RETHINKING RAG FOR TIME SERIES: DOES SIMILARITY REALLY LEAD TO PREDICTIVE UTILITY?

Anonymous authors
Paper under double-blind review

ABSTRACT

Historical patterns beyond the input context can provide useful information for time-series forecasting. Retrieval-augmented forecasting seeks to exploit such information by retrieving historical segment candidates whose contexts are similar to the query context. This raises a fundamental question: does similarity reliably indicate the *predictive utility* of a historical candidate? We examine it from three perspectives: (1) how prevalent useful historical candidates are, (2) whether similarity-based retrieval can identify them, and (3) whether their predictive utility remains consistent across augmentation interfaces. Across 24 experimental settings, our empirical results reveal several important findings. Useful candidates are common: 38.32–56.23% of candidates improve the forecast, with the best candidates yielding gains of 24.56–71.01%. Surprisingly, similarity-based retrieval underperforms random sampling in 83.33% of retrieval comparisons. Predictive utility is also strongly interface-dependent: in 55.56% of cross-interface transfers, historical candidates selected as optimal under the source interface yield negative gains under the target interface. Our findings suggest that similarity alone is insufficient for retrieval, and future retrieval methods should better account for which historical candidates are actually useful under the chosen augmentation interface.

1 INTRODUCTION

Time-series forecasting aims to predict future values from past observations, and recent time-series foundation models (TSFMs) enable zero-shot forecasting through large-scale pretraining on diverse time series (Woo et al., 2024; Ansari et al., 2025). At inference time, these models typically receive a finite input context, while the available historical record may extend far beyond it. Consequently, potentially useful historical patterns outside this context are not directly available to the model for the current forecast. Retrieval-augmented generation (RAG) for forecasting provides a way to access this broader history by retrieving historical candidates from an external historical collection and incorporating them into the forecasting process (Yang et al., 2024; Han et al., 2025; Tire et al., 2026; Ning et al., 2025; Lee et al., 2026). A central design choice in this framework is how to select historical candidates. Most existing methods rely, explicitly or implicitly, on the similarity between the candidate context and the query context to estimate relevance for the current forecast. This choice is intuitive, but it leaves a pivotal question unanswered: whether retrieval similarity is actually aligned with a candidate's predictive utility.

Similarity and predictive utility describe different properties of a historical candidate respectively. Similarity can be computed from the query context and candidate context before the query future is observed. Predictive utility instead measures how much the candidate actually improves forecasting performance. Therefore a highly similar candidate may provide little benefit or even harm to forecasting, as illustrated in Figure 1. This mismatch is especially plausible in highly-fluctuating time series, where similar contexts can be followed by different continuations (Zhou et al., 2026; Lee et al., 2026). Thus, from the perspective of predictive utility, similarity may be inappropriate for selecting candidates.

With RGA based forecasting, the retrieved candidate must be incorporated into the forecasting model. And existing methods use different augmentation interfaces for this purpose (Han et al., 2025; Ning et al., 2025; Lee et al., 2026). For example, historical candidates can be concatenated with the input, provided through additional channels, or integrated through separate modules. These

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.

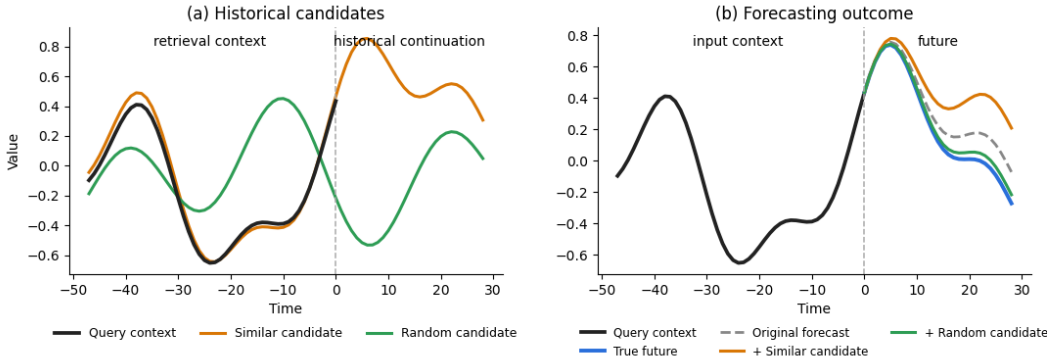


Figure 1: Illustration of the mismatch between retrieval similarity and predictive utility.

interfaces determine how the candidate interacts with the query context and the forecasting model. Predictive utility may therefore change across augmentation interfaces, and the same historical candidate can be beneficial under one interface but harmful under another.

In this work, we measure the predictive utility of a historical candidate with the reduction of forecasting error obtained by incorporating it into a given forecast. This utility is evaluated for a specific query with a forecasting model and an augmentation interface, and a certain error metric. This formulation allows us to study three research questions. **RQ1: Availability.** How common are useful historical candidates, and how much can the best candidates improve forecasting? **RQ2: Retrievability.** If useful candidates exist, can similarity-based retrieval prioritize them more effectively than random sampling? **RQ3: Interface dependence.** If a candidate is useful under one augmentation interface, does it remain useful for others?

Our empirical study reveals several important findings. First, useful historical candidates are widely available. Across 24 dataset–system settings, 38.32–56.23% of candidates improve forecasting, while Oracle@1000 yields gains of 24.56–71.01%. Second, similarity does not reliably identify these candidates. Across 6 similarity-based retrievers and 24 settings, similarity-based retrieval fails to outperform matched random sampling in 120 of 144 comparisons, and similarity-based rankings show little agreement with predictive utility. Third, predictive utility does not remain stable across augmentation interfaces. Utility rankings are weakly aligned across interfaces, and 20 of 36 cross-interface transfers produce negative aggregate gains even when the transferred candidates are optimal under the source interface. These findings expose two distinct limitations of similarity-based retrieval augmented for time series: *useful candidates can be difficult to identify by similarity*, and *their utility can change with different augmentation interface.*

Our contributions are threefold:

- **We introduce a utility-based framework for analyzing retrieval-augmented time series forecasting.** We formalize candidate-level predictive utility and use it to distinguish three questions that are otherwise conflated in end-to-end evaluation: whether useful candidates exist, whether similarity can retrieve them, and whether their utility transfers across augmentation interfaces.
- **We uncover a substantial gap between candidate availability and similarity-based retrievability.** Although useful candidates are common and can provide large oracle gains, similarity-based retrieval fails to outperform matched random sampling in 83.33% of comparisons, with weak agreement between similarity rankings and predictive utility. This shows that the major challenge is how to retrieve the useful history data.
- **We show that predictive utility is specific to the augmentation interface.** Under controlled cross-interface evaluation, we keep the forecasting model, queries, and candidate pools fixed and change only the augmentation interface. Utility rankings remain weakly aligned, and 55.56% of source-oracle transfers yield negative aggregate gains for the target interface. This demonstrates that each RAG interface needs special retrieval design.

2 Related Work

2.1 Zero-shot forecasting with frozen foundation models

Time-series foundation models (TSFMs) provide strong zero-shot forecasting capabilities across heterogeneous domains. Chronos-2 supports univariate, multivariate, and covariate-informed forecasting (Ansari et al., 2025), while Moirai learns a universal forecaster from heterogeneous time series (Woo et al., 2024). Chronos-Bolt uses patched inputs and direct multi-step quantile prediction (Ansari et al., 2024) and serves as the original forecasting backbone of TS-RAG (Ning et al., 2025).

2.2 Retrieval-augmented forecasting

History selection. Existing retrieval-augmented forecasters define historical relevance in several ways. RAF (Tire et al., 2026) and TS-RAG (Ning et al., 2025) retrieve in pretrained representation spaces , whereas RAEF (Villa Serna et al., 2026) moves retrieval directly to the input space. TimeRAF (Zhang et al., 2024) learns the retriever end to end, while SARAF (Zhou et al., 2026) augments temporal similarity with stationarity-aware diversification. These designs motivate our RQ2 comparison across raw-space similarity, elastic temporal distance, frozen-embedding similarity, and summary-statistic similarity. We use these future-blind signals to test how strongly observable similarity aligns with candidate-level predictive utility rather than to optimize another retriever.

History consumption. Retrieved histories are also incorporated into forecasting systems through substantially different mechanisms. RAF exposes the retrieved historical context and continuation to the forecaster through sequence-level input augmentation, and RAEF emphasizes concatenation-based aggregation that preserves the temporal structure of retrieved examples (Tire et al., 2026; Villa Serna et al., 2026). TS-RAG instead introduces an Adaptive Retrieval Mixer that fuses retrieved patterns with internal TSFM representations (Ning et al., 2025). TimeRAF uses Channel Prompting for knowledge integration, while Cross-RAG models query–retrieval interactions through cross-attention before combining retrieved information with the query (Zhang et al., 2024; Lee et al., 2026). Our Pre and Mid interfaces follow existing approaches to sequence concatenation and representation fusion. We also include a training-free Cov interface using the forecaster's native covariate pathway to test how the same candidate's utility varies across interfaces.

Retrieval relevance and predictive utility. The distinction between similarity and downstream usefulness is already reflected in forecasting methods such as TimeRAF (Zhang et al., 2024) and SARAF (Zhou et al., 2026). Related work in language modeling likewise uses downstream model feedback to improve retrieval, as in REPLUG (Shi et al., 2024). Our controlled candidate-level analysis complements these methods by distinguishing historical availability, similarity-based retrievability, and interface dependence.

3 Predictive Utility and Diagnostic Framework

3.1 Forecasting setup and candidate utility

At each forecast origin t, the forecaster observes a context window of length C and predicts the next H values. For a variable x, we denote the query context and its future as follows:

$$q_t = (x_{t-C}, \ldots, x_{t-1}), \qquad y_t = (x_t, \ldots, x_{t+H-1}).$$

A historical candidate $h = (c, z)$ is drawn from the training interval of the same variable, where c is a length-C candidate context and z is the observed H-step continuation immediately following c. A retriever ranks historical candidates using only the query context q and candidate context c. After selection, the full historical candidate h may be provided to the forecaster through an augmentation interface, while the query future y remains hidden and is used only for post-hoc evaluation.

Let $F(q)$ denote the baseline forecast and $F_a(q, h)$ the forecast obtained after candidate h is incorporated through augmentation interface a. For error metric m, we define the *predictive utility* of h for query q as the reduction in forecast error relative to the baseline:

$$U^{(m)}_{F,a}(q, h) = m(y, F(q)) - m(y, F_a(q, h)) .$$

Thus $U_{F,a}^{(m)}(q,h) > 0$ means that the candidate improves the forecast. By construction, utility is defined relative to a particular query, forecasting model, augmentation interface, and error metric. We omit the metric superscript when m is fixed.

3.2 THREE DIAGNOSTICS

Let $\mathcal{H}_B(q)$ denote the candidate pool of B historical candidates for query q. We study three questions: whether useful historical candidates exist in the pool, whether similarity-based retrieval can identify them, and whether their utility remains consistent across augmentation interfaces.

RQ1: Availability. Availability concerns two questions: how often useful candidates occur and how much improvement the best candidate can provide. We measure them by

$$P_{F,a}^{+}(q) = \frac{1}{B} \sum_{h \in \mathcal{H}_B(q)} \mathbf{1}[U_{F,a}(q,h) > 0], \qquad U_{F,a}^{\mathrm{Oracle@}B}(q) = \max_{h \in \mathcal{H}_B(q)} U_{F,a}(q,h).$$

Here, $P_{F,a}^{+}(q)$ is the fraction of positive-utility candidates in the pool, while $U_{F,a}^{\mathrm{Oracle@}B}(q)$ is the maximum utility available in the pool. We also examine the full utility distribution to characterize candidate availability beyond these two summaries.

RQ2: Retrievability. A retriever r assigns each candidate a score $S_r(q,h)$. Its Top-K retrieved set and the best utility contained in that set are

$$\mathcal{R}_{r,K}(q) = \mathrm{TopK}_{h \in \mathcal{H}_B(q)}\, S_r(q,h), \qquad U_{r,F,a}^{\mathrm{Best@}K}(q) = \max_{h \in \mathcal{R}_{r,K}(q)} U_{F,a}(q,h).$$

Alongside Best@K, we report Hit@K, which indicates whether $\mathcal{R}_{r,K}(q)$ contains at least one positive-utility candidate. Beyond these Top-K metrics, we compute the Spearman correlation between retrieval scores and predictive utility over the full candidate pool:

$$\rho_{r,F,a}(q) = \mathrm{Spearman}(S_r(q,\cdot), U_{F,a}(q,\cdot))\,.$$

The oracle gap,

$$U_{F,a}^{\mathrm{Oracle@}B}(q) - U_{r,F,a}^{\mathrm{Best@}K}(q),$$

measures the utility remaining between the retrieved set and the full-pool oracle. As a baseline, all retrieval methods are compared against same-pool Random@K.

RQ3: Interface dependence. RQ3 holds the query, candidate pool, and forecasting backbone fixed, and varies only the augmentation interface. For interface a, define the positive-utility candidate set for query q as

$$\mathcal{P}_a(q) = \{h \in \mathcal{H}_B(q) : U_{F,a}(q,h) > 0\}.$$

For two interfaces a_1 and a_2, we first measure whether they agree on which candidates are useful:

$$J_q(a_1,a_2) = \frac{|\mathcal{P}_{a_1}(q) \cap \mathcal{P}_{a_2}(q)|}{|\mathcal{P}_{a_1}(q) \cup \mathcal{P}_{a_2}(q)|}.$$

We then compare how the two interfaces rank candidates by utility:

$$\rho_q(a_1,a_2) = \mathrm{Spearman}(U_{F,a_1}(q,\cdot), U_{F,a_2}(q,\cdot))\,.$$

Finally, we test whether the best candidate under one interface remains useful under another. For source interface a_1, let

$$h_{a_1}^{*}(q) = \arg\max_{h \in \mathcal{H}_B(q)} U_{F,a_1}(q,h)$$

denote the Oracle@B candidate. We then evaluate the same candidate under target interface a_2:

$$U_{a_1 \to a_2}^{\mathrm{Transfer}}(q) = U_{F,a_2}\big(q, h_{a_1}^{*}(q)\big)\,.$$

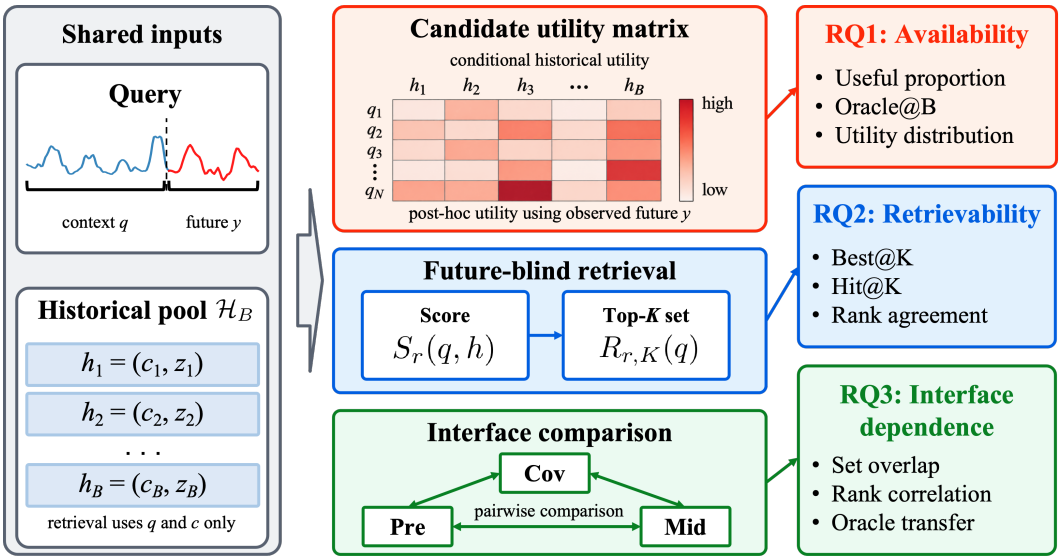


Figure 2: Diagnostic framework for retrieval-augmented forecasting.

4 Experimental Protocol

4.1 Data and paired sampling

We use the ETT benchmarks (ETTh1, ETTh2, ETTm1, and ETTm2) (Zhou et al., 2021), Weather as used in Autoformer (Wu et al., 2021), and Exchange (Lai et al., 2018), with context length $C = 512$ and forecasting horizon $H = 64$. Each candidate contains $C + H = 576$ observations, with its context matching the query length, as in the main experimental setting of TS-RAG (Ning et al., 2025). Retrieval is restricted to the same variable as the query. For each variable, we use $B = 1000$ historical candidates from the training interval and $Q = 1000$ test queries, exhaustively evaluating their 10^6 query–candidate pairs for each forecasting system. For controlled comparisons, we keep the queries and candidate pools matched across forecasting systems, augmentation interfaces, and retrieval rules. Further details are provided in Appendix A.

4.2 Forecasting systems and augmentation interfaces

We use Chronos-2 (Ansari et al., 2025), a widely used time-series foundation model with native support for covariate-informed forecasting, as our primary backbone. This support enables a training-free Cov interface alongside Pre and Mid, allowing controlled comparisons across augmentation interfaces with a fixed backbone. We compare three augmentation interfaces: **Cov** supplies the aligned historical candidate through the model's native covariate pathway; **Pre** prepends the complete candidate to the query, representing sequence-level augmentation (Tire et al., 2026; Villa Serna et al., 2026); and **Mid** injects the candidate through an intermediate representation-level fusion module adapted from TS-RAG (Ning et al., 2025). We denote each forecasting system by its backbone and augmentation interface, abbreviating Chronos-2 as C2 and Chronos-Bolt as Bolt (e.g., C2-Cov and Bolt-Mid).

RQ1 evaluates C2-Cov, C2-Pre, C2-Mid, and Bolt-Mid on all six datasets, with Moirai-{Cov,Pre,Mid} on ETTh1 as a backbone-robustness probe. RQ2 reuses the four core utility matrices, while RQ3 fixes Chronos-2, queries, and candidate pools and varies only Cov, Pre, and Mid.

4.3 Retrieval signals

RQ2 compares six retrieval signals over the same candidate pool, with a retrieval budget of $K = 10$. Raw-Cosine and Raw-L2 compare the query and candidate contexts directly. DTW-PAA64-Z applies dynamic time warping (DTW) (Sakoe & Chiba, 1978) after per-window normal-

Table 1: RQ1 under MSE. Each cell reports useful-candidate proportion / Oracle@1000 gain (%).

Dataset	C2-Cov	C2-Pre	C2-Mid	Bolt-Mid
ETTh1	41.35 / 58.13	54.08 / 36.10	48.92 / 24.56	53.05 / 25.60
ETTh2	39.29 / 56.26	51.19 / 37.62	39.57 / 35.00	47.06 / 37.52
ETTm1	44.52 / 71.01	48.15 / 67.90	45.26 / 29.14	45.94 / 44.34
ETTm2	43.48 / 60.42	50.01 / 54.23	43.08 / 32.01	44.77 / 40.31
Weather	52.81 / 66.87	45.45 / 68.99	38.32 / 28.02	43.68 / 52.64
Exchange	50.55 / 53.66	56.23 / 41.52	41.08 / 48.35	47.81 / 43.94

ization and piecewise aggregate approximation (PAA) (Keogh et al., 2001). Embedding-Cosine and Embedding-L2 compare representations from a shared frozen Chronos-2 encoder. Statistics-L2 compares ten standardized summary features, listed in Appendix A.

4.4 Metrics and reporting

MSE (Mean Squared Error) is the primary forecasting metric, with MAE (Mean Absolute Error) used as a robustness check. For percentage gains, we first aggregate errors across queries within each variable, compute the relative reduction, and then average equally across variables. Random@10 is computed exactly for uniform sampling without replacement from the same candidate pool. RQ3 agreement metrics are averaged over valid queries within each variable and then equally across variables. Further metric and aggregation details are provided in Appendix A.

5 How Common and Beneficial Are Useful Historical Candidates? (RQ1)

Table 1 summarizes historical availability under MSE.

Useful candidates are common and offer substantial oracle gains. Across all 24 dataset–system settings, 38.32%–56.23% of historical candidates improve forecasting, and Oracle@1000 achieves gains of 24.56%–71.01% (Table 1). Useful candidates are therefore common in the evaluated pools, and selecting the best candidate for each query can substantially reduce forecasting error. These oracle gains reveal substantial room for improvement through candidate selection. Whether this opportunity can be realized by similarity-based retrieval is the focus of RQ2.

Prevalence and oracle gains favor different interfaces. On all datasets except Weather, C2-Pre has the highest useful-candidate proportion, whereas C2-Cov achieves the largest oracle gain. On ETTh1, C2-Pre has a higher useful-candidate proportion than C2-Cov (54.08% vs. 41.35%) but a smaller oracle gain (36.10% vs. 58.13%). Figure 3 illustrates this distinction: C2-Pre has a lower CDF at zero, but C2-Cov's CDF falls below it in the high-gain region. C2-Cov thus has fewer useful candidates overall but a larger share of high-gain candidates, consistent with its larger oracle gain. Its heavier negative tail also indicates greater downside. Useful-candidate prevalence and oracle gain therefore characterize complementary aspects of historical availability: how often candidates help and how much the best candidates can help.

6 Can Similarity-Based Retrieval Prioritize Useful Historical Candidates? (RQ2)

RQ1 has established that useful historical candidates exists, and RQ2 asks whether similarity-based retrieval identifies them more effectively than random sampling. We compare the best utility within each retrieved Top-10 set (Best@10) with that of 10 candidates sampled from the same pool (Random@10). Table 2 reports the retriever with the highest Best@10 in each setting, selected in hindsight from the six tested signals.

Similarity-based retrieval rarely outperforms random sampling. Even the best-in-hindsight retriever falls below Random@10 in 17 of the 24 dataset–system settings. In the remaining seven,

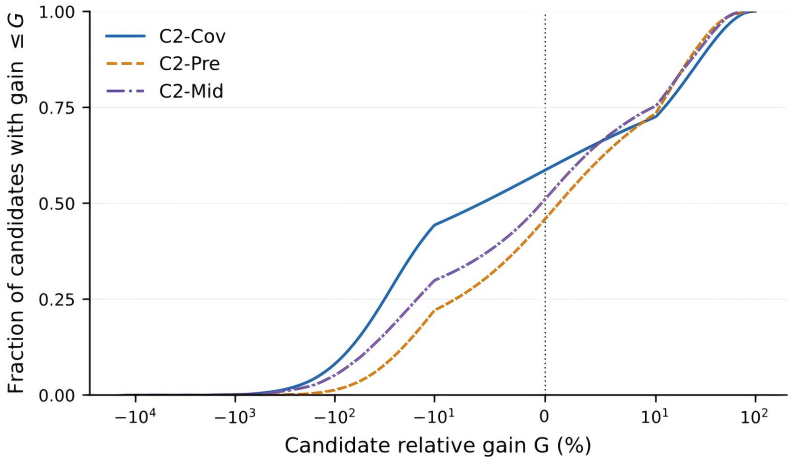


Figure 3: Variable-balanced empirical CDFs of candidate-level relative gain on ETTh1. The dotted line marks zero gain; the x-axis uses a symmetric-log scale.

Table 2: RQ2 under MSE. r^* is the best-in-hindsight retriever by Best@10. Best/Random are gains (%), Gap is Oracle@1000–Best@10 in percentage points, Hit is Hit@10, and ρ is retrieval–utility Spearman correlation. Bold indicates the larger gain within each Best/Random pair.

Dataset	System	r^*	Best / Random	Gap	Hit	ρ
ETTh1	C2-Cov	Emb-L2	29.80 / **32.21**	28.33	0.898	0.014
	C2-Pre	Emb-Cos	**20.46** / 19.60	15.64	0.911	0.001
	C2-Mid	Raw-Cos	10.12 / **12.51**	14.45	0.695	0.000
	Bolt-Mid	Emb-L2	12.46 / **14.16**	13.14	0.926	-0.014
ETTh2	C2-Cov	DTW	**29.39** / 28.19	26.87	0.824	-0.006
	C2-Pre	Emb-Cos	19.28 / **19.62**	18.34	0.859	0.014
	C2-Mid	Raw-L2	8.24 / **10.51**	26.76	0.582	0.022
	Bolt-Mid	Emb-Cos	16.58 / **17.97**	20.94	0.846	0.006
ETTm1	C2-Cov	Raw-Cos	**47.37** / 43.34	23.64	0.891	0.004
	C2-Pre	Emb-Cos	**42.22** / 41.48	25.68	0.888	-0.008
	C2-Mid	Stat-L2	12.06 / **13.31**	17.07	0.661	0.004
	Bolt-Mid	Stat-L2	26.34 / **27.88**	18.00	0.871	0.001
ETTm2	C2-Cov	DTW	**34.23** / 32.48	26.19	0.863	-0.009
	C2-Pre	Emb-L2	29.66 / **30.95**	24.58	0.855	-0.008
	C2-Mid	Stat-L2	9.16 / **11.89**	22.85	0.642	0.007
	Bolt-Mid	Stat-L2	22.32 / **22.67**	17.99	0.858	-0.004
Weather	C2-Cov	Raw-Cos	**43.69** / 40.23	23.18	0.919	-0.005
	C2-Pre	DTW	41.83 / **44.88**	27.16	0.840	0.010
	C2-Mid	Raw-L2	10.22 / **11.50**	17.80	0.545	-0.005
	Bolt-Mid	Raw-L2	**33.93** / 32.08	18.71	0.852	0.034
Exchange	C2-Cov	Emb-Cos	21.73 / **24.81**	31.92	0.850	-0.006
	C2-Pre	Emb-Cos	15.85 / **23.90**	25.67	0.744	0.001
	C2-Mid	Emb-Cos	16.07 / **33.59**	32.28	0.604	0.072
	Bolt-Mid	Emb-L2	21.10 / **32.08**	22.84	0.790	0.110

its advantage is limited to 0.74–4.03 percentage points. Across all settings, Best@10 remains 13.14–32.28 points below Oracle@1000. Thus, although useful candidates are available, the tested similarity signals generally provide no advantage over random sampling in the best utility captured.

The limitation extends across retrieval signals. Figure 4 shows the results for all six retrievers: 120 of the 144 retriever–setting pairs fail to outperform Random@10. Positive differences occur only in a minority of settings, and no retrieval signal consistently improves over the random baseline. On Exchange, all tested retrievers fall below Random@10 for all the four systems. Thus the limited benefit of similarity-based retrieval extends beyond any single retrieval signal or forecasting system.

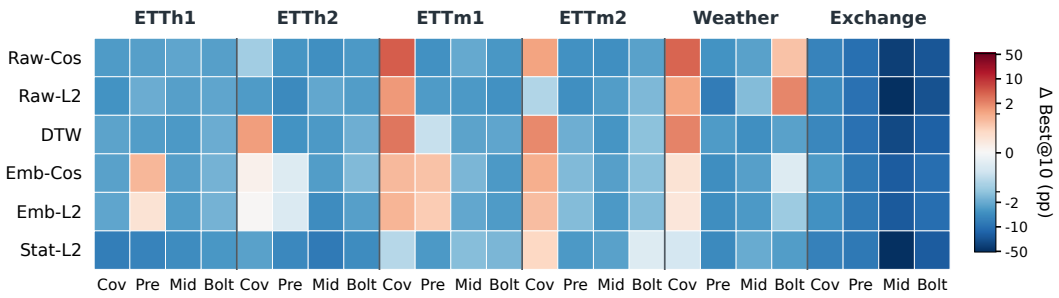


Figure 4: Best@10 minus same-pool Random@10 across six retrieval signals and 24 dataset–system settings. Values are percentage-point differences; red is positive and blue is negative. Colors use a zero-centered symmetric-log scale.

Table 3: RQ3 under MSE with Chronos-2. J is useful-set Jaccard overlap (%), ρ_U is utility-rank Spearman correlation, and each transfer cell reports transferred gain / target Oracle@1000 gain (%).

Dataset	A/B	J	ρ_U	$A \to B$	$B \to A$
ETTh1	Cov / Pre	31.24	0.048	3.09 / 36.10	-3.55 / 58.13
	Cov / Mid	25.62	0.000	-6.47 / 24.56	-18.75 / 58.13
	Pre / Mid	32.69	0.059	-0.75 / 24.56	2.69 / 36.10
ETTh2	Cov / Pre	27.67	0.025	2.15 / 37.62	-4.68 / 56.26
	Cov / Mid	20.60	0.010	-13.81 / 35.00	-25.78 / 56.26
	Pre / Mid	24.73	0.053	-12.53 / 35.00	-4.17 / 37.62
ETTm1	Cov / Pre	31.70	0.057	4.47 / 67.90	6.45 / 71.01
	Cov / Mid	26.10	0.004	-4.95 / 29.14	-8.02 / 71.01
	Pre / Mid	29.45	0.088	-0.38 / 29.14	0.23 / 67.90
ETTm2	Cov / Pre	30.48	0.041	5.00 / 54.23	2.49 / 60.42
	Cov / Mid	24.12	0.002	-8.05 / 32.01	-14.70 / 60.42
	Pre / Mid	26.73	0.073	-3.76 / 32.01	-6.10 / 54.23
Weather	Cov / Pre	34.03	0.062	3.14 / 68.99	11.96 / 66.87
	Cov / Mid	23.50	0.000	-20.39 / 28.02	0.90 / 66.87
	Pre / Mid	22.37	0.056	-13.18 / 28.02	2.22 / 68.99
Exchange	Cov / Pre	37.52	0.067	5.57 / 41.52	2.11 / 53.66
	Cov / Mid	28.67	0.033	-92.02 / 48.35	0.67 / 53.66
	Pre / Mid	29.70	0.165	-32.57 / 48.35	8.32 / 41.52

Finding a useful candidate does not imply ranking candidates by utility. The best-in-hindsight retrievers attain Hit@10 values of 0.545–0.926, indicating that their retrieved sets often contain at least one useful candidate. However, useful candidates are already common in the underlying pools, so Hit@10 alone does not establish an advantage from similarity-based selection. Retrieval scores also show weak rank agreement with utility: $|\rho| < 0.05$ in 22 of the 24 settings. Together, these results show that similarity can retrieve useful candidates without reliably prioritizing those that provide larger forecasting gains.

7 Does Predictive Utility Transfer Across Augmentation Interfaces? (RQ3)

RQ3 asks whether a candidate that is useful under one augmentation interface remains useful under another. We hold the forecasting backbone (Chronos-2), queries, and candidate pools fixed, varying only the interface among Cov, Pre, and Mid. We compare useful-candidate sets and utility rankings across interfaces, then evaluate whether candidates selected by the source-interface oracle retain their gains under the target interface.

Candidate usefulness and rankings vary across interfaces. Across the 18 dataset–interface-pair comparisons, the Jaccard overlap between useful-candidate sets ranges from 20.60% to 37.52%, and utility-rank correlations satisfy $|\rho_U| < 0.1$ in 17 cases (Table 3). Figure 5 illustrates this dependence on ETTh1: many query–candidate pairs lie in opposite-sign quadrants, where the same history im-

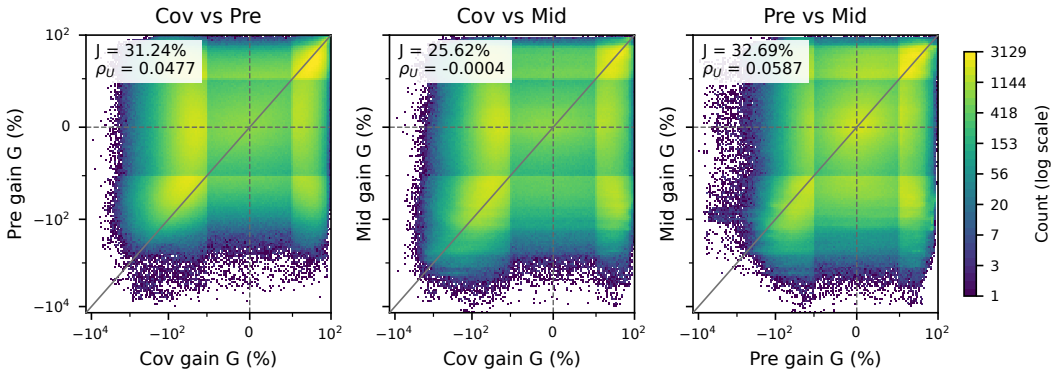


Figure 5: Paired candidate utilities across the three Chronos-2 augmentation interfaces on ETTh1. Hexbin color shows matched-pair count on a log scale; axes use the same symmetric-log scale. Dashed lines mark zero gain, the diagonal marks equal utility, and insets report J and ρ_U.

proves forecasting under one interface but worsens it under another. Thus, changing the interface affects both which candidates are useful and how they rank by utility.

Source-optimal candidates often yield negative gains after transfer. Selecting the best candidate under the source interface produces negative aggregate gains under the target interface in 20 of the 36 directional transfers. All 12 transfers from Cov or Pre to Mid yield negative gains, despite Mid's own Oracle@1000 gains of 24.56%–48.35%. Even in the 16 positive transfers, the gain amounts to at most 20.0% of the target-interface oracle gain. These results show that candidates optimal under one interface cannot be assumed to remain useful under another, even when useful candidates are available in the same pool.

8 Limitations

Historical data and window configuration. We evaluate six datasets using pools of $B = 1000$ candidates drawn from the training interval of the same variable, with context length $C = 512$ and continuation length $H = 64$. Whether the findings persist with different pool sizes, window lengths, or retrieval from other variables and domains remains untested.

Statistical uncertainty. The reported comparisons are descriptive point estimates without confidence intervals or significance tests. Although queries and candidate pools are matched across comparisons, uncertainty associated with query and candidate-pool sampling is not quantified.

9 Conclusion

We examine retrieval-augmented time series forecasting through candidate-level predictive utility from three aspects including candidate availability, similarity-based retrievability, and interface dependence. Useful candidates are common in the evaluated pools and offer substantial oracle gains, yet the tested similarity-based retrievers generally fail to outperform matched random sampling. Moreover, candidates selected as optimal under one interface often become harmful in aggregate under another. Our results challenge the reliability of similarity as a surrogate for predictive utility, and that predictive utility is interface-dependent rather than intrinsic to a historical candidate. These findings motivate developing proxies that better anticipate forecasting gains without access to future outcomes, and jointly designing candidate selection and augmentation interfaces for the target forecasting task.

Our code is available: https://anonymous.4open.science/r/ICLR27-E7F0

AI USE STATEMENT

We used generative AI tools to assist with manuscript editing, structural revision, and clarity of presentation. All AI-assisted content was reviewed and verified by the authors against the implemented methods, experimental outputs, and cited literature. The authors take full responsibility for the final content of this work, including all text, claims, code, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide the formal definitions of all diagnostics in Section 3 and the experimental protocol in Section 4. Appendix A specifies the candidate construction, augmentation interfaces, retrieval signals, metrics, and aggregation procedures. Appendices B–D provide the complete robustness and supplementary results for RQ1–RQ3, including the full retrieval comparison and cross-interface analyses. All reported comparisons use matched query and candidate manifests where required by the experimental design.

REFERENCES

Abdul Fatir Ansari, Caner Turkmen, Oleksandr Shchur, and Lorenzo Stella. Fast and accurate zero-shot forecasting with Chronos-Bolt and AutoGluon. AWS Machine Learning Blog, December 2024. URL https://aws.amazon.com/blogs/machine-learning/fast-and-accurate-zero-shot-forecasting-with-chronos-bolt-and-autogluon/. Official model release, December 2, 2024.

Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, Mononito Goswami, Shubham Kapoor, Danielle C. Maddix, Pablo Guerron, Tony Hu, Junming Yin, Nick Erickson, Prateek Mutalik Desai, Hao Wang, Huzefa Rangwala, George Karypis, Yuyang Wang, and Michael Bohlke-Schneider. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv:2510.15821*, 2025. doi: 10.48550/arXiv.2510.15821. URL https://arxiv.org/abs/2510.15821.

Sungwon Han, Seungeon Lee, Meeyoung Cha, Sercan O Arik, and Jinsung Yoon. Retrieval augmented time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 21774–21797. PMLR, 2025. URL https://proceedings.mlr.press/v267/han25d.html.

Eamonn Keogh, Kaushik Chakrabarti, Michael Pazzani, and Sharad Mehrotra. Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and Information Systems*, 3:263–286, 2001. doi: 10.1007/PL00011669. URL https://link.springer.com/article/10.1007/PL00011669.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 95–104. ACM, 2018. doi: 10.1145/3209978.3210006. URL https://arxiv.org/abs/1703.07015.

Seunghan Lee, Jaehoon Lee, Jun Seo, Sungdong Yoo, Minjae Kim, Tae Yoon Lim, Dongwan Kang, Hwanil Choi, SoonYoung Lee, and Wonbin Ahn. Not all retrievals are useful: Cross-attention for input-aware RAG in time series forecasting. *arXiv preprint arXiv:2603.14709*, 2026. doi: 10.48550/arXiv.2603.14709. URL https://arxiv.org/abs/2603.14709v3. Version 3; KDD Workshop on Mining and Learning from Time Series 2026.

Kanghui Ning, Zijie Pan, Yu Liu, Yushan Jiang, James Zhang, Kashif Rasul, Anderson Schneider, Lintao Ma, Yuriy Nevmyvaka, and Dongjin Song. TS-RAG: Retrieval-augmented generation based time series foundation models are stronger zero-shot forecaster. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5448. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/eed25c037bc08afcbefab6f7a6b700e0-Abstract-Conference.html.

Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49, 1978. doi: 10.1109/TASSP.1978.1163055. URL https://doi.org/10.1109/TASSP.1978.1163055.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. REPLUG: Retrieval-augmented black-box language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 8371–8384. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.naacl-long.463. URL https://aclanthology.org/2024.naacl-long.463/.

Kutay Tire, Ege Onur Taga, Muhammed Emrullah Ildiz, and Samet Oymak. Retrieval augmented time series forecasting. In *Proceedings of the 29th International Conference on Artificial Intelligence and Statistics*, volume 300 of *Proceedings of Machine Learning Research*, pp. 4294–4302. PMLR, 2026. URL https://proceedings.mlr.press/v300/tire26a.html.

Juan Pablo Villa Serna, Rohan Asthana, and Vasileios Belagiannis. Model-agnostic retrieval-augmented extended forecasting for time series. *arXiv preprint arXiv:2608.14054*, 2026. doi: 10.48550/arXiv.2608.14054. URL https://arxiv.org/abs/2608.14054.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53140–53164. PMLR, 2024. URL https://proceedings.mlr.press/v235/woo24a.html.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/bcc0d400288793e8bdcd7c19a8ac0c2b-Abstract.html.

Silin Yang, Dong Wang, Haoqi Zheng, and Ruochun Jin. TimeRAG: Boosting LLM time series forecasting via retrieval-augmented generation. *arXiv preprint arXiv:2412.16643*, 2024. doi: 10.48550/arXiv.2412.16643. URL https://arxiv.org/abs/2412.16643.

Huanyu Zhang, Chang Xu, Yi-Fan Zhang, Zhang Zhang, Liang Wang, Jiang Bian, and Tieniu Tan. TimeRAF: Retrieval-augmented foundation model for zero-shot time series forecasting. *arXiv preprint arXiv:2412.20810*, 2024. doi: 10.48550/arXiv.2412.20810. URL https://arxiv.org/abs/2412.20810.

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. doi: 10.1609/aaai.v35i12.17325. URL https://ojs.aaai.org/index.php/AAAI/article/view/17325.

Shiqiao Zhou, Holger Schöner, Zipeng Wu, Edouard Fouché, IAG Wilson, and Shuo Wang. Stationarity-aware retrieval-augmented time series forecasting. *arXiv preprint arXiv:2606.04135*, 2026. doi: 10.48550/arXiv.2606.04135. URL https://arxiv.org/abs/2606.04135. Accepted to the KDD 2026 research track.

A Experimental Details

A.1 Datasets and matched candidate pools

The main experiments use the ETT benchmarks (ETTh1, ETTh2, ETTm1, and ETTm2) (Zhou et al., 2021), Weather (Wu et al., 2021), and Exchange (Lai et al., 2018), with context length $C = 512$ and forecasting horizon $H = 64$. Sampling is performed separately for each variable in each dataset, and retrieval is restricted to the same variable as the query.

For each variable, we randomly sample $Q = 1000$ test queries and, separately, $B = 1000$ historical candidates from the training interval. Both the context and continuation of every historical candidate lie entirely within the training interval. The sampled historical candidates form a single pool shared by all Q queries for that variable. We exhaustively evaluate all $Q \times B = 10^6$ query–candidate pairs for each forecasting system.

Once sampled, the query set and candidate pool are held fixed and reused across all forecasting backbones, augmentation interfaces, and retrieval rules evaluated on that dataset and variable. This ensures that differences between systems are evaluated on the same query–candidate pairs. Query futures are used only for post-hoc evaluation, including utility computation, oracle selection, and cross-interface transfer analysis; they are not available to the similarity-based retrievers or the forecasting models.

A.2 Historical candidates and information boundaries

Each historical candidate contains $C + H = 576$ consecutive observations: a context of length 512 followed by its already observed continuation of length 64. Candidate selection uses only the query context and candidate context. After selection, the historical continuation may be supplied to the forecasting system through the corresponding augmentation interface.

Oracle@1000 selects the highest-utility candidate from the sampled set of 1000 historical candidates associated with each query. It therefore quantifies the best single-candidate opportunity within the evaluated finite pool.

A.3 Augmentation interfaces and forecasting systems

We evaluate three augmentation interfaces that differ in how the selected historical candidate is presented to the forecaster.

Covariate interface (Cov). The candidate context and its historical continuation are aligned with the query as auxiliary covariate information.

Prefix interface (Pre). The complete historical candidate is prepended to the query. Before model-specific tokenization or patching, the resulting numerical sequence contains $2C + H = 1088$ observations.

Intermediate interface (Mid). The candidate is injected through an intermediate fusion module adapted from TS-RAG (Ning et al., 2025). C2-Mid combines this interface with Chronos-2, whereas Bolt-Mid retains the original Chronos-Bolt TS-RAG backbone and augmentation structure.

For Bolt-Mid, the same selected candidate is repeated ten times to match the input structure expected by the augmentation module. These copies constitute a single candidate-level intervention and are distinct from RQ2 Top-10 retrieval, which contains ten distinct historical candidates.

A.4 Retrieval signals

RQ2 evaluates six future-blind retrieval signals over the same $B = 1000$ candidate pools used to construct the utility matrices. Retrieval scores are oriented consistently so that larger values always indicate higher retrieval priority; distance measures are therefore negated before rank-based evaluation.

Table 4: Retrieval signals used in RQ2.

Retriever	Compared representation	Raw quantity	Main preprocessing
Raw-Cos	Raw query/candidate contexts	Cosine similarity	Shared length-512 windows
Raw-L2	Raw query/candidate contexts	Negative L2 distance	Shared length-512 windows
DTW	Raw query/candidate contexts	Negative DTW distance	Per-window normalization; PAA $512 \to 64$
Emb-Cos	Frozen Chronos-2 representations	Cosine similarity	Shared representation extractor
Emb-L2	Frozen Chronos-2 representations	Negative L2 distance	Shared representation extractor
Stat-L2	Ten statistical descriptors	Negative standardized L2 distance	Training-period feature standardization

Statistics-L2 uses ten descriptors: mean, standard deviation, normalized slope, autocorrelation at lags 1, 24, and 168, dominant period, spectral centroid, spectral entropy, and peak-power share. Feature scaling is estimated from the training-period candidate population. DTW uses per-window normalization followed by PAA compression from 512 to 64 points. Embedding-based retrieval uses a shared frozen Chronos-2 representation extractor throughout the RQ2 comparison.

A.5 Metrics and aggregation

MSE is the primary forecasting metric, with MAE used as a robustness check. Candidate usefulness and candidate rankings are defined using absolute utility U. Percentage gains reported in the main text use aggregated errors: errors are first aggregated across queries within each variable, converted to a relative reduction, and then macro-averaged equally across variables. Query-wise percentage gains are reported only as supplementary diagnostics because ratios computed separately for individual queries can become unstable when baseline errors are small.

For RQ2, Hit@10 is the fraction of queries for which the retrieved set contains at least one positive-utility candidate. Best@10 evaluates the highest-utility candidate among the ten retrieved histories. Retrieval–utility Spearman correlation is computed within each query over all $B = 1000$ candidates.

Random@10 uses uniform sampling without replacement from the same $B = 1000$ candidate pool. If a query contains P positive-utility candidates, its exact random Hit@K probability is

$$\Pr(\text{Hit@}K) = 1 - \frac{\binom{B-P}{K}}{\binom{B}{K}}.$$

For Random Best@K, let

$$u_{(1)} \le u_{(2)} \le \cdots \le u_{(B)}$$

denote the candidate utilities for one query in ascending order. Under uniform sampling of K candidates without replacement, the exact expected maximum utility is

$$\mathbb{E}[\text{Best@}K] = \sum_{j=K}^{B} u_{(j)} \frac{\binom{j-1}{K-1}}{\binom{B}{K}}.$$

The corresponding expected augmented error is obtained from this expected utility. Errors are then aggregated across queries within each variable, converted to a relative gain against the aggregated baseline error, and macro-averaged equally across variables, using the same aggregation scheme as the deterministic retrievers.

For RQ3, define the positive-utility candidate set for query q under interface a as

$$\mathcal{P}_a(q) = \{h \in \mathcal{H}_B(q) : U_{F,a}(q,h) > 0\}.$$

For each query and interface pair A/B, useful-set overlap is

$$J_q(A,B) = \frac{|\mathcal{P}_A(q) \cap \mathcal{P}_B(q)|}{|\mathcal{P}_A(q) \cup \mathcal{P}_B(q)|},$$

and candidate-ranking agreement over the same 1000 candidates is

$$\rho_q^U(A,B) = \text{Spearman}(U_{F,A}(q,\cdot), U_{F,B}(q,\cdot)).$$

Table 5: RQ1 under MAE. Each cell reports useful-candidate proportion / Oracle@1000 aggregated-error gain (%).

Dataset	C2-Cov	C2-Pre	C2-Mid	Bolt-Mid
ETTh1	40.67 / 35.15	53.66 / 20.57	46.61 / 12.65	49.48 / 14.99
ETTh2	38.45 / 34.38	46.74 / 21.85	36.78 / 17.46	42.68 / 21.71
ETTm1	44.13 / 46.87	47.77 / 43.84	44.42 / 17.11	44.45 / 28.45
ETTm2	43.44 / 40.51	43.96 / 35.18	38.54 / 17.23	40.65 / 26.27
Weather	53.40 / 50.71	44.68 / 52.25	35.69 / 17.52	41.85 / 36.98
Exchange	50.34 / 42.07	56.04 / 30.85	41.10 / 36.33	47.26 / 34.48

Table 6: ETTh1 backbone robustness for RQ1. Useful is the useful-candidate proportion and Oracle is the finite-pool Oracle@1000 gain. All values are percentages.

System	Useful (MSE)	Oracle (MSE)	Useful (MAE)	Oracle (MAE)
Moirai-Cov	49.79	26.08	49.40	15.11
Moirai-Pre	49.51	35.39	48.32	20.71
Moirai-Mid	40.09	37.58	37.96	21.94

For both metrics, valid query-level values are first averaged within each variable and then macro-averaged equally across variables. Queries for which both positive-utility sets are empty are excluded from the Jaccard average. Queries for which either utility vector is constant are excluded from the corresponding Spearman average.

Directional source-oracle transfer selects a highest-utility candidate under the source interface and evaluates that same query–candidate pair under the target interface. All reported comparisons are descriptive point estimates; no confidence intervals or significance tests are reported.

B RQ1 Supplementary Results

B.1 MAE robustness

Table 5 repeats RQ1 using MAE-based candidate utility. Each cell reports useful-candidate proportion and Oracle@1000 aggregated-error gain.

The MAE results preserve the main RQ1 pattern. Across all 24 dataset–system settings, useful-candidate proportions range from 35.69% to 56.04%, and every finite candidate pool contains a positive oracle opportunity, with Oracle@1000 gains of 12.65–52.25%. In five of the six datasets, C2-Pre again has the highest useful-candidate proportion while C2-Cov has the largest oracle gain; Weather reverses this pattern. The separation between opportunity density and opportunity magnitude therefore persists under MAE.

B.2 Backbone robustness on ETTh1

The main controlled interface comparison uses Chronos-2. We additionally evaluate RQ1 on ETTh1 with other forecasting backbones to test whether finite-pool historical opportunity is specific to the primary model.

For Moirai, useful-candidate proportions range from 40.09% to 49.79% under MSE, while Oracle@1000 gains range from 26.08% to 37.58%. Moirai-Cov has the largest useful-candidate proportion, whereas Moirai-Mid has the largest oracle gain, again separating opportunity density from opportunity magnitude. The same qualitative separation is preserved under MAE, and all three interfaces retain positive finite-pool oracle gains.

B.3 Bolt-Mid variable-level decomposition

Dataset-level Bolt-Mid values in the main text are macro-averaged across variables. Table 7 decomposes the ETTh1 and ETTh2 MSE results into their seven constituent variables. Every variable retains a positive Oracle@1000 gain: 20.61–46.82% on ETTh1 and 33.06–43.20% on ETTh2. The dataset-level opportunity estimates are therefore not driven by improvement on a single variable.

Table 7: Bolt-Mid variable-level MSE results. Gain is the aggregated-error Oracle@1000 reduction (%).

	ETTh1			ETTh2		
Variable	Baseline	Oracle	Gain	Baseline	Oracle	Gain
HUFL	24.6368	19.5584	20.61	45.3492	25.9419	42.80
HULL	0.8804	0.6792	22.85	7.4690	4.9998	33.06
MUFL	22.7485	17.8619	21.48	39.2907	22.3164	43.20
MULL	0.6097	0.4801	21.26	5.4458	3.3347	38.76
LUFL	0.5327	0.4137	22.34	4.0721	2.6481	34.97
LULL	0.0432	0.0329	23.85	0.6343	0.4058	36.03
OT	4.4560	2.3696	46.82	12.5159	8.2817	33.83

Table 8: RQ2 Best@10 aggregated-error gain (%) under MSE. Random@10 is the exact expected Best@10 under uniform sampling without replacement from the same $B = 1000$ candidate pool.

Dataset	System	Raw-Cos	Raw-L2	DTW	Emb-Cos	Emb-L2	Stat-L2	Random
ETTh1	C2-Cov	29.18	28.45	29.72	29.56	29.80	25.14	32.21
	C2-Pre	16.81	17.61	16.67	20.46	19.97	13.59	19.60
	C2-Mid	10.12	9.71	9.21	9.75	9.58	7.79	12.51
	Bolt-Mid	11.34	11.72	12.17	12.46	12.46	10.79	14.16
ETTh2	C2-Cov	27.31	25.10	29.39	28.32	28.23	25.58	28.19
	C2-Pre	15.98	14.75	15.73	19.28	19.25	14.48	19.62
	C2-Mid	6.29	8.24	7.19	7.49	5.94	2.61	10.51
	Bolt-Mid	14.66	15.01	16.15	16.58	15.17	13.36	17.97
ETTm1	C2-Cov	47.37	44.66	45.89	44.17	44.20	42.61	43.34
	C2-Pre	37.45	38.44	40.88	42.22	42.12	38.23	41.48
	C2-Mid	11.11	10.02	10.78	11.77	10.99	12.06	13.31
	Bolt-Mid	24.40	23.93	25.44	24.68	24.69	26.34	27.88
ETTm2	C2-Cov	33.57	31.72	34.23	33.41	33.26	33.00	32.48
	C2-Pre	27.03	26.82	29.09	29.59	29.66	27.61	30.95
	C2-Mid	7.62	8.88	8.36	9.13	8.64	9.16	11.89
	Bolt-Mid	20.06	21.23	21.57	21.49	21.36	22.32	22.67
Weather	C2-Cov	43.69	41.23	42.18	40.59	40.54	39.78	40.23
	C2-Pre	41.05	37.43	41.83	40.41	40.36	39.98	44.88
	C2-Mid	8.84	10.22	7.22	8.76	8.19	9.46	11.50
	Bolt-Mid	32.83	33.93	29.40	31.74	31.16	29.19	32.08
Exchange	C2-Cov	18.71	19.91	19.40	21.73	20.82	18.36	24.81
	C2-Pre	13.43	13.70	13.79	15.85	15.66	15.61	23.90
	C2-Mid	-1.98	-18.56	6.48	16.07	14.83	-18.79	33.59
	Bolt-Mid	12.06	9.57	16.00	20.85	21.10	14.53	32.08

Because Bolt-Mid uses a different forecasting backbone and baseline from Chronos-2, these values are interpreted within system.

C RQ2 Complete Retrieval Results

The main text reports, for each dataset–system setting, the best-in-hindsight retriever r^* according to Best@10. This section reports all six retrieval signals on the same candidate pools, allowing the upper-envelope summary in the main text to be examined against the complete retrieval results.

C.1 Best@10

Across the $24 \times 6 = 144$ retriever–setting combinations, only 24 Best@10 point estimates exceed the corresponding same-pool Random@10 baseline. No individual retrieval signal exceeds Random@10 in more than 6 of the 24 dataset–system settings: Emb-Cos and Emb-L2 do so in 6 settings each, Raw-Cos and DTW in 4 each, Raw-L2 in 3, and Stat-L2 in 1. The complete grid therefore reinforces the main-text result that weak candidate-set enrichment is not specific to a single similarity family.

Table 9: RQ2 Hit@10 under MSE for all six retrieval signals and the exact same-pool Random@10 baseline.

Dataset	System	Raw-Cos	Raw-L2	DTW	Emb-Cos	Emb-L2	Stat-L2	Random
ETTh1	C2-Cov	0.872	0.871	0.859	0.891	0.898	0.790	0.894
	C2-Pre	0.877	0.900	0.880	0.911	0.907	0.847	0.920
	C2-Mid	0.695	0.697	0.680	0.696	0.696	0.659	0.730
	Bolt-Mid	0.899	0.918	0.921	0.926	0.926	0.864	0.934
ETTh2	C2-Cov	0.784	0.774	0.824	0.850	0.845	0.753	0.867
	C2-Pre	0.780	0.778	0.784	0.859	0.855	0.755	0.858
	C2-Mid	0.581	0.582	0.585	0.613	0.587	0.560	0.669
	Bolt-Mid	0.799	0.808	0.859	0.846	0.842	0.806	0.873
ETTm1	C2-Cov	0.891	0.876	0.875	0.887	0.882	0.861	0.883
	C2-Pre	0.877	0.880	0.866	0.888	0.886	0.848	0.880
	C2-Mid	0.646	0.643	0.649	0.650	0.649	0.661	0.681
	Bolt-Mid	0.915	0.917	0.896	0.899	0.896	0.871	0.883
ETTm2	C2-Cov	0.846	0.858	0.863	0.880	0.871	0.845	0.877
	C2-Pre	0.797	0.839	0.835	0.866	0.855	0.825	0.858
	C2-Mid	0.613	0.659	0.629	0.641	0.627	0.642	0.665
	Bolt-Mid	0.848	0.878	0.869	0.878	0.865	0.858	0.861
Weather	C2-Cov	0.919	0.883	0.891	0.892	0.890	0.873	0.905
	C2-Pre	0.832	0.852	0.840	0.857	0.856	0.852	0.862
	C2-Mid	0.540	0.545	0.534	0.537	0.533	0.549	0.565
	Bolt-Mid	0.849	0.852	0.827	0.844	0.842	0.832	0.854
Exchange	C2-Cov	0.745	0.758	0.803	0.850	0.845	0.785	0.906
	C2-Pre	0.691	0.699	0.716	0.744	0.744	0.743	0.872
	C2-Mid	0.509	0.468	0.544	0.604	0.605	0.507	0.727
	Bolt-Mid	0.694	0.602	0.713	0.781	0.790	0.655	0.878

C.2 HIT@10

Similarity retrieval also provides little consistent enrichment under Hit@10. Across the 144 retriever–setting combinations, only 18 have a Hit@10 point estimate above the corresponding random baseline. The relatively high absolute Hit@10 values in several settings are consistent with the high prevalence of positive-utility candidates observed in RQ1 and therefore do not by themselves indicate utility-aware ranking.

C.3 RETRIEVAL–UTILITY RANK AGREEMENT

The complete correlation grid is concentrated near zero. Among the 144 retriever–setting combinations, 100 have $|\rho| \leq 0.02$, 129 have $|\rho| < 0.05$, and 138 have $|\rho| < 0.1$; the largest observed absolute correlation is 0.113. The weak retrieval–utility alignment reported for the best-in-hindsight retrievers in the main text therefore extends across the full set of tested similarity signals.

C.4 MAE RETRIEVAL ROBUSTNESS

Table 11 repeats the Best@10 versus Random@10 comparison using MAE-based candidate utility. For each setting, the reported Best@10 value is the best-in-hindsight result among the same six retrieval signals evaluated in the MSE analysis.

The MAE comparison reproduces the main RQ2 pattern: the best-in-hindsight tested retriever exceeds same-pool Random@10 in only 7 of the 24 dataset–system settings. All four Exchange settings remain below the random baseline. Weak similarity-based retrievability is therefore preserved when candidate utility is defined using MAE.

D RQ3 SUPPLEMENTARY RESULTS

The main text reports variable-balanced query-wise useful-set overlap and utility-rank agreement together with aggregated-error source-oracle transfer under MSE. This section examines aggregation sensitivity for directional transfer and repeats the RQ3 analysis under MAE.

Table 10: Query-wise Spearman correlation between retrieval score and candidate utility under MSE, macro-averaged within each dataset–system setting. Retrieval scores are oriented so that larger values correspond to higher retrieval priority.

Dataset	System	Raw-Cos	Raw-L2	DTW	Emb-Cos	Emb-L2	Stat-L2
ETTh1	C2-Cov	0.001	-0.003	-0.015	0.007	0.014	-0.015
	C2-Pre	0.011	0.041	-0.019	0.001	-0.005	-0.025
	C2-Mid	0.000	0.016	0.008	0.002	0.002	0.011
	Bolt-Mid	-0.010	-0.012	0.008	-0.023	-0.014	0.004
ETTh2	C2-Cov	0.021	-0.016	-0.006	0.001	0.023	-0.032
	C2-Pre	-0.026	0.010	-0.000	0.014	0.000	0.021
	C2-Mid	-0.011	0.022	0.027	0.002	-0.002	0.028
	Bolt-Mid	0.003	0.014	0.056	0.006	0.013	0.026
ETTm1	C2-Cov	0.004	0.002	0.007	0.006	0.005	0.004
	C2-Pre	0.051	0.037	-0.004	-0.008	-0.006	0.001
	C2-Mid	-0.007	0.007	0.001	-0.004	0.005	0.004
	Bolt-Mid	0.021	0.022	0.006	0.002	-0.004	0.001
ETTm2	C2-Cov	0.007	-0.001	-0.009	-0.013	-0.002	-0.024
	C2-Pre	0.004	-0.005	0.023	0.002	-0.008	0.013
	C2-Mid	0.004	0.005	0.017	-0.003	0.005	0.007
	Bolt-Mid	0.068	0.009	0.024	0.020	0.026	-0.004
Weather	C2-Cov	-0.005	-0.009	-0.008	-0.010	-0.006	-0.011
	C2-Pre	-0.009	0.076	0.010	0.015	0.011	0.051
	C2-Mid	0.038	-0.005	0.040	0.047	0.049	0.012
	Bolt-Mid	0.113	0.034	0.074	0.106	0.104	0.029
Exchange	C2-Cov	-0.011	0.008	-0.005	-0.006	-0.010	0.007
	C2-Pre	0.025	0.010	0.025	0.001	-0.006	0.004
	C2-Mid	-0.018	-0.006	0.051	0.072	0.100	-0.012
	Bolt-Mid	0.012	0.039	0.092	0.100	0.110	0.040

Table 11: RQ2 Best@10 under MAE. Each cell reports the best tested retriever's Best@10 / same-pool Random@10 gain (%).

Dataset	C2-Cov	C2-Pre	C2-Mid	Bolt-Mid
ETTh1	15.82 / 17.05	10.89 / 10.53	4.08 / 5.43	7.07 / 8.10
ETTh2	16.01 / 15.76	10.51 / 10.55	3.44 / 4.78	8.84 / 9.04
ETTm1	26.91 / 24.93	23.60 / 23.46	6.11 / 7.03	15.40 / 16.48
ETTm2	20.78 / 19.99	17.55 / 18.26	4.46 / 5.51	13.28 / 13.63
Weather	29.37 / 27.07	27.71 / 30.26	5.90 / 6.51	22.44 / 21.15
Exchange	15.15 / 17.78	10.86 / 16.75	11.32 / 22.95	14.22 / 22.93

D.1 Query-wise source-oracle transfer

Table 12 recomputes directional transfer by first converting each query to a percentage gain and then averaging these query-level ratios. This aggregation is included as a sensitivity diagnostic because division by query-specific baseline error can assign extreme weight to queries with very small denominators.

The denominator sensitivity is visible directly in the results: all 36 directional mean query-wise percentage gains are negative, and several settings produce extremely large magnitudes. These values motivate the aggregated-error formulation used in the main text. We therefore use the query-wise table to assess aggregation sensitivity rather than to quantify the magnitude of cross-interface transfer effects.

D.2 MAE cross-interface robustness

MAE preserves the interface-dependence pattern observed under MSE. Across the 18 dataset–interface pairs, J ranges from 19.38% to 37.22%, while $|\rho_U| \leq 0.164$ in every comparison and 17 of 18 correlations remain below 0.1. Of the 36 directional source-oracle transfers, 24 produce negative target gains. Cross-interface utility instability is therefore preserved when forecast error is measured using MAE.

Table 12: RQ3 directional source-oracle transfer using mean query-wise percentage gain. For A/B, the last two columns report $A \to B$ and $B \to A$, respectively.

Dataset	A/B	$A \to B$	$B \to A$
ETTh1	Cov / Pre	-0.76	-18.31
	Cov / Mid	-20.46	-45.54
	Pre / Mid	-15.80	-0.55
ETTh2	Cov / Pre	-21.63	-32.22
	Cov / Mid	-559.76	-151.08
	Pre / Mid	-415.07	-33.95
ETTm1	Cov / Pre	-42.48	-72.79
	Cov / Mid	-51.21	-131.03
	Pre / Mid	-40.39	-47.48
ETTm2	Cov / Pre	-1.10×10^4	-100.89
	Cov / Mid	-851.13	-618.80
	Pre / Mid	-273.43	-980.98
Weather	Cov / Pre	-5.82×10^{15}	-43.05
	Cov / Mid	-2.72×10^6	-91.20
	Pre / Mid	-7.39×10^5	-6.24×10^{15}
Exchange	Cov / Pre	-12.22	-17.35
	Cov / Mid	-475.67	-24.55
	Pre / Mid	-208.15	-15.88

Table 13: RQ3 under MAE with Chronos-2 fixed. J is the variable-balanced average of query-wise positive-utility-set Jaccard overlap (%), and ρ_U is the variable-balanced average of query-wise Spearman correlations between candidate utility rankings over the same 1000 candidates. Directional cells report transferred gain / target Oracle@1000 gain (%) using aggregated errors.

Dataset	A/B	J	ρ_U	$A \to B$	$B \to A$
ETTh1	Cov / Pre	30.52	0.047	1.58 / 20.57	-2.01 / 35.15
	Cov / Mid	24.09	0.004	-4.18 / 12.65	-9.52 / 35.15
	Pre / Mid	30.85	0.060	-1.87 / 12.65	1.31 / 20.57
ETTh2	Cov / Pre	25.90	0.027	-0.09 / 21.85	-3.38 / 34.38
	Cov / Mid	19.38	0.020	-8.97 / 17.46	-12.20 / 34.38
	Pre / Mid	22.45	0.048	-7.67 / 17.46	-3.28 / 21.85
ETTm1	Cov / Pre	31.36	0.058	1.43 / 43.84	1.44 / 46.87
	Cov / Mid	25.65	0.004	-4.40 / 17.11	-8.28 / 46.87
	Pre / Mid	28.78	0.088	-1.80 / 17.11	-1.17 / 43.84
ETTm2	Cov / Pre	28.29	0.041	0.95 / 35.18	-0.09 / 40.51
	Cov / Mid	22.26	0.016	-6.93 / 17.23	-7.54 / 40.51
	Pre / Mid	23.09	0.071	-4.03 / 17.23	-1.98 / 35.18
Weather	Cov / Pre	33.74	0.062	0.69 / 52.25	6.69 / 50.71
	Cov / Mid	22.38	0.004	-12.21 / 17.52	0.69 / 50.71
	Pre / Mid	20.73	0.060	-9.06 / 17.52	1.02 / 52.25
Exchange	Cov / Pre	37.22	0.067	3.76 / 30.85	0.86 / 42.07
	Cov / Mid	28.53	0.033	-33.28 / 36.33	-0.68 / 42.07
	Pre / Mid	29.63	0.164	-12.93 / 36.33	4.90 / 30.85