. The full NeST post-hoc correction pipeline, including TFA-Fisher scoring, pattern-critical neuron identification, and OBS compensation, completes in under 2 minutes per model per dataset and introduces no additional GPU-hours to the reported budget.

D Additional Empirical Evidence of Structural Forgetting

The main paper reports structural forgetting for TimesFM 2.0 and Timer-XL on five financial datasets (Table 1). This appendix extends that evidence along two axes: additional qualitative visualizations across further TSFM backbones and downstream datasets, and a comprehensive quantitative table covering all three TSFMs across the remaining fine-tuning datasets. Together, the results confirm that structural forgetting is not confined to any particular architecture or data domain but is a universal consequence of unconstrained fine-tuning.

D.1 Visualizations across More TSFMs and Datasets

Figures 8–11 visualize representative predictions before and after fine-tuning across additional TSFM–dataset combinations. In every case, fine-tuning systematically degrades structural properties: predicted trends flatten or reverse, autocorrelation structure collapses, and spectral composition diverges from the ground truth. These qualitative breakdowns are consistent with the pattern shown in Figure 2 of the main paper and persist across typical TSFM architectures, as well as across more downstream datasets.

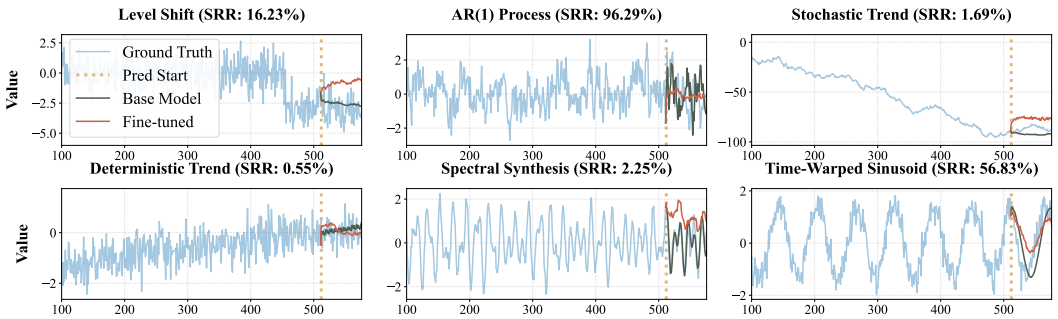


Figure 8: Structural forgetting in Chronos after fine-tuning on the Gold dataset across six retained pattern types. Pattern-specific structural collapse is qualitatively consistent with that observed in TimesFM 2.0 (Figure 2).

D.2 Quantitative Results across More TSFMs and Datasets

Table 7 extends Table 1 to cover the remaining downstream datasets for TimesFM 2.0 and Timer-XL, and all nine datasets for Chronos. Structural forgetting is pervasive: across all entries, the mean Structural Retention Rate falls below 40%, with the majority of individual pattern scores dropping

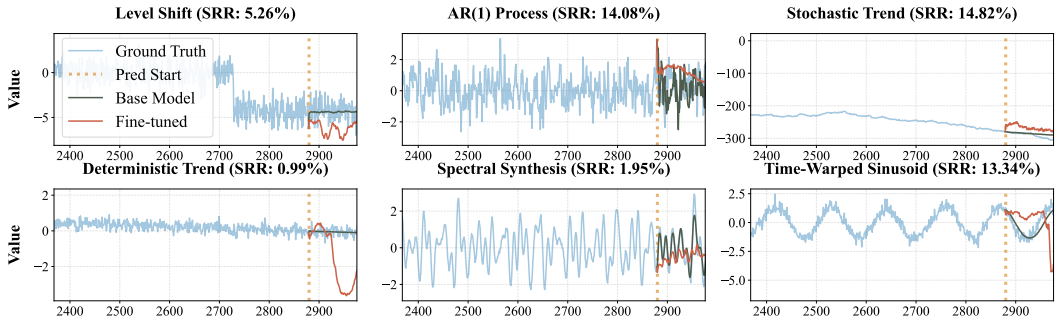


Figure 9: Structural forgetting in Timer-XL after fine-tuning on the Gold dataset across six retained pattern types.

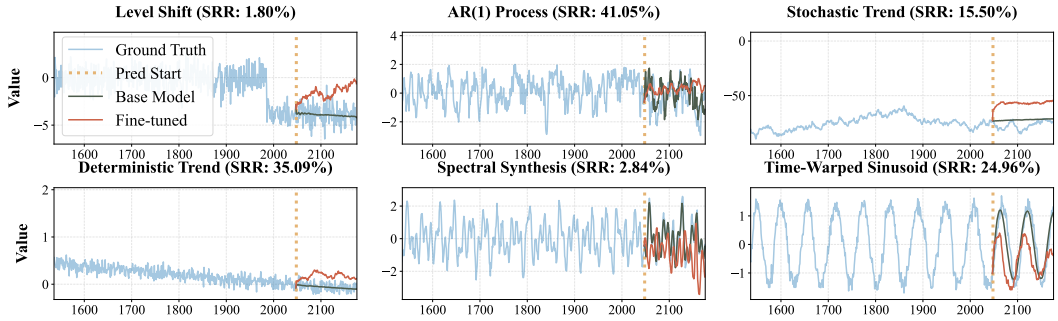


Figure 10: Structural forgetting in TimesFM 2.0 after fine-tuning on the SPX500 dataset across six retained pattern types.

beneath the 35% threshold (highlighted in red). The AR(1) category consistently achieves higher retention than distributional-shift and spectral categories across all models and datasets, likely because short-range autocorrelation is partially preserved by residual sequential dependencies in the fine-tuned weights. Notably, ETT-family datasets (ETTh1, ETTm1) induce severe structural forgetting despite yielding modest task gains, indicating that even low-magnitude fine-tuning suffices to erase structural representations when the target distribution diverges substantially from the pre-training corpus.

E Extended Mechanistic Analysis

E.1 Full-Trajectory Sign Conflict Ratio

The main paper reports SCR distributions at early-phase checkpoints (steps 50, 100, 150, 200) and notes that late-phase measurements revert toward 0.5, precluding their use as evidence of directional conflict. Figure 12 provides the full fine-tuning trajectory to substantiate this claim. We compute the layer-averaged SCR for $\mathcal{A}^{pc}$ and $\mathcal{A}^{\text{other}}$ at every epoch under the same experimental setup as Section 3.2.1 (TimesFM 2.0, Gold dataset, three independent runs). The trajectory reveals a consistent two-phase pattern. In the *early phase* (epochs 1–3), the SCR of $\mathcal{A}^{pc}$ is above 0.5, corroborating the directional conflict reported in the main paper. This elevation reflects the opposition between task gradients and structural preservation objectives before the structural signal in $\mathcal{A}^{pc}$ has been substantially degraded. In the *late phase* (epochs 4 onward), the SCR of $\mathcal{A}^{pc}$ declines monotonically back toward 0.5, converging to the same level as $\mathcal{A}^{\text{other}}$. This convergence, however, does not indicate resolution of the conflict: rather, it reflects the progressive erasure of structural information from pattern-critical

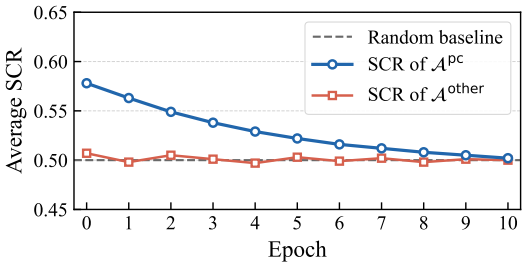


Figure 12: Full-trajectory SCR of $\mathcal{A}^{pc}$ and $\mathcal{A}^{\text{other}}$.

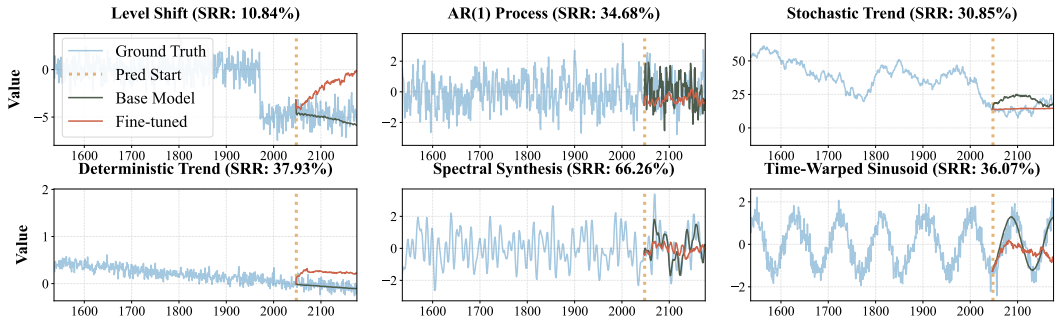


Figure 11: Structural forgetting in TimesFM 2.0 after fine-tuning on the Crude Oil dataset across six retained pattern types.

Table 7: Extended structural forgetting quantification for TimesFM 2.0, Timer-XL, and Chronos across the remaining fine-tuning datasets not reported in Table 1. Values indicate Retention Rate $<35\%$.

Model & Dataset		Δ Task Gain↑ (%)		Structural Retention Rate↑ (%)						
		MSE	MAE	LS	AR	ST	DT	SS	TW	**AVG**
TimesFM	Apple	+9.63	+6.58	44.76	50.23	39.53	27.35	71.87	45.17	46.49
	Natural Gas	+6.21	+4.07	17.59	53.83	59.84	49.82	50.33	58.29	48.28
	ETTh1	+1.81	+0.52	32.14	48.52	29.86	34.11	41.29	31.58	36.25
	ETTm1	+9.08	+3.15	28.41	32.47	36.57	21.17	38.92	48.64	34.36
Timer-XL	Apple	+6.42	+4.11	29.17	48.22	27.84	24.16	34.39	41.52	34.22
	Natural Gas	+4.12	+2.84	39.15	45.73	51.22	42.67	52.11	41.38	45.38
	ETTh1	+1.01	+0.65	25.69	43.87	31.21	28.52	44.14	32.64	34.34
	ETTm1	+0.76	+0.21	36.27	31.83	29.54	27.81	33.49	44.53	33.91
Chronos	Apple	+8.55	+5.27	22.13	41.59	28.42	19.65	35.29	42.63	31.62
	Crude Oil	+4.22	+2.94	38.56	49.12	34.29	31.52	40.81	33.47	37.96
	Exchange Rate	+3.18	+1.89	29.47	42.26	25.18	24.68	32.76	40.55	32.48
	Gold	+5.44	+3.73	33.14	45.62	36.85	29.59	41.28	48.26	39.12
	MSFT	+7.28	+4.85	24.52	40.22	22.89	18.47	31.53	41.16	29.80
	Natural Gas	+6.83	+4.58	27.67	33.76	39.12	32.49	38.56	43.46	35.84
	SPX500	+8.16	+5.58	21.48	41.52	26.32	17.85	29.48	40.82	29.58
	ETTh1	+2.13	+0.37	28.51	34.21	31.48	26.75	35.87	42.56	33.23
	ETTm1	+7.16	+2.45	25.21	41.48	28.66	21.33	33.12	44.28	32.35

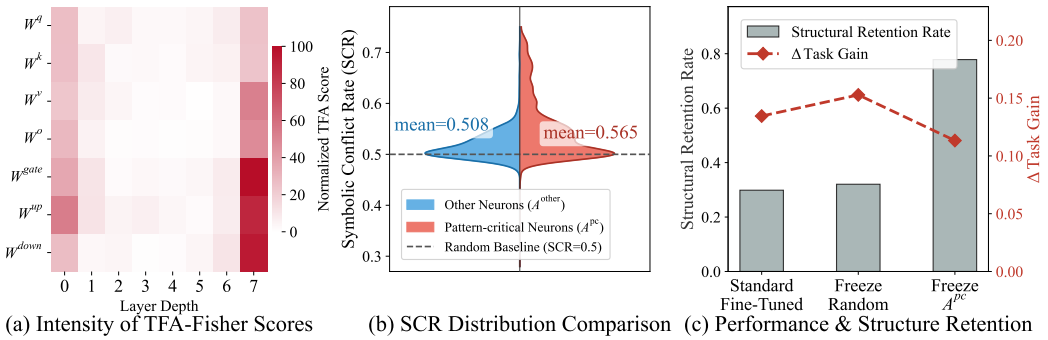
Note: **LS**: Level Shift, **AR**: AR(1) Process, **ST**: Stochastic Trend, **DT**: Deterministic Trend, **SS**: Spectral Synthesis, **TW**: Time-Warped Sinusoid.

neurons, so that $\mathcal{L}_{\text{TFA}}$ gradients lose their structural sensitivity and become effectively random in sign. The SCR of $\mathcal{A}^{\text{other}}$ remains near 0.5 throughout, confirming that non-critical neurons exhibit no systematic gradient alignment with either objective at any stage of training. This trajectory therefore validates the exclusive focus on early-phase SCR in the main analysis.

We also considered gradient cosine similarity as an alternative measure of directional conflict between $\nabla_\theta \mathcal{L}_{\text{task}}$ and $\nabla_\theta \mathcal{L}_{\text{TFA}}$. However, in high-dimensional parameter spaces, the cosine similarity between two random vectors concentrates sharply around zero as dimension increases; in practice, all per-neuron cosine similarity values collapsed to near zero regardless of epoch or neuron group, providing no discriminative signal. SCR avoids this concentration effect by aggregating element-wise sign agreements rather than computing a single dot-product similarity, and retains a meaningful reference point: a value above 0.5 indicates systematic directional opposition that cannot arise by chance, making it a more suitable indicator of gradient conflict in high-dimensional TSFM settings.

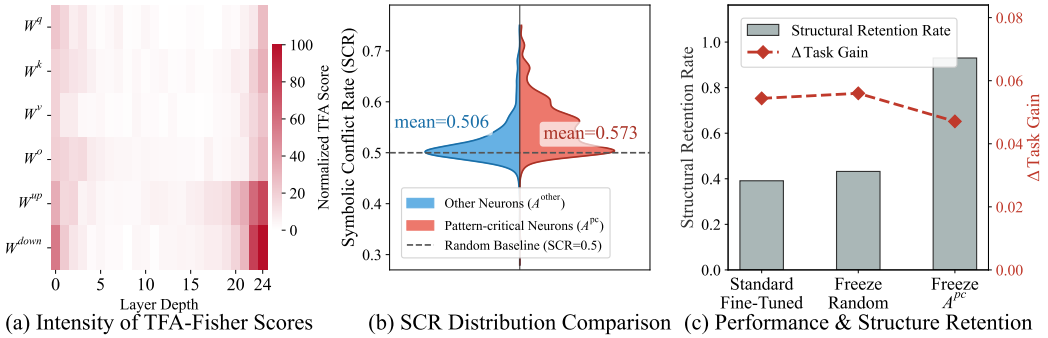
E.2 More Results of Section 3

We apply the same mechanistic analysis from Section 3 to Timer-XL and Chronos. As shown in Figures 13 and 14, the observed patterns are consistent across both models. TFA-Fisher importance concentrates in the early and terminal MLP layers with relatively sparse representation in intermediate layers (panel a). The SCR distributions (panel b) show that $\mathcal{A}^{pc}$ is shifted noticeably above 0.5 at early checkpoints, while $\mathcal{A}^{\text{other}}$ remains centered near 0.5 throughout, with no systematic directional bias. In the freeze ablation (panel c), *Freeze* $\mathcal{A}^{pc}$ recovers a substantially higher Structural Retention Rate than Vanilla SFT with only a small reduction in task gain, whereas *Freeze Random* provides no meaningful structural benefit. These observations hold across both architectures and are consistent with what was reported for TimesFM 2.0 in the main paper.



(a) Intensity of TFA-Fisher Scores (b) SCR Distribution Comparison (c) Performance & Structure Retention

Figure 13: Analysis of why fine-tuning destroys structural awareness in Timer-XL (Gold dataset).



(a) Intensity of TFA-Fisher Scores (b) SCR Distribution Comparison (c) Performance & Structure Retention

Figure 14: Analysis of why fine-tuning destroys structural awareness in Chronos (Gold dataset).

F Additional Results of NeST

Table 2 in the main paper evaluates NeST and all baselines on TimesFM 2.0 across three financial datasets. Here we extend these experiments along two orthogonal axes: (i) two additional TSFMs—Timer-XL and Chronos—on the same three datasets, and (ii) three additional datasets—Apple, ETTh1, and ETTm1—on TimesFM 2.0. Across all configurations, NeST consistently achieves the highest Structural Retention Rate while preserving task gain at a level comparable to Vanilla SFT. Baselines either fail to recover structural capacity sufficiently or do so at the cost of substantially reduced task improvement.

F.1 Additional Results across models

Tables 8 and 9 report results for Timer-XL and Chronos, respectively. Despite notable differences in model architecture and pre-training scale, both exhibit severe structural forgetting under Vanilla SFT, with average SRR falling below 45% in all cases. NeST recovers structural retention to above 75% on Timer-XL and above 80% on Chronos across all three datasets, while task gains remain within

0.5% of Vanilla SFT in nearly every setting. Among baselines, LoRA provides the most consistent structural benefit, consistent with its parameter-efficient nature reducing gradient interference, but its SRR still falls 15–25 percentage points short of NeST. EWC and LwF impose regularization costs that dampen task gain without achieving comparable structural recovery.

Table 8: Comparison of downstream task performance (Δ Task Gain) and Structural Retention Rate on Timer-XL. **Best results** are in bold and second-best are underlined.

Dataset & Method		Δ Task Gain↑ (%)		Structural Retention Rate↑ (%)						
		MSE	MAE	LS	AR	ST	DT	SS	TW	**AVG**
Gold	Vanilla SFT	+13.45	+6.82	23.17	53.08	19.63	16.41	37.07	29.72	29.85
	LoRA	+9.12	+4.66	54.80	**70.53**	35.26	41.60	55.92	49.72	51.30
	Rehearsal	+11.74	+5.89	38.27	58.54	26.65	25.27	43.79	39.41	38.66
	EWC	+5.87	+2.94	43.84	65.31	26.67	31.11	48.95	38.13	42.34
	LwF	+7.36	+3.70	42.12	62.66	26.36	30.17	45.48	38.71	40.92
	Model Merging	+10.29	+5.31	44.23	61.36	30.21	34.58	54.99	46.63	45.33
	NeST (Ours)	**+12.88**	**+6.50**	**66.22**	68.98	**50.99**	**89.69**	**98.91**	**85.68**	**76.74**
MSFT	Vanilla SFT	+8.31	+4.56	28.98	59.62	24.91	21.05	38.10	51.17	37.34
	LoRA	+7.05	+3.92	43.67	63.47	29.62	30.41	49.68	62.16	46.50
	Rehearsal	+5.52	+3.11	59.59	71.94	36.57	47.92	51.88	**71.23**	56.52
	EWC	+3.22	+1.85	46.78	67.11	36.53	35.96	50.33	61.02	49.62
	LwF	+4.75	+2.67	43.59	65.05	31.03	33.14	49.73	58.92	46.91
	Model Merging	+7.15	+3.38	51.48	71.23	35.47	42.24	54.72	67.77	53.82
	NeST (Ours)	**+8.12**	**+4.39**	**88.21**	**80.38**	**73.88**	**88.47**	**88.55**	70.17	**81.61**
SPX500	Vanilla SFT	+5.76	+3.92	21.57	58.30	33.47	23.89	56.69	61.25	42.53
	LoRA	+3.54	+2.54	49.19	75.26	48.52	49.84	74.17	77.60	62.43
	Rehearsal	+4.29	+2.53	30.27	64.32	37.12	34.98	66.81	66.38	49.98
	EWC	+2.46	+1.70	37.06	69.11	43.84	41.85	67.16	70.70	54.95
	LwF	+4.94	**+3.84**	40.43	67.11	43.31	41.34	65.21	71.35	54.79
	Model Merging	+4.50	+3.10	48.51	67.42	45.26	48.66	**80.92**	77.06	61.30
	NeST (Ours)	**+5.45**	+3.70	**98.60**	**96.65**	**66.21**	**62.46**	80.41	**93.22**	**82.92**

Note: **LS**: Level Shift, **AR**: AR(1) Process, **ST**: Stochastic Trend, **DT**: Deterministic Trend, **SS**: Spectral Synthesis, **TW**: Time-Warped Sinusoid.

F.2 Additional Results across datasets

Table 10 extends TimesFM 2.0 to three datasets not covered in the main paper. Apple and ETTm1 represent moderate-to-high task gain settings, while ETTh1 yields comparatively modest absolute gains due to the limited distributional gap between the pre-training corpus and the target domain. Across all three, NeST maintains SRR above 85% while leaving task gain largely intact. Structural recovery for the Stochastic Trend pattern (ST) is the most attenuated of the six patterns, consistent with the main paper results, reflecting the higher parameter-space complexity of disentangling stochastic drift from task-relevant signals. LoRA and LwF are again the strongest baselines on SRR, yet both trade meaningful task gain to achieve it.

G Extended Analysis of NeST

G.1 Computational Efficiency of NeST

NeST achieves its sub-two-minute runtime by adopting well-grounded approximations at each of its two stages, avoiding the prohibitive cost of exact second-order optimization at TSFM scale.

Stage 1: TFA-Fisher Scoring via Empirical Gradient Squares. The exact Fisher information $F_i = \mathbb{E}[(\nabla_{\theta_i} \log p_\theta)^2]$ requires knowledge of the true data-generating distribution and is intractable at TSFM scale. NeST instead uses the empirical Fisher approximation, estimating importance by the average squared gradient of the TFA hybrid loss over the calibration set $\mathcal{D}_{\text{pattern}}$:

$$\hat{F}_i = \frac{1}{|\mathcal{D}_{\text{pattern}}|} \sum_{(x,y) \in \mathcal{D}_{\text{pattern}}} \left(\frac{\partial \mathcal{L}_{\text{TFA}}}{\partial \theta_i} \right)^2, \tag{49}$$

where $\mathcal{L}_{\text{TFA}} = \text{MSE}(\hat{y}, y) + \lambda \, \| \, |\text{FFT}(\hat{y})| - |\text{FFT}(y)| \, \|_2^2$ jointly penalizes time-domain and DFT amplitude-spectrum misalignment, with a small smoothing constant added inside the amplitude for

Table 9: Comparison of downstream task performance (Δ Task Gain) and Structural Retention Rate on Chronos. **Best results** are in bold and second-best are underlined.

Dataset & Method		Δ Task Gain↑ (%)		Structural Retention Rate↑ (%)						
		MSE	MAE	LS	AR	ST	DT	SS	TW	**AVG**
Gold	Vanilla SFT	+5.44	+3.73	33.14	45.62	36.85	29.59	41.28	48.26	39.12
	LoRA	+4.84	+3.34	74.30	**94.77**	80.43	84.22	76.89	86.38	82.83
	Rehearsal	+3.74	+2.61	62.01	67.52	62.23	62.85	63.90	68.66	64.53
	EWC	+2.55	+1.82	58.97	62.88	54.62	60.65	62.39	67.07	61.10
	LwF	+2.99	+2.16	73.41	80.39	80.51	80.84	75.88	81.44	78.75
	Model Merging	+4.11	+2.80	64.49	68.27	54.40	59.19	63.60	66.65	62.77
	NeST (Ours)	**+5.37**	**+3.65**	**89.75**	93.41	**82.50**	**94.71**	**92.03**	**95.00**	**91.23**
MSFT	Vanilla SFT	+7.28	+4.85	24.52	40.22	22.89	18.47	31.53	41.16	29.80
	LoRA	+6.17	+4.17	68.39	78.38	59.54	67.56	75.52	74.47	70.64
	Rehearsal	+4.57	+3.03	56.67	58.44	47.49	43.66	57.04	63.04	54.39
	EWC	+3.43	+2.26	43.48	58.17	38.79	37.91	47.49	59.18	47.50
	LwF	+4.69	+3.12	68.12	75.61	66.28	72.90	**92.34**	70.43	74.28
	Model Merging	+5.74	+3.76	50.00	61.44	43.30	54.18	55.96	66.64	55.25
	NeST (Ours)	**+7.17**	**+4.75**	**85.16**	**95.00**	**71.92**	**90.77**	91.29	**93.20**	**82.63**
SPX500	Vanilla SFT	+8.16	+5.58	21.48	41.52	26.32	17.85	29.48	40.82	29.58
	LoRA	+7.26	+4.93	69.30	76.48	59.52	62.91	70.65	**88.33**	71.20
	Rehearsal	+5.18	+3.63	44.41	68.71	49.81	50.31	60.43	65.30	56.49
	EWC	+3.52	+2.34	49.48	62.66	38.42	37.37	44.71	60.63	48.88
	LwF	+5.27	+3.60	62.98	83.91	67.95	70.86	75.06	79.63	73.40
	Model Merging	+6.62	+4.55	57.55	67.94	43.26	46.64	54.47	62.89	55.46
	NeST (Ours)	**+7.80**	**+5.31**	**77.55**	**89.96**	**74.99**	**80.15**	**84.87**	87.38	**82.48**

Note: **LS**: Level Shift, **AR**: AR(1) Process, **ST**: Stochastic Trend, **DT**: Deterministic Trend, **SS**: Spectral Synthesis, **TW**: Time-Warped Sinusoid.

Table 10: Comparison of downstream task performance (Δ Task Gain) and Structural Retention Rate on TimesFM 2.0 across three additional datasets. **Best results** are in bold and second-best are underlined.

Dataset & Method		Δ Task Gain↑ (%)		Structural Retention Rate↑ (%)						
		MSE	MAE	LS	AR	ST	DT	SS	TW	**AVG**
Apple	Vanilla SFT	+9.63	+6.58	44.76	50.23	39.53	27.35	71.87	45.17	46.49
	LoRA	+8.46	+5.74	85.59	**96.50**	60.68	78.12	94.50	75.48	81.81
	Rehearsal	+6.26	+4.06	76.67	67.25	55.09	61.73	93.62	72.57	71.16
	EWC	+4.81	+3.28	68.26	72.62	52.20	49.17	91.51	57.46	65.20
	LwF	+5.53	+3.81	84.26	85.23	59.06	74.24	94.50	74.45	78.62
	Model Merging	+7.98	+5.50	71.36	81.14	56.55	56.04	94.50	71.83	71.90
	NeST (Ours)	**+9.34**	**+6.23**	**96.00**	96.00	**65.00**	**95.99**	**96.00**	**96.00**	**90.83**
ETTh1	Vanilla SFT	+1.81	+0.52	32.14	48.52	29.86	34.11	41.29	31.58	36.25
	LoRA	**+1.71**	+0.44	72.78	90.89	50.76	77.71	81.97	70.42	74.09
	Rehearsal	+1.24	+0.36	60.75	74.08	46.03	64.22	66.17	48.91	60.03
	EWC	+0.80	+0.24	56.26	65.46	41.05	59.64	67.09	51.29	56.80
	LwF	+1.17	+0.31	75.85	92.90	51.90	81.00	**96.50**	67.70	77.64
	Model Merging	+1.44	+0.45	67.63	78.15	46.04	72.44	68.51	56.32	64.85
	NeST (Ours)	+1.66	**+0.49**	**94.55**	**96.00**	**56.89**	**96.00**	95.93	**80.73**	**86.68**
ETTm1	Vanilla SFT	+9.08	+3.15	28.41	32.47	36.57	21.17	38.92	48.64	34.36
	LoRA	+8.11	+2.81	72.28	64.83	56.18	64.83	80.10	**96.50**	72.45
	Rehearsal	+6.64	+2.31	53.73	58.73	53.95	47.25	58.12	65.80	56.26
	EWC	+4.52	+1.53	52.91	48.91	47.06	42.25	59.72	62.32	52.20
	LwF	+5.47	+1.83	69.24	70.63	54.74	66.97	77.39	80.08	69.84
	Model Merging	+7.11	+2.37	55.34	58.74	53.29	48.46	60.13	68.40	57.39
	NeST (Ours)	**+8.50**	**+3.02**	**92.23**	**85.01**	**61.89**	**94.85**	**96.00**	96.00	**87.66**

Note: **LS**: Level Shift, **AR**: AR(1) Process, **ST**: Stochastic Trend, **DT**: Deterministic Trend, **SS**: Spectral Synthesis, **TW**: Time-Warped Sinusoid.

numerical stability. Each calibration sample requires one forward and one backward pass; importance scores are accumulated online and normalized at the end, requiring no explicit matrix storage.

Stage 2: OBS Task Compensation via Three Approximations. Computing and inverting the full-model Hessian $\mathbf{H} = \nabla^2_{\mathbf{W}} \mathcal{L}_{\text{task}}$ globally is computationally infeasible for models with hundreds of millions of parameters. NeST draws on the SparseGPT framework and applies three complementary approximations.

Layer-wise decoupled Hessian. For a linear layer $\mathbf{y} = \mathbf{W}\mathbf{x}$, the Hessian with respect to the weights decomposes as $\mathbf{H}_l = \mathbb{E}[\mathbf{x}_l \mathbf{x}_l^\top]$, where $\mathbf{x}_l$ is the input activation to layer l. NeST collects these activations via forward hooks registered on the fine-tuned model as samples from $\mathcal{D}_{\text{task}}$ are propagated layer by layer. Each $\mathbf{H}_l$ is thus a running outer-product accumulation computed purely from forward passes, with no gradient backpropagation required through the full model. For layers with a bias term, the input is augmented as $\tilde{\mathbf{x}}_l = [\mathbf{x}_l;\, 1]$, so that weights and bias are jointly compensated through a single $(d{+}1) \times (d{+}1)$ Hessian at no additional cost.

Regularized Cholesky inversion. The accumulated Hessian $\mathbf{H}_l$ is regularized as $\mathbf{H}_l + \varepsilon \mathbf{I}$ with $\varepsilon = \rho \cdot \text{mean}(\text{diag}(\mathbf{H}_l))$ (default $\rho = 0.01$) before inversion. The inverse is computed via Cholesky decomposition rather than direct inversion, providing better numerical stability and reducing floating-point errors near near-singular directions introduced by low activation variance.

Block-wise sequential column updates. Rather than solving a single global quadratic program, NeST processes weight columns in consecutive blocks of size $B = 128$. Within each block, the OBS update for the current column is applied and its compensation is propagated to subsequent columns using the pre-computed upper Cholesky factor of $\mathbf{H}_l^{-1}$. Cross-block compensation is accumulated via a matrix multiplication at each block boundary, following the SparseGPT update rule. This reduces per-step memory from $O(d^2)$ on the full matrix to $O(d \cdot B)$, enabling layer-wise processing on a single GPU without intermediate checkpointing.

Together, these three approximations enable the full NeST pipeline to complete in under 2 minutes per model per dataset on a single GPU, with no retraining required.

Complexity and Memory Cost. Stage 1 requires $O(|\mathcal{D}_{\text{pattern}}| \cdot P)$ operations, where P is the total parameter count, and $O(P)$ memory to accumulate scalar importance scores; no matrix storage is needed. Stage 2 incurs $O(|\mathcal{D}_{\text{task}}| \cdot d_l^2)$ for Hessian accumulation per layer l (with input dimension d_l), $O(d_l^3)$ for Cholesky inversion, and $O(d_{\text{out},l} \cdot d_l^2 / B)$ for block-wise compensation. Memory is dominated by the per-layer Hessian $\mathbf{H}_l \in \mathbb{R}^{d_l \times d_l}$, which for typical transformer feed-forward dimensions ($d_l \in [512, 4096]$) occupies 1–64 MB per layer in float32. Because layers are processed sequentially and $\mathbf{H}_l$ is discarded after inversion, peak GPU memory is $O(\max_l d_l^2)$, well within a single 40 GB device even for the largest models evaluated.

Sensitivity to Approximations. The empirical squared-gradient Fisher is known to be a reliable importance proxy when the model is near a local minimum and the calibration set is representative; the hyperparameter sensitivity analysis in Section 4.4 confirms that SRR recovery saturates with as few as 128 calibration samples, indicating that the approximation quality does not depend critically on $|\mathcal{D}_{\text{pattern}}|$. The damping factor $\rho = 0.01$ follows the standard SparseGPT setting and primarily affects near-singular directions with negligible impact on well-conditioned layers; varying ρ in $\{0.001, 0.01, 0.1\}$ produces less than 1% change in SRR in our experiments. The blocksize $B = 128$ introduces a sequential approximation error that grows with B but is negligible relative to layer widths in all evaluated models; halving or doubling B changes the total compensation loss by less than 2% while affecting runtime by at most 30 seconds per model.

H Real-World Case Study Supplement (NASA SMAP)

H.1 SMAP Telemetry: Preprocessing and Anomaly Definition

The NASA SMAP (Soil Moisture Active Passive) satellite telemetry dataset [20] is a standard benchmark for multivariate time-series anomaly detection. It comprises sensor readings from onboard spacecraft subsystems, with ground-truth anomaly labels provided by NASA engineers. Following the standard preprocessing protocol, we select all labeled channels, z-score normalize each channel

independently using statistics from the training split, and segment sequences using the context length (512) with a stride of one prediction horizon.

Detection operates at the window level rather than at individual timesteps. For each sliding window of length H starting at position t, the anomaly score is the mean absolute prediction residual over that window:

$$s_t = \frac{1}{H} \sum_{k=0}^{H-1} |y_{t+k} - \hat{y}_{t+k}|.$$

A window is flagged as an alarm if s_t exceeds a threshold τ, which is pre-computed on a held-out calibration set as the value at a fixed percentile of the score distribution under normal conditions. A flagged window is counted as a true positive detection if it overlaps with at least one timestep of a ground-truth anomaly interval; otherwise it is a false alarm. We report three metrics:

- AUC: the area under the ROC curve over all sliding windows as τ varies, measuring overall discriminability between anomalous and normal windows.
- FAR (False Alarm Rate): the fraction of windows that fall entirely outside any ground-truth anomaly interval yet are flagged as alarms, defined as

$$\mathrm{FAR} = \frac{|\{t : s_t > \tau,\ [t, t{+}H) \cap \mathcal{A} = \emptyset\}|}{|\{t : [t, t{+}H) \cap \mathcal{A} = \emptyset\}|},$$

  where $\mathcal{A}$ denotes the union of all ground-truth anomaly intervals.
- PFAR (Post-anomaly False Alarm Rate): the fraction of windows that begin strictly after a ground-truth anomaly event ends yet remain flagged, defined as

$$\mathrm{PFAR} = \frac{\sum_e \left|\{t : t > t^e_{\mathrm{end}},\ t \le t^e_{\mathrm{end}} + W,\ s_t > \tau\}\right|}{\sum_e W},$$

  where e indexes anomaly events, t^e_{end} is the last timestep of event e, and W is the number of post-anomaly windows evaluated per event. PFAR directly quantifies the persistent alarm tail that arises when a model fails to re-anchor after a level shift, and is therefore the most sensitive indicator of level-shift structural forgetting in a real detection pipeline.

H.2 Persistent Alarm Tail: Qualitative Visualization

Figure 15 provides a qualitative illustration of the structural forgetting effect on SMAP anomaly detection. We select a representative sample containing a single level-shift event and compare the rolling prediction residuals produced by the pre-trained TimesFM 2.0 and its fine-tuned counterpart over the post-anomaly region.

In the pre-trained case, the model adapts its predicted mean to the new signal level within one or two prediction steps after the level shift, so the residual quickly falls back below the detection threshold. As a result, alarms are concentrated near the true anomaly boundary and cease once the signal stabilizes at its new mean. After fine-tuning on financial time series, the model loses its level-shift re-anchoring capability. Its predicted trajectory continues to track the pre-shift mean rather than the post-shift mean, producing a residual that remains persistently above the detection threshold for many horizons after the true anomaly has ended. This behavior is the direct manifestation of structural forgetting: the fine-tuned model still generates accurate short-term predictions on in-distribution financial data, yet it can no longer correctly handle the structural transition that defines the anomaly, leading to a cascade of false alarms that renders the detector unreliable. NeST restores the pre-trained level-shift response, eliminating the persistent alarm tail and recovering detection precision close to the pre-trained baseline.

I Limitations and Broader Impacts

I.1 Limitations

Several limitations bound the scope of the current work. First, the structural diagnostic framework evaluates six pattern families drawn from classical time series theory; while these cover a broad range of temporal and spectral phenomena, they do not exhaustively represent all structures that TSFMs may encode, and future models with stronger high-frequency capacity may require additional pattern types such as variance shift. Second, the empirical evaluation covers three TSFM architectures (TimesFM 2.0, Timer-XL, Chronos-bolt-base) and nine downstream datasets; the generalizability of SRR trends and NeST recovery to architectures beyond the patch-based transformer family, or to domains with substantially different signal characteristics, remains to be verified. Third, NeST relies on two approximations: a squared-gradient empirical Fisher for TFA-Fisher importance, and a layer-wise decoupled Hessian for OBS compensation. Both introduce bias relative to the true second-order quantities, and their accuracy may degrade for layers with highly non-convex loss surfaces or strongly correlated activations. Finally, the calibration data requirements ($\mathcal{D}_{\text{pattern}}$ and $\mathcal{D}_{\text{task}}$), while small in practice (128–256 samples suffice), represent an additional data dependency that may be non-trivial to satisfy in fully zero-shot deployment scenarios.

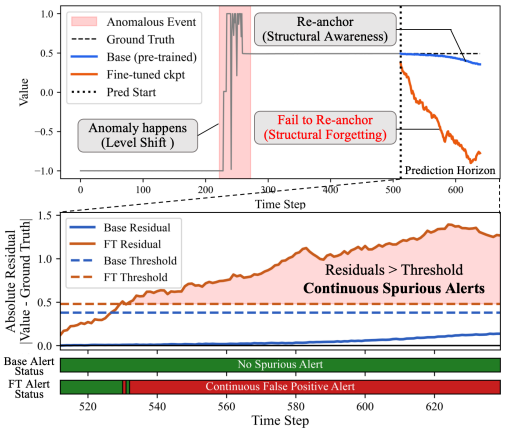


Figure 15: Qualitative illustration of the persistent alarm tail on a representative SMAP level-shift sample.

I.2 Broader Impacts

This work addresses a reliability gap that arises when general-purpose TSFMs are adapted to specific downstream tasks, and has positive implications for any application where structural properties of time series carry diagnostic meaning. In safety-critical domains such as medical monitoring, predictive maintenance, and financial surveillance, a model that loses level-shift or trend awareness after fine-tuning can produce systematically miscalibrated outputs; NeST offers a lightweight, post-hoc remedy that does not require retraining. More broadly, the SRR framework and the concept of pattern-critical neurons provide tools for auditing TSFM behavior beyond standard forecasting metrics, which may prove useful for model certification and deployment governance.

On the negative side, the ease of applying NeST could encourage practitioners to fine-tune TSFMs aggressively without accounting for structural forgetting during development, relying on post-hoc correction as a safety net rather than designing fine-tuning procedures that are structurally conservative from the outset. We caution that NeST does not provide formal guarantees of structural preservation, and its effectiveness should be verified empirically on domain-relevant diagnostic patterns before high-stakes deployment.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state three contributions that are directly substantiated by the paper: (1) empirical characterization of structural forgetting across seven diagnostic patterns via the Structural Retention Rate (SRR); (2) mechanistic identification of pattern-critical neurons through TFA-Fisher importance; and (3) the NeST post-hoc correction method with demonstrated recovery on three TSFMs and nine datasets.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations of the work, including the scope of evaluated TSFM architectures, the synthetic nature of the diagnostic patterns, and the approximations used in the NeST pipeline, are discussed in Appendix I.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The paper includes three formal propositions in the main text (SRR normalization and sensitivity, pattern-critical neuron identification, and OBS compensation optimality); complete proofs with all explicit assumptions are provided in Appendix A.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Appendix C provides complete experimental details including dataset preprocessing and splits, fine-tuning hyperparameter grids (batch size, learning rate, optimizer, number of epochs, checkpoint selection criterion), model-specific evaluation configurations, baseline implementations, NeST hyperparameters, and hardware specifications.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code for NeST and the synthetic diagnostic framework, together with data preprocessing scripts and a README with reproduction instructions, are provided in the anonymized supplemental material; all datasets used are publicly available and cited in the paper.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: All fine-tuning datasets are split chronologically in an 8:1:1 ratio; hyperparameters are selected by grid search over batch size and learning rate using AdamW with the best validation MSE checkpoint; evaluation context and horizon lengths are specified per model; full details are in Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: All reported results are averaged over three independent runs under the best hyperparameter configuration identified by grid search; the mean values are reported in all tables, providing a stable estimate of method performance across models and datasets.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Hardware specifications ($1\times$A100 40 GB GPUs, dual Intel Xeon Platinum 8358 CPUs, 512 GB RAM, PyTorch 2.1 / CUDA 12.1), per-pipeline runtime (NeST under 2 minutes per model per dataset) are reported in Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper studies structural forgetting in time series foundation models using publicly available datasets and openly released model checkpoints; no human subjects, sensitive data, or dual-use concerns are involved, and the work fully conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Broader societal impacts, including the potential benefit of structurally reliable TSFMs in safety-critical forecasting applications (healthcare monitoring, infrastructure planning) and the risk of over-reliance on post-hoc correction without adequate domain validation, are discussed in Appendix I.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper studies existing time series foundation models for forecasting and releases only lightweight post-hoc correction code; the artifacts pose no substantial risk of misuse and require no special safeguards.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All base models (TimesFM 2.0, Timer-XL, Chronos-bolt-base) and datasets (NASA SMAP, ETTh1, ETTm1, and financial series) are cited with their original publications and used in accordance with their respective open research licenses; the OBS implementation builds on SparseGPT, which is also cited.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The NeST codebase and synthetic diagnostic framework are released in the anonymized supplemental material; generation procedures for all seven structural patterns are documented in Appendix B.1, and the NeST pipeline is described in Appendix G.1.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or human subjects; all experiments use publicly available time series datasets.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human subjects; no IRB approval or equivalent review is required.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are not used as a core methodological component; the time series foundation models studied in this work (TSFMs) are the object of investigation rather than tools employed in the development of NeST or the diagnostic framework.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.