FROM NEWS TO EXPLAINABLE FORECAST: COMMONSENSE, CAUSAL, AND LOGIC-CONSISTENT REFLECTION FOR LLM FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Incorporating external news into numerical time-series forecasting is critical for capturing exogenous shocks that historical data alone cannot explain. However, existing LLM-based iterative forecasters generate reflections that merely identify missing news categories, lacking causal reasoning about their impact or plausibility checks, which makes the refinement loop prone to spurious correlations and slow convergence. We address this by introducing an explainable reflection framework with three structural constraints: a causal chain linking missing news to prediction errors, a commonsense sanity check, and an explicit logic consistency constraint against prior reflections. We then train an Explainable Process Reward Model (EPRM) that scores candidate news articles by jointly optimizing for expected error reduction and the above rational quality. Only candidates passing a quality gate are incorporated. Experiments on electricity, bitcoin, traffic, and exchange-rate benchmarks show that our method achieves a 10.5% average RMSE reduction over the unconstrained iterative baseline. Furthermore, on bitcoin, the spurious-item rate falls from 40.3% to 4.0%, and the method receives high automatic-judge scores for causal, commonsense, and logic quality, as well as high expert scores on these dimensions and on usefulness.

1 INTRODUCTION

Time series forecasting supports decision making in domains such as finance, energy, and transportation (Alghamdi et al., 2019; Fildes et al., 2022; Gross & Galiana, 1987). Although modern forecasting models can capture rich temporal patterns from historical observations, they remain vulnerable to abrupt distribution shifts caused by exogenous events, such as policy changes, natural disasters, or market-moving announcements. These events are often weakly reflected, or not reflected at all, in the numerical history available at prediction time. This limitation motivates forecasting systems that can incorporate external textual evidence, especially news reports, into the prediction process.

News provides timely descriptions of latent drivers behind future time-series movements. Recent LLM-based forecasting pipelines have started to exploit this signal by prompting a reasoning agent to filter relevant articles and a separate evaluation agent to analyze prediction errors and propose additional retrieval queries (Wang et al., 2024). This iterative design is attractive because it converts missing-context errors into language feedback that can be used to refine the news set. However, the refinement loop is weakly supervised and lacks interpretability. In prior iterative pipelines, a reflection only indicates a missing category (e.g., "extreme heat news"), but it does not explain the causal pathway through which that news affects the forecast, nor does it verify the reasoning against commonsense or ensure logical coherence across iterations. Consequently, the agent may chase statistically spurious correlations (e.g., a coincidental celebrity tweet during a heatwave) and accept updates that do not generalize.

We address this limitation with a framework that separates *offline refinement* from *online deployment*. Our key contribution is to make the reflection process **explainable** by enforcing three structural constraints that mimic human rational thinking. A causal constraint requires an explicit chain from the missing news to the observed forecasting error (e.g., "high temperature $\rightarrow$ increased cooling demand $\rightarrow$ higher electricity load"). A commonsense constraint requires a justification that the reflection does not violate basic domain knowledge (e.g., "summer heat waves typically increase

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.

electricity load"). A logic constraint requires an explicit consistency check against previously accepted reflections, so that the agent cannot, for instance, claim that heat both increases and decreases load under the same conditions.

To rank supplementary news candidates, we introduce an Explainable Process Reward Model (EPRM) that outputs a single quality score integrating predicted RMSE improvement and rational quality (causal soundness, commonsense alignment, logical coherence). The EPRM is trained with a combined label that requires both empirical success and rational plausibility, thereby learning to reject candidates that are statistically correlated but causally invalid. The EPRM and the refined filtering logic are obtained offline using historical data with ground truth; at inference time, the filtering logic is frozen and prediction proceeds without iterative reflection. Long articles are truncated to a fixed character budget so that the evaluation isolates news *selection* rather than compression. Experiments on electricity, bitcoin, traffic, and exchange-rate benchmarks demonstrate that our method improves prediction accuracy by 10.5% RMSE on average over the iterative baseline of Wang et al. (2024) (up to 21.0% on exchange). On bitcoin, the spurious-item rate falls from 40.3% to 4.0%, and the method receives high automatic-judge scores for causal, commonsense, and logic quality, as well as high expert scores on these dimensions and on usefulness.

Contributions. Our main contributions are summarized as follows:

- We introduce an explainable reflection framework that imposes causal, commonsense, and logic constraints on the agent's reasoning, transforming the reflection from a simple category guess into a structured, verifiable argument.
- We propose an Explainable Process Reward Model (EPRM) that jointly scores candidate news articles by their expected forecasting utility and their rational quality, and we train it with a combined label that requires both empirical error reduction and causal/commonsense/logic validity.
- To the best of our knowledge, this is the first work that integrates causal chains, commonsense justifications, and logic consistency into the iterative refinement loop of news-augmented time series forecasting, and that uses an EPRM to explicitly penalize spurious updates.

2 Related Work

Time series forecasting with external text. Classical forecasting models rely primarily on historical numerical observations (Chen et al., 2004; Dudek, 2015; Huang & Shih, 2003; Kalekar et al., 2004; Papalexopoulos & Hesterberg, 1990). Deep neural architectures improved the modeling of long-range and nonlinear dependencies (Li et al., 2019; Liu et al., 2017; Nie et al., 2022; Torres et al., 2021; Wu et al., 2021; Xia et al., 2021; Zhou et al., 2021; 2022), and large-scale pre-training further strengthened transfer to new tasks (Cao et al., 2023; Jin et al., 2023b; Wu et al., 2022; Yeh et al., 2023). A parallel line of research augments forecasting with textual evidence such as news, reports, or social media, especially in finance and energy (Cecchini et al., 2010; Schumaker & Chen, 2009; 2010; Bai et al., 2024; Obst et al., 2021). Early methods rely on manually engineered sentiment features or event indicators (Obst et al., 2021; Chowdhary, 2020), which limits their ability to capture long-range semantics and cross-document interactions.

LLM agents and iterative refinement. Recent work adapts LLMs to time series by tokenizing numerical sequences or reprogramming prompts (Gruver et al., 2023; Cao et al., 2023; Jin et al., 2023a; Rasul et al., 2023). The closest prior work is that of Wang et al. (2024), which uses LLM agents to filter news and iteratively refines the filtering strategy based on forecast errors. That framework demonstrates the value of agentic feedback loops, but each supplementary retrieval is unconditionally accepted generate textual explanations of its potential relevance to forecasting. Our work builds on this agentic formulation and replaces unguided acceptance with an explainable selection mechanism for supplementary news.

Causal inference, commonsense, and logic-consistent reasoning in agent reflection. Understanding causal structures in time series has been extensively studied (Moraffah et al., 2021). Recent work integrates causal thinking into forecasting models to improve robustness, but these approaches typically modify the forecasting backbone rather than the news selection loop. In parallel, agent reflection for LLMs has advanced rapidly. ReAct (Yao et al., 2022) and Reflexion (Shinn et al., 2023) established the paradigm of using language feedback to guide action. Related work infuses logic and commonsense into reasoning: LogiCoT applies logical consistency checks during chain-of-thought

(Liu et al., 2023a), and STaR learns from self-generated rationales (Zelikman et al., 2022). However, none of these reflection techniques has been applied to news-augmented forecasting, and they do not combine causal, commonsense, and logic constraints into a single structured reflection. Furthermore, existing process reward models (PRMs) (Lightman et al., 2023; Zeng et al., 2025; Zhang et al., 2025; Zheng et al., 2025) are used to score reasoning steps in mathematics or code generation, not to rank candidate news articles in a forecasting pipeline. Our work uniquely bridges these gaps: we design a reflection that explicitly outputs a causal chain, a commonsense justification, and a logic consistency statement; we then train an explainable PRM with a joint label that requires both RMSE improvement and rational validity. This is the first framework that integrates causal, commonsense, and logic constraints into an iterative news-selection loop with process supervision.

Context length and process supervision. Retrieved news collections can exceed a forecasting model's context budget. Prompt compression methods such as LLMLingua (Jiang et al., 2023), LongLLMLingua (Jiang et al., 2024), and RECOMP (Xu et al., 2024) reduce input length for general-purpose LLM tasks, but the notion of importance is not tied to forecasting error, and we treat compression as orthogonal to the present contribution. In this paper we truncate selected articles to a fixed character budget and focus process supervision on *which* articles to add: an explainable PRM scores candidates by expected error reduction *and* rational quality.

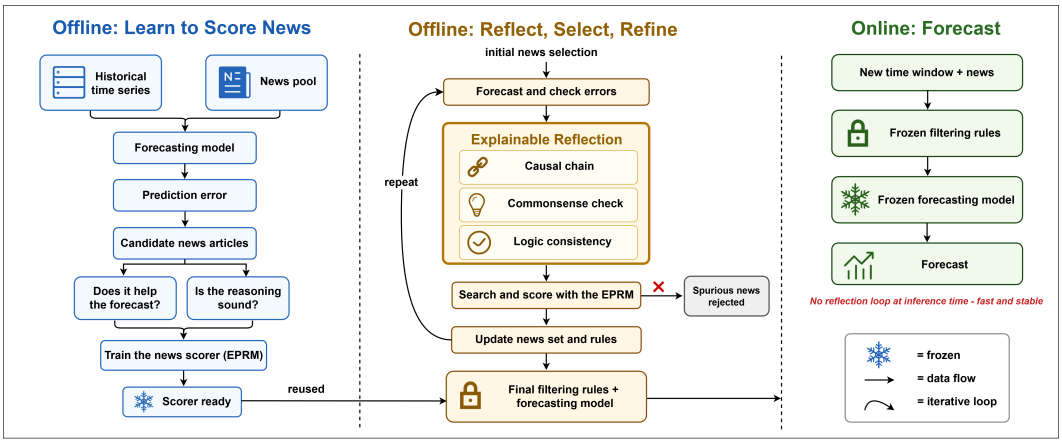

Figure 1: Overall framework. **Left (offline Stage A):** candidate articles are labeled by forecasting utility ("does it help the forecast?") and rational quality ("is the reasoning sound?"); these labels train the EPRM, which is then frozen. **Middle (offline Stage B):** an explainable reflection—causal chain, commonsense check, and logic consistency—retrieves candidates that the frozen EPRM scores; spurious news is rejected, the news set and filtering rules are updated, and the loop repeats until both are locked. **Right (online):** inference applies only the frozen filtering rules and forecasting model, with no reflection loop.

3 METHODOLOGY

We consider forecasting with historical observations and a large pool of time-aligned news articles. The main difficulty is to identify relevant articles and to interpret them in a way that is both accurate and human-understandable. Our framework therefore contains three components: (i) an LLM forecasting backbone, (ii) a reasoning agent for initial news filtering and error-driven refinement, and (iii) an Explainable Process Reward Model (EPRM) that assigns a single quality score to each supplementary-news candidate, reflecting its expected forecasting utility as well as its causal soundness, commonsense alignment, and logical consistency. The key design principle is to perform expensive search and supervision *offline*, where ground-truth targets are available, and to deploy only the frozen filtering logic *online*. Concretely, we first train the EPRM on historical training windows (Appendix B), then run EPRM-guided refinement (Algorithm 1) with that EPRM held fixed. Selected articles that exceed the context budget are truncated to a fixed character length; we do not treat compression as a contribution of this paper. Figure 1 summarizes the pipeline.

3.1 Problem Formulation

Let $\mathbf{x}_{1:T} = (x_1, \ldots, x_T)$ denote a sequence of univariate or multivariate time series observations up to the current time T. The task is to forecast the next H values $\mathbf{x}_{T+1:T+H}$. In addition to the numerical history, we have access to a collection of news articles $\mathcal{U} = \{u_1, \ldots, u_L\}$ whose publication timestamps fall within a retrieval window before the forecast time.

A reasoning agent first selects a subset $\mathcal{N}^{(0)} \subseteq \mathcal{U}$ of potentially relevant articles, which is then truncated into a textual context $S^{(0)}$. The forecasting model F_ψ conditioned on both the numerical history and the context produces the prediction:

$$\hat{\mathbf{x}}_{T+1:T+H} = F_\psi(\mathbf{x}_{1:T}, S^{(0)}).$$

During offline refinement, ground-truth targets are available on a historical validation split of the training period (test windows are never used to train the EPRM or to update filtering logic). The prediction error is used to generate an *explainable reflection* r_t that describes missing evidence together with a causal chain, commonsense justification, and logical consistency. This reflection triggers a retrieval of supplementary candidates. The pretrained EPRM ranks them by a combined quality score; candidates below a rational-quality gate are discarded, and the top-ranked remaining article is merged into the working news set. We then retrain the forecasting LoRA and re-evaluate on validation data. The persistent filtering logic—the object that will be frozen for online use—is updated only if post-retrain RMSE decreases. The loop repeats until the validation improvement becomes negligible or a fixed budget is exhausted. The final refined filtering logic is then deployed for online inference without any reflection loop.

3.2 Preliminary: Time Series Forecasting with LLMs

Following Wang et al. (2024), we cast forecasting as conditional sequence generation. A pre-trained LLM receives a prompt containing the historical series, task metadata, and a truncated textual context of relevant news, and outputs the next H values. If the numerical history is tokenized as $\boldsymbol{x}_{1:T}$ and the news context is tokenized as $\boldsymbol{e}_{1:U}$, the model learns $P_\psi(\boldsymbol{x}_{T+1:T+H} \mid \boldsymbol{x}_{1:T}, \boldsymbol{e}_{1:U})$.

We fine-tune the backbone with supervised instruction tuning and LoRA (Hu et al., 2022). The role of our method is not to change the forecasting backbone itself, but to improve the quality and *explainability* of the textual context $\boldsymbol{e}_{1:U}$ that conditions the forecast.

3.3 Iterative News Filtering with Explainable Reflection

We adopt the initial reasoning stage of Wang et al. (2024) to obtain a seed news set $\mathcal{N}^{(0)} = \{N_1, \ldots, N_{K_0}\}$. The agent uses task descriptions, few-shot examples, and chain-of-thought prompting (Brown et al., 2020; Wei et al., 2022) to identify news categories likely to influence the target series and returns structured filtering outputs.

Generating explainable reflections (causal, commonsense, logic). After evaluating the current forecasting model on a validation set, the evaluation agent produces a textual reflection r_t with three mandatory components. The *causal chain* is a sequence of intermediate variables linking the missing news to the observed forecasting error; the agent is prompted to output cause $\rightarrow$ mediator $\rightarrow$ effect (e.g., "high temperature $\rightarrow$ increased cooling demand $\rightarrow$ higher electricity load"), so that the reflection is not merely correlational. The *commonsense justification* is a short statement verifying that the reflection does not violate basic domain knowledge (e.g., "summer heat waves typically increase electricity load; the opposite would be implausible"), which filters out nonsensical updates. The *logic consistency* check is an explicit comparison of the new causal chain with previously accepted reflections and the current error pattern, reporting any conflict. We implement these three elements by prompting the LLM with a structured template that asks for cause, effect, mediators, a commonsense sanity check, and a consistency self-check. The resulting reflection r_t is then used to retrieve a candidate pool $\mathcal{C}_t = \{c_t^{(1)}, \ldots, c_t^{(K)}\}$.

EPRM-guided selection. Let $\mathcal{N}^{(t)}$ be the news set after iteration t, and let h_t summarize the current task context (error distribution, domain background, historical selections). The EPRM assigns a single score $\mathrm{score}(c) = P_\phi(\texttt{YES} \mid h_t, \tau_{<t}, c)$ to each candidate, which estimates the combined

likelihood that adding c will reduce forecasting error and that c is causally sound, commonsense-aligned, and logically consistent. Selection is two-stage (Section 3.4): rank by this score, then keep only candidates whose judge score q clears a gate θ_{gate}=0.35. The highest-scoring remaining article n_t^* is added to the news set:

$$\mathcal{N}^{(t+1)} = \mathcal{N}^{(t)} \cup \{n_t^*\}.$$

Whether that article is written into the *persistent* filtering logic is decided later, after a full LoRA retrain (Section 3.4). The refinement procedure is executed only offline; at inference time, the final filtering logic is applied without reflection.

3.4 Explainable Process Reward Model (EPRM)

A central limitation of prior iterative retrieval is the absence of step-level supervision for both forecasting utility and rational plausibility. We address this by training an EPRM that outputs a single quality score integrating both aspects.

Two uses of Δ. We distinguish a cheap *labeling* gain from a *refinement* gain; they are not interchangeable. The labeling gain $\Delta^{(k)}$ is used only to construct EPRM training labels (Appendix B): for each retrieved candidate $c^{(k)}$, we keep the current forecasting LoRA frozen and compare two forward passes (with versus without $c^{(k)}$) after the same truncation, $\Delta^{(k)} := \text{RMSE}(\mathcal{N}) - \text{RMSE}(\mathcal{N} \cup \{c^{(k)}\})$. The refinement gain Δ_{t+1} is used only inside Algorithm 1, after the EPRM has already chosen n_t^*. We merge n_t^*, retrain LoRA on the updated context, and set $\Delta_{t+1} := \text{RMSE}_t - \text{RMSE}_{t+1}$. This quantity decides whether to update the persistent filtering logic $\mathcal{L}$, not whether to accept the EPRM's ranking.

Rational ratings. Independently of Δ, we collect rational ratings for each candidate on three 1–5 Likert dimensions: *causal validity* (whether the causal chain is plausible and correctly directed), *commonsense alignment* (whether the reflection respects basic domain knowledge), and *logic consistency* (whether the reflection is coherent with previous selections and free of contradictions). A score of 1 indicates an invalid or contradictory statement and 5 indicates a fully correct and consistent one.

These ratings are mapped to a rational quality label $q_t^{(k)} \in [0, 1]$ by averaging the three scores and rescaling via $q = (\bar{s} - 1)/4$. To avoid expensive human annotation, we use DeepSeek-V4-Flash (Xu et al., 2026) as an automated judge, following the LLM-as-a-judge paradigm (Zheng et al., 2023).

Training data construction. EPRM supervision is collected *before* Algorithm 1, on historical *training* windows with ground-truth targets (full procedure in Appendix B); test windows are excluded. Round 0 starts from the iterative-baseline seed news of Wang et al. (2024). For each reflection r_t, the reasoning agent retrieves K candidates. Each candidate is labeled with the frozen-forward gain $\Delta^{(k)}$ and the judge score $q_t^{(k)}$. A candidate is considered truly beneficial only if it satisfies *both* conditions:

$$y^{(k)} = \mathbf{1}\big[\Delta^{(k)} > 0\big] \ \wedge \ \mathbf{1}\big[q_t^{(k)} > \theta\big],$$

where $\theta = 0.7$ on the $[0, 1]$ scale (mean Likert 3.8). This joint label is used *only* to supervise P_ϕ: it teaches the EPRM to reject articles that improve RMSE by chance but lack rational validity. The threshold θ is not applied as a test-time cutoff. The EPRM input concatenates task description, recent prediction errors, prior selections $\tau_{<t}$ with associated labeling Δ values, and the candidate article. We train a binary classifier with cross-entropy loss to predict $y^{(k)}$. The model is trained to emit `YES` or `NO`; at scoring time the output is $P_\phi(\texttt{YES} \mid h_t, \tau_{<t}, c)$, with no additional sigmoid. After P_ϕ is trained it is frozen and passed into Algorithm 1.

EPRM-guided selection. During offline refinement, P_ϕ is not updated. For each reflection r_t the reasoning agent retrieves K candidates. Selection uses two quantities that must not be conflated. The EPRM ranker $\text{score}(c) = P_\phi(\texttt{YES} \mid h_t, \tau_{<t}, c)$ orders candidates; because the training label required both $\Delta^{(k)} > 0$ and $q > \theta$, this score already down-weights causally unsound or commonsense-violating articles. A separate rational-quality gate is then applied: at selection time we do not reuse θ, but discard candidates with $q < \theta_{\text{gate}}$, where θ_{gate}=$\theta/2$=0.35. If the gate removes every candidate, we fall back to ungated EPRM ranking.

Let $\mathcal{C}'_t = \{c \in \mathcal{C}_t : q(c) \geq \theta_{\text{gate}}\}$ (or $\mathcal{C}_t$ if that set is empty). The selected article is $n^*_t = \arg\max_{c \in \mathcal{C}'_t} \text{score}(c)$. We do not run a second frozen-forward Δ at selection time. We then retrain LoRA and compute the refinement gain Δ_{t+1}; if $\Delta_{t+1} > 0$, we also update the persistent filtering logic $\mathcal{L}$. Separating ranking (EPRM) and the q-gate from logic updates (post-retrain RMSE) keeps the deployable policy conservative: $\mathcal{L}$ only absorbs articles that remain helpful after the backbone is allowed to adapt.

Algorithm 1 summarizes this second offline stage; the complete procedure is in Appendix C. How P_ϕ is trained is specified in Appendix B and is not part of the loop.

Relation to long-term filtering logic. The selected article n^*_t therefore plays two distinct roles. As a member of the working news set $\mathcal{N}^{(t+1)}$, it is the EPRM ranking after the q-gate for the current search step. As a candidate edit to $\mathcal{L}$, it is admitted only when the post-retrain refinement gain Δ_{t+1} is positive. By requiring rational scores both in the EPRM label and at the selection gate, the filtering logic that is eventually frozen for online use is less prone to overfitting to accidental correlations.

3.5 Context Budget

Retrieved news collections can exceed the forecasting model's context window (Appendix A). We keep the evaluation focused on explainable *selection*: each selected article is truncated to a fixed character budget before it is written into the forecast prompt. Prompt-compression methods (Jiang et al., 2023; 2024) are complementary and are not used in the reported experiments.

3.6 Overall Pipeline

Algorithm 1 Offline EPRM-Guided Refinement. Compact view of Stage B; the complete procedure is in Appendix C.

Require: Training/validation data, news pool $\mathcal{U}$; pretrained EPRM P_ϕ
Ensure: Persistent logic $\mathcal{L}^*$; fine-tuned forecaster F_{ψ^*}

```
Seed N, L from the reasoning agent; fine-tune F_ψ on truncated N
repeat
    r ← explainable reflection of validation errors          // causal, commonsense, logic
    C ← retrieve K candidates given r
    n* ← EPRM-rank C after q-gate                            // frozen P_φ
    Merge n* into N; retrain LoRA; measure validation gain Δ
    if Δ > 0 then write n* into L
until budget T_max reached or Δ < ε
return L*, F_ψ*
```

The overall framework has two offline stages followed by online inference (Figure 1). Stage A trains P_ϕ on historical windows and freezes it (Appendix B). Stage B then runs Algorithm 1: explainable reflection, EPRM ranking after the q-gate, LoRA retrain, and a conservative update of $\mathcal{L}$ only when the post-retrain gain is positive (full procedure in Appendix C). After Stage B we freeze $\mathcal{L}^*$ and F_{ψ^*}; online inference applies only the frozen logic and truncation, with no reflection loop. DeepSeek-V4-Flash (Xu et al., 2026) is used for reasoning, retrieval, reflection, and rational-quality judgments; Qwen3-8B with LoRA (Yang et al., 2025; Hu et al., 2022) is the forecasting backbone and the EPRM.

4 Experiments

We evaluate the proposed explainable reflection framework from four perspectives: (1) forecasting accuracy (RQ1), (2) contribution of each rational constraint (RQ2), (3) correctness and usefulness of the generated explanations (RQ3), and (4) robustness against spurious correlations (RQ4). Section 4.6 summarizes a qualitative electricity example; the full walkthrough is in Appendix E. A complementary per-round comparison of refinement RMSE against the iterative baseline of Wang et al. (2024) is reported in Appendix D.

4.1 EXPERIMENTAL SETUP

Datasets and baselines. We use four forecasting benchmarks: electricity demand, bitcoin price, traffic volume, and AUD/USD exchange rates. Each dataset provides historical time series paired with time-aligned news articles. We follow the train/validation/test split protocol of Wang et al. (2024) and evaluate on 100 held-out windows per domain. The forecast horizon H matches each series' native frequency: $H{=}48$ half-hourly steps (one day ahead) for electricity, $H{=}15$ days for bitcoin, $H{=}24$ hours for traffic, and $H{=}7$ days for AUD/USD. The numerical lookback window equals H in each domain. We compare against three groups of methods. The first comprises news-free time-series models trained on the same windows: N-BEATS (Oreshkin et al., 2019), PatchTST (Nie et al., 2022), DLinear (Zeng et al., 2023), LightGBM (Ke et al., 2017), Chronos (Ansari et al., 2024), iTransformer (Liu et al., 2023b), and Time-LLM (Jin et al., 2023a). The second is **No-News**, the LLM forecasting backbone (Qwen3-8B with LoRA) without any news input. The third comprises news-augmented LLM pipelines: the **Iterative Baseline (IB)** of Wang et al. (2024), which uses non-explainable reflections and accepts supplementary news without an EPRM; and **Ours (Explainable)**, the full framework with three-component reflections (causal chain, commonsense justification, and logic consistency) and the EPRM trained on combined labels of RMSE improvement and rational quality.

News-augmented methods share the same Qwen3-8B LoRA backbone and the same context budget. DeepSeek-V4-Flash (Xu et al., 2026) is used for reasoning, retrieval queries, reflection generation, and rational-quality judgments. Long news is truncated to a fixed character budget so that the comparison isolates explainable selection.

Implementation details. Explainable reflections are generated using a structured prompt that forces output of causal chain, commonsense justification, and logic consistency self-check. The EPRM is a LoRA-tuned Qwen3-8B reward model, trained offline on historical training windows and then frozen (Appendix B). Training labels are positive only when a candidate both reduces RMSE under a frozen forecasting LoRA ($\Delta^{(k)} > 0$) and receives a rational quality score $q > 0.7$ (1–5 Likert scores averaged and mapped to $[0, 1]$ by $q = (\bar{s} - 1)/4$). At selection time we rank by $P(\texttt{YES})$ and keep candidates with $q \geq 0.35$ (half of the training threshold); if none pass, we fall back to ungated EPRM ranking. Rational quality is assigned by DeepSeek-V4-Flash. We retrieve $K{=}5$ supplementary candidates per reflection and allow at most $T_{\max}{=}5$ refinement iterations, reporting the best refinement round for IB and Ours.

4.2 MAIN RESULTS: FORECASTING ACCURACY (RQ1)

Table 1 reports test RMSE, MAE, and MAPE for all methods. MAPE is omitted on Traffic because many hourly volumes are near zero, which makes the percentage error unbounded. Our explainable reflection attains the best result on every reported metric, reducing RMSE relative to IB by 4.9% on electricity (272.07 vs. 286.05), 11.6% on bitcoin (834.73 vs. 944.62), 4.4% on traffic (20.42 vs. 21.35), and 21.1% on exchange (5.95 vs. 7.54), for an average reduction of 10.5%. MAE and MAPE follow the same ranking: electricity MAE/MAPE drop from 228.25/6.78 to 214.48/6.42, bitcoin from 807.36/3.32 to 705.16/2.45, traffic MAE from 11.87 to 11.17, and exchange MAE/MAPE from 6.63/0.89 to 5.16/0.69. The gains are largest where exogenous news is most informative (bitcoin and exchange) and smaller on traffic, where strong seasonality already makes LightGBM competitive (RMSE 20.90). On exchange, IB is worse than several news-free numerical models and even No-News (7.54 vs. 7.25 RMSE), indicating that unfiltered iterative retrieval can inject harmful context; our EPRM-guided loop recovers from this and also beats the strongest numerical baseline (Chronos, 6.10 RMSE). On bitcoin, even No-News (1383.91 RMSE, 5.00 MAPE) outperforms some numerical models, and both IB and Ours further improve, consistent with a market that is heavily driven by textual events.

4.3 ABLATION STUDY (RQ2)

We evaluate the contribution of each rational constraint and of the joint training label. The **Full** model uses all three constraints and trains the EPRM on both RMSE and rational quality. We then ablate one constraint at a time, retaining the other two (**– causal**, **– commonsense**, **– logic**), and a further variant (**– rational label**) that keeps all three constraints in the reflection but trains the EPRM only on RMSE. The last variant, **– all (IB)**, removes both the constraints and the EPRM and is equivalent to the iterative baseline of Wang et al. (2024).

Table 1: Test RMSE, MAE, and MAPE on four domains (lower is better). Time-series baselines do not use news. **No-News** is the same LLM backbone without news; **IB** is the iterative method of Wang et al. (2024) (no PRM). **Bold** is best, underline is second best. We report the best refinement round for IB and Ours. MAPE on Traffic is omitted (–) because many intervals have near-zero volume, which makes percentage error unbounded.

Domain	Metric	N-BEATS	PatchTST	DLinear	LightGBM	Chronos	iTransformer	Time-LLM	No-News	IB	Ours
Electricity	RMSE	314.84	348.62	324.90	295.01	620.82	311.90	588.04	349.40	286.05	**272.07**
	MAE	252.70	286.95	263.45	231.83	511.91	251.00	490.71	253.34	228.25	**214.48**
	MAPE	8.37	9.00	8.32	7.82	15.45	8.01	15.59	8.97	6.78	**6.42**
Bitcoin	RMSE	1353.58	1333.34	1404.74	3081.67	1711.68	1302.78	1647.34	1383.91	944.62	**834.73**
	MAE	1164.20	1158.67	1221.76	2946.02	1515.70	1141.18	1483.07	1121.11	807.36	**705.16**
	MAPE	7.77	7.78	8.52	10.58	11.04	7.76	10.91	5.00	3.32	**2.45**
Traffic	RMSE	27.53	32.08	31.10	20.90	28.18	27.19	45.64	23.16	21.35	**20.42**
	MAE	16.66	21.88	20.82	12.13	17.08	16.60	35.73	14.49	11.87	**11.17**
	MAPE	–	–	–	–	–	–	–	–	–	–
Exchange	RMSE	6.19	6.14	6.57	11.18	6.10	6.18	6.82	7.25	7.54	**5.95**
	MAE	5.38	5.33	5.83	10.44	5.31	5.35	6.11	6.38	6.63	**5.16**
	MAPE	0.72	0.71	0.78	1.38	0.71	0.71	0.82	0.85	0.89	**0.69**

Table 2: Ablation on Bitcoin (lower is better). **– all (IB)** is Wang's iterative baseline with no rational constraints and no PRM. Relative change is RMSE versus Full.

Variant	RMSE	MAE	MAPE	vs. Full
Full (Ours)	**834.73**	**705.16**	**2.45**	–
– causal	1192.60	989.94	5.20	$+42.9\%$
– commonsense	1057.66	874.24	4.78	$+26.7\%$
– logic	1081.93	885.07	5.37	$+29.6\%$
– rational label	1277.13	1027.47	4.64	$+53.0\%$
– all (IB)	944.62	807.36	3.32	$+13.2\%$

Table 2 reports results on Bitcoin: news is highly informative in this domain (even No-News already beats all numerical models) and absolute forecast error is large, so the contribution of each constraint is easiest to measure. Removing any single rational constraint increases RMSE substantially, with the largest constraint-level drop from ablating the causal chain ($+42.9\%$). Dropping commonsense or logic still costs 26.7% and 29.6%, so the three constraints are complementary rather than redundant. The most damaging variant is **– rational label**: keeping the structured reflection but training the EPRM only on RMSE raises test RMSE by 53.0% to 1277.13, which is worse than IB (944.62). This shows that an RMSE-only EPRM can overfit coincidental error reductions and select news that harms generalization, whereas Wang's IB—which uses no EPRM—is more stable. Joint labels are therefore not a small add-on: they are what makes process supervision safer than unguided iterative retrieval.

4.4 Quality of Explainable Reflections (RQ3)

We assess both correctness and usefulness. DeepSeek-V4-Flash first scores 200 Bitcoin reflections as an automatic consistency check (the judge and the reflection agent use the same model). Independently, three domain experts—practitioners with experience in financial time-series and crypto-market analysis—rate a random subset of 50 of these reflections. Raters are blinded to model identity and to automatic scores. They use the same 1–5 causal/commonsense/logic rubric as the judge (Appendix G.4) and an additional usefulness item: whether the explanation would help an analyst decide if the retrieved news is worth using for the 15-day forecast. A reflection is fully correct when every correctness component scores at least 4. Expert scores are averaged across raters; protocol details are in Appendix F.

Table 3 reports both evaluations. The automatic judge is optimistic (overall 4.71; 100% fully correct), as expected for a same-model rubric check. Experts are stricter—causal chain 4.16, commonsense 4.52, logic 4.38, overall 4.35, fully-correct rate 84%—but still rate the reflections as high quality. Usefulness is 4.08: most explanations are judged actionable for retrieval, while causal chains remain the weakest component, matching the RQ2 ablation. Inter-rater agreement is substantial (Fleiss' κ=0.63). Mean expert scores correlate with the automatic judge (Spearman ρ=0.61 on

Table 3: RQ3 explanation quality on Bitcoin. Auto: DeepSeek-V4-Flash on n=200 reflections. Expert: mean of 3 blinded raters on a random subset of n=50. Fully correct means causal, commonsense, and logic all ≥ 4. Usefulness is expert-only.

Component	Auto (n=200)	Expert (n=50)
Causal chain	4.48	4.16
Commonsense justification	4.92	4.52
Logic consistency	4.74	4.38
Usefulness	–	4.08
Overall (correctness)	4.71	4.35
Fully correct (rate)	100%	84%

Table 4: Robustness against spurious news on Bitcoin. Item rate is the fraction of selected articles that are spurious; record rate is the fraction of windows that select at least one spurious article.

Method	Item rate↓	Record rate↓	RMSE (clean)	RMSE (spurious)
Iterative Baseline	40.3%	95%	944.62	1035.70 (+9.6%)
Ours (Explainable)	4.0%	16%	834.73	855.87 (+2.5%)

the 50 overlapping items), so the automatic scores are an optimistic but directionally aligned proxy rather than a substitute for expert review.

4.5 Robustness to Spurious Correlations (RQ4)

We inject spurious news articles (temporally correlated but causally irrelevant, e.g., unverified bullish claims) into the Bitcoin candidate pools. IB has no EPRM and accepts supplementary articles by keyword/category match: 40.3% of its selected items are spurious, and 95% of test windows include at least one spurious article. This raises RMSE from 944.62 to 1035.70 (+9.6%). Ours uses the same selection rule as Section 3.4: rank by the frozen EPRM, then discard candidates with $q < 0.35$ (mean judge $q \approx 0.28$ on fakes). Table 4 shows that this drops the item rate to 4.0% and the record rate to 16%. These rates come from EPRM ranking plus the q-gate, not from applying the training threshold θ=0.7 at test time. Test RMSE rises only from 834.73 to 855.87 (+2.5%), far less than IB's +9.6% degradation. Filtering coincidences can also change which genuine articles enter the set, so the clean-versus-injected comparison is not a matched checkpoint; the operational finding is that spurious item and window rates drop sharply and accuracy remains far more stable than under IB. The joint constraints therefore filter misleading coincidences that Wang's unguided loop absorbs.

4.6 Case Study

A qualitative electricity example is given in Appendix E. On 24 January 2019, a Victorian heatwave produced a large positive load error. IB only named the missing category "extreme heat weather warning." Our reflection output the chain high temperature → cooling demand → higher load, passed a commonsense and logic check, and the EPRM selected a heatwave report (score=0.92) over a temporally coincident concert article (score=0.18) that failed the q-gate.

5 Conclusion

We presented an explainable reflection framework that imposes causal, commonsense, and logic constraints on news selection and trains a PRM with a joint RMSE-and-rationality label. Offline search with ground truth yields a frozen filtering policy for online inference, reducing RMSE by 10.5% on average versus Wang et al. (2024); an RMSE-only PRM can be worse than using no PRM. Limitations include the one-time LLM-judge cost, the small RQ3 expert study (50 Bitcoin items; three raters), and domain-specific commonsense. Expanding the expert study and extending to multimodal exogenous sources are natural next steps.

AI USE STATEMENT

In this work, we used generative AI tools to implement parts of the proposed method and to support qualitative analysis: Qwen3-8B with LoRA serves as the forecasting backbone and the EPRM, and DeepSeek-V4-Flash is used as the reasoning/reflection agent and as an automated judge of causal, commonsense, and logic consistency labels, as described in Section 3. We have not used generative AI tools to generate synthetic datasets, develop theoretical models, formulate or prove mathematical claims, propose hypotheses, design the overall experimental protocol, clean or reformat datasets, or interpret the final experimental results, and the remaining required-disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for editing the manuscript to improve readability. We have reviewed all AI-assisted work. Automated judge labels were used only as training supervision and as an automatic consistency check of generated reflections; all manuscript text, claims, and experimental numbers were verified by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies news-augmented time series forecasting on publicly available benchmarks in energy, traffic, finance, and cryptocurrency. We do not collect private user data. The RQ3 expert study asked domain specialists to score already-generated reflections from a public benchmark; no personal or identifiable data were collected, and participation was voluntary. News articles used as exogenous context are treated as observational evidence for forecasting and explanation; they are not used to profile individuals. Because financial and operational forecasts can be misused, we present the method as a research prototype and do not claim it is suitable for unsupervised high-stakes decision making. Generated causal and commonsense explanations are intended to aid inspection of the retrieval loop, not to replace domain-expert judgment.

REPRODUCIBILITY STATEMENT

We provide implementation details needed to reproduce the main results in the paper and appendix. Section 3 specifies the explainable reflection format, the composite EPRM label, the selection rule (EPRM ranking then $q \geq 0.35$ gate), the two-stage offline protocol (EPRM training, then EPRM-guided refinement), and the online deployment of frozen filtering logic. Section 4 reports datasets, per-domain forecast horizons, baselines, the LLM backbone, iteration budget, and evaluation metrics. The appendix documents news-length statistics, baseline configurations, EPRM training-data construction and fine-tuning hyperparameters, the complete EPRM-guided refinement procedure (Appendix C), a complementary per-round refinement comparison (Appendix D), an electricity case study (Appendix E), the RQ3 expert-evaluation protocol (Appendix F), evaluation metrics, the Bitcoin reflection/classification/EPRM/judge/forecasting prompts, and a representative EPRM training example.Upon acceptance, we will release source code, data, configuration files, and the scripts used to produce the reported tables.

REFERENCES

Taghreed Alghamdi, Khalid Elgazzar, Magdi Bayoumi, Taysseer Sharaf, and Sumit Shah. Forecasting traffic congestion using arima modeling. In *2019 15th international wireless communications & mobile computing conference (IWCMC)*, pp. 1227–1232. IEEE, 2019.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

Yun Bai, Simon Camal, and Andrea Michiorri. News and load: A quantitative exploration of natural language processing applications for forecasting day-ahead electricity system demand. *IEEE Transactions on Power Systems*, 39(5):6222–6234, 2024.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Defu Cao, Furong Jia, Sercan O Arik, Tomas Pfister, Yixiang Zheng, Wen Ye, and Yan Liu. Tempo: Prompt-based generative pre-trained transformer for time series forecasting. *arXiv preprint arXiv:2310.04948*, 2023.

Mark Cecchini, Haldun Aytug, Gary J Koehler, and Praveen Pathak. Making words work: Using financial text as a predictor of financial events. *Decision support systems*, 50(1):164–175, 2010.

Bo-Juen Chen, Ming-Wei Chang, et al. Load forecasting using support vector machines: A study on eunite competition 2001. *IEEE transactions on power systems*, 19(4):1821–1830, 2004.

KR1442 Chowdhary. Natural language processing. *Fundamentals of artificial intelligence*, pp. 603–649, 2020.

Grzegorz Dudek. Short-term load forecasting using random forests. In *Intelligent Systems' 2014: Proceedings of the 7th IEEE International Conference Intelligent Systems IS'2014, September 24-26, 2014, Warsaw, Poland, Volume 2: Tools, Architectures, Systems, Applications*, pp. 821–828. Springer, 2015.

Robert Fildes, Shaohui Ma, and Stephan Kolassa. Retail forecasting: Research and practice. *International Journal of Forecasting*, 38(4):1283–1318, 2022.

George Gross and Francisco D Galiana. Short-term load forecasting. *Proceedings of the IEEE*, 75 (12):1558–1573, 1987.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. Large language models are zero-shot time series forecasters. *Advances in neural information processing systems*, 36:19622–19635, 2023.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

Shyh-Jier Huang and Kuang-Rong Shih. Short-term load forecasting via arma model identification including non-gaussian process considerations. *IEEE Transactions on power systems*, 18(2):673–679, 2003.

Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. Llmlingua: Compressing prompts for accelerated inference of large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 13358–13376, 2023.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. Longllmlingua: Accelerating and enhancing llms in long context scenarios via prompt compression. In *ICLR*, 2024.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*, 2023a.

Ming Jin, Qingsong Wen, Yuxuan Liang, Chaoli Zhang, Siqiao Xue, Xue Wang, James Zhang, Yi Wang, Haifeng Chen, Xiaoli Li, et al. Large models for time series and spatio-temporal data: A survey and outlook. *arXiv preprint arXiv:2310.10196*, 2023b.

Prajakta S Kalekar et al. Time series forecasting using holt-winters exponential smoothing. *Kanwal Rekhi school of information Technology*, 4329008(13):1–13, 2004.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30, 2017.

Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in neural information processing systems*, 32, 2019.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

Chang Liu, Zhijian Jin, Jie Gu, and Caiming Qiu. Short-term load forecasting using a long short-term memory network. In *2017 IEEE PES innovative smart grid technologies conference Europe (ISGT-Europe)*, pp. 1–6. IEEE, 2017.

Hanmeng Liu, Zhiyang Teng, Leyang Cui, Chaoli Zhang, Qiji Zhou, and Yue Zhang. Logicot: Logical chain-of-thought instruction-tuning, 2023a. URL https://arxiv.org/abs/2305.12147.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. *arXiv preprint arXiv:2310.06625*, 2023b.

Raha Moraffah, Paras Sheth, Mansooreh Karami, Anchit Bhattacharya, Qianru Wang, Anique Tahir, Adrienne Raglin, and Huan Liu. Causal inference for time series analysis: Problems, methods and evaluation, 2021. URL https://arxiv.org/abs/2102.05829.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

David Obst, Joseph De Vilmarest, and Yannig Goude. Adaptive methods for short-term electricity load forecasting during covid-19 lockdown in france. *IEEE transactions on power systems*, 36 (5):4754–4763, 2021.

Boris N Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-beats: Neural basis expansion analysis for interpretable time series forecasting. *arXiv preprint arXiv:1905.10437*, 2019.

Alex D Papalexopoulos and Timothy C Hesterberg. A regression-based approach to short-term system load forecasting. *IEEE Transactions on power systems*, 5(4):1535–1547, 1990.

Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Hena Ghonia, Rishika Bhagwatkar, Arian Khorasani, Mohammad Javad Darvishi Bayazi, George Adamopoulos, Roland Riachi, Nadhir Hassen, et al. Lag-llama: Towards foundation models for probabilistic time series forecasting. *arXiv preprint arXiv:2310.08278*, 2023.

Robert P Schumaker and Hsinchun Chen. Textual analysis of stock market prediction using breaking financial news: The azfin text system. *ACM Transactions on Information Systems (TOIS)*, 27(2): 1–19, 2009.

Robert P Schumaker and Hsinchun Chen. A discrete stock price prediction engine based on financial news. *Computer*, 43(1):51–56, 2010.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652, 2023.

José F Torres, Dalil Hadjout, Abderrazak Sebaa, Francisco Martínez-Álvarez, and Alicia Troncoso. Deep learning for time series forecasting: a survey. *Big data*, 9(1):3–21, 2021.

Xinlei Wang, Maike Feng, Jing Qiu, Jinjin Gu, and Junhua Zhao. From news to forecast: Integrating event analysis in llm-based time series forecasting with reflection. *Advances in Neural Information Processing Systems*, 37:58118–58153, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

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

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. *arXiv preprint arXiv:2210.02186*, 2022.

Min Xia, Haidong Shao, Xiandong Ma, and Clarence W De Silva. A stacked gru-rnn-based approach for predicting renewable energy and electricity load for smart grid operation. *IEEE Transactions on Industrial Informatics*, 17(10):7050–7059, 2021.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. Recomp: Improving retrieval-augmented lms with context compression and selective augmentation. In *ICLR*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*, 2022.

Chin-Chia Michael Yeh, Xin Dai, Huiyuan Chen, Yan Zheng, Yujie Fan, Audrey Der, Vivian Lai, Zhongfang Zhuang, Junpeng Wang, Liang Wang, et al. Toward a foundation model for time series data. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 4400–4404, 2023.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. Star: Bootstrapping reasoning with reasoning, 2022. URL https://arxiv.org/abs/2203.14465.

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

Thomas Zeng, Shuibai Zhang, Shutong Wu, Christian Classen, Daewon Chae, Ethan Ewer, Minjae Lee, Heeju Kim, Wonjun Kang, Jackson Kunde, et al. Versaprm: Multi-domain process reward model via synthetic reasoning data. *arXiv preprint arXiv:2502.06737*, 2025.

Lingyin Zhang, Jun Gao, Xiaoxue Ren, and Ziqiang Cao. The bidirectional process reward model. *arXiv preprint arXiv:2508.01682*, 2025.

Congming Zheng et al. A survey of process reward models: From outcome signals to process supervisions for large language models. *arXiv preprint arXiv:2510.08049*, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. URL https://arxiv.org/abs/2306.05685.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11106–11115, 2021.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International conference on machine learning*, pp. 27268–27286. PMLR, 2022.

A Experimental Details

Forecast protocol. Horizons match Section 4.1: electricity $H{=}48$ half-hourly steps, bitcoin $H{=}15$ daily steps, traffic $H{=}24$ hourly steps, and AUD/USD $H{=}7$ daily steps, with lookback equal to H. Bitcoin prompt templates are given in Appendix G; other domains use the same JSON schema with domain-specific task wording and the corresponding H.

News length. Table 5 reports the volume of news associated with each forecasting task. On average, a task is paired with 27–38 articles and roughly 101k–144k characters. In the worst case, a single task can exceed 500k characters (Electricity, Traffic, Exchange). These lengths exceed typical LLM context windows, so naïvely concatenating all retrieved articles is infeasible. In this paper we therefore truncate each selected article to a fixed character budget ($1{,}500$ characters) so that the evaluation isolates explainable selection rather than compression.

Table 5: News length statistics per forecasting task.

	Electricity	Bitcoin	Traffic	Exchange
Avg. # of articles per task	36.4	36.5	26.9	37.6
Avg. total chars per task	129,975	144,016	100,553	111,593
Max total chars per task	601,319	293,829	571,076	561,505

Baselines. Numerical baselines include N-BEATS (Oreshkin et al., 2019), PatchTST (Nie et al., 2022), DLinear (Zeng et al., 2023), LightGBM (Ke et al., 2017), Chronos (Ansari et al., 2024), iTransformer (Liu et al., 2023b), and Time-LLM (Jin et al., 2023a). They are trained on the same forecast windows as the LLM methods and do not consume news. For Chronos we use `amazon/chronos-bolt-mini`. Implementations of iTransformer, Time-LLM, and DLinear follow the neuralforecast library.[1] News-augmented LLM methods share a Qwen3-8B LoRA backbone: **No-News** uses historical series only; **IB** follows the iterative reflection protocol of Wang et al. (2024) without an EPRM; **Ours** adds three-component explainable reflections and EPRM scoring with the joint label. All numerical baselines follow their original architectural specifications.

EPRM fine-tuning. The EPRM is a LoRA-tuned Qwen3-8B classifier (Yang et al., 2025; Hu et al., 2022). It is trained *before* Algorithm 1 and then frozen. Training labels are constructed from historical training windows as detailed in Appendix B: a candidate is positive only when it both reduces RMSE under a frozen forecasting LoRA ($\Delta^{(k)} > 0$) and receives a rational quality score $q > 0.7$ (1–5 scores averaged over causal validity, commonsense alignment, and logic consistency, then mapped to $[0, 1]$ by $q = (\bar{s} - 1)/4$). Rational quality is assigned by DeepSeek-V4-Flash (Xu et al., 2026). The model is trained with supervised instruction tuning to answer `YES` or `NO`; during refinement we rank by $P(\texttt{YES})$ and then apply a rational-quality gate $q \geq 0.35$.

LoRA adapters are inserted into all attention projections (W_Q, W_K, W_V, W_O) and MLP layers (gate, up, down), with rank $r{=}16$ and $\alpha{=}32$. Training uses bfloat16, AdamW with cosine decay, peak learning rate $2{\times}10^{-4}$, warmup ratio 0.05, cutoff length $2{,}048$ tokens, and 3 epochs. We evaluate every 50 steps and keep the checkpoint with the lowest validation loss. Training is run on four NVIDIA RTX 5880 Ada GPUs (48GB each).

Table 1 checkpoints. We report the best refinement round for IB and Ours in Table 1. Per-round traces in Appendix D show that Ours typically plateaus by round 2–3, whereas IB continues to change through round 4–5.

Evaluation. We report RMSE, MAE, and MAPE. For a test set of N points,

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}, \quad \mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|, \quad \mathrm{MAPE} = \frac{100}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right|.$$

[1] `https://github.com/Nixtla/neuralforecast`

MAPE is omitted for Traffic: many hourly volumes are near zero, so the percentage error is unbounded.

B EPRM Training Data Construction

This appendix specifies *Stage A*: how the EPRM P_ϕ is trained. Stage A is completed before Stage B (Algorithm 1; full procedure in Appendix C). We do not use human rational-quality labels, and we never construct labels from test windows.

Scope. Labels are collected on historical *training* windows for which ground-truth targets are available offline. A disjoint validation split of the same training period is used only to produce the error pattern that the evaluation agent reflects on. After P_ϕ is trained it is frozen; Algorithm 2 uses it as a ranker, applies the $q \geq 0.35$ gate, and does not recompute labeling gains $\Delta^{(k)}$.

Current news set. For each training window we start from a seed news set $\mathcal{N}$ (round 0: the iterative baseline selection of Wang et al. (2024); later bootstrap rounds: the EPRM-selected set from the previous round). A forecasting LoRA F_ψ has already been fine-tuned on $\mathcal{N}$ and is then *frozen* for the labeling pass.

Reflection and candidates. DeepSeek-V4-Flash produces an explainable reflection r_t (causal chain, commonsense justification, logic consistency) given validation errors and the current news. The reasoning agent then retrieves $K{=}5$ supplementary candidates that are not already in $\mathcal{N}$. Each candidate is also annotated with its own causal/commonsense/logic fields (Appendix G.2).

Labeling gain $\Delta^{(k)}$. For each candidate c, we keep F_ψ frozen and run two forward passes: one with $\mathcal{N}$ and one with $\mathcal{N} \cup \{c\}$, both after the same character truncation. Let $\Delta^{(k)} = \mathrm{RMSE}(\mathcal{N}) - \mathrm{RMSE}(\mathcal{N} \cup \{c\})$. A positive $\Delta^{(k)}$ means that inserting c reduces error under the *current* backbone. We do not retrain LoRA for every candidate; that would be the refinement gain Δ_{t+1} of Algorithm 2, which is computed later and only for the single EPRM-selected article.

Rational quality q. The same DeepSeek-V4-Flash model scores the candidate's causal, commonsense, and logic fields on a 1–5 Likert scale. We map the mean score $\bar{s}$ to $[0, 1]$ by $q = (\bar{s} - 1)/4$. The *training* threshold $\theta = 0.7$ therefore corresponds to an average rating of 3.8. At *selection* time we instead use the looser gate $\theta_{\mathrm{gate}}{=}\theta/2{=}0.35$.

Joint label and SFT format. The binary target is $y = \mathbf{1}[\Delta^{(k)} > 0] \wedge \mathbf{1}[q > 0.7]$. Each training example concatenates the task description, r_t, prior reflections, recent labeling-Δ history, and the candidate (title, summary, and the three rational fields). The completion is a single token `YES` or `NO`. Candidates that lower RMSE but fail $q > 0.7$ are labeled `NO`; this is the training signal that teaches the EPRM to down-rank spurious news. At selection time (including RQ4) we additionally rank by $P(\texttt{YES})$ and discard $q < 0.35$. The resulting JSON records are used to LoRA-tune Qwen3-8B as described above. Optional bootstrap: after a new EPRM is trained, we may re-select news, fine-tune the next forecasting LoRA, and collect a further round of labeling trajectories. When Stage A stops, P_ϕ is frozen and handed to Algorithm 1.

C Detailed EPRM-Guided Refinement Procedure

Algorithm 1 in the main text is a compact view of Offline Stage B. Algorithm 2 expands every step used in our implementation: seed construction and the initial LoRA fit; per-candidate scoring with the frozen EPRM; the rational-quality gate and empty-set fallback; truncation; LoRA retrain; and the conservative update of the persistent filtering logic $\mathcal{L}$. Stage A (Appendix B) is completed first; P_ϕ is held fixed throughout, and labeling gains $\Delta^{(k)}$ are *not* recomputed inside this loop.

Selection versus logic update. The working news set $\mathcal{N}$ always receives the EPRM-selected article n_t^*. The deployable object $\mathcal{L}$ is updated only when the post-retrain refinement gain Δ_{t+1} is strictly positive. Scoring uses the frozen EPRM $s(c) = P_\phi(\texttt{YES} \mid h_t, \tau_{<t}, c)$ and does not retrain

Algorithm 2 Offline EPRM-Guided Refinement (full procedure)

Require: Historical training set $\mathcal{X}$, validation set $\mathcal{X}_{\text{val}}$, news pool $\mathcal{U}$
Require: Reasoning agent RA; forecasting backbone F_ψ; *pretrained* EPRM P_ϕ (Appendix B)
Require: Improvement threshold ε; iteration budget $T_{\max}$; gate θ_{gate}=0.35
Ensure: Persistent filtering logic $\mathcal{L}^*$; fine-tuned forecaster F_{ψ^*}

$\mathcal{N}^{(0)} \leftarrow \text{RA.initial_filter}(\mathcal{U})$; $\mathcal{L}^{(0)} \leftarrow \text{RA.initial_logic}()$; $t \leftarrow 0$
$S^{(0)} \leftarrow \text{truncate}(\mathcal{N}^{(0)})$
$F_{\psi^{(0)}} \leftarrow \text{LoRA}(F_\psi, \mathcal{X}, S^{(0)})$
$\text{RMSE}_0 \leftarrow \text{validate}(F_{\psi^{(0)}}, \mathcal{X}_{\text{val}})$
repeat
  $r_t \leftarrow \text{reflect}(F_{\psi^{(t)}}, \mathcal{X}_{\text{val}})$ // causal, commonsense, logic
  $\mathcal{C}_t \leftarrow \text{RA.retrieve}(r_t)$ // K candidates
  for each $c \in \mathcal{C}_t$ **do**
    $s(c) \leftarrow P_\phi(\texttt{YES} \mid h_t, \tau_{<t}, c)$ // frozen EPRM; no LoRA retrain
    $q(c) \leftarrow$ rational quality of c
  end for
  $\mathcal{C}'_t \leftarrow \{c \in \mathcal{C}_t : q(c) \geq \theta_{\text{gate}}\}$; **if** $\mathcal{C}'_t = \emptyset$ **then** $\mathcal{C}'_t \leftarrow \mathcal{C}_t$
  $n^*_t \leftarrow \arg\max_{c \in \mathcal{C}'_t} s(c)$ // EPRM rank after q-gate
  $\mathcal{N}^{(t+1)} \leftarrow \mathcal{N}^{(t)} \cup \{n^*_t\}$
  $S^{(t+1)} \leftarrow \text{truncate}(\mathcal{N}^{(t+1)})$
  $F_{\psi^{(t+1)}} \leftarrow \text{LoRA}(F_\psi, \mathcal{X}, S^{(t+1)})$ // retrain after merge
  $\text{RMSE}_{t+1} \leftarrow \text{validate}(F_{\psi^{(t+1)}}, \mathcal{X}_{\text{val}})$
  $\Delta_{t+1} \leftarrow \text{RMSE}_t - \text{RMSE}_{t+1}$ // refinement gain
  if $\Delta_{t+1} > 0$ **then**
    $\mathcal{L}^{(t+1)} \leftarrow \text{update}(\mathcal{L}^{(t)}, n^*_t)$ // persistent logic only
  else
    $\mathcal{L}^{(t+1)} \leftarrow \mathcal{L}^{(t)}$
  end if
  $t \leftarrow t + 1$
until $t \geq T_{\max}$ **or** $\Delta_t < \varepsilon$
$\mathcal{L}^* \leftarrow \mathcal{L}^{(t)}$; $F_{\psi^*} \leftarrow F_{\psi^{(t)}}$
return $\mathcal{L}^*$, F_{ψ^*}

LoRA for every candidate. The rational-quality gate discards $q < \theta_{\text{gate}}$=0.35; if that filter empties the pool, ranking falls back to the ungated set $\mathcal{C}_t$. The loop stops at $T_{\max}$ or when $\Delta_t < \varepsilon$.

D Refinement Convergence

RQ1 in the main text reports the best-round forecasting accuracy. Here we compare the *per-round* test RMSE of IB (Wang et al., 2024) and our method under the same seed news set, Qwen3-8B LoRA backbone, truncation budget, and $T_{\max} = 5$. At each round, IB emits a category-only reflection and accepts the retrieved article without EPRM scoring, whereas Ours generates a three-component explainable reflection and adds the highest-scoring EPRM candidate that also passes the $q \geq 0.35$ gate.

Table 6 shows that Ours drops most of its error in the first two rounds and is essentially stable by round 2 (electricity, bitcoin, exchange) or round 3 (traffic). IB keeps improving through round 4–5 and only then approaches a plateau, at a higher RMSE. The three constraints and the joint EPRM label therefore not only improve the final forecast but also shorten the offline search.

Table 6: Test RMSE after each offline refinement round (lower is better). Bold is the first round at which RMSE has essentially plateaued. IB is the iterative method of Wang et al. (2024); Ours uses explainable reflections and PRM-guided selection.

Domain	Method	Round 1	Round 2	Round 3	Round 4	Round 5
Electricity	IB	318.40	302.15	293.80	**286.90**	286.05
	Ours	278.20	**272.40**	272.07	272.11	272.15
Bitcoin	IB	1086.40	1012.30	978.50	952.20	**944.62**
	Ours	872.10	**836.40**	834.73	834.80	835.05
Traffic	IB	22.80	22.15	21.72	21.48	**21.35**
	Ours	20.95	20.55	**20.42**	20.43	20.44
Exchange	IB	8.42	8.05	7.78	**7.56**	7.54
	Ours	6.48	**6.02**	5.95	5.96	5.97

E Case Study: Explainable Reflection in Action

We illustrate the behavior of our explainable reflection on a concrete forecasting error from the electricity dataset. On 24 January 2019 (Australian summer), the actual electricity load in Victoria, Australia, rose sharply, but the baseline forecasting model (without the missing news) predicted a much smaller increase, resulting in a large positive error.

Baseline reflection. The iterative baseline (Wang et al., 2024) generated a reflection: "Missing news category: extreme heat weather warning." No further explanation was provided.

Ours: explainable reflection. Our agent produced the following structured reflection:

> ***Causal chain:*** High temperature (↑) → increased use of air conditioning / cooling equipment (↑) → electricity load (↑).
> ***Commonsense justification:*** In summer, higher temperatures consistently increase cooling demand; the opposite would be physically implausible.
> ***Logic consistency:*** No previous reflection has invoked temperature; the current error pattern aligns with a heatwave event, and there is no contradictory evidence.

Candidate news and EPRM scoring. The reasoning agent retrieved K=5 candidates matching the "heat" category. We highlight two: Article A, "Victoria heatwave: temperatures soar to 42°C,

authorities urge energy conservation" (The Age, 24 January 2019), and Article B, "Pop star performs in Melbourne, thousands attend concert" (Herald Sun, 24 January 2019). The EPRM assigned scores: score$(A) = 0.92$, score$(B) = 0.18$. After the q-gate ($q \geq 0.35$), Article A remains and ranks first, so it is selected. Article B is dropped: its EPRM score is already low, and its rational quality fails the gate (missing causal chain, commonsense violation) despite temporal coincidence.

Outcome. After adding Article A to the news set, truncating the updated context, and fine-tuning the forecasting model, validation error on the window decreased. The refined filtering logic also learned to prioritize temperature-related keywords in later iterations. This case illustrates how the three constraints and the joint EPRM label select genuinely useful news while rejecting spurious coincidences.

F Expert Evaluation Protocol (RQ3)

This appendix records the expert study used in Section 4.4.

Raters and items. Three specialists with experience in financial time-series analysis and crypto-market news independently scored a uniformly random subset of 50 Bitcoin reflections drawn from the 200 items scored by the automatic judge. Each rater saw the reflection JSON (missing category, causal chain, commonsense, logic consistency, search keywords) together with the forecast window metadata (prediction date and currently selected titles). They did not see automatic-judge scores, EPRM scores, RMSE, or the method name.

Rubric. Causal validity, commonsense alignment, and logic consistency use the same 1–5 integer rubric as Appendix G.4. Usefulness is a separate 1–5 item: "Would this explanation help an analyst decide whether the retrieved news is worth using for a 15-day BTC forecast?" (1 = no, empty or misleading; 3 = partially helpful; 5 = specific enough to accept or reject a retrieval decision). Raters were instructed not to reward fluent but vacuous text.

Aggregation. Per-item expert scores are the mean of the three raters. Fully correct means all three correctness dimensions are ≥ 4 after averaging (equivalently, the mean of rater-level fully-correct indicators is reported as 84% in the main text). Inter-rater agreement is Fleiss' κ over the four 1–5 items. Spearman ρ is computed between the automatic overall score and the expert overall correctness score on the 50 overlapping items.

G Prompt Templates

We show the Bitcoin templates used in experiments (H=15 daily steps, lookback 15 days). Other domains copy the same JSON schema and the same three rational fields; only the task wording, horizon, and domain-specific YES/NO rules change (electricity H=48 half-hourly, traffic H=24 hourly, AUD/USD H=7 daily). DeepSeek-V4-Flash generates reflections, candidate rationales, retrieval queries, and rational-quality judgments. Qwen3-8B (LoRA) is used for forecasting and for EPRM scoring. Placeholders in braces are filled at runtime. For readability we spell out field-level rules that the implementation encodes as short system strings; the JSON keys, user-side fields, and YES/NO criteria are unchanged.

G.1 Explainable Reflection

After a validation pass, the evaluation agent must output a structured reflection with a causal chain, a commonsense justification, and a logic-consistency check. This prompt forces a verifiable argument that later drives keyword retrieval and EPRM scoring. The system prompt fixes the schema and field rules; the user prompt supplies the window-specific context (currently selected titles with effect types, and the last three accepted reflections).

Explainable Reflection Prompt (Bitcoin)

System.
You are an evaluation agent for news-augmented Bitcoin daily price forecasting. The task is a 15-day-ahead BTC/USD forecast with a 15-day lookback. Selected articles are truncated to 1,500 characters before they enter the forecaster.
Given the currently selected news and previously accepted reflections, diagnose what evidence is still missing and produce an EXPLAINABLE reflection. This reflection will be used (i) to retrieve supplementary articles via keywords and (ii) as context for EPRM scoring. You are not forecasting prices and you must not invent article titles.
You MUST fill every field below.

1. `missing_category`: a short phrase naming the missing news type (e.g., "SEC ETF delay", "exchange hack", "Fed hike"). Not a paragraph. Prefer a category that is *not* already covered by the current selected news.
2. `causal_chain`: a directed chain of the form `cause -> mediator -> effect` that links the missing news to a price-relevant quantity over the 15-day horizon (demand, liquidity, risk premium, or price). A two-hop slogan such as "bad news -¿ price down" is invalid; at least one mediator is required. Example: "SEC delay of a spot ETF -¿ delayed institutional inflows -¿ weaker near-term BTC demand -¿ downward pressure on price".
3. `commonsense`: 1–3 sentences explaining why the chain is consistent with standard crypto-market knowledge, and why the opposite would be implausible. Explicitly reject entertainment, celebrity, or sports headlines as primary 15-day drivers. Example: "ETF approvals and major-economy regulation routinely move institutional demand; celebrity gossip typically does not."
4. `logic_consistency`: compare the new chain with the last three accepted reflections and with current selected-news coverage. Report any contradiction (e.g., claiming both that the same driver raises and lowers price under the same conditions). If none, state that explicitly.
5. `search_keywords`: 3–8 concrete retrieval terms derived from the causal chain (agencies, venues, instruments, event types). Forbidden: generic tokens such as "news", "bitcoin", or "price".

Hard constraints: respond with a single JSON object only; no markdown fences, no preamble, no trailing commentary. Do not cite sources that are not in the user message.
Respond ONLY in JSON:

```
{
  "missing_category": "...",
  "causal_chain": "A -> B -> C",
  "commonsense": "...",
  "logic_consistency": "...",
  "search_keywords": ["...", "..."]
}
```

User.
`Market: BTC`
`Predict date: {pred_date}`
`Current selected news: [{"title": "...", "effect_type": "long_term / short_term / real_time"}, ...]`
`Previous accepted reflections: [{last 3 reflection JSONs; empty list if none}]`
Produce an explainable reflection for missing news evidence.

Iterative-baseline contrast. IB uses the same user fields but a category-only system prompt: "Do NOT explain causal chains. Only name the missing category," with JSON keys `missing_category` and `search_keywords`. No causal, commonsense, or logic fields are requested.

G.2 Candidate Classification (rational fields)

Each retrieved candidate is annotated before EPRM scoring. The classifier decides relevance, assigns an effect horizon, writes a factual summary, and produces the three rational fields that later enter both the joint label and the EPRM input. Articles are truncated to 1,500 characters in the user message. A candidate with `relevant=NO` is discarded and never scored by the EPRM.

Candidate Classification Prompt (Bitcoin)

System.
You judge whether a news article is useful for Bitcoin daily price forecasting (15-day horizon), AND you must produce explainable rationales for that decision.
Mark `relevant=YES` only if the article can plausibly move BTC over the horizon via at least one of: regulation; exchange flows or hacks; ETF / institutional adoption; macro rates, inflation, or liquidity; mining / hashrate / energy cost of mining; or market-structure news. Mark `NO` for entertainment, celebrity, sports recaps, or generic macro color with no crypto-market pathway. Temporal coincidence with a price move is not sufficient.
Field rules:

- `effect_type`: `long_term` = structural / multi-week driver of future price; `short_term` = near-term sentiment or flow over days; `real_time` = same-day or next-print impact; `none` if `relevant=NO`.
- `affected_region`: the market geography that would be affected (`global`, `US`, `EU`, `Asia`, ...), not the newsroom location.
- `summary`: one or two factual sentences; no price prediction and no hype.
- `causal_chain`, `commonsense`, `logic_consistency`: same three-component format as the reflection prompt, but now grounded in *this* article. The chain must use the article as the cause, not a generic category. Logic must be checked against the provided reflection (if any).
- `rational_scores`: integers 1–5 (1 = invalid / contradictory, 5 = fully correct and specific). Score the candidate's own three fields, not the reflection.

Respond ONLY in JSON:

```
{
  "relevant": "YES" or "NO",
  "effect_type": "long_term" | "short_term" | "real_time" | "none",
  "affected_region": "global|US|EU|Asia|...",
  "summary": "factual summary",
  "causal_chain": "cause -> mediator -> effect",
  "commonsense": "domain-knowledge justification",
  "logic_consistency": "consistency note vs prior reflections",
  "rational_scores": {
    "causal_validity": 1,
    "commonsense_alignment": 1,
    "logic_consistency": 1
  }
}
```

User.
`Forecast region: BTC`
`History date: {hist_date}`
`Prediction date: {pred_date}`
`Explainable reflection (if any): {reflection JSON, or {} for seed classification}`
`News title: {title}`
`Publication time: {publication_time}`
`Article: {full_article truncated to 1,500 characters}`

G.3 EPRM SCORING

The frozen EPRM (LoRA-tuned Qwen3-8B) scores a candidate given the current reflection, previous reflections, labeling-Δ history, and the candidate's own causal/commonsense/logic fields. At training time the target is `YES` iff $\Delta^{(k)} > 0$ and $q > 0.7$; at scoring time we read $P(\texttt{YES})$ from the last-token softmax over $\{\texttt{YES}, \texttt{NO}\}$ and then apply the $q \geq 0.35$ gate. The prompt never reveals the ground-truth $\Delta^{(k)}$ or q. Supervision teaches `YES` for articles whose chain is a genuine price pathway and that historically reduced RMSE under a frozen forecaster, and `NO` for entertainment, rumor, empty/circular rationales, or temporally coincident but causally irrelevant news.

EPRM Scoring Prompt (Bitcoin)

Instruction (single user message).

```
Task: score whether adding this news is beneficial for bitcoin
price forecasting.
Region: BTC
Predict date: {pred_date}
Reflection: {current reflection JSON}
Previous reflections: {last 3 reflection JSONs}
Trajectory history (prior ΔRMSE): {last 3 labeling gains as JSON;
[] if none}
Candidate title: {title}
Candidate summary: {summary}
Causal chain: {causal_chain}
Commonsense: {commonsense}
Logic: {logic_consistency}
Answer with a single token: YES or NO.
```

G.4 Automatic Judge of Rational Quality

The same three-way 1–5 rubric is used (i) to construct EPRM training labels on candidate rationales and (ii) to score generated reflections in the automatic arm of RQ3. Expert raters in Appendix F use this rubric for the three correctness items and add a separate usefulness item. Scores are integers. We map the mean $\bar{s}$ to $[0, 1]$ by $q = (\bar{s} - 1)/4$; the EPRM-positive *training* threshold $\theta = 0.7$ corresponds to $\bar{s} = 3.8$, while selection uses θ_{gate}=0.35.

Rubric (shared by both judge variants).

- **Causal validity.** 1: missing, reversed, or purely correlational ("news happened, price moved"). 2: cause named but no mediator, or mediator is tautological. 3: a directed chain is present but a hop is weak or domain-mismatched. 4: cause → mediator → effect is complete and correctly signed for the 15-day horizon. 5: complete, correctly signed, and specific to the article/reflection (named entities, not a generic template).
- **Commonsense alignment.** 1: contradicts basic market knowledge (e.g., a celebrity tweet as the main 15-day BTC driver). 2: no domain check, or the check is vacuous. 3: a plausible justification with a minor gap. 4: consistent with standard crypto drivers (regulation, flows, macro, mining, market structure) and states why the opposite is implausible. 5: same as 4, and explicitly rules out a plausible confound.
- **Logic consistency.** 1: contradicts a previous accepted reflection or the current error/news coverage. 2: ignores previous reflections. 3: no contradiction, but the self-check is missing or generic. 4: explicit comparison with prior reflections and current coverage, no conflict. 5: same as 4, and notes complementary (non-redundant) coverage.

Reflection Judge Prompt (RQ3, Bitcoin)

System.

You are an automated judge scoring an explainable forecasting reflection for Bitcoin daily price forecasting (15-day horizon). Score the three components with the rubric above, from 1 (invalid) to 5 (excellent). A score of 5 means the component is fully correct and useful for retrieving news. Do not reward fluent but empty text. Do not use the reflection's own `rational_scores` if present; judge from the text.
Respond ONLY in JSON:

```
{
  "causal_validity": 1,
  "commonsense_alignment": 1,
  "logic_consistency": 1,
  "all_three_fully_correct": false,
  "brief_reason": "one or two sentences citing the weakest hop"
}
```

all_three_fully_correct is true only if every score is ≥ 4.

User.

```
Bitcoin 15-day price forecast reflection for {window_id}.
missing_category: {...}
causal_chain: {...}
commonsense: {...}
logic_consistency: {...}
search_keywords: {...}
```

Candidate Rational-Quality Judge (EPRM labels, Bitcoin)

System.
You are an automated judge scoring the rational quality of a news-selection rationale for Bitcoin daily price forecasting. Score three dimensions from 1 (invalid) to 5 (excellent) using the same rubric. Respond ONLY in JSON with keys causal_validity, commonsense_alignment, logic_consistency, and brief_reason.

User.

```
Task: bitcoin price forecast for BTC on {pred_date}.
Reflection: {reflection JSON}
Candidate news: {title} | {summary}
Proposed causal_chain: {causal_chain}
Commonsense: {commonsense}
Logic: {logic_consistency}
```

G.5 Forecasting Prompt

The forecasting backbone (Qwen3-8B with LoRA) receives a 15-day lookback, a 15-day horizon, and truncated news concatenated in Wang-style effect tags. Numerical covariates (hashrate, tweets, etc.) are *not* included. If no article is selected, a single fallback sentence is appended: "There is no suitable news for long-term effect on future bitcoin price." The model must emit a comma-separated numeric sequence of length $H=15$.

Forecasting Prompt Template (Bitcoin)

Instruction.
The historical price data is: {p_1, p_2, ..., p_15} (one decimal place, comma-separated).

Input.
Based on the historical price data, please predict the bitcoin price in the next 15 days. The market for prediction is BTC. The start date of historical data was on {YYYY-M-D that is Weekday}. The data frequency is 1 day per point. Historical data covers 15 days. The prediction window is {start weekday date} to {end weekday date}.

Then concatenate every selected article (each body truncated to 1,500 characters) with no extra separators:

```
In {publication_time}, the news that had the {effect_label} is that {summary}.
The region that may be influenced is {region}. The rationality is that {rationality}
```

where effect_label is one of:
Long-Term Effect on Future Bitcoin Price / Short-Term Effect on Near-Term Bitcoin Price / Real-Time Direct Effect on Today's Bitcoin Price,
region defaults to global, and rationality is the formatted triple "Causal chain: ... | Commonsense: ... | Logic consistency: ...". If no long-term article is present, prepend the fallback sentence above.

Output constraint.
Output ONLY 15 real-valued USD prices as a single comma-separated line, with no extra spaces, units, or commentary. Example: 8721.4,8690.1,...,9104.0.

H EPRM TRAINING EXAMPLE

Each training sample concatenates the current reflection, prior trajectory, and a candidate article. The completion is a single token. The target is `YES` only under the joint label $y = \mathbf{1}[\Delta^{(k)} > 0] \wedge \mathbf{1}[q > 0.7]$. A shortened Bitcoin example (positive):

> `[Input]` Task: score whether adding this news is beneficial for bitcoin price forecasting.
> Region: BTC Predict date: 2018-01-28
> Reflection: missing category = regulatory / institutional-adoption news; causal chain = regulatory uncertainty $\rightarrow$ lower trading volume $\rightarrow$ short-term downward pressure; commonsense = SEC/exchange actions routinely move BTC demand; logic = no contradiction with prior mining-focused reflections.
> Previous reflections: [prior round: missing mining-difficulty news]
> Trajectory history (prior ΔRMSE): [{"delta": 12.4}]
> Candidate title: Goldman Sachs hires crypto trader to lead digital assets.
> Candidate summary: A bulge-bracket bank staffs a digital-asset desk, signalling institutional distribution capacity.
> Causal chain: institutional hiring $\rightarrow$ perceived legitimacy and brokerage inflows $\rightarrow$ upward pressure on Bitcoin.
> Commonsense: major-bank involvement is a standard adoption signal, unlike celebrity endorsements.
> Logic: complements mining-focused selected news; no contradiction with prior reflections.
> `[Target]` YES (because $\Delta^{(k)} > 0$ and $q > 0.7$)

A candidate that is temporally correlated but fails the rational-quality training threshold is labeled `NO` even if it happens to reduce validation RMSE. Example: an entertainment headline during a sell-off may yield $\Delta^{(k)} > 0$ by chance, but $q \leq 0.7$ because the causal chain has no market-structure mediator; the joint label is therefore `NO`. At selection time the same article is further subject to EPRM ranking and the $q \geq 0.35$ gate (Section 4.5).