R3-TTA: TEST-TIME ADAPTATION FROM MATURED OUTCOMES FOR POST-EVENT MARKET PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Standard Test-Time Adaptation assumes that an unlabelled query carries a useful learning signal. We show this assumption fails in market response prediction, where outcomes have not occurred at prediction time and posts arriving within the same time bar share a single future return, causing standard gradient adapters to severely degrade performance relative to the frozen baseline. We propose **R**egime-**R**outed **R**etrieval **T**est-**T**ime **A**daptation (R3-TTA), which grounds adaptation in historical events with matured outcomes. For each query bar, R3-TTA retrieves past matured bars routed by arrival timing and lagged market state, fits a prior-shrunk Bernoulli response model with a closed-form minimizer, and fuses this local estimate with the frozen prediction in logit space. By design, zero parameters are updated, no signals are extracted from unlabelled test batches, and shared outcomes are counted strictly once. Fusing the local nonparametric estimate with the frozen prediction drives the primary performance gains, far outperforming naive bias-offset adjustments. On post-event market prediction, R3-TTA consistently improves performance across diverse architectures and walk-forward evaluations achieving substantial gains on tabular models where all gradient adapters fail. Without retuning, R3-TTA seamlessly transfers to broad market assets. We release the aligned post–market panel with a reproducible content tagging layer.

1 INTRODUCTION

Markets respond to online posts, and a predictor of that response must track conditions that change from one event to the next. Political communication is a prominent case: posts can precede sharp repricing, but most arrive during quiet periods, and similar arrivals are followed by different responses in different market regimes (Baker et al., 2016; Colonescu, 2018; Brans & Scholtens, 2020). Test-time adaptation (TTA) is the natural tool for such a predictor, and it is standard in deployed vision and time-series systems (Wang et al., 2022; Niu et al., 2022; Kim et al., 2025). However, post-event prediction removes the signal that TTA learns from. Existing methods typically advance along two pathways: ❶ self-supervised or entropy objectives built from the unlabeled query (Sun et al., 2020; Wang et al., 2021; Niu et al., 2023), and ❷ normalization statistics re-estimated from the test batch (Wang et al., 2021; Niu et al., 2022). In post-event markets, both pathways lack a reliable signal, since the market response is unknown at prediction time.

While these methods have achieved strong results, their performance here is constrained by a fundamental limitation: **the adaptation signal is drawn from the query**. Existing approaches suffer from two issues in this setting. ❶ They learn from an outcome that has yet to happen, so the only forward-looking signal is the model's own confidence. ❷ They treat posts as independent inputs, while many posts share the outcome of one bar, so an adaptation batch is often a single row or a set of correlated copies. The resulting adapter is harmful: on a BatchNorm backbone, the standard entropy adapter falls far below its frozen starting point. Figure 1 contrasts this query-driven loop (middle) with adaptation that reaches backward into matured outcomes (right).

In this work, we argue that adaptation in this regime should reach *backward* into history, where the outcomes of past events have already matured. We propose **R**egime-**R**outed **R**etrieval **T**est-**T**ime **A**daptation (R3-TTA), which turns each prediction into a small supervised problem on those matured

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

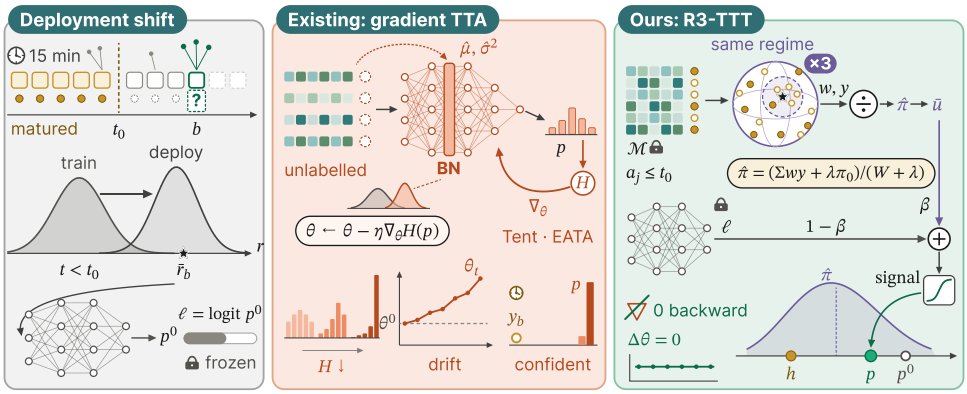

Figure 1: Comparison of Test-Time Adaptation Strategies.

outcomes. The core insight of R3-TTA lies in **backward-reaching supervision**: labels that exist before deployment replace the missing query label, so nothing is estimated from the test batch. The gain these labels bring is consistent with **variance reduction**. The frozen model has already absorbed what the shared features say about its own bias, and the gain is not recovered by a bias-offset operator built from the same kernel. Local averages over the same matured labels therefore reach a similar level, and R3-TTA attains it with a closed-form, well-posed estimate. A prior-shrunk average delivers this estimate in closed form, and the adaptation step needs no gradient.

R3-TTA runs in three steps **Regime-routed retrieval** finds historical bars that match the query bar by arrival timing, by lagged market state, and by the union of the two, so the retrieved outcomes come from the same regime as the query. **Closed-form local estimation** fits a prior-shrunk Bernoulli model to the retrieved labels. The minimizer is one division, so there is no learning rate, stopping rule or backward pass. **Fixed fusion** blends the local estimate with the frozen prediction in logit space and averages over a fixed grid of settings, with half the weight on the joint routing space. We test R3-TTA on Donald Trump's Truth Social posts, which we use as a case study of an event stream that moves global markets and comes with structured data. The task is to predict how each post affects the stocks in the SPY (S&P 500) and crypto markets constituent universe. On three tree and neural backbones and four frozen language-model backbones, R3-TTA improves all seven, and the confidence interval excludes zero on six. It beats every gradient adapter on the neural backbones where those adapters apply. It stays above the frozen anchor in a pre-deployment walk-forward, a forward deployment and an untuned crypto transfer, and no fold falls below the frozen model. Its accuracy matches adaptive neighbor fusion (AdaNPC) within sampling noise, while it uses a well-posed closed-form estimate, twelve fixed members, no validation search and zero backward passes.

Our main contributions are as follows: ❶ **Trump trade dataset.** To best of our knowledge this is the first quantitative dataset that aligns President Donald Trump's Truth Social posts with bar-level outcomes of stocks, together with a reproducible content-tagging layer. ❷ **Closed-form adaptation.** We propose R3-TTA, which adapts through backward-reaching supervision with a closed-form scalar solution, no backward pass, no test-batch statistic, and a fixed memory that counts each shared outcome once. ❸ **Mechanism design and universal efficacy.** R3-TTAexhibits strong generalizability by universally enhancing predictive efficacy over the frozen baseline across seven diverse backbones, walk-forward testing, forward deployment, and cross markets.

2 Related Work

Event-driven Market Prediction links news and social-media text to asset prices (Xu & Cohen, 2018; Xu et al., 2021; Zhou et al., 2021). Presidential posts move equity prices and volatility (Ge et al., 2019; Colonescu, 2018; Kinyua et al., 2021; Perico Ortiz, 2023), and event-study methods motivate precise alignment and clustered inference (Boehmer et al., 1991; Kolari & Pynnönen, 2010). Recently, language models read news for return prediction (Lopez-Lira & Tang, 2023) and act as trading agents (Zhang et al., 2024), while benchmarks evaluate forecasts through investment returns (Hu et al., 2026). These methods improve the forecaster itself. However, a forecaster fixed at deployment still faces unseen market regimes, and its accuracy depends on how it adapts to them.

Table 1: Taxonomy of related work vs. R3-TTA. Batch Statistics (BS), matured labels (Lab.), no backward pass (BP), fixed memory (Mem.), one reference row per outcome (Row).

Method	BS	Lab.	BP	Mem.	Row	Method	BS	Lab.	BP	Mem.	Row
TTT (Sun et al., 2020)	✓	✗	✗	✓	✗	PETSA (Medeiros et al., 2025)	✓	✓	✗	✓	✗
TTT-MAE (Gandelsman et al., 2022)	✓	✗	✗	✓	✗	SOLID (Chen et al., 2024)	✓	✓	✗	✓	✗
Tent (Wang et al., 2021)	✗	✗	✗	✓	✗	T3A (Iwasawa & Matsuo, 2021)	✗	✗	✓	✗	✗
EATA (Niu et al., 2022)	✗	✗	✗	✓	✗	AdaNPC† (Zhang et al., 2023)	✓	✗	✓	✗	✗
SAR (Niu et al., 2023)	✗	✗	✗	✓	✗	TDA† (Karmanov et al., 2024)	✓	✗	✓	✗	✗
DeYO (Lee et al., 2024)	✗	✗	✗	✓	✗	Tip-Adapter (Zhang et al., 2022)	✓	✓	✓	✓	✗
CoTTA (Wang et al., 2022)	✗	✓	✓	✗	✗	ProKeR (Bendou et al., 2025)	✓	✓	✓	✓	✗
RoTTA (Yuan et al., 2023)	✗	✗	✓	✓	✗	kNN-LM (Khandelwal et al., 2020)	✓	✓	✓	✓	✗
ROID (Marsden et al., 2024)	✗	✗	✗	✓	✓	TabR‡ (Gorishniy et al., 2024)	✓	✓	✗	✓	✗
NOTE (Gong et al., 2022)	✓	✗	✓	✗	✗	ModernNCA‡ (Ye et al., 2025)	✓	✓	✓	✓	✗
UnMix-TNS (Tomar et al., 2024)	✗	✗	✓	✗	✗	LoCalPFN‡ (Thomas et al., 2024)	✓	✗	✓	✓	✗
TAFAS (Kim et al., 2025)	✓	✓	✗	✓	✗	**R3-TTA (ours)**	✓	✓	✓	✓	✓

†Memory stores test features with predicted pseudo-labels. ‡Supervised retrieval models.

Test-time Adaptation updates a trained model from unlabeled test inputs (Sun et al., 2020; Gandelsman et al., 2022). Existing methods typically follow two strategies: entropy and self-training methods minimize prediction uncertainty and filter unreliable samples (Wang et al., 2021; Niu et al., 2022; 2023; Lee et al., 2024; Wang et al., 2022; Marsden et al., 2024), while normalization methods re-estimate or repair batch statistics (Gong et al., 2022; Yuan et al., 2023; Tomar et al., 2024). Time-series variants further exploit partially observed targets or contextually similar history (Kim et al., 2025; Medeiros et al., 2025; Chen et al., 2024). Despite these advances, these methods adapt parameters or normalization statistics at test time. In post-event markets, a batch often holds a single outcome or correlated copies of it, so such updates become unreliable.

Retrieval-based Prediction corrects a model with labeled reference examples, as in kNN-LM, which interpolates a frozen model with a distance-weighted label vote (Khandelwal et al., 2020). Caches, prototypes and kernel predictors adapt classifiers similarly (Iwasawa & Matsuo, 2021; Zhang et al., 2022; Karmanov et al., 2024; Bendou et al., 2025). On tabular data, neighbor retrieval yields strong supervised and in-context predictors (Gorishniy et al., 2024; Ye et al., 2025; Thomas et al., 2024; Hollmann et al., 2025). The closest work, AdaNPC (Zhang et al., 2023), grows its memory with test features and predicted labels. Unlike prior methods, R3-TTA uses outcomes matured before deployment. It maintains a fixed memory with one outcome per bar, routes queries by market regime, and fits a prior-shrunk Bernoulli probability in closed form.

3 Methodology

3.1 Preliminaries and Overview

Method Overview. R3-TTA adapts a frozen probabilistic forecaster to each market bar with outcomes that matured before deployment. Given a complete query bar, the frozen predictor and a fixed memory of historical bars, R3-TTA produces an adapted probability for each post that updates no parameter and reads no deployment outcome. As architecture shown in Figure 2, R3-TTA consists of four fixed stages: ❶ **Frozen prediction** supplies the slow logit, which carries everything learned before deployment (3.2). ❷ **Regime-routed retrieval** matches the query to historical *bars* in three routing spaces that encode arrival timing, lagged market state and their union (3.2). ❸ **Closed-form local estimation** fits a prior-shrunk Bernoulli model to the retrieved labels, and its minimizer is a single division (3.3). ❹ **Fixed fusion and ensembling** blends the local logit with the slow logit at a fixed weight and averages a fixed-weight member grid in probability (3.4). The operator estimates nothing from the test batch and accepts any probabilistic backbone.

Problem Definition. The prediction unit is a market bar, and at its opening the predictor may use only past information. Let $\mathcal{E}$ be the post set and $\mathcal{A}$ the instrument universe, $|\mathcal{A}| = 39$. A post $e \in \mathcal{E}$ has publication time t_e and observable metadata x_e. Each instrument has 15-minute bars b with opening and closing timestamps $\mathrm{op}(b)$ and $\mathrm{cl}(b)$, and $Z_{b'}$ is the market data of bar b', observable at $\mathrm{cl}(b')$. We align a post to the first bar opening at or after it, $b(e,i) = \min\{b : \mathrm{op}(b) \geq t_e\}$ on instrument i's grid. We write $b(e) = b(e, \mathrm{SPY})$ and call $E_b = \{e : b(e) = b\}$ the *complete query bar*. The information at the opening of bar b is

$$\mathcal{H}_b = \sigma\big((Z_{b'})_{\mathrm{cl}(b') \leq \mathrm{op}(b)},\ (x_e, t_e)_{t_e \leq \mathrm{op}(b)}\big), \tag{1}$$

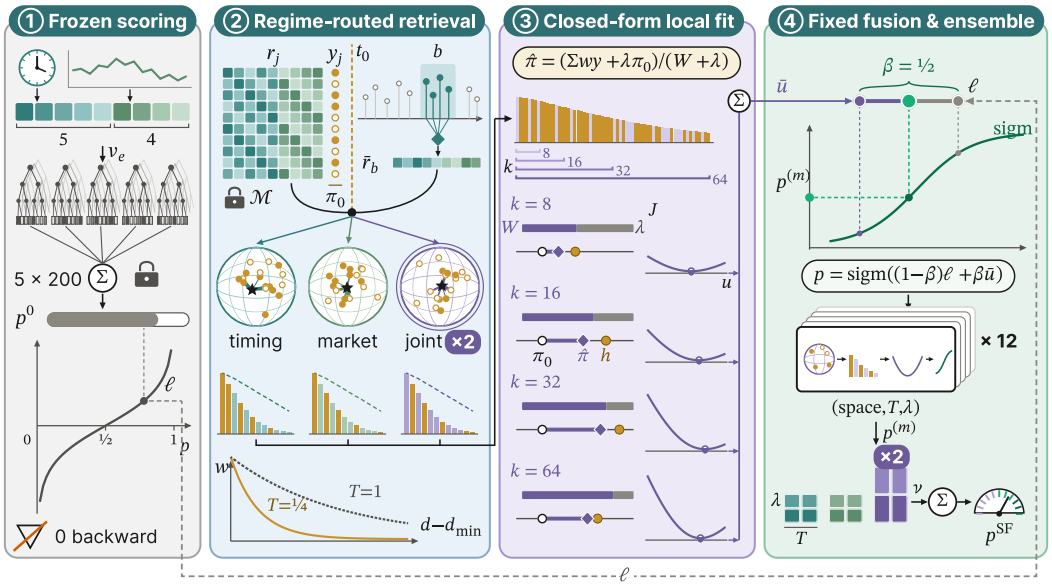

Figure 2: Overview of R3-TTA. Four stages frozen forecaster adapts: ❶ the slow logit ℓ of the frozen model, ❷ bar-level routing in three spaces over a fixed memory, ❸ closed-form shrunk fits on nested neighborhoods, and ❹ fixed-β fusion with a joint-weighted member average.

where $\sigma(\cdot)$ denotes the generated sigma-field. The two requirements enforce temporal precedence and make no claim about the causal effect of a post .

Requirement 1 (Measurability). *For each deployment bar b, every score assigned to a post $e \in E_b$ and every asset weight entered at* $\mathrm{op}(b)$ *is $\mathcal{H}_b$-measurable.*

Requirement 2 (Immutability). *At deployment time t_0 the memory and all fitted transformations are fixed. Every memory item j has label-availability time $a_j \le t_0$, and no deployment outcome or pseudo-label enters the memory after t_0.*

Targets. The event label records whether the absolute SPY response around the aligned bar exceeds a fixed threshold. For bars whose neighbors $b-1$ and $b+3$ exist on the grid,

$$r_b^{\mathrm{evt}} = C_{b+3,\mathrm{SPY}}/C_{b-1,\mathrm{SPY}} - 1, \qquad y_b = \mathbf{1}\{10^4|r_b^{\mathrm{evt}}| > \kappa_{\mathrm{evt}}\}, \qquad y_e = y_{b(e)}, \tag{2}$$

where $C_{b,i}$ is the bar-close price and $\kappa_{\mathrm{evt}} = 17.15$ bps is the training-period median of $10^4|r_b^{\mathrm{evt}}|$. The label therefore measures magnitude and ignores direction. Its nominal one-hour window can span an overnight gap, so the asset target uses each instrument's own executable open-to-close return.

Evaluation Unit. All posts of one bar share one label, so we average scores within a bar before computing AUC, and the memory holds one item per bar. We leverage bar-level market clearing as an intrinsic noise filter that suppresses high-frequency rhetorical volatility, thereby stabilizing the supervision signal for robust downstream adaptation. The panel contains 8,678 aligned posts: 5,277 before 2025-10-01, 1,566 in 2025-Q4 for validation, and 1,835 test posts on 525 bars over 68 trading days in 2026 (Table 7, Appendix D). The clean event model reads five timing and four lagged-market features and no text, so the adaptation gain reflects timing and market regime.

3.2 Regime-Routed Retrieval: One Matured Outcome per Bar

Frozen Prediction. R3-TTA starts from the frozen forecast and retrieves, for each query bar, the matured outcomes most relevant to its regime. The frozen model already reflects everything learned before deployment, so retrieval leaves that prediction intact. Let v_e be the clean feature vector of post e and p_e^0 its frozen probability. For $p \in (0,1)$, we write $\mathrm{logit}(p) = \log[p/(1-p)]$ with inverse $\mathrm{sigm}(u) = (1+e^{-u})^{-1}$, and call $\ell_e = \mathrm{logit}(p_e^0)$ the *slow* logit. In the tree implementation, $p_e^0 = \frac{1}{5}\sum_{s=1}^{5} f_{\theta_s}(v_e)$ averages five pre-deployment gradient-boosted classifiers of 200 depth-4 trees. We clip probabilities to $[10^{-6}, 1-10^{-6}]$ before the logit.

Bar-level Memory. Retrieval operates on bars, so a burst of posts cannot multiply the contribution of one outcome. In a routing space of dimension d_r, memory item j stores the standardized mean routing vector $r_j \in \mathbb{R}^{d_r}$ of one historical bar and its matured label y_j. We fit the standardizer to the historical bar means before deployment. The memory $\mathcal{M} = \{(r_j, y_j)\}_{j=1}^{M}$ satisfies Requirement 2 and has label prior $\pi_0 = M^{-1} \sum_j y_j \in (0, 1)$. We form the query by averaging the post-level routing vectors of the complete bar, $\bar{r}_b = |E_b|^{-1} \sum_{e \in E_b} r_e$. We use three routing spaces: timing ($d_r = 5$), lagged market state ($d_r = 4$) and their union ($d_r = 9$).

Neighborhood Weights. Let $\mathcal{N}_k(b) \subseteq \{1, \dots, M\}$ index the k memory items nearest to $\bar{r}_b$ under $d(r, r') = \|r - r'\|_2^2 / d_r$, with a fixed tie-breaking rule (all items if $M < k$), and let $d_b^{\min} = \min_j d(\bar{r}_b, r_j)$. With temperature $T > 0$, each retrieved item receives the weight

$$w_{bj} = \exp\{-[d(\bar{r}_b, r_j) - d_b^{\min}]/T\}, \qquad W_b(k, T) = \sum_{j \in \mathcal{N}_k(b)} w_{bj}, \tag{3}$$

where $W_b(k, T)$ is the total retrieved weight. The nearest item has weight one, so $W_b(k, T) \geq 1$. We divide by d_r so that one temperature grid serves all three spaces. Appendix A shows how the routing coordinates change the neighborhood of a fixed query.

3.3 Closed-Form Local Estimation: Adaptation without a Gradient Step

R3-TTA derives a closed-form local estimate from retrieved labels, bypassing fragile gradient steps on immature query labels. Because regime-matched historical labels have already matured before deployment, retrieval supplies the requisite supervision. Our core insight is that these labels primarily yield variance reduction: since the frozen model has already absorbed the structural bias, the historical neighborhood offers a lower-variance estimate of the event rate. Consequently, local averages over matured labels achieve comparable gains, whereas a bias-offset operator on the same kernel fails to reproduce them.

Local Objective. The natural objective for such an estimate is a shrunk, weighted Bernoulli likelihood, and its structure removes optimization altogether. Specifically, the scalar *fast* logit u minimizes

$$J_{b,k}(u) = \sum_{j \in \mathcal{N}_k(b)} w_{bj} \, \mathrm{CE}\big(y_j, \mathrm{sigm}(u)\big) + \lambda \, \mathrm{CE}\big(\pi_0, \mathrm{sigm}(u)\big), \tag{4}$$

where $\mathrm{CE}(q, p) = -q \log p - (1 - q) \log(1 - p)$ is the binary cross-entropy and $\lambda > 0$ is the prior strength.

Proposition 1 (Unique closed-form fast state)**.** *Assume $M \geq 1$, $\lambda > 0$ and $\pi_0 \in (0, 1)$. For every query bar b and every $k \geq 1$, eq. (4) has the unique finite minimizer*

$$u_{b,k} = \mathrm{logit}(\widehat{\pi}_{b,k}), \qquad \widehat{\pi}_{b,k} = \frac{\sum_{j \in \mathcal{N}_k(b)} w_{bj} y_j + \lambda \pi_0}{W_b(k, T) + \lambda}. \tag{5}$$

The proof (Appendix A) uses $J'_{b,k}(u) = (W_b(k, T) + \lambda)(\mathrm{sigm}(u) - \widehat{\pi}_{b,k})$ and $J''_{b,k} > 0$. It also treats the boundary case $\lambda = 0$, where a unanimous neighborhood has no finite minimizer. With $h_{b,k} = W_b(k, T)^{-1} \sum_{j \in \mathcal{N}_k(b)} w_{bj} y_j$ the weighted neighbor vote, the minimizer can be written as

$$\widehat{\pi}_{b,k} = \alpha_{b,k} \, h_{b,k} + (1 - \alpha_{b,k}) \, \pi_0, \qquad \alpha_{b,k} = \frac{W_b(k, T)}{W_b(k, T) + \lambda} \in (0, 1), \tag{6}$$

so the fast state is the vote shrunk toward the memory prior. It equals the posterior mean of a Beta–Bernoulli model with weighted observations and pseudo-counts $\lambda\pi_0$ and $\lambda(1 - \pi_0)$. The prior therefore keeps the fast logit finite when the neighborhood is small or unanimous, and its share $1 - \alpha_{b,k}$ falls as retrieved weight grows As Figure 3b shows the estimate design.

Nested Scales. The solution depends on the routing space, T and λ. We index this triple as a member $m = (\text{space}, T, \lambda)$ and write $u_{b,k}^{(m)}$. We solve eq. (5) at four nested scales $\mathcal{K} = \{8, 16, 32, 64\}$ read off one distance-ranked support, and average the logits,

$$\bar{u}_b^{(m)} = \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} u_{b,k}^{(m)}, \tag{7}$$

which combines the nearest outcomes with a broader historical context at no extra retrieval cost. In short, adaptation reduces to weighted counts of matured outcomes and one division per scale.

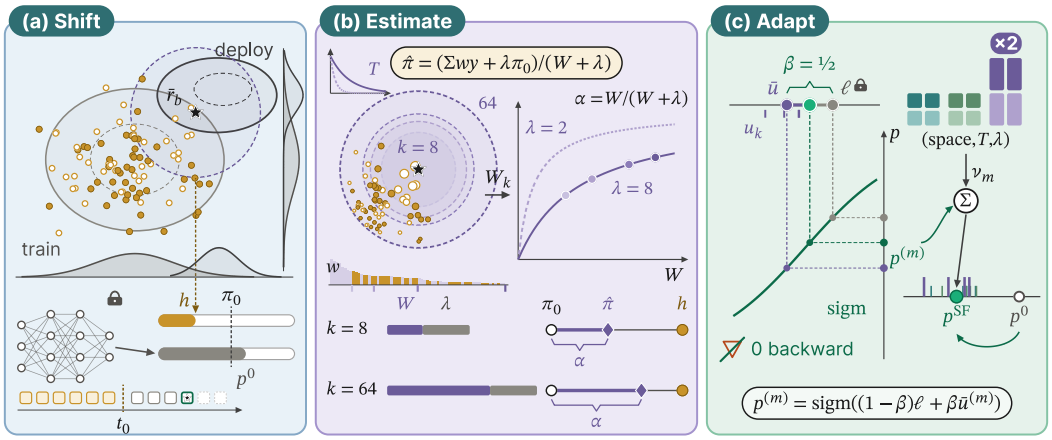

Figure 3: Shift, Estimate, Adapt Design. (a) **Shift:** Matured historical labels near query $\bar{r}_b$ reveal a local rate h shifting away from the frozen model's prior π_0 and prediction p^0. (b) **Estimate:** Distance-weighted retrieval aggregates local labels to interpolate the adapted estimate $\hat{\pi}$ between prior π_0 and local rate h. (c) **Adapt:** Multi-scale ensemble logits combine with structural anchor ℓ to output the final prediction p^{SF} in a single forward pass.

3.4 Fixed Fusion: Anchored on the Frozen Forecast

R3-TTA anchors its forecast on the frozen prediction and blends in the local estimate at a fixed weight. The local estimate is informative but noisy, while the frozen prediction is stable but regime-blind, so the forecast blends the two. For member m, we interpolate the two in logit space at a fixed blend weight,

$$p_e^{(m)} = \mathrm{sigm}\big((1-\beta)\,\ell_e + \beta\,\bar{u}_{b(e)}^{(m)}\big), \qquad \beta = \tfrac{1}{2}, \tag{8}$$

where β sets how far the forecast moves toward the local estimate, and λ sets how far the local estimate moves from the prior (Figure 3b,c). Since ℓ_e is post-specific and $\bar{u}_b^{(m)}$ is shared by the bar, the adapted logit is an increasing affine map of ℓ_e with slope $1-\beta$ and bar-level intercept $\beta\bar{u}_b^{(m)}$. Adaptation therefore re-ranks bars against one another and keeps the frozen order within a bar.

Selection-free Ensemble. We form the member grid $\mathcal{G}$ by crossing the three routing spaces with $T \in \{0.25, 1\}$ and $\lambda \in \{2, 8\}$. These values are fixed before deployment, and Section 4.3 studies their sensitivity. We then average the $|\mathcal{G}| = 12$ members in probability with fixed weights,

$$p_e^{\mathrm{SF}} = \frac{\sum_{m\in\mathcal{G}} \nu_m\, p_e^{(m)}}{\sum_{m\in\mathcal{G}} \nu_m}, \qquad \nu_m = \begin{cases} 2, & m \text{ routes in the joint space,} \\ 1, & \text{otherwise,} \end{cases} \tag{9}$$

where ν_m gives the joint space, which contains both coordinate sets, half of the ensemble weight. The four scales are averaged in logit space (eq. (7)) and the 12 members in probability space (eq. (9)). Every member is kept, and no score on a deployment panel sets a weight, which makes the ensemble *selection-free*. Section 4.2 shows why averaging over routing spaces is preferable to selecting one. The blend weight is the same constant for every bar and member, and Section 4.2 evaluates a data-dependent alternative.

Proposition 2 (Invariance across complete bars)**.** *Fix the backbone in deterministic inference mode, the memory, all fitted transformations and the tie-breaking rule. For any complete query bar E_b, the prediction assigned to each post $e \in E_b$ is invariant, in exact arithmetic, to the order in which deployment bars are processed, to how they are batched, and to which other deployment bars are processed alongside E_b.*

Every quantity above is a function of the bar's own inputs and the fixed memory, so the operator is deployment-stationary. It stays unchanged along the deployment stream, and a prediction can be audited from the bar alone (proof in Appendix A).

Computational Complexity. Each query bar retrieves up to 256 candidates per routing space, shared by all members of that space. A brute-force scan costs $O(Md_r)$ per space, and an index would

reduce this cost while leaving the scalar update unchanged. R3-TTA requires zero backward passes and no validation configuration search. Under a common timing harness on the 2025-Q4 validation deployment, it deploys at 0.107 ms per bar, and the distance scan over the three routing spaces dominates this cost. Algorithm 1 in Appendix A summarizes the procedure.

4 EXPERIMENTS

❶ **Data and protocol.** Development ends on 2025-09-30, validation is 2025-Q4 and the test panel spans 2026-01-02 to 2026-04-10. The frozen model and the immutable memory are fitted on all pre-2026 data; validation is used for training but never for grid weighting. Neural architectures and every comparator's hyperparameters are selected on validation alone. Two further panels are a nine-fold pre-deployment walk-forward (1,594 bars, nothing at or after 2026-01-01 read) and a transfer panel of 40 crypto perpetuals. Trump post based on Truth media information with timestep. (Detail in Appendix D) ❷ **Baselines.** We compare with seven gradient or batch-statistic adapters (Tent, ETA, EATA, SAR, CoTTA, ROID and a statistics-only control) and with TAFAS, which receives delayed targets (Wang et al., 2021; Niu et al., 2022; 2023; Wang et al., 2022; Marsden et al., 2024; Kim et al., 2025), all on matched MLP+BatchNorm and FT-Transformer backbones. Retrieval comparators share the same matured history: T3A, nearest centroid, kNN and AdaNPC votes, Nadaraya–Watson, AdaNPC fusion, and kNN-LM in a validation-selected and a fixed-grid form (Iwasawa & Matsuo, 2021; Zhang et al., 2023; Khandelwal et al., 2020). ❸ **Implementation.** R3-TTA uses the fixed 12-member grid of Sec 3.4 on every backbone with nothing tuned per backbone; gradient comparators adapt on the posts of one bar as their batch. ❹ **Metrics.** Bar-level AUC, computed after averaging scores within each outcome bar, is primary; bar Brier score measures probability error. Intervals are paired 5,000-replicate five-day moving-block bootstraps clustered by day, seed 42.

4.1 MAIN RESULTS

Table 2: Prediction bar AUC on the 2026 test panel for three frozen backbones. × marks arms that do not apply; **bold** marks the best value per column.

Method	GBT	MLP+BN	FT-Transformer
Frozen model	0.7679	0.7514	0.7226
Gradient and batch-statistic test-time adaptation (unlabelled query)			
Tent (Wang et al., 2021)	×	0.5486	0.7223
ETA (Niu et al., 2022)	×	0.5799	0.7250
EATA (Niu et al., 2022)	×	0.6246	0.7240
SAR (Niu et al., 2023)	×	0.5927	0.7225
CoTTA (Wang et al., 2022)	×	0.6279	0.7219
ROID (Marsden et al., 2024)	×	0.6075	0.7219
BN statistics only	×	0.6276	×
TAFAS (Kim et al., 2025)	×	0.7514	0.7227
LAME (Boudiaf et al., 2022)	×	0.7488	0.7212
Retrieval and reference-based adaptation (matured history)			
T3A (Iwasawa & Matsuo, 2021)	0.7500	0.7401	0.7255
Nearest centroid	0.7682	0.7456	0.7265
kNN vote	0.7527	0.7527	0.7527
AdaNPC vote (Zhang et al., 2023)	0.7536	0.7536	0.7536
Nadaraya–Watson	0.7775	0.7591	0.7513
kNN-LM, selected (Khandelwal et al., 2020)	0.7792	0.7678	0.7529
AdaNPC, fused	0.7655	0.7802	0.7408
kNN-LM, fixed grid	0.7808	0.7726	0.7554
R3-TTA (ours)	**0.7852**	**0.7788**	**0.7483**

Comparison with test-time adaptation. Gradient-based test-time adaptation fails in this regime, whereas R3-TTA consistently delivers robust performance gains across all settings (Figure 4a). In Table 2, R3-TTA improves on the frozen model in every column, with every paired interval excluding zero, while every gradient adapter is at or below its frozen model: on the MLP+BN backbone the damage is severe, with Tent losing 0.2028 AUC (dropping from 0.7514 to 0.5486) and

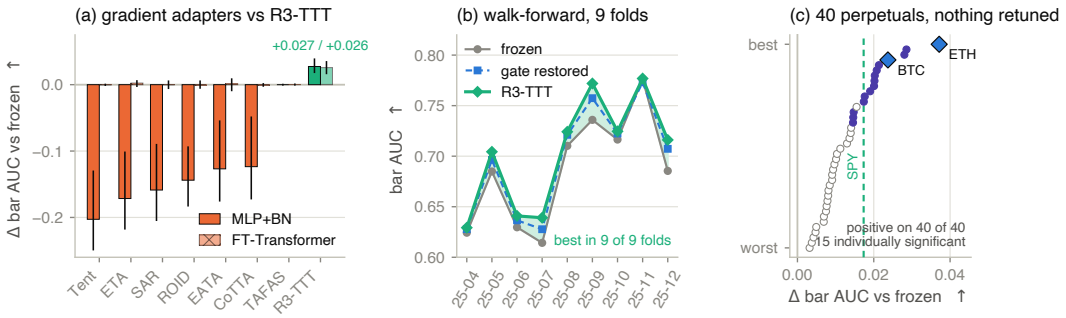

Figure 4: R3-TTA improves over the frozen model in every evaluation setting. **(a)** Paired bar-AUC gain on the two neural backbones. **(b)** Per-fold walk-forward AUC. **(c)** Gain on 40 perpetuals; filled markers exclude zero, diamonds are the declared references, the dashed line is the SPY gain.

BN statistics alone losing 0.1238, whereas on the FT-Transformer backbone the same methods are inert within 0.003 of frozen. TAFAS, which is designed for delayed targets, stays strictly at frozen performance (0.7514 and 0.7227). Every paired R3-TTA-minus-gradient interval excludes zero on both backbones. The contrast between performance degradation during batch statistics estimation and complete inertia in its absence serves as the clear signature of normalization-induced covariate shift.

Comparison with retrieval. Among methods reading matured history, predicting from retrieved labels alone (T3A, kNN, AdaNPC votes) generally falls below or merely matches the frozen baseline; the value lies in fusing the local estimate with the frozen forecast. R3-TTA achieves the top AUC across all three backbones (0.7852, 0.7788, and 0.7483). At matched ensemble size, it outperforms the selected kNN-LM (0.7792, 0.7678, 0.7529) and exceeds the fixed grid kNN-LM on GBT (0.7852 vs. 0.7808) and MLP+BN (0.7788 vs. 0.7726), trailing slightly only on FT-Transformer (0.7483 vs. 0.7554).

Frozen language backbones. Four frozen checkpoints supply text features from the complete post to a validation-selected readout over the same nine state features. Table 3 collects all seven backbones: the gain over the matched frozen predictor is positive on every one, reaching 0.8023 with Seed-OSS-36B, and three of the four language rows survive the declared 20-contrast correction. These rows support adaptation with frozen parameters and the value of the R3-TTA.

Table 3: AUC of frozen backbones vs. R3-TTAAdapt.

	Backbone	Frozen	R3-TTA	Gain ↑	95% CI
Tabular and neural					
1	GBT (5-seed)	0.7679	**0.7852**	+0.0173	[+0.0081, +0.0256]
2	MLP + BatchNorm	0.7514	**0.7788**	+0.0274	[+0.0187, +0.0384]
3	FT-Transformer	0.7226	**0.7483**	+0.0257	[+0.0169, +0.0344]
Frozen language backbones, 1,024-token context					
4	Seed-OSS-36B	0.7909	**0.8023**	+0.0114	[+0.0005, +0.0208]
5	Qwen3.8-27B	0.7793	**0.7943**	+0.0151	[+0.0029, +0.0292]
6	Qwen3.6-27B	0.7666	**0.7827**	+0.0161	[+0.0042, +0.0267]
7	Qwen3.6-35B-A3B (MoE)	0.7818	**0.7926**	+0.0107	[−0.0025, +0.0240]

4.2 Ablation Study

Table 4 evaluates each variant on **SEED-OSS-36B**. The full R3-TTA achieves optimal performance (0.8023 Test AUC), with every ablation strictly degrading accuracy ($\Delta < 0$). **Reliability Weighting.** Replacing our default blending mechanism with neighbor-agreement weighting ($\beta\rho_b^{(m)}$, Eq. (10)) lowers performance across all panels, confirming that fixed reliability weights fail to adapt to cross-panel shift. **Routing Spaces.** Restricting routing to any single space (market state, timing, or joint) degrades test AUC. Furthermore, omitting joint emphasis underperforms multi-space averaging, demonstrating the necessity of joint representation in non-stationary regimes. **Shrinkage, Fusion, and Ensembling.** Removing prior shrinkage, replacing logit fusion etc. ensembling consistently hurts

Table 4: All ablations of R3-TTA on the three panels (bar AUC). Δ is the test-panel difference from R3-TTA with its paired 95% interval.

Configuration	Walk-forward	Validation	Test	Test Δ [95% CI]
Frozen model	0.6970	0.7406	0.7679	-0.0344 $[-0.0429, -0.0259]$
+ neighbor-agreement reliability weight	0.7012	0.7430	0.7749	-0.0274 $[-0.0340, -0.0208]$
− space averaging (market state only)	0.6940	0.7247	0.7750	-0.0273 $[-0.0421, -0.0125]$
− space averaging (timing only)	0.7116	0.7531	0.7824	-0.0199 $[-0.0280, -0.0118]$
− space averaging (joint only)	0.7201	0.7501	0.7831	-0.0192 $[-0.0244, -0.0140]$
− joint emphasis (uniform spaces)	0.7131	0.7496	0.7852	-0.0171 $[-0.0192, -0.0150]$
− prior shrinkage (raw clipped vote)	0.7177	0.7413	0.7881	-0.0142 $[-0.0216, -0.0068]$
− logit fusion (probability mixture)	0.7196	0.7496	0.7847	-0.0176 $[-0.0207, -0.0145]$
− multi-scale ($k = 32$ only)	0.7164	0.7527	0.7862	-0.0161 $[-0.0188, -0.0134]$
− temperature average ($T = 0.25$ only)	0.7150	0.7505	0.7850	-0.0173 $[-0.0181, -0.0165]$
− temperature average ($T = 1$ only)	0.7156	0.7501	0.7847	-0.0176 $[-0.0183, -0.0169]$
− prior-strength average ($\lambda = 2$ only)	0.7178	0.7493	0.7860	-0.0163 $[-0.0178, -0.0148]$
− prior-strength average ($\lambda = 8$ only)	0.7128	0.7501	0.7837	-0.0186 $[-0.0203, -0.0169]$
R3-TTA	**0.7254**	**0.7904**	**0.8023**	—

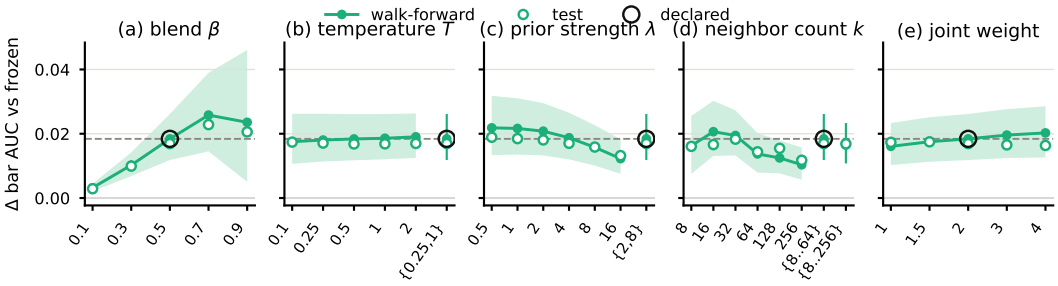

Figure 5: Sensitivity analysis of the gain of R3-TTA over frozen on the walk-forward (line) and test (hollow) panels, where each panel varies one setting and rings mark the declared values.

generalization. This validates that shrinkage filtering, logit-level integration, and hyperparameter averaging collectively stabilize predictions against noise.

4.3 Sensitivity Analysis

Figure 5 illustrates the sensitivity of R3-TTA across key hyperparameters blend β, prior strength λ, temperature, neighborhood scale k, and joint-space weight, on both the walk-forward and test panels. The gain over the frozen baseline remains strictly positive across all configurations, bounded below by $+0.0029$. While temperature and joint-space weight exhibit minimal performance impact ($\Delta < 0.0012$), prior strength and scale selection prove more critical: overly strong priors ($\lambda = 16$) or overly large neighborhood sizes noticeably attenuate gains, justifying our multi-scale averaging strategy. The blending parameter β exerts the strongest influence, as it directly controls the shift toward local estimates; setting $\beta = 0.1$ reduces the test gain to its lower bound.

5 Conclusion

This paper introduces R3-TTA, a closed-form test-time adaptation framework addressing target delay under shared supervision. By leveraging matured historical outcomes retrieved via arrival timing and market regime, R3-TTA solves a prior-shrunk Bernoulli problem through a lightweight, single-division operator. Across seven model backbones, pre-deployment walk-forward setups, forward deployments, and crypto perpetuals, R3-TTA consistently outperforms both frozen models and gradient-based adapters. Ultimately, R3-TTA offers a stable, non-parametric approach to test-time adaptation under delayed feedback, achieving robust generalization via localized variance reduction.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods and to provide feedback on research methodology and experiments. We have not used generative AI tools to generate synthetic data sets, develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, or interpret results. Translation, dataset cleaning and reformatting, and qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to create and edit software code, such as experiment and evaluation scripts, and to edit the text of the paper for readability. We have reviewed all AI-assisted work. All AI-assisted code was read, executed, and tested by the authors, and all reported results were checked against the saved experiment logs. Feedback from AI tools on methodology and experiments was treated as suggestions; the research questions, method design, and conclusions were determined by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of AI.

ETHICS STATEMENT

The study uses public posts and market data. Automated trading based on political speech can amplify market volatility. The analysis evaluates predictive associations and does not constitute a trading recommendation.

REPRODUCIBILITY STATEMENT

Appendix D specifies the data, targets and fitting environments. Appendix A gives the derivations, Algorithm 1 and implementation conventions, and Appendix B documents the evaluation protocol and comparator implementations. Two requirements are load-bearing. First, nearest-neighbor ties are broken by `argpartition` over a BLAS Gram matrix, so all reported numbers pin BLAS thread counts to one and each track asserts that the frozen and reference AUCs reproduce to 10^{-9} before scoring. Second, the forward-window protocol was hashed before any forward row was read. Code and predictions are intended for release. The source code is in https://anonymous.4open.science/r/R3TTA

REFERENCES

Scott R. Baker, Nicholas Bloom, and Steven J. Davis. Measuring economic policy uncertainty. *The Quarterly Journal of Economics*, 131(4):1593–1636, 2016.

Yassir Bendou, Amine Ouasfi, Vincent Gripon, and Adnane Boukhayma. ProKeR: A kernel perspective on few-shot adaptation of large vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25092–25102, 2025. doi: 10.1109/CVPR52734.2025.02336.

Ekkehart Boehmer, Jim Masumeci, and Annette B. Poulsen. Event-study methodology under conditions of event-induced variance. *Journal of Financial Economics*, 30(2):253–272, 1991. doi: 10.1016/0304-405X(91)90032-F.

Malik Boudiaf, Romain Mueller, Ismail Ben Ayed, and Luca Bertinetto. Parameter-free online test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Heleen Brans and Bert Scholtens. Under his thumb: The effect of president donald trump's twitter messages on the US stock market. *PLOS ONE*, 15(3):e0229931, 2020. doi: 10.1371/journal.pone.0229931.

Mouxiang Chen, Lefei Shen, Han Fu, Zhuo Li, Jianling Sun, and Chenghao Liu. Calibration of time-series forecasting: Detecting and adapting context-driven distribution shift. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 341–352, 2024.

Constantin Colonescu. The effects of Donald Trump's tweets on US financial and foreign exchange markets. *Athens Journal of Business & Economics*, 4(4):375–388, 2018. doi: 10.30958/ajbe.4-4-2.

Yossi Gandelsman, Yu Sun, Xinlei Chen, and Alexei A. Efros. Test-time training with masked autoencoders. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022.

Qi Ge, Alexander Kurov, and Marketa Halova Wolfe. Do investors care about presidential company-specific tweets? *Journal of Financial Research*, 42(2):213–242, 2019.

Taesik Gong, Jongheon Jeong, Taewon Kim, Yewon Kim, Jinwoo Shin, and Sung-Ju Lee. NOTE: Robust continual test-time adaptation against temporal correlation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022.

Yury Gorishniy, Ivan Rubachev, Nikolay Kartashev, Daniil Shlenskii, Akim Kotelnikov, and Artem Babenko. TabR: Tabular deep learning meets nearest neighbors. In *International Conference on Learning Representations (ICLR)*, 2024.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025.

Yifan Hu, Yuante Li, Peiyuan Liu, Yuxia Zhu, Naiqi Li, Tao Dai, Shutao Xia, Dawei Cheng, and Changjun Jiang. A comprehensive and practical benchmark for financial time series forecasting. *Frontiers of Computer Science*, 20(10):2010629, 2026. doi: 10.1007/s11704-026-51064-5.

Yusuke Iwasawa and Yutaka Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Adilbek Karmanov, Dayan Guan, Shijian Lu, Abdulmotaleb El Saddik, and Eric Xing. Efficient test-time adaptation of vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. Training-free Dynamic Adapter; arXiv:2403.18293.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations (ICLR)*, 2020.

HyunGi Kim, Siwon Kim, Jisoo Mok, and Sungroh Yoon. Battling the non-stationarity in time series forecasting via test-time adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 17868–17876, 2025. doi: 10.1609/aaai.v39i17.33965. arXiv:2501.04970.

Johnson D. Kinyua, Charles Mutigwe, Daniel J. Cushing, and Michael Poggi. An analysis of the impact of President Trump's tweets on the DJIA and S&P 500 using machine learning and sentiment analysis. *Journal of Behavioral and Experimental Finance*, 29:100447, 2021.

James W. Kolari and Seppo Pynnönen. Event study testing with cross-sectional correlation of abnormal returns. *The Review of Financial Studies*, 23(11):3996–4025, 2010. doi: 10.1093/rfs/hhq072.

Jonghyun Lee, Dahuin Jung, Saehyung Lee, Junsung Park, Juhyeon Shin, Uiwon Hwang, and Sungroh Yoon. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. In *International Conference on Learning Representations (ICLR)*, 2024.

Alejandro Lopez-Lira and Yuehua Tang. Can ChatGPT forecast stock price movements? return predictability and large language models. *SSRN Working Paper 4412788*, 2023.

Robert A. Marsden, Mario Döbler, and Bin Yang. Universal test-time adaptation through weight ensembling, diversity weighting, and prior correction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2543–2553, 2024. doi: 10.1109/WACV57701.2024.00254.

Heitor R. Medeiros, Hossein Sharifi-Noghabi, Gabriel L. Oliveira, and Saghar Irandoust. Accurate parameter-efficient test-time adaptation for time series forecasting. *arXiv preprint arXiv:2506.23424*, 2025.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, 2022.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. In *International Conference on Learning Representations (ICLR)*, 2023.

Daniel Perico Ortiz. Economic policy statements, social media, and stock market uncertainty: An analysis of donald trump's tweets. *Journal of Economics and Finance*, 47(2):333–367, 2023. doi: 10.1007/s12197-022-09608-5.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pp. 9229–9248, 2020.

Valentin Thomas, Junwei Ma, Rasa Hosseinzadeh, Keyvan Golestan, Guangwei Yu, Maksims Volkovs, and Anthony Caterini. Retrieval & fine-tuning for in-context tabular models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Devavrat Tomar, Guillaume Vray, Jean-Philippe Thiran, and Behzad Bozorgtabar. Un-mixing test-time normalization statistics: Combatting label temporal correlation. In *International Conference on Learning Representations (ICLR)*, 2024.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations (ICLR)*, 2021.

Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Wentao Xu, Weiqing Liu, Chang Xu, Jiang Bian, Jian Yin, and Tie-Yan Liu. REST: Relational event-driven stock trend forecasting. In *Proceedings of the Web Conference (WWW)*, pp. 1–10, 2021.

Yumo Xu and Shay B. Cohen. Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1970–1979, 2018.

Han-Jia Ye, Huai-Hong Yin, De-Chuan Zhan, and Wei-Lun Chao. Revisiting nearest neighbor for tabular data: A deep tabular baseline two decades later. In *International Conference on Learning Representations (ICLR)*, 2025.

Longhui Yuan, Binhui Xie, and Shuang Li. Robust test-time adaptation in dynamic scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15922–15932, 2023.

Renrui Zhang, Wei Zhang, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-Adapter: Training-free adaption of CLIP for few-shot classification. In *European Conference on Computer Vision (ECCV)*, pp. 493–510, 2022. doi: 10.1007/978-3-031-19833-5_29.

Wentao Zhang, Lingxuan Zhao, Haochong Xia, Shuo Sun, Jiaze Sun, Molei Qin, Xinyi Li, Yuqing Zhao, Yilei Zhao, Xinyu Cai, Longtao Zheng, Xinrun Wang, and Bo An. A multimodal foundation agent for financial trading: Tool-augmented, diversified, and generalist. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 4314–4325, 2024.

Yi-Fan Zhang, Xue Wang, Kexin Jin, Kun Yuan, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan. AdaNPC: Exploring non-parametric classifier for test-time adaptation. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 41647–41676. PMLR, 2023.

Zhihan Zhou, Liqian Ma, and Han Liu. Trade the event: Corporate events detection for news-based event-driven trading. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021. URL https://aclanthology.org/2021.findings-acl.186.

Appendix contents

A Method Details and Proofs **14**

A.1 Tree Implementation 14

A.2 Notation 14

A.3 Label Maturity 14

A.4 Numerical Conventions and Retrieval Order 14

A.5 Routing Geometry 15

A.6 Proofs 15

A.7 Algorithm 16

B Evaluation Protocol and Comparators **16**

B.1 Declared Protocol 16

B.2 Reliability-Weight Ablation 16

B.3 Comparator Implementations 17

C Limitation **17**

D Data, Evaluation Unit and Experiment Scope **17**

D.1 Trump-Post Archive 17

D.2 Time Alignment 17

D.3 Prediction Environments 18

D.4 Evaluation Unit 18

D.5 Paired Inference 18

D.6 Reproduction Artifacts 18

A Method Details and Proofs

A.1 Tree Implementation

In the clean event benchmark the frozen predictor is a five-seed gradient-boosted ensemble on the five timing and four lagged-market features, and the same nine features, pooled per bar, form the retrieval query (Figure 2). Seed averaging precedes the logit transform, so ℓ_e is the logit of the mean seed probability, whereas member averaging follows probability fusion (eq. (9)).

A.2 Notation

Table 5 collects the symbols used in the method. We write the logistic function as sigm and reserve $\sigma(\cdot)$ for a generated sigma-field.

Table 5: Notation for the model and targets. Each quantity is defined where first used.

Symbol	Meaning
$e, t_e, x_e;\ i;\ b$	Post, publication time, observable metadata; instrument; aligned bar.
$\mathrm{op}(b), \mathrm{cl}(b), O_{b,i}, C_{b,i}$	Bar-open/close timestamps and prices.
$E_b, \mathcal{E}, \mathcal{A}, \mathcal{H}_b$	Same-bar posts, all posts, asset universe and available information.
r_b^{evt}	Nominal event return.
$\kappa_{\mathrm{evt}};\ y$	Training-fitted threshold in bps; corresponding binary exceedance label.
$v_e; \theta_s$	Event features; frozen event-seed parameters.
$p_e^0, \ell_e, p_e^{(m)}, p_e^{\mathrm{SF}}$	Frozen probability/logit, member probability and ensemble probability.
$\mathrm{sigm}, \mathrm{logit}, \mathrm{CE}$	Logistic function, log-odds transform and binary cross-entropy.
$t_0, a_j, \mathcal{M}, M, \pi_0$	Deployment time, label availability, fixed memory, its size and label prior.
$r_j, r_e, \bar{r}_b, d_r, d, \mathcal{N}_k(b)$	Standardized memory/event vectors, pooled query, dimension, distance and neighbor set.
$k, \mathcal{K}, T, \lambda, w_{bj}, W_b(k,T)$	Neighbor count/scales, temperature, prior mass, retrieval weight/total.
$J_{b,k}, \widehat{\pi}_{b,k}, u_{b,k}, \bar{u}_b$	Local objective, fast probability/logit and scale-averaged logit.
$h_{b,k}, \alpha_{b,k}, \beta, \mathcal{G}, m, \nu_m$	Weighted neighbor vote, data share of the shrunk estimate, fixed blend, member grid, member index and fixed member weight.
ρ_b	Neighbor-agreement reliability weight, evaluated as an ablation (Section 4.2).

A.3 Label Maturity

Forward alignment permits exact timestamp matches. The event return runs from the previous completed SPY close to the close three observed bars after the aligned bar. Its label is therefore available at the latter close, which can occur more than one hour after entry when the interval crosses a market gap. The historical event cache sets `available_time` to aligned time plus one hour. The maturity audit identifies 15 rows on one bar at the 2025-04-01 boundary and one row/bar each at 2025-07-01 and 2025-10-01 whose actual labels mature after the nominal boundary. No such row occurs at 2026-01-01. Purging these bars from both fitting labels and memory leaves the headline unchanged.

A.4 Numerical Conventions and Retrieval Order

The clean event logit clips probabilities to $[10^{-6}, 1-10^{-6}]$. Exponents cap the nonnegative scaled distance gap at 700. The fast state is also computed from a clipped probability, and clipping is inactive when the shrunk probability already lies in $[10^{-6}, 1-10^{-6}]$. The minimizer in Proposition 1 (proof in Appendix A.6) is the mathematical solution before clipping. All transformations use the memory or fitting window specified for that environment and no later data.

The retrieval implementation keeps memory rows in a fixed chronological order, obtains up to 256 candidates by partial selection, and stably sorts those candidates by distance. This convention fixes candidate ordering within the recorded implementation. Stable sorting after partial selection leaves the order among all equidistant memory items to the implementation, without an earliest-time rule. Reproduction therefore retains the candidate budget, memory order and retrieval implementation.

A.5 Routing Geometry

Figure 6 shows how the choice of routing coordinates changes the retrieved neighborhood while leaving the query and historical memory fixed.

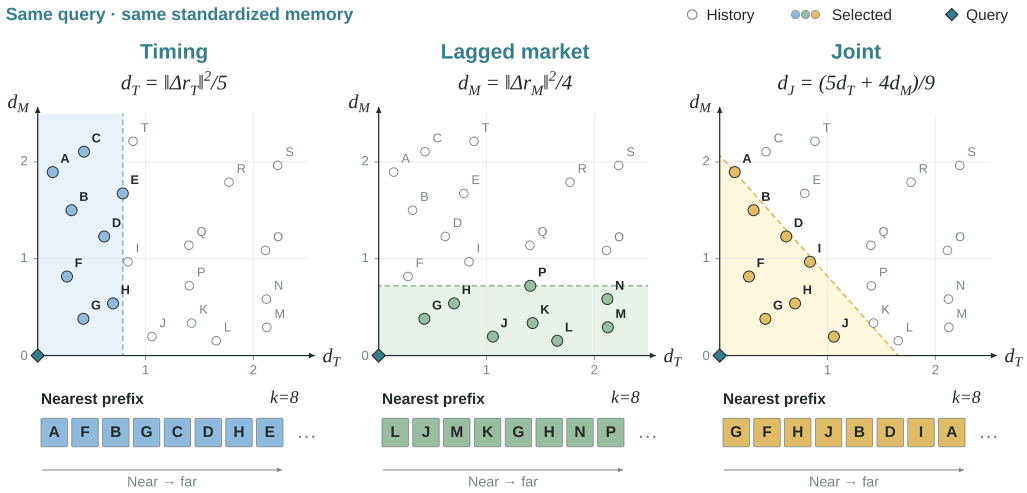

Figure 6: **Routing changes which historical bars are nearby.** The same query and memory are compared using five timing coordinates, four lagged-market coordinates, or their union. Each plotted point records its timing and market distance from the query. With common coordinate-wise standardization, the joint distance is $d_J = (5d_T + 4d_M)/9$. The shaded regions and ordered eight-neighbor prefixes show the resulting retrieval choices. This is a synthetic distance-component diagram, not a learned embedding or an empirical projection.

A.6 Proofs

Proof of Proposition 1. For every retrieved item $w_{bj} > 0$, and the nearest retrieved item has weight 1, hence $W_b(k,T) \geq 1 > 0$. Write $s(u) = \mathrm{sigm}(u)$ and use $\frac{d}{du}\mathrm{CE}(q, s(u)) = s(u) - q$. Differentiating eq. (4),

$$J'_{b,k}(u) = \sum_{j \in \mathcal{N}_k(b)} w_{bj}\big(s(u) - y_j\big) + \lambda\big(s(u) - \pi_0\big) = \big(W_b(k,T) + \lambda\big)\big(s(u) - \widehat{\pi}_{b,k}\big),$$

$$J''_{b,k}(u) = \big(W_b(k,T) + \lambda\big)\, s(u)\big(1 - s(u)\big) > 0 \qquad \text{for all } u \in \mathbb{R},$$

with $\widehat{\pi}_{b,k}$ as in eq. (5). Because $\lambda > 0$, $\pi_0 \in (0,1)$ and $W_b(k,T) > 0$, the numerator of $\widehat{\pi}_{b,k}$ lies strictly between 0 and its denominator, so $0 < \widehat{\pi}_{b,k} < 1$. The equation $J'_{b,k}(u) = 0$ therefore has the unique finite solution $u = \mathrm{logit}(\widehat{\pi}_{b,k})$, and strict convexity makes it the unique global minimizer.

Substituting $\sum_{j \in \mathcal{N}_k(b)} w_{bj} y_j = W_b(k,T)\, h_{b,k}$ into eq. (5) gives the convex combination in eq. (6), with $\alpha_{b,k} = W_b(k,T)/(W_b(k,T) + \lambda) \in (0,1)$.

At $\lambda = 0$ the objective is a weighted Bernoulli cross-entropy with target mean $h_{b,k} = W_b(k,T)^{-1} \sum_{j \in \mathcal{N}_k(b)} w_{bj} y_j$, the weighted vote of eq. (6). If $0 < h_{b,k} < 1$, equivalently if both labels occur among the retrieved items, the unique finite minimizer is $u = \mathrm{logit}(h_{b,k})$. If all retrieved labels are one, $J_{b,k}(u) = W_b(k,T)\log(1 + e^{-u})$ decreases to 0 as $u \to \infty$ and has no finite minimizer. The all-zero case is symmetric, with the infimum attained only as $u \to -\infty$. □

Figure 3 distinguishes prior shrinkage of the local estimate (panel b) from its fusion with the frozen forecast (panel c).

Proof of Proposition 2. Fix the backbone in deterministic inference mode, the fitted transformations, the memory and the tie-breaking rule. For a complete query bar E_b, the construction computes the

bar-level routing vector $\bar{r}_b$, evaluates distances from $\bar{r}_b$ to the fixed memory vectors $\{r_j\}_{j=1}^{M}$, selects the index sets $\mathcal{N}_k(b)$, forms the weights w_{bj}, computes the fast states $u_{b,k}^{(m)}$ and their averages $\bar{u}_b^{(m)}$, and combines each $\bar{u}_b^{(m)}$ with the post-specific frozen logits ℓ_e, $e \in E_b$. Each of these quantities is a deterministic function of $(E_b, \{\ell_e : e \in E_b\}, \mathcal{M})$ alone, and none depends on the contents, order, predictions or batch statistics of any other deployment bar. In exact arithmetic the prediction assigned to each $e \in E_b$ is therefore invariant to the order in which deployment bars are processed, to how they are batched, and to which other bars are processed alongside E_b. The statement concerns exact arithmetic. Across numerical implementations, finite precision changes roundoff, and near exact distance ties the tie handling is implementation-dependent. It also excludes changes to the contents of E_b, updates to the memory, and any sequential rule that feeds past deployment outputs into future inputs. □

A.7 Algorithm

Algorithm 1 lists the steps of R3-TTA for one complete aligned bar.

Algorithm 1 R3-TTA for one complete aligned bar

Require: Frozen predictor, immutable memory $\mathcal{M}$, prior π_0, fitted standardizers, complete bar E_b, grid $\mathcal{G}$, scales $\mathcal{K}$, fixed $\beta = \frac{1}{2}$

1: Compute frozen probabilities p_e^0 and slow logits ℓ_e for $e \in E_b$
2: **for** each routing space **do**
3: Standardize and pool the query $\bar{r}_b$, then retrieve and order up to 256 candidates with fixed tie-breaking
4: **for** each (T, λ), i.e. each member m of this space **do**
5: Form weights by eq. (3); for each $k \in \mathcal{K}$ use the length-k candidate prefix and compute $u_{b,k}^{(m)}$ by eq. (5)
6: Average the four logits $u_{b,k}^{(m)}$ into $\bar{u}_b^{(m)}$ by eq. (7); compute $p_e^{(m)}$, $e \in E_b$, by eq. (8)
7: **end for**
8: **end for**
9: Return the weighted probability average p_e^{SF} of eq. (9) (weight $\nu_m = 2$ for joint-space members, 1 otherwise) before observing the bar's outcome, and leave all fitted state unchanged

B Evaluation Protocol and Comparators

B.1 Declared Protocol

All hyperparameters of R3-TTA, including the weight of the joint routing space, were fixed on the pre-deployment walk-forward and validation panels and declared in a protocol file before the test read reported here, and the forward-window evaluation protocol was hashed before it was run. The declaration is identified by the SHA-256 of its JSON body with sorted keys, `f4063cad...8002`. Every entry point pins one BLAS thread, and every reference arm reproduces its previously computed level exactly before any adapter is scored.

B.2 Reliability-Weight Ablation

The ablation of Section 4.2 replaces the fixed blend β in eq. (8) by $\beta\rho_b^{(m)}$, where ρ is an agreement score derived from the unshrunk vote $h_{b,k}^{(m)} = W_b(k,T)^{-1} \sum_{j \in \mathcal{N}_k(b)} w_{bj} y_j$ and the across-scale logit dispersion,

$$c_b^{(m)} = \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} 2\left|h_{b,k}^{(m)} - \tfrac{1}{2}\right|, \qquad \rho_b^{(m)} = \frac{c_b^{(m)}}{1 + \sqrt{|\mathcal{K}|^{-1} \sum_{k \in \mathcal{K}} \left(u_{b,k}^{(m)} - \bar{u}_b^{(m)}\right)^2}} \in [0,1], \tag{10}$$

so that the local estimate is trusted more when neighbors agree in label and scales agree in logit.

B.3 Comparator Implementations

Table 6: Implementation scope of the prediction baselines. Each row names the retained mechanism and the adaptation needed for binary tabular event data. Configuration selection uses validation.

Method	Implementation and declared change
AdaNPC	Cosine neighbor voting and confidence-based pseudo-label growth are retained. Routing vectors replace the penultimate neural representation for the GBT port; no BatchNorm update is available there.
T3A	Low-entropy support retention and cosine-to-centroid scores are retained. The GBT port initialises supports using frozen predictions on memory bars because the tree has no linear classifier head.
LAME	The reference Laplacian optimizer and affinities are retained. A singleton bar has an empty nearest-neighbor affinity.
Tent	Entropy minimization updates normalization affine parameters. The table separately includes a BatchNorm-statistics-only control that takes no gradient step.
EATA / ETA	Filtering is retained; the similarity threshold is selected for binary outputs. EATA includes the anti-forgetting term, whereas ETA omits it.
SAR	Sharpness-aware updates and reset logic are retained. The reset constant is selected on validation because its original scaling targets many-class classification.
CoTTA / ROID	Their adaptation mechanisms are retained; feature dropout and Gaussian jitter provide tabular augmentations. This augmentation choice differs from the image setting.
TAFAS	Cross-entropy replaces the forecasting loss. The label arrives atomically, so progressive partial-ground-truth feedback and the original Prediction Adjustment module are omitted.

Every comparator uses the frozen model of its column, the same nine pre-entry features and the same matured memory, and none observes a deployment label. Comparator hyperparameters are selected on validation. AdaNPC fusion uses the configuration selected on each backbone's own validation panel.

C Limitation

While R3-TTA demonstrates consistent non-parametric gains, its scope warrants further exploration. Empirical evaluation is currently restricted to US equities and cryptocurrency perpetuals; validating the framework across a broader spectrum of global asset classes remains future work.

D Data, Evaluation Unit and Experiment Scope

D.1 Trump-Post Archive

The source collection contains timestamped public posts, text, post identifiers, URLs, media and repost indicators, and engagement snapshots. The shorter archive contains 8,847 unique post records from 2024-11-01 to 2026-04-10 (UTC). A longer archive contains 32,436 records from 2022-02-14 to 2026-04-10. These are overlapping source collections rather than additive samples. The final aligned panel used in the main experiments contains 8,678 posts (Table 7). Session-based exports are clock-time filters, and alignment to observed market bars determines eligibility for the empirical study.

D.2 Time Alignment

Post times are converted to the source's New York market clock with daylight-saving handling. Each post is aligned to the first observed bar whose opening timestamp is at or after publication, including exact matches and extended hours. Closed-market gaps are skipped, so the nominal event label can include an overnight or weekend return. The implementation uses the observed bar grid without a separate holiday calendar. Engagement snapshots are excluded from the clean event predictor and routing features, as their values become known after publication.

D.3 Prediction Environments

Table 7: **Data of the event prediction task.** Event rows are posts. Observed dates include extended-hours event bars.

Quantity	Development	Validation	Test
Period	Before 2025-Q4	2025-Q4	2026 Jan–Apr
Event posts	5,277	1,566	1,835
Distinct outcome bars	1,842	407	525
Mean posts per bar	2.86	3.85	3.50
Observed evaluation dates	—	63	68

The clean event input has five timing features and four lagged-market features. Its primary test contains 1,835 posts, 525 outcome bars and 68 observed dates. The five-seed tree and immutable memory use pre-2026 data. The neural architectures are reused from an earlier validation sweep that included engagement features. Both neural backbones are refitted, and every comparator's settings are selected, on the clean feature panel. Each adaptation gain is measured against the corresponding frozen model. The pre-deployment walk-forward has nine monthly folds from 2025-04 to 2025-12. Each fold refits the frozen model and rebuilds the memory from rows strictly before its month and pools to $1{,}594$ scored bars. The forward window is a single fresh deployment at 2026-04-16, with the frozen model and memory built from every row before 2026-04-11, and it scores 758 bars to 2026-08-25. The crypto panel transplants the equity protocol unchanged to 40 Binance USD-margined perpetuals (851 test bars each), aligning posts from their raw publication time, as that grid runs continuously.

D.4 Evaluation Unit

Multiple posts can share one market outcome. In test, 995 positive and 840 negative post rows create 835,800 positive–negative pairs, and 188 positive and 337 negative outcome bars create 63,356 pairs. Post-level AUC thus counts 13.19 times as many pairs, with 71.2% of pair mass attached to the busiest 5% of bars. We compute primary AUC and Brier score from within-bar mean predictions, giving each outcome bar one observation. Post AUC is secondary. The memory follows the same convention, with one pooled routing vector and one label per historical outcome bar.

D.5 Paired Inference

The compared arms use the same observations. Reported 95% intervals use 5,000 paired moving-block bootstrap replicates with blocks of five observed dates and seed 42, carrying all observations of each sampled date together. One-sided tail probabilities are the bootstrap fraction of paired differences at or below zero. The forward-window contrasts form one declared family with a Holm correction.

D.6 Reproduction Artifacts

The experiment artifacts contain the clean event predictions, the matched neural implementations, the walk-forward, forward-window and crypto predictions, and the ablation and sensitivity traces. The shared post and market inputs remain read-only. Appendices A.3 and A.4 specify the actual-maturity sensitivity, retrieval convention and numerical boundary cases.