SIMILARITY IS NOT FUNCTIONAL ALIGNMENT: WHAT TIME-SERIES–LANGUAGE ALIGNMENT MUST ESTABLISH

Anonymous authors
Paper under double-blind review

ABSTRACT

Time-series–language alignment is commonly evaluated through retrieval and representational similarity. These scores measure correspondence between paired series and descriptions; they do not by themselves certify whether a statement is supported by the observed history, adds new conditional-mean information about the future, or improves a fitted predictor. We formalize these distinctions as Functional Alignment (FA), which separates representation correspondence, historical support, predictive innovation, and fitted-model assistance. Our central result is an identification limit: with the history–future process held fixed, two ways of selecting true historical statements can induce the same history–text distribution and the same support labels, yet different predictive innovation. No correspondence measure of fixed encoders can therefore tell the two apart, and a history-derived feature with zero innovation can still assist a restricted predictor. Across 20 encoder configurations, retrieval reaches 51.8 to 156.4 times chance, yet the strongest retriever has temporal factual-reversal accuracy of 0.4984. Executable Claim Alignment (ECA), which checks typed interpretations against observations, reaches domain-macro accuracy 0.9874 in controlled verification across 11 held-out CaTS domains. On Time-MMD, task-specific projection training nearly doubles Linear CKA, yet its forecasting increments over matched untrained projections range from -3.53% to 5.67% across six backbones. Functional claims about alignment require evidence matched to the target being claimed.

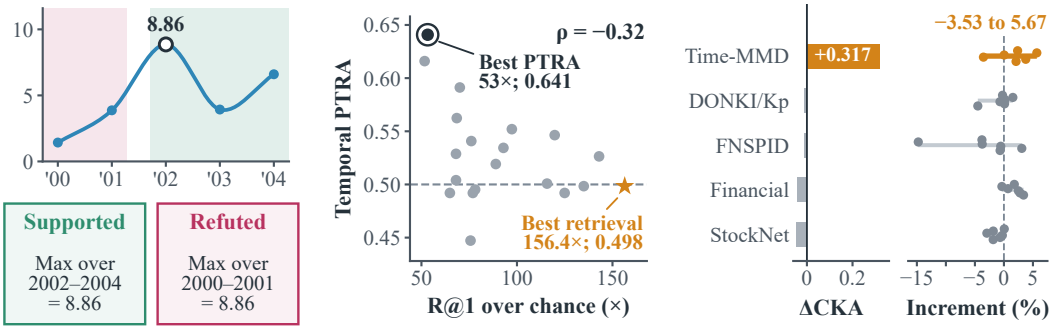
1 INTRODUCTION

Time-series–language alignment promises description, factual verification, and forecasting with textual context. Following image–text contrastive learning (Radford et al., 2021), series and text encoders are trained on paired corpora (Liu et al., 2024a; Chen et al., 2025). Retrieval, CKA (Kornblith et al., 2019), and mutual-kNN overlap (Huh et al., 2024) measure their correspondence, which recent work examines across encoder scales and modalities (Yashwante & Yu, 2026). Similarity measures correspondence; it does not by itself certify whether a statement is supported by the observed history, adds new conditional-mean information about the future, or improves a fitted predictor. We argue that representation correspondence is insufficient as a general criterion of functional alignment, and that some of these functional claims cannot be settled by correspondence evidence at all.

First, numerical and temporal claims are checkable. In Figure 1(a), "the maximum value from 2002 to 2004 is 8.86" is true; changing the interval to "2000 to 2001" makes it false. Paired contrastive training rewards matching inputs without explicitly supervising truth under such minimal edits. Vision–language studies likewise distinguish retrieval performance from compositional understanding (Yuksekgonul et al., 2023). For numerical and temporal claims, checking the stated commitments against observations directly tests whether a matching score respects factual support.

Second, the predictive value of language depends on how statements are selected. Two statements can both accurately describe the same history, yet choosing between them can reveal an external signal related to the future. What a statement says about the past does not determine what its selection reveals about the future. Conversely, a statement generated as a fixed function of the history adds no new information, but can help a restricted forecaster use information already present. Ex-

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.



(a) Temporal edit flips support. **(b)** Retrieval ≠ factual ranking. **(c)** Geometry ≠ forecasting gain.

Figure 1: **Correspondence does not certify functional success.** **(a)** A minimal temporal edit reverses support. **(b)** The strongest of 20 retrievers has temporal PTRA 0.4984 (chance: 0.5). **(c)** Time-MMD: Linear CKA rises by 0.317, but forecasting increments range from -3.53% to 5.67% across six backbones.

ternal context can also inform the future without being verifiable from the history (Williams et al., 2025). These possibilities distinguish factual correctness, new predictive information, and the gains a particular model can realize.

We formalize what alignment must establish as Functional Alignment (FA), separating four targets: representation correspondence of the encoders, historical support of a statement by its history, predictive innovation in the future's conditional mean beyond the history, and fitted-model assistance to a trained forecaster (Section 3). The separation rests on an identification result. We hold the history–future process fixed and change only which of two true historical statements is reported. An independent selector and a future-relevant selector induce the same population history–text joint distribution and support status, but different predictive innovation (Proposition 1). Some cross-target implications are therefore not identified in general: for claims about new conditional-mean information, even complete knowledge of the paired distribution is insufficient, and no improvement of a correspondence metric can resolve the question.

FA turns alignment evaluation into three cross-target questions, each with its own test. RQ1 asks whether correspondence tracks historical support. Minimal factual reversals test the matching score's sensitivity to support, and Executable Claim Alignment (ECA) supplies a direct support assessment by checking typed candidate interpretations against observations, in controlled CaTS verification (Zhou et al., 2026) and on naturally occurring TSVer claims (Strong & Vlachos, 2025). RQ2 asks whether support determines innovation and assistance; controlled processes make innovation analytic. RQ3 asks whether stronger correspondence implies greater forecasting assistance; comparisons against matched untrained projection heads distinguish text gains from training increments.

Our first contribution is a target decomposition with an identification limit (Section 3, RQ2): historical support and the paired history–text law do not determine predictive innovation, since two supported selection mechanisms have analytic innovations of 0 and 1.5625 despite identical history–text distributions. A separate construction supplies a redundant feature at zero innovation that reduces a restricted predictor's test MSE by 0.0894, so fitted assistance can arise without new information.

The second contribution shows that correspondence and factual support require separate tests (RQ1). Across 20 configurations, paired-caption retrieval reaches 51.8 to 156.4 times chance, yet the strongest retriever has temporal Pairwise Truth-Reversal Accuracy (PTRA) of 0.4984, near the 0.50 chance reference (Figure 1(b)). In a 21-configuration analysis including a split-controlled TVL-style reference, none of the 18 tested associations between nine correspondence measures and two reversal targets survives joint correction. Used as a direct support instrument, ECA recovers controlled support with domain-macro accuracy 0.9874.

The third contribution separates text gains from representation-state increments (RQ3). On Time-MMD, task-specific training raises Linear CKA from 0.34 to 0.66, while the six corresponding increments range from -3.53% to 5.67% (Figure 1(c)). Across all 30 cells of five tasks and six

backbones, three increments survive correction, two negative and one positive. A descriptive fixed-grid synthesis gives -0.35% (95% CI $[-1.11, 0.40]\%$) with $I^2 = 66.9\%$, that is, little additional fitted utility and substantial heterogeneity instead of a common direction.

2 Related Work

Representation correspondence. Contrastive learning aligns paired inputs in a shared space, as exemplified by CLIP (Radford et al., 2021) and language-supervised time-series classification in TS-CLIP (Chen et al., 2025). CKA measures representational similarity (Kornblith et al., 2019), while the Platonic Representation Hypothesis links model scaling to convergence in representational structure (Huh et al., 2024). TVL (Yashwante & Yu, 2026) studies alignment among time series, visual plots, and text, distinguishing global geometry from local correspondence. Ding et al. (2021) assess whether representation similarity measures track changes in model behavior. Vision–language studies also distinguish compositional understanding from retrieval performance (Yuksekgonul et al., 2023) and expose biases in negative-caption benchmarks (Hsieh et al., 2023). Our focus is which factual and predictive claims can be inferred from cross-modal correspondence.

Executable semantics and factual verification. Truth-conditional captioning grounds descriptions in programs evaluated on time series (Jhamtani & Berg-Kirkpatrick, 2021), while Mori et al. (2022) translate temporal claims into queries for verification. CaTS-Bench evaluates captioning and numerical reasoning (Zhou et al., 2026); TSVer supplies real-world claims with annotated time-series evidence and verdicts (Strong & Vlachos, 2025). ECA checks typed candidate interpretations against observations to assess historical support: what a statement says about the observed history, not what its selection reveals about the future.

Language-augmented forecasting. Time-MMD pairs numerical histories with textual context (Liu et al., 2024a), while Context is Key constructs forecasting tasks that require textual information (Williams et al., 2025). Zhang et al. (2025a) study how forecasting gains depend on model capacity, data, and complementary text; Zheng et al. (2026) generate semi-synthetic contexts with verifiable information beyond numerical histories. These studies identify when text improves forecasting; we ask what such gains establish about alignment by measuring correspondence and forecasting on the same representation states.

Predictive information and fitted-model utility. Algorithm-agnostic variable importance compares population-optimal predictiveness with and without a feature (Williamson et al., 2023), whereas model-class reliance examines how feature reliance varies across well-performing predictors (Fisher et al., 2019). FA connects these population- and model-level views to historical support and representation correspondence: the reported true statement can change predictive innovation at a fixed history–text law and support, and a history-derived feature can help a restricted predictor at zero innovation. These results distinguish what language states, what its selection reveals, and what a fitted model can use.

3 Functional Alignment

Functional Alignment (FA) decomposes the evidence needed for claims beyond representation correspondence, rather than prescribing a single metric or requiring every application to establish all four targets. We separate the objects behind these claims (Table 1), show that even the full history–text distribution can leave predictive innovation undetermined (Proposition 1), and specify the tests used for RQ1–RQ3. Let X denote numerical information available at the forecast origin, S an accompanying statement, and Y the future target.

3.1 Four Targets

Representation correspondence. Correspondence describes structure shared by encoded modalities across a population of pairs. Contrastive training rewards ranking paired inputs above in-batch negatives. Retrieval, global geometry (CKA), and local neighborhood agreement (mutual-kNN; Appendix A.3) measure aspects of this structure; they do not assign truth labels to statements.

Table 1: **Four targets of Functional Alignment.** Each is a property of a different object and needs its own evidence; the contrastive objectives studied here optimize only correspondence.

Target	Property of	Question	Direct evidence	Tested in
Correspondence	Encoder pair	Do paired inputs match?	Retrieval, CKA, mutual-kNN	§4.1, §4.3
Historical support	Statement–history pair	Is s true of x?	Checks against observations	§4.1
Predictive innovation	Law of (X, S, Y)	Does S shift $\mathbb{E}[Y \mid X]$?	Analytic U_2 in known processes	§4.2
Fitted-model assistance	Fitted predictor	Does $\widehat{f}$ gain from S?	Matched risk contrasts	§4.2, §4.3

Historical support. For a realized pair (s, x), $G(s, x)$ is SUPPORTED, REFUTED, or UNRESOLVED, according to whether the supplied observations establish, contradict, or leave undetermined the claim's numerical and temporal commitments. A single literal or interval can reverse support on the same history. Claims requiring unobserved future or external evidence remain UNRESOLVED, even when informative. We test support sensitivity by asking whether a matching score ranks a supported statement above its minimally edited refuted counterpart; this evaluates that readout, not every possible decoder of the embeddings.

Predictive innovation. Innovation concerns what S adds about Y once X is known. Under squared loss and $\mathbb{E}\|Y\|_2^2 < \infty$, define

$$U_2(S; Y \mid X) = \mathbb{E}\Big[\|\mathbb{E}[Y \mid X, S] - \mathbb{E}[Y \mid X]\|_2^2\Big]. \tag{1}$$

This equals the reduction in Bayes squared-error risk from adding S to X; it captures conditional-mean information, not all conditional dependence. If $S = h(X)$ for fixed measurable h, then $U_2 = 0$. Historical support does not imply this deterministic relation: the choice among several true statements can depend on information absent from X.

Fitted-model assistance. Experiments observe fitted predictors rather than Bayes-optimal ones. For input V, let $R(\widehat{f}_V) = \mathbb{E}\|Y - \widehat{f}_V(V)\|_2^2$, with Bayes risk $R^*(V)$ and excess risk $\delta_V = R(\widehat{f}_V) - R^*(V) \geq 0$. On the same evaluation population,

$$A_{\text{fit}} = R(\widehat{f}_X) - R(\widehat{f}_{X,S}) = U_2(S; Y \mid X) + \delta_X - \delta_{X,S}. \tag{2}$$

Relative to these raw-input Bayes risks, excess risk includes representation loss as well as approximation, estimation, and optimization effects. Thus redundant information can help a restricted predictor at $U_2 = 0$, while small fitted gains need not imply small innovation. Test-risk contrasts estimate assistance for the fitted predictors being compared.

3.2 SUPPORT AND PAIRED CORRESPONDENCE DO NOT IDENTIFY INNOVATION

The limit concerns which true statement is reported. Let Z, W be independent uniform signs, independent of X, and keep the process $Y = m(X) + \beta Z + \epsilon$ fixed, with independent zero-mean noise and $\beta \neq 0$. For each history x, let $s(x, -1) \neq s(x, +1)$ be two true statements, for example reports of the observed values of channels 1 and 2.

Proposition 1 (Non-identification of innovation)**.** *Let $S_0 = s(X, W)$ and $S_1 = s(X, Z)$. Then $P(X, S_0) = P(X, S_1)$, both mechanisms share $P(X, Y)$, $G(S_j, X) =$ SUPPORTED almost surely, and $U_2(S_0; Y \mid X) = 0$ while $U_2(S_1; Y \mid X) = \beta^2$. Hence every quantity determined by the paired law and the support labels, including every correspondence measure of fixed encoders, is identical under the two mechanisms.*

Proof sketch. Conditional on $X = x$, both mechanisms select each statement with probability $1/2$, so the paired laws coincide. Because W is independent of (Z, ϵ), $\mathbb{E}[Y \mid X, S_0] = m(X)$; the channel identity of S_1 reveals Z, so $\mathbb{E}[Y \mid X, S_1] = m(X) + \beta Z$. Substituting into Equation 1 gives the two values. □

The argument uses selection, not support status, and applies unchanged when both statements are UNRESOLVED. What a statement says about the past does not determine what its selection reveals about the future. Conversely, by Equation 2, a history-derived statement $S = h(X)$ has $U_2 = 0$

yet can yield $A_{\text{fit}} = \delta_X - \delta_{X,S} > 0$ for a restricted predictor. Section 4.2 instantiates both cases. U_2 is identified once outcomes are observed together with statements; what the paired law lacks is outcome-linked evidence, not a better similarity measure.

3.3 From Targets to Tests

RQ1: from matching to support. In-batch negatives need not contradict the paired caption, so their ranking objective does not directly supervise minimal factual distinctions. We hold the history and most wording fixed, reverse a numerical or temporal commitment, and test the ranking (Section 4.1).

Executable support assessment. ECA checks claims through up to 20 typed candidate programs: it binds candidates to observations, executes them, and aggregates their verdicts using a learned scorer. Candidate-level binding and execution traces expose which observations were checked; missing executable evidence or abstention can yield UNRESOLVED. This operationalizes historical support within the candidate language (Appendix C.1). Section 4.1 and Appendix C give the evidence and calibration settings.

RQ2: from support to innovation. Proposition 1 is an identification result, so benchmark data alone cannot test it. We construct processes in which U_2 and A_{fit} are analytic, vary how a supported statement is selected or which predictor receives it, and compare support status with both quantities (Section 4.2).

RQ3: from representation training to assistance. We measure correspondence and forecasting with the same frozen encoders and compare untrained with trained projection heads. State-specific numerical projections and separately fitted adapters require distinguishing within-state text gain from the additional text-related risk reduction over the untrained reference (Section 4.3). A separate interface adds an ECA summary $\phi(X, S)$ while retaining the original text input. Conditional on its frozen parameters, ϕ adds no innovation beyond raw (X, S); at the encoded interface, it can still expose information discarded by the encoders or make retained information easier to use.

4 Experiments

We answer RQ1–RQ3 in turn. Data and models are summarized below; full protocols are in Appendix A.

Data. Support tests use CaTS series–caption pairs (Zhou et al., 2026) and real TSVer claims with time-series evidence (Strong & Vlachos, 2025). Forecasting uses task-specific preparations of Time-MMD (Liu et al., 2024a), MTBench Financial (Chen et al., 2026), StockNet AAPL (Xu & Cohen, 2018), FNSPID A (Dong et al., 2024), and NASA DONKI notifications paired with GFZ Kp (NASA Community Coordinated Modeling Center, n.d.; Matzka et al., 2021).

Models. We pair T5 (Raffel et al., 2020), E5 (Wang et al., 2024a;b), and Qwen3-Embedding (Yang et al., 2025; Zhang et al., 2025b) with MOMENT, Chronos, and TimesFM in 20 configurations. Encoders are frozen; RQ1 projection heads use five seeds. Forecasting uses iTransformer (Liu et al., 2024b), TimeMixer (Wang et al., 2024c), PatchTST (Nie et al., 2023), MOMENT (Goswami et al., 2024), TimesFM (Das et al., 2024), and Chronos (Ansari et al., 2024). All forecasting conditions share evaluation samples, use validation-only checkpoint selection, and average three fitted seeds.

4.1 RQ1: Does Correspondence Track Historical Support?

From CaTS we construct 250 numerical and 250 temporal reversal pairs, each pairing a SUPPORTED statement with a minimally edited REFUTED alternative for the same series: numerical pairs change the stated maximum, and temporal pairs change the queried interval (Appendix B). Pairwise Truth-Reversal Accuracy (PTRA) is the fraction of pairs for which the series–text similarity ranks the SUPPORTED statement higher; ties count as errors, and chance is 0.5. Correspondence is measured by series-to-caption retrieval among the 4,005 CaTS test captions (chance R@1 $= 1/4005 \approx 0.00025$).

Table 2: **Correspondence and factual discrimination across 20 encoder configurations.** Mean ± sample SD over five projection seeds. R@1: series-to-caption retrieval among 4,005 captions (chance $1/4005 \approx 0.00025$). Pair cos.: mean cosine between the supported and refuted texts of a pair. PTRA: supported-over-refuted ranking accuracy over 250 pairs per reversal type (chance 0.50). Bold marks the highest R@1 and temporal PTRA.

		Numerical reversal		Temporal reversal	
Configuration	**R@1 ↑**	**Pair cos.**	**PTRA ↑**	**Pair cos.**	**PTRA ↑**
T5 family					
T5-small + MOMENT-base	0.0162 ± 0.0016	0.8770	0.5656 ± 0.0284	0.9209	0.4920 ± 0.0165
T5-small + MOMENT-large	0.0192 ± 0.0017	0.8849	0.5520 ± 0.0139	0.9294	0.4920 ± 0.0345
T5-small + Chronos-base	0.0312 ± 0.0011	0.9044	0.5392 ± 0.0292	0.9417	0.4920 ± 0.0194
T5-small + Chronos-large	0.0337 ± 0.0026	0.9063	0.5576 ± 0.0137	0.9482	0.4984 ± 0.0252
T5-3B + MOMENT-base	0.0190 ± 0.0026	0.8998	0.4680 ± 0.0235	0.8664	0.5408 ± 0.0232
T5-3B + TimesFM-500M	0.0357 ± 0.0028	0.8952	0.4856 ± 0.0217	0.8623	0.5264 ± 0.0248
E5 family					
E5-base + MOMENT-base	0.0171 ± 0.0014	0.8986	0.5104 ± 0.0143	0.8634	0.5624 ± 0.0448
E5-base + MOMENT-large	0.0170 ± 0.0020	0.9028	0.5000 ± 0.0160	0.8725	0.5040 ± 0.0511
E5-base + Chronos-base	0.0289 ± 0.0018	0.9191	0.5544 ± 0.0288	0.8978	0.5008 ± 0.0323
E5-base + Chronos-large	0.0299 ± 0.0020	0.9240	0.5464 ± 0.0317	0.9117	0.5464 ± 0.0626
E5-Mistral-7B + MOMENT-base	0.0222 ± 0.0003	0.7328	0.5064 ± 0.0143	0.8942	0.5192 ± 0.0052
Qwen family					
Qwen-0.6B + MOMENT-base	0.0133 ± 0.0004	0.8362	0.5552 ± 0.0211	0.7638	$\mathbf{0.6408} \pm 0.0276$
Qwen-0.6B + MOMENT-large	0.0129 ± 0.0012	0.8554	0.5312 ± 0.0242	0.7998	0.6160 ± 0.0179
Qwen-0.6B + Chronos-base	0.0243 ± 0.0014	0.9126	0.5960 ± 0.0305	0.8839	0.5520 ± 0.0427
Qwen-0.6B + Chronos-large	0.0232 ± 0.0026	0.9119	0.5808 ± 0.0175	0.8886	0.5344 ± 0.0477
Qwen-0.6B + TimesFM-500M	0.0175 ± 0.0013	0.8745	0.5904 ± 0.0205	0.8452	0.5912 ± 0.0484
Qwen-4B + MOMENT-base	0.0195 ± 0.0019	0.8542	0.5704 ± 0.0265	0.7909	0.4952 ± 0.0509
Qwen-4B + MOMENT-large	0.0170 ± 0.0003	0.8583	0.5664 ± 0.0243	0.7947	0.5288 ± 0.0286
Qwen-4B + Chronos-large	$\mathbf{0.0391} \pm 0.0018$	0.8537	0.5912 ± 0.0107	0.7103	0.4984 ± 0.0286
Qwen-8B + MOMENT-base	0.0189 ± 0.0019	0.7832	0.4888 ± 0.0033	0.8254	0.4472 ± 0.0173

Above-chance retrieval can coexist with weak factual discrimination. All 20 configurations retrieve above chance, with R@1 from 0.0129 to 0.0391 (51.8–156.4 times chance, calculated before rounding), yet the retrieval ordering does not carry over to factual discrimination (Table 2). The strongest retriever, Qwen-4B + Chronos-large, is near chance on temporal reversals (PTRA 0.4984), whereas Qwen-0.6B + MOMENT-base reaches 0.6408 with one of the lowest R@1 values (Figure 1(b)). Temporal PTRA spans 0.4472–0.6408 and numerical PTRA 0.4680–0.5960. Configuration-mean positive–negative text-pair cosines of 0.71–0.95 also show that text representations can remain close after truth reverses.

The mismatch persists across metrics and a capacity control. We compare nine measures (R@1, R@5, R@10, MRR, cosine margin, Procrustes disparity, kernel and debiased CKA, and mutual-kNN) with both PTRA targets over 21 configuration means, including a split-controlled TVL-style reference (Yashwante & Yu, 2026). None of the 18 tested associations survives joint Benjamini–Hochberg correction; resampling source groups over the 20 main configurations, the R@1–temporal PTRA correlation is negative, $\rho = -0.3269$ $[-0.5416, -0.1114]$ (Appendix B.2). In a capacity control for Qwen-4B + Chronos-large, a wider head raises R@1 from 0.0326 to 0.0400 while temporal PTRA remains near chance (0.4987 and 0.4960; Appendix B.4).

Factual sensitivity depends on the tested setting. On constrained transformations of 246 official TRUCE test series, whose repeated templates make every contrast oracle-checkable, three representation settings reach PTRA between 0.9350 and 1.0000 (Appendix B.4). In the CaTS description-sensitivity probe, varying density and explicitness does not raise PTRA above 0.57 (Appendix B.3). Representations can therefore separate factual edits in favorable settings, but strong correspondence alone does not establish this ability on the CaTS reversals.

Executable checks recover controlled support. Controlled CaTS supplies the correct series for 8,010 Supported/Refuted claims per scoring seed in 11 held-out domains. ECA achieves domain-macro accuracy 0.9874 (domain range: 0.9505–1.0000); original three-output matching and generic

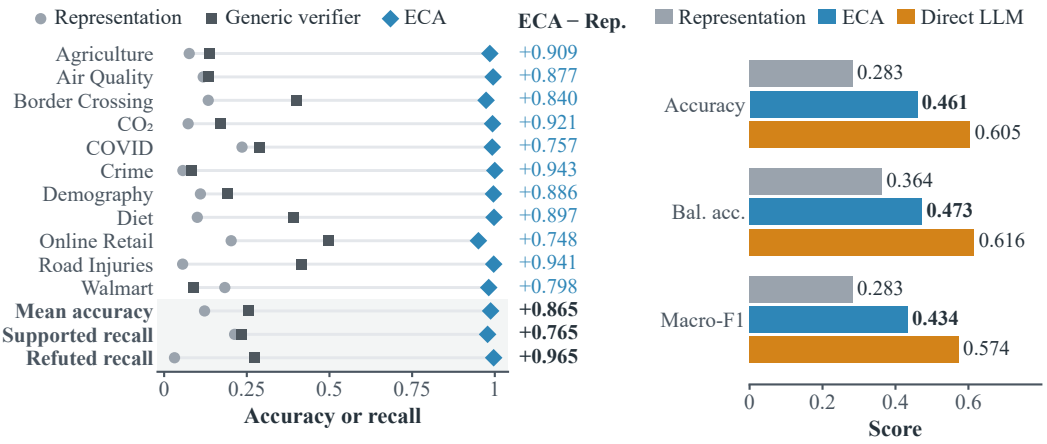


Figure 2: **Direct assessment of historical support. Left:** Known-X verification across 11 held-out CaTS domains; original verifiers allow UNRESOLVED outputs on binary gold labels. **Right:** Retrospective comparison on 304 Native TSVer claims. Higher is better; diagnostics are in the text and Appendix C.

neural verifiers score 0.1222 and 0.2538 (Figure 2, left). Their UNRESOLVED outputs count as errors on binary gold labels. A separate binary diagnostic uses frozen representations and inner-development selection: matching reaches 0.4973 and a linear head 0.6871. Thus a learned readout extracts support information missed by similarity, while remaining below controlled ECA (Table 7).

Native claims expose a harder verification problem. Native TSVer contains 304 claims across 43 source groups and includes UNRESOLVED. With source-disjoint scorer folds and historically reused calibration, ECA reaches Macro-F1 0.4337, compared with 0.2833 for matching and 0.5740 for the direct LLM verifier (Figure 2, right). The paired ECA–matching difference is 0.1505 with 95% source-group interval $[0.0700, 0.2437]$, and the ECA–LLM difference is -0.1403 $[-0.2493, -0.0361]$. This is a retrospective known-source system comparison: matching's features are constructed from whole series, the LLM requests contain claim-localized evidence, and ECA's reused calibration is not nested within the scorer folds. A label-conditioned candidate oracle reaches Macro-F1 0.8951, locating much of ECA's error in interpretation selection (Appendix C.3).

4.2 RQ2: DOES SUPPORT DETERMINE PREDICTIVE INNOVATION AND ASSISTANCE?

Controlled processes make innovation known while keeping the prediction task fixed. We assign support by construction and compute U_2 analytically, then vary statement selection, feature accessibility, or the replacement rule (Table 3).

Statement selection changes innovation at fixed paired law. We instantiate Proposition 1 with $X \sim \mathcal{N}(0, I_3)$, $m(X) = 0.7X_1 - 0.4X_2 + 0.2X_3$, $\beta = 1.25$, and independent noise with standard deviation 0.5. All four support–selector cells share the same numerical X/Y draws within each realization. Supported statements report an observed value from the channel chosen by $Q = W$ or $Q = Z$; Unresolved statements report an external state absent from the history. Within either support condition, switching selectors preserves $P(X, S)$ and changes U_2 from 0 to 1.5625 (Table 3(a)). Affine ridge predictors receive X or (X, Q), using 600 training and 2,500 test samples in each of 200 realizations. Their mean gains are -0.0032 and 1.5701, respectively, under either

Table 3: **Controlled functional separations.** U_2: analytic innovation; A_{fit}: test-MSE reduction from S (Equation 2). Means of 200 paired realizations; intervals in Appendix D.

Setting	Variant	U_2	A_{fit}
(a) Selection mechanism, either support status			
Selector	independent W	0.0000	-0.0032
	future-relevant Z	1.5625	1.5701
(b) Redundant feature $S = X^2$			
Restricted	$[x] \to [x, s]$	0	0.0894
Expressive	$[x, x^2, x^3] \to [\cdot, s]$	0	≈ 0

support condition. This finite-model check uses the structured selector; it does not assume that a language encoder has learned to extract it.

Assistance can arise without innovation. The second construction fixes $S = X^2$ and $Y = X^2 + \epsilon$, so $U_2 = 0$. Adding S reduces test MSE by 0.0894 for an affine predictor receiving only x, but has approximately zero effect when the predictor already receives $[x, x^2, x^3]$ (Table 3(b)). The gain is entirely $\delta_X - \delta_{X,S}$ in Equation 2: an explicit feature makes existing information accessible.

Replacement contrasts depend on the intervention. A donor contrast tests the predictive association removed by its replacement rule. At fixed $U_2 = 1.44$, training on unconditionally replaced statements raises test risk by 1.4291, whereas replacement within strata of the future-relevant variable changes it by -0.0011; both refitted models are evaluated on the original test statements. Within-stratum replacement preserves the predictive association, so its near-zero contrast does not identify zero innovation (Appendix D). Real-data time-legal donor contrasts likewise characterize their specified replacement protocol rather than a universal innovation null.

4.3 RQ3: Does Stronger Correspondence Imply Greater Forecasting Assistance?

Matched representation states. We use the same frozen T5-small/MOMENT-base encoders for correspondence and forecasting, in four states. RAW normalizes the frozen embeddings; RANDOM applies untrained projection heads; CATS uses heads trained on CaTS; TASK trains both heads on the current task's historical text–series pairs, excluding future targets. TASK starts from the same initialization as RANDOM, so RANDOM is the matched untrained reference. Within each state, adapters of identical architecture are fitted separately on common recipients with numerical-only (zero text branch), paired-text, or time-legal donor-text inputs. Projection and adapter seeds are paired; intervals condition on the fitted runs, and DONKI/Kp contributes six test clusters.

Text gain and training increment. Let $R_{r,N}$ and $R_{r,T}$ be the seed-averaged test risks of state r without and with paired text. The text gain and the training increment over RANDOM are

$$G_r = \frac{R_{r,N} - R_{r,T}}{R_{r,N}}, \qquad \Delta G_r = \frac{(R_{r,N} - R_{r,T}) - (R_{\mathrm{RANDOM},N} - R_{\mathrm{RANDOM},T})}{R_{\mathrm{RANDOM},N}}.$$

Tables report $100G_r$ and $100\Delta G_r$. The former is normalized fitted assistance within state r (Equation 2); the latter compares absolute assistance with RANDOM under a common normalization, not the difference of two text-gain percentages. Because TASK changes both projections and each state has its own validation-selected adapter, the numerical-only references differ across states; ΔG_r compares text-related risk reductions and does not isolate text-branch training.

Text gains and TASK–RANDOM increments are distinct. On Time-MMD, TASK text gains are positive for five of six backbones and pass correction for three (Table 4). Across all five tasks, three of 30 TASK increments pass the 30-cell BH correction: FNSPID/TimeMixer (-14.76%), FNSPID/PatchTST (-3.74%), and DONKI/Kp/TimeMixer (1.54%). No CATS increment passes its correction (Table 11, Appendix E.1). FNSPID/TimeMixer illustrates the attribution issue: RANDOM gains 15.45% from text and TASK 0.69%, so the TASK increment is negative.

Correspondence changes and forecasting increments. On Time-MMD, TASK raises Linear CKA from 0.340 under RANDOM to 0.658, while the six estimated forecasting increments range from -3.53% to 5.67%; none passes correction (Table 4; Figure 1(c)). On the other four tasks, TASK lowers CKA relative to RANDOM (Appendix E.2), while increment point estimates again differ in sign across backbones (Table 4). Because each task has one CKA change shared by six backbones, mixed point-estimate signs alone do not establish backbone-dependent effects.

Fixed-grid synthesis. Using the paired cluster-bootstrap uncertainty of each cell, a descriptive fixed-grid synthesis (random effects) of the 30 TASK increments is -0.353% (95% CI $[-1.108, 0.401]\%$), with $I^2 = 66.9\%$ and prediction interval $[-3.177, 2.470]\%$. The pooled interval covers zero and excludes grid-average increments above 0.401%; the prediction interval spans both signs. Because cells share datasets, recipients, and representation states, this synthesis describes

Table 4: **Text gain and TASK training increment (%).** RANDOM/TASK gains and TASK increments use seed-averaged risks and state-specific numerical references. Bold: $q < 0.05$ within the corresponding 30-cell BH family. RAW/CATS controls: Table 11.

Backbone	Measure	Time-MMD	Financial	StockNet	FNSPID	DONKI/Kp
iTransformer	RANDOM gain	2.59	−3.14	−1.09	1.61	0.33
	TASK gain	**4.93**	−1.35	−1.02	−2.16	0.09
	TASK increment	2.36	1.79	0.06	−3.79	−0.24
TimeMixer	RANDOM gain	7.91	0.65	−2.42	**15.45**	−0.19
	TASK gain	13.50	0.30	−4.26	0.69	1.35
	TASK increment	5.67	−0.35	−1.84	**−14.76**	**1.54**
PatchTST	RANDOM gain	6.72	−1.57	0.55	3.80	**0.98**
	TASK gain	**6.99**	−0.87	−2.37	0.06	1.13
	TASK increment	0.04	0.70	−2.94	**−3.74**	0.16
MOMENT	RANDOM gain	−0.43	−1.52	−1.02	−2.77	8.84
	TASK gain	−3.90	0.97	−1.28	−3.30	8.66
	TASK increment	−3.53	2.49	−0.27	−0.55	−0.68
TimesFM	RANDOM gain	6.16	−1.17	3.14	−10.51	**9.22**
	TASK gain	**9.67**	1.60	2.49	−7.42	**9.56**
	TASK increment	3.76	2.77	−0.66	3.08	0.11
Chronos	RANDOM gain	6.04	−1.73	0.57	−0.01	4.36
	TASK gain	7.86	1.64	−1.26	−0.67	−0.14
	TASK increment	2.09	3.38	−1.85	−0.66	−4.50

heterogeneity on the fixed grid; it is not an independent-study meta-analysis. Appendix E.2 reports the corresponding CATS summary and a component-sensitivity diagnostic.

Text gain and training increment answer different questions. For Time-MMD/PatchTST, TASK gains 6.99% from text with a paired 95% interval $[3.30, 10.74]\%$, and RANDOM gains 6.72%. Their absolute MSE reductions, 0.018228 and 0.018129 (Table 10), yield a training increment of 0.04%. The text gain is positive, whereas the increment's 95% interval, $[-6.73, 14.51]\%$, leaves its sign unresolved.

Executable-summary diagnostics. Deployed summaries add little assistance: with the Base interface, 90% effect intervals remain within the retrospective $\pm 1\%$ band when evaluation is restricted to recipients with candidates. A separate positive control supplies zero-padded future targets to the same PatchTST interface, reducing MSE by 85.35% on Time-MMD and 85.71% on Financial. This non-deployable control shows that the interface can use strong auxiliary signals. Appendix F gives further diagnostics.

5 Conclusion

Representation correspondence is insufficient as a general criterion of time-series–language alignment success. Proposition 1 makes the limitation explicit: two mechanisms can share the history–future process, the history–text distribution, and historical support, yet differ in predictive innovation because they select different true statements. The distinction is between what a statement asserts and what its selection reveals. Empirically, above-chance retrieval coexists with weak factual-reversal ranking, executable checks recover support in a controlled setting, and forecasting increments over matched untrained projections differ in sign across the fixed task–backbone grid.

The forecasting bridge uses one encoder pair, and training with factual contrasts and verified claims would test whether correspondence can become informative about function (Appendix G). Similarity remains a useful measure of correspondence; historical support calls for direct checks against observations, innovation for an identified information comparison, and fitted utility for matched risk contrasts. Functional claims require evidence matched to the target being claimed.

Ethics Statement

This work uses only publicly released data sets under their original terms of use and involves no human subjects or new human annotation. We are not aware of ethical concerns specific to this work.

Reproducibility Statement

The supplementary code contains the core implementation used in RQ1 and RQ3, including frozen encoder wrappers, projection heads and alignment metrics, the typed claim executor, residual forecasting adapters, and the paired cluster bootstrap with Benjamini–Hochberg correction. These are the actual implementations used in the reported experiments, rather than illustrative or pseudocode implementations. The encoder caches, candidate programs, and prediction files consumed by these components are large and are not included, and no trained weights are released. All datasets are publicly available; sources, splits, seeds, and hyperparameters are given in Appendix A and the experiment-specific appendices. The full codebase will be released upon publication.

AI Use Statement

We used large language models to polish the text and figures, to help design additional experiments, including checking whether control experiments suit the claims they test, and to assist with parts of the mathematical derivations. The authors verified all derivations, results, and text, and take full responsibility for the content of the paper. Language models used as experimental components are described in Section 4 and Appendices C and F.

References

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. Chronos: Learning the language of time series, 2024. URL https://arxiv.org/abs/2403.07815.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300, 01 1995. ISSN 0035-9246. doi: 10.1111/j.2517-6161.1995.tb02031.x. URL https://doi.org/10.1111/j.2517-6161.1995.tb02031.x.

Jialin Chen, Aosong Feng, Ziyu Zhao, Juan Garza, Gaukhar Nurbek, Cheng Qin, Ali Maatouk, Leandros Tassiulas, Yifeng Gao, and Rex Ying. MTBench: A multimodal time series benchmark for temporal reasoning and question answering, 2026. URL https://arxiv.org/abs/2503.16858v2. Version 2.

Ziwen Chen, Xiaoyuan Zhang, and Ming Zhu. TS-CLIP: Time series understanding by CLIP. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 4646–4664, Suzhou, China, nov 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.231. URL https://aclanthology.org/2025.emnlp-main.231/.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10148–10167. PMLR, 2024. URL https://proceedings.mlr.press/v235/das24c.html.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, Fucong Dai, Guangbo Hao, Guanting Chen, Guoai Cao, Guolai Meng, Guowei Li, Han Yu, Han Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoling Zhang,

Haoming Luo, Haoran Wei, Haotian Yuan, Haowei Zhang, Haowen Luo, Haoyu Chen, Haozhe Ji, Hengqing Zhang, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, J Yang, JQ Zhu, Jia Luo, Jia Song, Jia Yu, Jialiang Huang, Jialu Cai, Jian Liang, Jiangting Zhou, Jiasheng Ye, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jieyu Yang, Jin Chen, Jin Yan, Jingchang Chen, Jingli Zhou, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jingzi Zhou, Jinhua Zhu, Jiping Yu, Joseph Sun, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junmin Zheng, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Leyi Xia, Li Zhang, Liang Zhao, Lihua Guo, Lingxiao Luo, Linwang Ma, Linyan Zhu, Litong Wang, Liyu Cai, Liyue Zhang, Longhao Chen, MS Di, MY Xu, Max Mei, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Mingxu Zhou, Minmin Han, Ning Wang, Panpan Huang, Panpan Wang, Peixin Cong, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Qiwei Jiang, Rui Tian, Ruifan Xu, Ruijie Lu, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqian Chen, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, Ruyi Chen, SH Liu, Shanghao Lu, Shangmian Sun, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoheng Nie, Shaoqing Wu, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Shuying Yu, Songyang Zhou, Tao Ni, Tao Yun, Tian Jin, Tian Pei, Tian Ye, Tianle Lin, Tianran Ji, Tianyi Cui, Tianyuan Yue, Tingting Yu, Tun Wang, W Zhang, WL Xiao, Wangding Zeng, Wei An, Weilin Zhao, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjing Yao, Wenjun Gao, Wenkai Yang, Wenlve Huang, Wenqing Hou, Wentao Zhang, Wenting Ma, Xi Gao, Xiang He, Xiangwen Wang, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingchen Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyu Zhang, Xu Chen, Xuanyu Wang, Xuecheng Su, Xueyin Chen, Xuheng Lin, Xuwei Fu, YC Yan, YQ Wang, YW Ma, Yanfeng Luo, Yang Zhang, Yanhong Xu, Yanru Ma, Yanwen Huang, Yao Li, Yao Li, Yao Xu, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Shao, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yijia Wu, Yiliang Xiong, Yiling Ma, Ying He, Ying Tang, Ying Zhou, Yingjia Luo, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiang Zhang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, YuKun Li, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuanhao Li, Yuduan Wang, Yuehan Yang, Yuer Xu, Yuhan Wu, Yuhao Meng, Yuheng Zou, Yukun Zha, Yunfan Xiong, Yupeng Chen, Yuping Lin, Yuqian Cao, Yuqian Wang, Yushun Zhang, Yuting Yan, Yutong Lin, Yuxian Gu, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuxuan Zhou, Yuyang Zhou, Yuzhen Huang, ZF Wu, Zehao Wang, Zehua Zhao, Zehui Ren, Zekai Zhang, Zhangli Sha, Zhe Fu, Zhe Ju, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zheren Gao, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhixuan Chen, Zhiyu Wu, Zhizhou Ren, Zhongyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihua Qu, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Ziyi Wan, Zizheng Pan, and Zongqing Yao. DeepSeek-V4: Towards highly efficient million-token context intelligence, 2026. URL https://arxiv.org/abs/2606.19348.

Rebecca DerSimonian and Nan Laird. Meta-analysis in clinical trials. *Controlled Clinical Trials*, 7 (3):177–188, 1986.

Frances Ding, Jean-Stanislas Denain, and Jacob Steinhardt. Grounding representation similarity through statistical testing. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 1556–1568. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/0c0bf917c7942b5a08df71f9da626f97-Paper.pdf.

Zihan Dong, Xinyu Fan, and Zhiyuan Peng. FNSPID: A comprehensive financial news dataset in time series, 2024. URL https://arxiv.org/abs/2402.06698.

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, New York, 1993.

Aaron Fisher, Cynthia Rudin, and Francesca Dominici. All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *Journal of Machine Learning Research*, 20(177):1–81, 2019. URL https://www.jmlr.org/papers/v20/18-760.html.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A family of open time-series foundation models. In Ruslan Salakhutdinov, Zico

Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 16115–16152. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/goswami24a.html.

Julian PT Higgins and Simon G Thompson. Quantifying heterogeneity in a meta-analysis. *Statistics in Medicine*, 21(11):1539–1558, 2002.

Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Aniruddha Kembhavi, and Ranjay Krishna. SugarCrepe: Fixing hackable benchmarks for vision-language compositionality. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 31096–31116. Curran Associates, Inc., 2023. doi: 10.52202/075280-1355. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/63461de0b4cb760fc498e85b18a7fe81-Paper-Datasets_and_Benchmarks.pdf.

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. Position: The platonic representation hypothesis. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20617–20642. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/huh24a.html.

Harsh Jhamtani and Taylor Berg-Kirkpatrick. Truth-conditional captions for time series data. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 719–733, Online and Punta Cana, Dominican Republic, nov 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.55. URL https://aclanthology.org/2021.emnlp-main.55/.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3519–3529. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/kornblith19a.html.

Haoxin Liu, Shangqing Xu, Zhiyuan Zhao, Lingkai Kong, Harshavardhan Kamarthi, Aditya B. Sasanur, Megha Sharma, Jiaming Cui, Qingsong Wen, Chao Zhang, and B. Aditya Prakash. Time-MMD: Multi-domain multimodal dataset for time series analysis. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 77888–77933. Curran Associates, Inc., 2024a. doi: 10.52202/079017-2476. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/8e7768122f3eeec6d77cd2b424b72413-Paper-Datasets_and_Benchmarks_Track.pdf.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=JePfAI8fah.

Jürgen Matzka, Oliver Bronkalla, Katrin Tornow, Kirsten Elger, and Claudia Stolle. Geomagnetic Kp index. GFZ Data Services. Data publication, 2021. URL https://doi.org/10.5880/Kp.0001.

Marco Mori, Paolo Papotti, Luigi Bellomarini, and Oliver Giudice. Neural machine translation for fact-checking temporal claims. In Rami Aly, Christos Christodoulopoulos, Oana Cocarascu, Zhijiang Guo, Arpit Mittal, Michael Schlichtkrull, James Thorne, and Andreas Vlachos (eds.), *Proceedings of the Fifth Fact Extraction and VERification Workshop (FEVER)*, pp. 78–82, Dublin, Ireland, may 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.fever-1.8. URL https://aclanthology.org/2022.fever-1.8/.

NASA Community Coordinated Modeling Center. Space Weather Database Of Notifications, Knowledge, Information (DONKI). Online database and web application, n.d. URL https://ccmc.gsfc.nasa.gov/tools/DONKI/. NASA Goddard Space Flight Center. Accessed September 24, 2026.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Jbdc0vTOcol.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer, 2017. URL https://arxiv.org/abs/1709.07871.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Richard D Riley, Julian PT Higgins, and Jonathan J Deeks. Interpretation of random effects meta-analyses. *BMJ*, 342:d549, 2011.

Le Song, Alex Smola, Arthur Gretton, Justin Bedo, and Karsten Borgwardt. Feature selection via dependence maximization. *Journal of Machine Learning Research*, 13:1393–1434, 2012.

Marek Strong and Andreas Vlachos. TSVer: A benchmark for fact verification against time-series evidence. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 29906–29926, Suzhou, China, nov 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.1519. URL https://aclanthology.org/2025.emnlp-main.1519/.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training, 2024a. URL https://arxiv.org/abs/2212.03533v2. Version 2.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Improving text embeddings with large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11897–11916, Bangkok, Thailand, aug 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.642. URL https://aclanthology.org/2024.acl-long.642/.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Zhang, and Jun Zhou. TimeMixer: Decomposable multiscale mixing for time series forecasting. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 38626–38652, 2024c. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/a7ac8a21e5a27e7ab31a5f42a0117bdb-Paper-Conference.pdf.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers, 2020. URL https://arxiv.org/abs/2002.10957.

Andrew Robert Williams, Arjun Ashok, Étienne Marcotte, Valentina Zantedeschi, Jithendaraa Subramanian, Roland Riachi, James Requeima, Alexandre Lacoste, Irina Rish, Nicolas Chapados, and Alexandre Drouin. Context is key: A benchmark for forecasting with essential textual information. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 66887–66944. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/williams25a.html.

Brian D Williamson, Peter B Gilbert, Noah R Simon, and Marco Carone. A general framework for inference on algorithm-agnostic variable importance. *Journal of the American Statistical Association*, 118(543):1645–1658, 2023. doi: 10.1080/01621459.2021.2003200. URL https://doi.org/10.1080/01621459.2021.2003200.

Yumo Xu and Shay B. Cohen. Stock movement prediction from tweets and historical prices. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1970–1979, Melbourne, Australia, jul 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1183. URL https://aclanthology.org/P18-1183/.

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.

Pratham Yashwante and Rose Yu. Time series, vision, and language: Exploring the limits of alignment in contrastive representation spaces. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=TQi2IqvpIf.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=KRLUvxh8uaX.

Xiyuan Zhang, Boran Han, Haoyang Fang, Abdul Fatir Ansari, Shuai Zhang, Danielle C. Maddix, Cuixiong Hu, Andrew Gordon Wilson, Michael W. Mahoney, Hao Wang, Yan Liu, Huzefa Rangwala, George Karypis, and Bernie Wang. When does multimodality lead to better time series forecasting?, 2025a. URL https://arxiv.org/abs/2506.21611.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models, 2025b. URL https://arxiv.org/abs/2506.05176.

Vincent Zhihao Zheng, Étienne Marcotte, Arjun Ashok, Andrew Robert Williams, Lijun Sun, Alexandre Drouin, and Valentina Zantedeschi. Overcoming the modality gap in context-aided forecasting. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=NOHlqCnYot.

Luca Zhou, Pratham Yashwante, Marshall Fisher, Alessio Sampieri, Zihao Zhou, Fabio Galasso, and Rose Yu. CaTS-bench: Can language models describe time series? In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 34479–34519, San Diego, California, United States, jul 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.1722. URL https://aclanthology.org/2026.findings-acl.1722/.

Appendix Contents

A Experimental Setup **15**
- A.1 Datasets and Evaluation Populations 15
- A.2 Models and Representation Setup 16
- A.3 Measurements and Statistical Inference 16

B Factual Discrimination Across Controls **17**
- B.1 Factual-Reversal Construction 18
- B.2 Stability Across Seeds and Measures 18
- B.3 Sensitivity to Description Form 19
- B.4 Projection Capacity and a Positive Boundary 19

C Direct Verification and Its Error Sources **19**
- C.1 Executable Verification and Comparator Protocols 20
- C.2 Controlled and Native Verification 22
- C.3 Interpretation Diagnostics 22

D Controlled Functional Separation **22**
- D.1 Support, Predictive Innovation, and Fitted-Model Assistance 23
- D.2 Training Replacement and Predictive Risk 24

E Text Utility Across Representation States **24**
- E.1 Matched Representation States and Forecasting Gains 25
- E.2 Correspondence and Forecasting Diagnostics 25

F Utility of Executable Support Summaries **28**
- F.1 Summary Definition and Interface Controls 28
- F.2 Candidate Coverage and Binding Diagnostics 29
- F.3 Summary Formation and Positive Control 30

G Limitations and Future Work **31**

A Experimental Setup

A.1 Datasets and Evaluation Populations

CaTS (Zhou et al., 2026) provides 14,995 training, 1,000 development, and 4,005 official test caption–series pairs. The main retrieval evaluation uses all 4,005 test captions, while factual discrimination uses 250 numerical and 250 temporal reversal pairs constructed from the test set. Each reversal pair keeps the time series fixed and changes only the associated statement. Controlled verification gives the verifier the correct series and evaluates $4{,}005 \times 2 = 8{,}010$ Supported/Refuted claims per scoring seed across 11 held-out CaTS domains.

Native verification uses all 304 TSVer claims (Strong & Vlachos, 2025): 92 Supported, 164 Refuted, and 48 Not Enough Evidence claims, with the latter mapped to UNRESOLVED. The claims belong to 43 source groups and are partitioned into five source-group-disjoint outer folds. Comparator-specific evidence inputs are described in Appendix C.1.

We evaluate forecasting on task-specific preparations of Time-MMD (Liu et al., 2024a), MTBench Financial (Chen et al., 2026), StockNet AAPL (Xu & Cohen, 2018), FNSPID A (Dong et al., 2024), and DONKI/Kp based on NASA DONKI notifications and GFZ Kp observations (NASA Community Coordinated Modeling Center, n.d.; Matzka et al., 2021). Table 5 summarizes the numerical histories, targets, and matched populations.

Table 5: **Forecasting tasks and matched populations.** Train/validation/test entries report retained / available recipients after applying the common time-legal-donor mask. The retained population is shared across representation states, backbones, and input conditions, including numerical-only conditions.

Dataset	Numerical history	Target	Train	Val.	Test	Test clusters
Time-MMD	26 weekly OT values	Next four weekly OT values	2344/2345	645/646	645/646	162
Financial	7-day price window	One-day mean-price log return	448/449	150/151	149/150	134
StockNet	5 trading days	Next-day AAPL log return	297/298	97/98	98/99	98
FNSPID A	5 trading days	Next-day log return	190/191	61/62	63/64	63
DONKI/Kp	Eight 3-hour Kp values	Next eight 3-hour Kp values	170/171	45/46	45/46	6

Each task uses only text available at the forecast origin. Time-MMD documents come from the preceding 35-day window, Financial uses the article at its publication-time origin, StockNet and FNSPID use tweets or news from the five trading days before the origin, and DONKI/Kp uses event notifications available at the origin. The Financial target is $\log(\mathrm{mean}(P_{\mathrm{out}})/P_{\mathrm{in,last}})$ over a one-day output window; FNSPID A denotes Agilent Technologies (ticker `A`). A time-legal donor is non-self, belongs to the same split, satisfies the task's text requirements, and has an origin no later than the recipient. This mask removes one recipient from each split of every task, and all compared conditions use the same retained recipients.

A.2 Models and Representation Setup

The representation sweep pairs frozen text and time-series encoders in 20 configurations. Text encoders include T5-small and T5-3B (Raffel et al., 2020), E5-base-v2 (Wang et al., 2024a), E5-Mistral-7B (Wang et al., 2024b), and Qwen3-Embedding 0.6B, 4B, and 8B (Yang et al., 2025; Zhang et al., 2025b). Time-series encoders are MOMENT-base/large (Goswami et al., 2024), Chronos-base/large (Ansari et al., 2024), and TimesFM-2.0-500M (Das et al., 2024). Table 2 lists the 20 encoder pairs, abbreviating E5-base-v2 as E5-base, TimesFM-2.0-500M as TimesFM-500M, and Qwen3-Embedding-nB as Qwen-nB.

All encoders remain frozen. T5 and E5-base-v2 use mean pooling over valid tokens, and E5-Mistral-7B and Qwen3-Embedding use the last non-padding token. Text is truncated to 512 tokens, and CaTS series are linearly resampled to 512 points. The projection heads are described in Appendix B.2.

Forecasting uses six backbone families: iTransformer (Liu et al., 2024b), TimeMixer (Wang et al., 2024c), PatchTST (Nie et al., 2023), MOMENT (Goswami et al., 2024), TimesFM (Das et al., 2024), and Chronos (Ansari et al., 2024). iTransformer, TimeMixer, and PatchTST are fitted on each task with validation-based checkpoint selection; MOMENT-1-base is a frozen feature extractor with a linear probe; TimesFM-2.0-500M and Chronos-T5-base use frozen pretrained inference. For a fixed dataset, backbone, and seed, the backbone forecast is shared across matched adapter conditions.

A.3 Measurements and Statistical Inference

We follow the metric definitions of TVL (Yashwante & Yu, 2026) for the correspondence measures and add PTRA for factual discrimination. Let x_i and s_i, $i = 1, \ldots, n$, denote the L_2-normalized projected series and caption representations of n test pairs, and let $\mathrm{sim}(a, b) = a^\top b$ be their cosine similarity.

Retrieval and cosine margin. Retrieval ranks all captions by $\mathrm{sim}(x_i, \cdot)$ for each series query. R@k is the fraction of queries whose paired caption is among the k highest-ranked captions, and MRR is the mean reciprocal rank of the paired caption. Table 2 reports series-to-caption R@1, whereas the multimetric analysis in Appendix B.2 averages retrieval over both directions. The cosine margin measures how far matched pairs are separated from mismatched pairs:

$$\mathrm{Margin} = \frac{1}{n}\sum_{i=1}^{n}\Big(\mathrm{sim}(x_i, s_i) - \frac{1}{n-1}\sum_{j\neq i}\mathrm{sim}(x_i, s_j)\Big). \tag{3}$$

Pair cosine instead compares the text representations of the positive and edited negative statements of a reversal pair.

Procrustes disparity. Let $\widetilde{X}$ and $\widetilde{S}$ be the column-centered matrices whose rows are x_i and s_i, each scaled to unit Frobenius norm. Procrustes disparity (Ding et al., 2021) is the residual after the best orthogonal transformation,

$$d_{\mathrm{P}} = \min_{R^\top R = I} \|\widetilde{X}R - \widetilde{S}\|_F^2, \tag{4}$$

so smaller values indicate closer geometry.

CKA. For Gram matrices K and L computed on the two spaces, CKA (Kornblith et al., 2019) is

$$\mathrm{CKA}(K, L) = \frac{\langle HKH, HLH \rangle_F}{\|HKH\|_F \, \|HLH\|_F}, \qquad H = I - \tfrac{1}{n}\mathbf{1}\mathbf{1}^\top. \tag{5}$$

The CaTS analysis uses RBF CKA, with Gaussian kernels whose bandwidth is set by the median pairwise distance, and debiased linear CKA, which uses $K = XX^\top$ and $L = SS^\top$ with the unbiased HSIC estimator (Song et al., 2012) for the three inner products.

Mutual-kNN. Let $\mathcal{N}_k^X(i)$ and $\mathcal{N}_k^S(i)$ be the k cosine nearest neighbors of pair i in the two spaces, excluding the query. Mutual-kNN (Huh et al., 2024) is their mean overlap,

$$\mathrm{mKNN} = \frac{1}{nk} \sum_{i=1}^{n} \left| \mathcal{N}_k^X(i) \cap \mathcal{N}_k^S(i) \right|, \tag{6}$$

with $k = 5$. CKA and mutual-kNN use a fixed subsample of at most 1,500 CaTS test pairs. The forecasting CKA values in Appendix E.2 are computed on the task-specific test populations, so they should not be compared directly with the CaTS values.

PTRA. For reversal pair i, let x_i be the projected series and s_i^+ and s_i^- the projected supported and refuted statements for that series. PTRA over the $m = 250$ numerical or $m = 250$ temporal pairs is

$$\mathrm{PTRA} = \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}\left[\mathrm{sim}(x_i, s_i^+) > \mathrm{sim}(x_i, s_i^-)\right], \tag{7}$$

so ties count as errors and chance is 0.5. Table 2 reports means over five projection seeds.

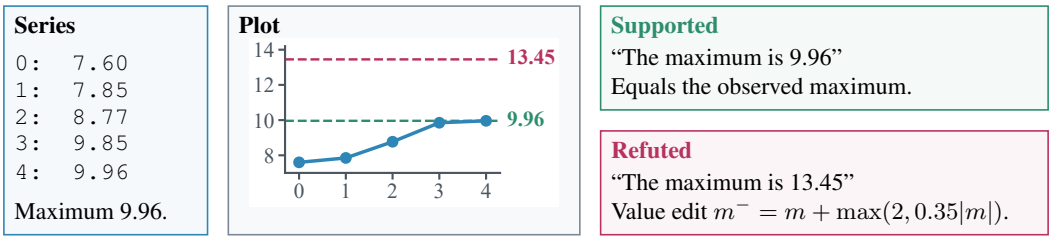
Uncertainty. Native TSVer metrics pool out-of-fold predictions over all 304 claims. Paired intervals resample the 43 source groups 10,000 times, using the same group draw for both sides of each contrast.

For forecasting, squared error is first averaged over target coordinates for each recipient and then over the three fitted seeds. Paired uncertainty is estimated by 10,000 cluster-bootstrap (Efron & Tibshirani, 1993) draws of these seed-averaged per-recipient losses, using the same draw for both conditions in a contrast. Time-MMD uses source-series-aware non-overlapping four-origin blocks, Financial uses frozen source/event groups, StockNet and FNSPID use singleton origins, and DONKI/Kp uses non-overlapping eight-origin blocks. The resulting test cluster counts are shown in Table 5; in particular, DONKI/Kp contains only six test clusters.

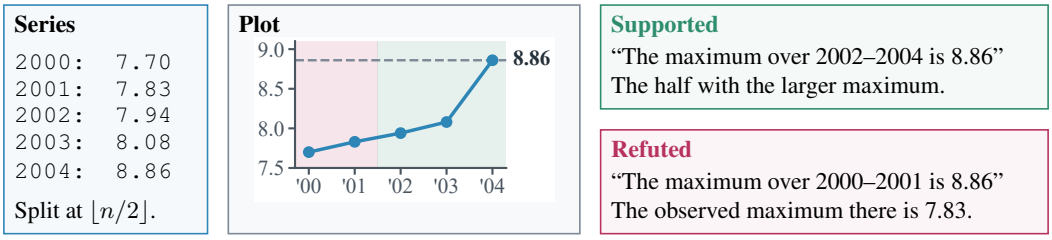
Multiple comparisons. We apply the Benjamini–Hochberg procedure (Benjamini & Hochberg, 1995) within each prespecified family of comparisons, and each section below states which comparisons its family contains.

B Factual Discrimination Across Controls

In this section, we test whether the gap between correspondence and factual discrimination in Section 4.1 depends on the reversal construction, the projection seed, the correspondence measure, the description form, or the projection capacity. We also report a constrained setting in which the same type of matching score does separate the factual alternatives.



(a) Value edit



(b) Interval edit

Figure 3: **Two factual-reversal constructions on a fixed series.** Each row shows one pair: the series, its plot, and the two statements. **(a)** The value edit changes the stated maximum. **(b)** The interval edit keeps the stated value and moves the interval. The labels 2000–2004 are synthetic indices rather than calendar dates.

B.1 Factual-Reversal Construction

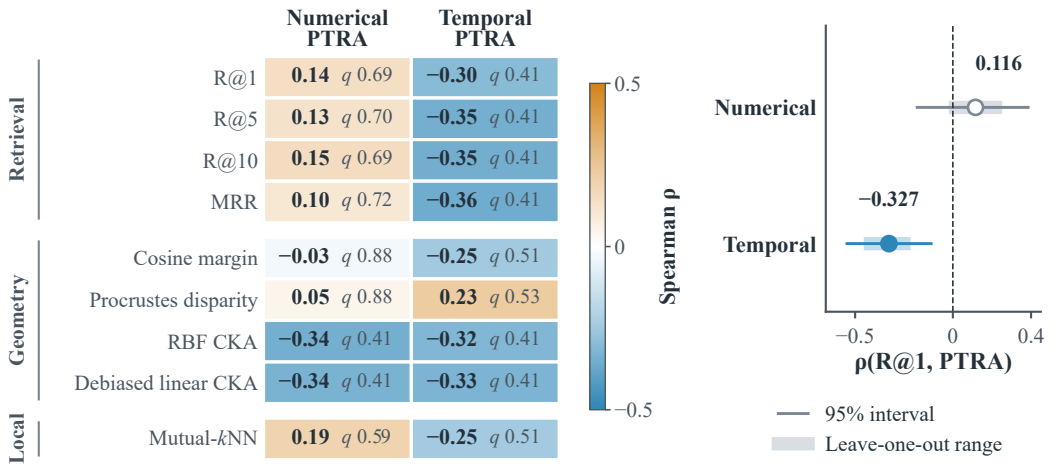
Each of the 500 CaTS test pairs keeps the series and entity fixed and contrasts a Supported statement with a minimally edited Refuted alternative, similar to hard negatives for vision–language compositionality (Yuksekgonul et al., 2023; Hsieh et al., 2023). The two statements have the same token count, literals use two decimals, and each edit changes one factor. A numerical reversal replaces the true maximum m by $m^- = m + \mathrm{sgn}_+(m)\max(2, 0.35|m|)$, with $\mathrm{sgn}_+(m) = 1$ for $m \geq 0$ and -1 otherwise. A temporal reversal splits the series at $\lfloor n/2 \rfloor$: the half with the larger maximum supplies the supported interval and the other half the refuted one. We discard pairs whose two maxima are not sufficiently separated.

B.2 Stability Across Seeds and Measures

In the main sweep, we map each frozen representation through $\mathrm{Linear}(d, 3072) \to \mathrm{LayerNorm} \to \mathrm{GELU} \to \mathrm{Dropout}(0.1) \to \mathrm{Linear}(3072, 1024)$ and L_2-normalize the output. Heads are trained with symmetric InfoNCE (Radford et al., 2021) (temperature 0.2), AdamW (learning rate 10^{-4}, weight decay 10^{-5}), batch size 256, and at most 50 epochs with patience 5 on development loss, using projection seeds 2024–2028.

The 21-configuration analysis adds a split-controlled TVL-style trimodal reference (Yashwante & Yu, 2026). It uses frozen SigLIP2-base, Qwen3-Embedding-0.6B, and Chronos-T5-base encoders with projection heads trained on our split instead of the released TVL checkpoint. Over three seeds, it reaches R@1 0.0220, numerical PTRA 0.5773, and temporal PTRA 0.5107. Retrieval measures in this analysis average both directions. We test each of the 18 associations with Spearman's ρ and 10,000 two-sided permutations, and apply one Benjamini–Hochberg correction to all 18.

None of the 18 tested associations survives joint correction (Figure 4(a)). All nine temporal correlations have the same direction, with closer correspondence going with lower temporal PTRA, while the numerical correlations change sign across measures. The lack of significance does not mean that the two quantities are unrelated. When we resample the 436 CaTS source groups jointly over the 20 main configurations (2,000 draws, fitted models held fixed), the temporal interval stays below zero, $\rho = -0.3269$ $[-0.5416, -0.1114]$, and the numerical interval includes zero, $\rho = 0.1158$ $[-0.1822, 0.3850]$ (Figure 4(b)). These intervals are not adjusted for multiplicity and do not change the joint 18-test BH result.



(a) Nine measures, joint BH over 18 tests

(b) Resampling 436 CaTS source groups

Figure 4: **Correspondence measures versus factual discrimination.** **(a)** Spearman ρ between each measure (rows) and numerical or temporal PTRA (columns) across 21 configurations, with the BH q value in each cell. Procrustes is a disparity, so lower values indicate closer geometry. **(b)** $\rho(\mathrm{R@1}, \mathrm{PTRA})$ over the 20 main configurations when the 436 CaTS source groups are resampled jointly: 95% interval (line) and leave-one-configuration-out range (band).

B.3 Sensitivity to Description Form

We express the same 500 reversal pairs in six description forms (Figure 5(a)) for T5-small + MOMENT-base, Qwen-4B + Chronos-large, and the TVL-style reference. The first four forms add observed context to the target claim, and the Generic-wording and Evidence-listed forms are wording controls.

Across the displayed means, PTRA is at most 0.5693 and does not increase monotonically with added context. For T5-small + MOMENT-base, the Extrema-added form raises R@1 over the Endpoints-added form from 0.0020 to 0.0037, while PTRA falls from 0.5573 to 0.5227. Because the forms change content and presentation together, these results measure sensitivity to description form and do not isolate the effect of textual richness.

B.4 Projection Capacity and a Positive Boundary

To test whether a larger projection head recovers factual ranking, we vary the head for three frozen encoder pairs and keep the factual evaluation fixed (Table 6). These are separate three-seed runs. The 3072-hidden head has the main-sweep architecture but is averaged over three seeds instead of five, so its mean need not equal the main-table value.

For T5-small + MOMENT-large, R@1 roughly doubles from the linear to the 3072-hidden head (0.0099 to 0.0203). Across all eleven variants, temporal PTRA stays between 0.4667 and 0.5120. For Qwen-4B + Chronos-large, the same change raises R@1 from 0.0326 to 0.0400 with temporal PTRA near chance (0.4987 versus 0.4960).

On 246 TRUCE pairs whose labels can be checked against the series (Jhamtani & Berg-Kirkpatrick, 2021), all three representation settings reach PTRA between 0.9350 and 1.0000 (Figure 6), while instance R@1 over the 246 candidates is at most 0.0088. Matching scores can therefore rank factual alternatives well when the contrast is tightly constrained, even at low retrieval, whereas on the CaTS reversals high retrieval does not produce this ranking.

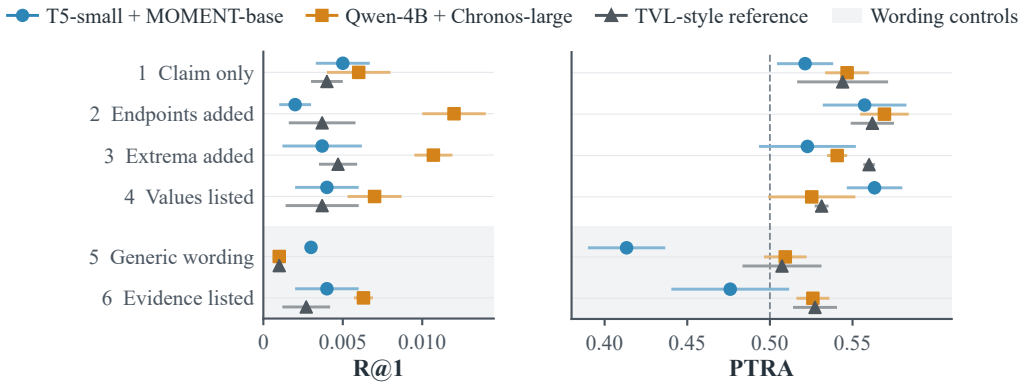
C Direct Verification and Its Error Sources

In this section, we describe how ECA verifies historical support and separate its errors into two sources, executable coverage and interpretation selection. Controlled CaTS (Zhou et al., 2026) tests

Series `7.60, 7.85, 8.77, 9.85, 9.96`
Target claim SUPPORTED maximum = 9.96; REFUTED maximum = 13.45

1 ***Claim only*** Target claim; no additional evidence	**2** ***Endpoints added*** Target claim + start = 7.60, end = 9.96	**3** ***Extrema added*** Target claim + start = 7.60, end = 9.96, min = 7.60, max = 9.96
4 ***Values listed*** Target claim + 7.60, 7.85, 8.77, 9.85, 9.96	**5** ***Generic wording*** "The time series exhibits / does not exhibit the described pattern."	**6** ***Evidence listed*** Indices 0–4, target claim, and evidence 7.60, 7.85, 8.77, 9.85, 9.96

(a) One reversal pair under the six description forms



(b) R@1 (c) PTRA

Figure 5: **Retrieval and factual ranking respond differently to description form. (a)** Forms 1–4 add observed context to the target claim; forms 5–6 are wording controls. **(b, c)** R@1 and PTRA for each form and encoder setting (means ± sample SD over three projection seeds); the shaded rows are the wording controls. PTRA is computed over all 500 pairs, and the dashed line marks chance.

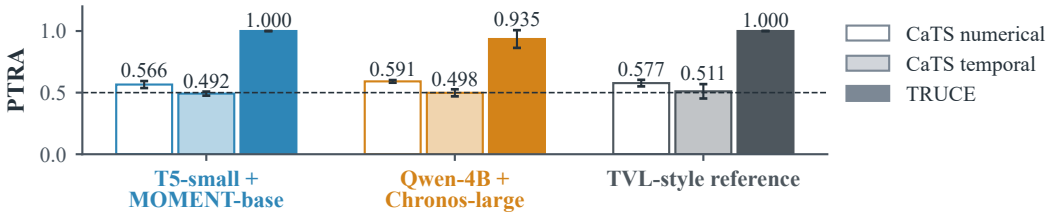


Figure 6: **Factual ranking on CaTS reversals versus constrained TRUCE contrasts.** PTRA on the CaTS reversals (five-seed main-sweep values; three seeds for the TVL-style reference) and on the TRUCE contrasts (three seeds). The dashed line marks chance.

the verifier when the correct series is given, and Native TSVer (Strong & Vlachos, 2025) asks the same support question on naturally occurring claims.

C.1 EXECUTABLE VERIFICATION AND COMPARATOR PROTOCOLS

Executable verification. ECA constructs at most 20 typed candidate programs per claim (Figure 7). A candidate specifies the referenced series, temporal scope, literals, operators, and comparisons; binding maps these arguments to the supplied observations, and execution returns SUPPORTED, REFUTED, or UNRESOLVED. Candidates rejected by the typed gate, or failing binding or execution, remain UNRESOLVED. Native candidate generation excludes the held-out source groups and uses the annotated reference series and time ranges.

Table 6: **Projection capacity.** Eleven head and learning-rate variants over three frozen encoder pairs (mean ± sample SD over three seeds). Heads use learning rate 10^{-4} unless noted; the 3072-hidden head matches the main-sweep architecture. Chance PTRA is 0.50.

Encoder pair and head	R@1	Numerical PTRA	Temporal PTRA
T5-small + MOMENT-base			
Linear	0.0112 ± 0.0013	0.5787 ± 0.0061	0.4667 ± 0.0083
1-hidden	0.0164 ± 0.0010	0.5493 ± 0.0254	0.5107 ± 0.0638
3072-hidden	0.0157 ± 0.0015	0.5840 ± 0.0346	0.5120 ± 0.0640
3072-hidden, lr 3×10^{-4}	0.0143 ± 0.0014	0.5800 ± 0.0349	0.4880 ± 0.0250
3072-hidden, lr 3×10^{-5}	0.0167 ± 0.0005	0.5720 ± 0.0212	0.5000 ± 0.0382
Qwen-4B + Chronos-large			
Linear	0.0326 ± 0.0007	0.6173 ± 0.0140	0.4987 ± 0.0201
1-hidden	0.0358 ± 0.0008	0.6040 ± 0.0139	0.4933 ± 0.0220
3072-hidden	0.0400 ± 0.0023	0.5853 ± 0.0115	0.4960 ± 0.0144
T5-small + MOMENT-large			
Linear	0.0099 ± 0.0010	0.5200 ± 0.0106	0.4773 ± 0.0122
1-hidden	0.0189 ± 0.0012	0.5467 ± 0.0266	0.4933 ± 0.0469
3072-hidden	0.0203 ± 0.0008	0.5667 ± 0.0101	0.5120 ± 0.0200

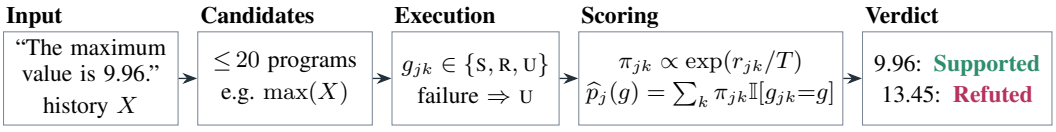


Figure 7: **Executable Claim Alignment.** A claim and its history are mapped to typed candidate programs, each program is bound to the observations and executed, and a learned scorer converts candidate scores into execution mass over the three verdicts (Equation 8). Executing the example on the history of Figure 3(a) returns SUPPORTED; its numerical reversal returns REFUTED.

Candidate scoring. For candidate z_{jk} of claim a_j, ECA assigns $r_{jk} = f_\theta(\xi(a_j, z_{jk}, X))$ and converts the candidate scores into execution mass

$$\pi_{jk} = \frac{\exp(r_{jk}/T)}{\sum_\ell \exp(r_{j\ell}/T)}, \qquad \widehat{p}_j(g) = \sum_k \pi_{jk} \mathbb{I}[g_{jk} = g]. \tag{8}$$

The scorer uses claim, program, and binding features; the Native model adds lexical-overlap and binding indicators. It is trained with a pairwise softplus ranking loss (margin 0.25) between candidates equivalent to an accepted training program and the remaining candidates, plus an emission loss with weight 0.25 on candidates whose execution matches the training verdict. The Native scorer is a 128–64 GELU MLP trained with AdamW for 200 epochs.

Native inference. Native verification uses five source-group-disjoint outer folds and three scorer seeds. Candidate masses are evaluated at temperature 16; fold-specific abstention thresholds are fixed before the outer-fold predictions. A low maximum class mass returns UNRESOLVED; otherwise the maximum-mass class wins, and the three scorer predictions are combined by majority vote, with a three-way tie mapped to UNRESOLVED. The scorer folds are source-disjoint. The temperature and abstention thresholds come from historically reused calibration and are not re-estimated inside each outer fold, so we treat the Native results as a retrospective known-source system comparison.

Comparators. Representation matching uses frozen T5-small/MOMENT-base features and cosine score $s = z_X^\top z_S$. Two thresholds selected on an inner-development split map high scores to SUPPORTED, low scores to REFUTED, and intermediate scores to UNRESOLVED. The generic verifier is a 4096–512–3 MLP on $[z_X; z_S; |z_X - z_S|; z_X \odot z_S]$. The direct LLM verifier uses `deepseek-v4-pro` (DeepSeek-AI et al., 2026) and receives the claim together with serialized observations from its annotated evidence range, with no tools or web search; all 304 responses parse to a single verdict. Because the representation comparator uses whole-series features and the direct LLM receives claim-localized evidence, the Native systems do not receive matched inputs.

Table 7: **Controlled and Native verification. (a)** Two-class readouts on controlled representations. **(b)** Native verification on 304 TSVer claims. The final column reports ECA minus each comparator in Macro-F1. Intervals use 10,000 paired resamples of the 43 source groups.

(a) Controlled CaTS, two-class readouts

Readout	Macro acc.	Pooled acc.
Thresholded matching	0.4973	0.4954
Fitted linear head	0.6871	0.6762

(b) Native TSVer

Method	Accuracy	Bal. acc.	Macro-F1	ECA − comparator [95% CI]
Representation	0.2829	0.3637	0.2833	0.1505 [0.0700, 0.2437]
Generic verifier	0.1579	0.3333	0.0909	0.3428 [0.2905, 0.3980]
ECA	0.4605	0.4731	0.4337	—
Direct LLM verifier	0.6053	0.6162	0.5740	−0.1403 [−0.2493, −0.0361]

C.2 Controlled and Native Verification

Controlled verification. Controlled CaTS contains 8,010 binary claims per scoring seed (42, 43, 44) across 11 held-out domains, and outer-test examples are excluded from compiler fitting and calibration. ECA reaches domain-macro accuracy 0.987393 (Figure 2, left) and pooled accuracy 0.989263. To check whether support can be read from the frozen T5-small/MOMENT-base representations, we remove the UNRESOLVED output and fit two binary readouts on the same claims, thresholded matching and a linear head over $[z_X; z_S; |z_X - z_S|; z_X \odot z_S]$, both selected by inner-development balanced accuracy. The linear head raises macro accuracy from 0.4973 to 0.6871 (Table 7(a)). Since these readouts have two outputs, their accuracies are not directly comparable with the three-output systems in Figure 2.

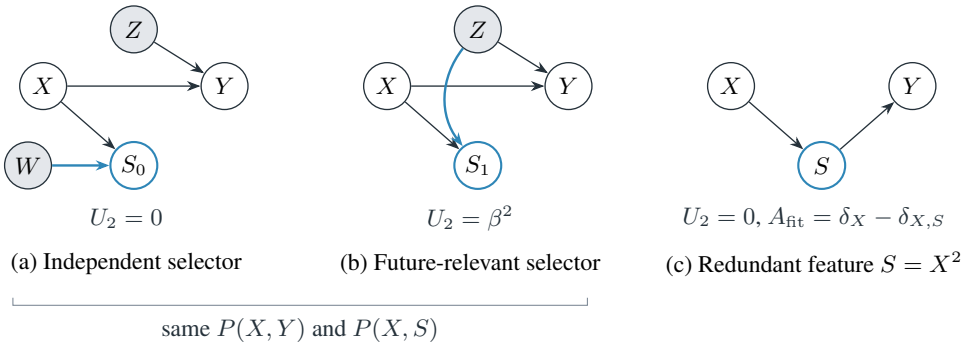
Native verification. All four systems in Table 7(b) are evaluated on the same 304 claims. The generic verifier predicts UNRESOLVED for every claim. For reference, a constant REFUTED rule has accuracy 0.5395, balanced accuracy 0.3333, and Macro-F1 0.2336.

C.3 Interpretation Diagnostics

Interpretation selection. ECA resolves 194 of the 304 Native claims (coverage 0.6382), and 83 of the resolved verdicts are wrong (conditional error 0.4278). A label-conditioned candidate-emission oracle reaches Macro-F1 0.8951, so some candidate often emits the reference verdict. Because the oracle uses the label, we use it only to measure candidate availability. To test a different selection rule, we keep candidates, binding, execution, gate, and scorer fixed and give a DeepSeek V4 Flash selector (temperature zero, reasoning disabled) the claim and candidate metadata, without verdicts, scorer values, or labels. When the selection is valid, its one-hot weights q_L replace the ECA weights π_E by $\frac{1}{2}\pi_E + \frac{1}{2}q_L$. The selector chooses a candidate for 72 claims and abstains on 232. Reweighting changes 28 final Native verdicts and moves Macro-F1 from 0.4337 to 0.4448 (+0.0110, 95% interval $[-0.0281, 0.0423]$), with coverage 0.6941 and conditional error 0.4455.

D Controlled Functional Separation

In this section, we use controlled constructions to separate historical support, predictive innovation, and fitted assistance, three quantities that can coincide in special cases. The first construction holds the numerical history–future process fixed and changes only how a statement is selected, so historical support stays the same while predictive innovation changes. The second adds a feature that is redundant at the population level but useful to a restricted fitted predictor. The third shows what a training-replacement contrast measures.



(a) Independent selector (b) Future-relevant selector (c) Redundant feature $S = X^2$

Figure 8: **Generating structure of the controlled constructions.** Blue nodes are the emitted statements and blue edges their selection mechanism; shaded nodes are latent signs. **(a, b)** Instances of Proposition 1: the two selectors induce the same history–statement law and the same support labels, yet only S_1 carries the future-relevant sign Z. **(c)** A history-derived feature has no innovation but can reduce the risk of a restricted predictor.

Table 8: **Controlled separation of support, innovation, and assistance. (a)** Historical support and statement selection; assistance is the fitted test-risk reduction from adding Q, over 200 realizations. **(b)** Redundant feature at zero innovation.

(a) Historical support and statement selection

Historical verdict	**Selector**	U_2	**Assistance**	**95% interval**
Supported or Unresolved	Independent	0	-0.003155	$[-0.003859, -0.002452]$
Supported or Unresolved	Future-relevant	1.5625	1.570143	$[1.566149, 1.574136]$

(b) Redundant feature at zero innovation

Predictor	U_2	**Assistance [95% CI]**
Restricted	0	$0.089400\ [0.089114, 0.089685]$
Expressive	0	$\approx 0\ [-8.52, 3.77] \times 10^{-13}$

D.1 Support, Predictive Innovation, and Fitted-Model Assistance

Risk decomposition. All constructions use Equations 1 and 2: fitted assistance equals U_2 plus the difference in excess risks of the two fitted predictors. Under squared error, U_2 is the algorithm-agnostic importance of S given X (Williamson et al., 2023), whereas the excess-risk term depends on the predictor class, as in model-class reliance (Fisher et al., 2019). $U_2 = 0$ means that the two conditional means are equal almost surely; it does not require conditional independence of S and Y given X.

Statement selection. We complete here the specification of the constructions in Section 4.2. In the selection construction (Figure 8(a, b)), Z, W are independent Rademacher signs, $Y = 0.7X_1 - 0.4X_2 + 0.2X_3 + 1.25Z + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.5^2)$, and a supported statement reports a true observed value from channel 1 or 2, with the selector deciding which one is emitted. Verdicts are assigned by construction, not by ECA. Table 8(a) gives the intervals behind Table 3(a); the two verdict conditions give identical values because they share the X/Y draws and the selector.

Redundant feature. In the construction of Figure 8(c), $X \sim \text{Uniform}[-1, 1]$, $S = X^2$, and $Y = S + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.1^2)$. Both predictors standardize training inputs and use ridge penalty 10^{-6} with an unpenalized intercept, over 200 realizations of 400 training and 3,000 test observations; intervals are the mean ± 1.96 standard errors (Table 8(b)).

Table 9: **Predictive innovation and training-replacement targets.** $D^{\mathrm{pop}}_{\mathrm{train}}$ is the population replacement contrast of Equation 12; D_{train} is the estimate over 200 realizations with its 95% interval. Both fitted models are evaluated on the original test statements.

Construction	U_2	$D^{\mathrm{pop}}_{\mathrm{train}}$	D_{train} [95% interval]
Independent-language null	0	0	0.0003 [−0.0002, 0.0007]
Unconditional donor, $\beta = 0$	1.44	1.44	1.4291 [1.3879, 1.4702]
Within-Z donor, $\beta = 0$	1.44	0	−0.0011 [−0.0029, 0.0008]
Within-Z donor, $\beta = 1.2$	3.24	0.1475	0.1599 [0.1448, 0.1749]

D.2 Training Replacement and Predictive Risk

A training-replacement contrast asks a different question: how much predictive performance is lost when the training statements are replaced under a specified rule while evaluation remains on the original test population. If $\mathcal{D}^M$ denotes the donor-replaced training set and P the original test population, define

$$D_{\mathrm{train}} = R_P(\widehat{f}_{\mathcal{D}^M}) - R_P(\widehat{f}_{\mathcal{D}}). \tag{9}$$

The two models are fitted separately (affine ridge, penalty 10^{-6}) and evaluated on the same original test statements, over 200 realizations of 500 training and 500 test observations. Within-Z donors are non-self permutations; no donor crosses the training/test boundary.

Population targets of the replacement rules. For the non-null constructions, $Z \in \{-1, 1\}$ with equal probability, X is independent of Z, and

$$P(S = 1 \mid Z = 1) = 0.8, \qquad P(S = 1 \mid Z = -1) = 0.2, \qquad Y = X + 2Z + \beta S + \varepsilon, \tag{10}$$

where $\varepsilon \sim \mathcal{N}(0, 0.5^2)$ is independent. The predictor receives X and S, but not Z. Since $\mathbb{E}[Z \mid S] = 1.2S - 0.6$ and $\mathrm{Var}(S) = 1/4$,

$$\mathbb{E}[Y \mid X, S] = X - 1.2 + (2.4 + \beta)S, \qquad U_2(S; Y \mid X) = \frac{(2.4 + \beta)^2}{4}. \tag{11}$$

Thus the $\beta = 0$ construction already has innovation 1.44, while $\beta = 1.2$ has innovation 3.24.

For within-Z replacement, draw S^M independently of S conditional on the same Z. Then $\mathbb{E}[S \mid S^M] = 0.32 + 0.36S^M$, so the population predictor trained under the replacement law is

$$f_M(X, s) = X - 1.2 + 0.32\beta + (2.4 + 0.36\beta)s.$$

Evaluating this predictor on the original S gives

$$D^{\mathrm{pop}}_{\mathrm{train}} = \mathbb{E}\Big[\big(f_M(X, S) - \mathbb{E}[Y \mid X, S]\big)^2\Big] = 0.1024\beta^2. \tag{12}$$

Hence within-Z replacement has population contrast 0 when $\beta = 0$ despite $U_2 = 1.44$, and 0.147456 when $\beta = 1.2$. Unconditional replacement instead removes the association between S and Z, giving population contrast 1.44 when $\beta = 0$.

In the independent-language null, $S \sim \mathrm{Bernoulli}(1/2)$ is independent of the target-generating variables, so $U_2 = 0$ and the estimated replacement contrast is 0.0003 (Table 9). In the two $\beta = 0$ constructions, where $U_2 = 1.44$, unconditional replacement gives 1.4291 and within-Z replacement −0.0011, matching their population targets.

E Text Utility Across Representation States

In this section, we connect correspondence to forecasting through one frozen encoder pair. We separate the text gain within one representation state, which is the quantity usually reported for text-aided forecasting (Zhang et al., 2025a; Williams et al., 2025), from the increment of a trained projection over its matched untrained reference.

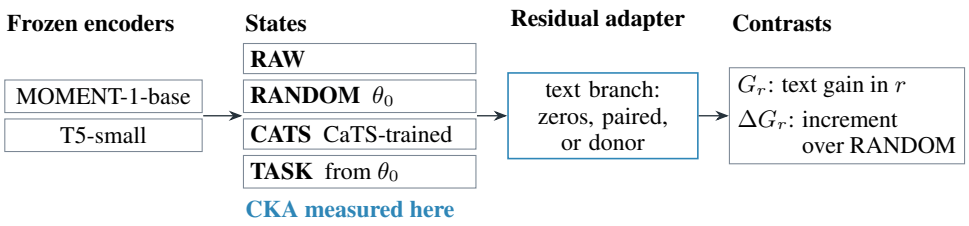


Figure 9: **One pipeline for correspondence and forecasting.** The same frozen encoders feed four representation states; correspondence is measured on these embeddings, and each state feeds separately fitted residual adapters whose text branch receives zeros, paired text, or time-legal donor text. RANDOM and TASK share the initialization θ_0, so RANDOM is the matched untrained reference for the increment ΔG_r.

E.1 Matched Representation States and Forecasting Gains

Representation states. The bridge (Figure 9) uses frozen MOMENT-1-base (Goswami et al., 2024) and T5-small (Raffel et al., 2020) encoders. Series are resampled to 512 points; text is encoded in chunks of at most 512 tokens and combined using token-count weights. We compare four representation states: RAW normalizes the frozen embeddings and pads them to 1024 dimensions; RANDOM applies untrained projection heads; CATS uses projection heads trained on CaTS; and TASK trains projection heads on eligible historical text–series pairs from the current forecasting task. RANDOM and TASK start from the same projection initialization, so comparing them isolates task-specific projection training. The TASK checkpoints are separate from the CaTS-trained projections of the factual-reversal sweep.

RANDOM, CATS, and TASK use the projection architecture of Appendix B.2. TASK uses symmetric multi-positive contrastive training (temperature 0.2), treating examples with the same text hash as positives, with checkpoints selected on validation contrastive loss; future forecasting targets are excluded from projection training and selection.

Forecasting interface. Each residual adapter receives numerical covariates, the backbone forecast, and the state-specific series representation; its text branch receives zeros, paired text, or time-legal donor text. Matched conditions share recipients, architecture (branch widths 128/64/32, fusion 128/64), initialization, and example order but are fitted separately with AdamW and validation-MSE selection. Projection seeds $(2024, 2025, 2026)$ are paired with adapter seeds $(101, 211, 307)$.

Text gain and training increment. Numerical-only risks can differ across representation states because the zero-text adapter still receives the state-specific series representation and is fitted and selected separately. We therefore distinguish a within-state text gain from the increment over the RANDOM reference. For seed-averaged numerical-only and paired-text risks,

$$G_r = \frac{\overline{R}_{r,N} - \overline{R}_{r,T}}{\overline{R}_{r,N}}, \tag{13}$$

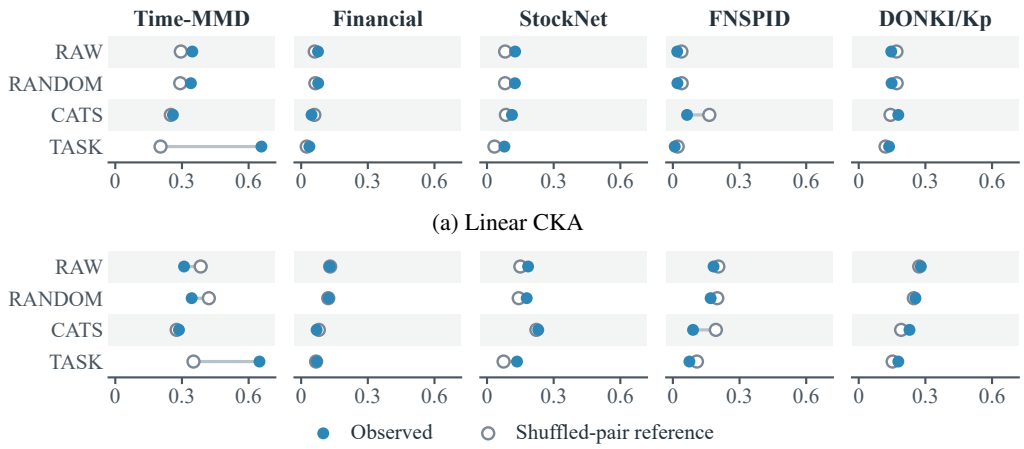
$$\Delta G_r = \frac{(\overline{R}_{r,N} - \overline{R}_{r,T}) - (\overline{R}_{\mathrm{RANDOM},N} - \overline{R}_{\mathrm{RANDOM},T})}{\overline{R}_{\mathrm{RANDOM},N}}. \tag{14}$$

Tables report $100G_r$ and $100\Delta G_r$; ΔG_r compares absolute risk reductions under a common normalization, so it differs from the difference between two percentage gains. Table 11 adds RAW and CATS to the RANDOM and TASK results of Table 4.

No CATS increment has $q < 0.05$ within its 30-cell correction family (Table 11).

E.2 Correspondence and Forecasting Diagnostics

Geometry of the representation states. Correspondence is measured on the same projected embeddings that feed the adapters, once per task and state. On Time-MMD (Liu et al., 2024a), TASK raises Linear CKA from 0.34016 under RANDOM to 0.65753; on the other four tasks it lowers Linear CKA (Figure 10). The shuffled-pair reference permutes the cross-modal pairing within a state and keeps the embeddings fixed, so it tests pairing, whereas RANDOM tests projection training.



(a) Linear CKA

(b) RBF CKA

Figure 10: **Observed correspondence relative to shuffled pairing.** Filled and hollow points denote observed and shuffled-pair CKA, respectively, averaged over three projection seeds. Lines pair measurements within each task and state. Geometry is measured once per task/state and shared across its six forecasting backbones.

Table 10: **Time-MMD/PatchTST separates correspondence, text gain, training increment, and pairing sensitivity.** Panel (a) reports Linear CKA and seed-averaged risks. Panel (b) reports the corresponding relative effects with paired 95% cluster intervals and q values from their correction families.

(a) Correspondence and absolute risks

State	Linear CKA	Numerical MSE	Paired-text MSE	Donor-text MSE
RAW	0.347	0.247199	0.243938	0.260856
RANDOM	0.340	0.269599	0.251470	0.251685
CATS	0.259	0.265444	0.270490	0.263291
TASK	0.658	0.260870	0.242642	0.267266

(b) Fitted-risk contrasts

State	Effect (%)	95% interval	q
Within-state text gain			
RAW	+1.32	[−18.99, 12.72]	0.932
RANDOM	+6.72	[−4.70, 12.21]	0.393
CATS	−1.90	[−14.75, 6.89]	0.941
TASK	+6.99	[3.30, 10.74]	0.010
Training increment relative to RANDOM			
CATS	−8.60	[−17.36, 1.22]	0.472
TASK	+0.04	[−6.73, 14.51]	0.996
Paired text versus time-legal donor text			
RAW	+6.49	[−16.38, 20.29]	0.668
RANDOM	+0.09	[−7.27, 5.21]	0.970
CATS	−2.73	[−10.69, 7.93]	0.921
TASK	+9.21	[−0.65, 16.51]	0.233

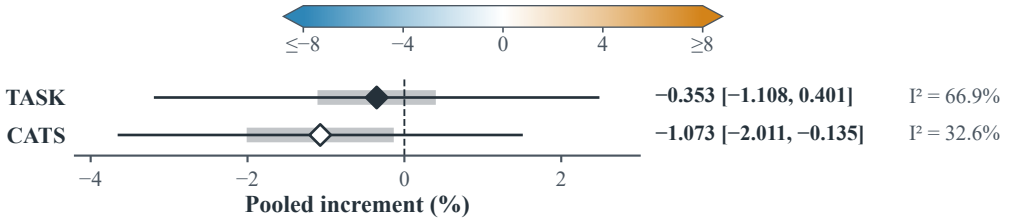
Time-MMD/PatchTST example. Table 10 expands the Time-MMD/PatchTST example to all four states. Pairing sensitivity $(\overline{R}_{r,\mathrm{DONOR}} - \overline{R}_{r,\mathrm{TRUE}})/\overline{R}_{r,\mathrm{DONOR}}$ is positive when correctly paired text beats time-legal donor text.

Task	ΔCKA	iTransformer	TimeMixer	PatchTST	MOMENT	TimesFM	Chronos
Time-MMD	+0.317	2.36	5.67	0.04	−3.53	3.76	2.09
DONKI/Kp	−0.011	−0.24	**1.54**	0.16	−0.68	0.11	−4.50
FNSPID	−0.012	−3.79	**−14.76**	**−3.74**	−0.55	3.08	−0.66
Financial	−0.040	1.79	−0.35	0.70	2.49	2.77	3.38
StockNet	−0.047	0.06	−1.84	−2.94	−0.27	−0.66	−1.85

(a) TASK

Time-MMD	+0.317	0.81	−6.08	−8.60	2.37	−6.28	0.23
DONKI/Kp	−0.011	−0.37	0.39	−0.83	−0.19	2.74	−4.49
FNSPID	−0.012	−5.07	−9.11	1.78	−5.01	−4.05	0.65
Financial	−0.040	0.72	−4.84	0.77	0.41	−1.66	−2.64
StockNet	−0.047	0.21	−6.17	−13.26	2.61	−2.48	0.49

(b) CATS



(c) Random-effects summaries over the 30 cells

Figure 11: **Training increments over the fixed task–backbone grid. (a, b)** TASK and CATS increments $100\Delta G_r$ (%) for each task (rows) and backbone (columns). Tasks are ordered by $\Delta\text{CKA} = \text{CKA}_{\text{TASK}} - \text{CKA}_{\text{RANDOM}}$, given beside each task name; the colour scale saturates at ± 8, and outlined cells mark TASK increments with $q < 0.05$. **(c)** Pooled increment with its 95% interval (thick) and prediction interval (thin).

With paired text, RANDOM and TASK reduce MSE by 0.018129 and 0.018228. Their difference, 0.000099, gives the 0.04% increment after RANDOM normalization. The TASK text gain of 6.99% therefore comes with an increment near zero, because RANDOM already obtains almost the same risk reduction.

Fixed-grid synthesis. The 30 task–backbone cells share datasets, recipients, and representation states, so we summarize them descriptively with DerSimonian–Laird random effects (DerSimonian & Laird, 1986), using the standard deviation of each cell's paired cluster-bootstrap draws as its standard error. Heterogeneity is reported as I^2 (Higgins & Thompson, 2002), and prediction intervals (Riley et al., 2011) use the normal approximation $\hat{\mu} \pm 1.96\{\hat{\tau}^2 + \text{se}(\hat{\mu})^2\}^{1/2}$. When tasks are ordered by the TASK-minus-RANDOM change in Linear CKA (Figure 11(a)), the increments do not follow this order: the six Time-MMD TASK increments range from -3.53% to 5.67% and overlap those of Financial, where TASK lowers CKA.

The synthesis (Figure 11(c)) gives a TASK pooled increment of -0.353% (95% CI $[-1.108, 0.401]\%$; $I^2 = 66.9\%$; prediction interval $[-3.177, 2.470]\%$) and a CATS pooled increment of -1.073% (95% CI $[-2.011, -0.135]\%$; $I^2 = 32.6\%$; prediction interval $[-3.639, 1.493]\%$).

Projection-component sensitivity. To see which head drives the change in CKA, we recombine the separately trained RANDOM and TASK series and text heads. On Time-MMD, Linear CKA is 0.3402 for RANDOM/RANDOM (series/text), 0.4380 for RANDOM/TASK, 0.4667 for TASK/RANDOM, and 0.6575 for TASK/TASK. We refit the adapters of the mixed states over six backbones and three adapter seeds. With RANDOM/RANDOM as reference, the interaction $\Delta G_{\text{T,T}} - \Delta G_{\text{R,T}} - \Delta G_{\text{T,R}} + \Delta G_{\text{R,R}}$ (series head first; R: RANDOM, T: TASK) ranges from -1.998% to $+12.399\%$ across backbones; the three combinations and the interaction form one 24-comparison BH family. Four comparisons have $q < 0.05$: $\Delta G_{\text{T,R}}$ for iTransformer ($+3.37\%$)

Table 11: **RAW and CATS controls (%).** Gains and increments follow Equations 13–14. Bold marks $q < 0.05$ within the corresponding 30-cell correction family.

Backbone	Measure	Time-MMD	Financial	StockNet	FNSPID	DONKI/Kp
iTransformer	RAW gain	−2.14	0.52	−0.45	−0.72	0.21
	CATS gain	**3.35**	−2.38	−0.86	−3.18	−0.03
	CATS increment	0.81	0.72	0.21	−5.07	−0.37
TimeMixer	RAW gain	1.20	3.42	−0.67	1.30	1.76
	CATS gain	1.93	−4.28	−8.45	5.62	0.20
	CATS increment	−6.08	−4.84	−6.17	−9.11	0.39
PatchTST	RAW gain	1.32	2.24	−0.86	0.73	−0.12
	CATS gain	−1.90	−0.79	−12.74	4.95	0.15
	CATS increment	−8.60	0.77	−13.26	1.78	−0.83
MOMENT	RAW gain	−6.43	0.92	0.66	−0.75	8.03
	CATS gain	1.78	−1.09	1.53	−7.35	**8.93**
	CATS increment	2.37	0.41	2.61	−5.01	−0.19
TimesFM	RAW gain	2.76	2.70	3.56	−1.38	**12.35**
	CATS gain	−0.13	−2.85	0.66	−13.99	**11.76**
	CATS increment	−6.28	−1.66	−2.48	−4.05	2.74
Chronos	RAW gain	−2.48	3.00	−1.41	0.58	4.20
	CATS gain	5.76	−4.42	1.02	0.60	−0.13
	CATS increment	0.23	−2.64	0.49	0.65	−4.49

and MOMENT (-9.59%), and the interaction for MOMENT ($+7.80\%$) and TimesFM ($+7.81\%$). Because the heads are recombined after training, we treat this analysis as a component-sensitivity diagnostic.

F Utility of Executable Support Summaries

The deployed ECA summaries add little fitted assistance (Section 4.3). In this section, we test whether the fusion interface, candidate coverage, summary form, or interpretation weights explain this result. All conditions keep the original text, so the experiments measure the fitted utility that a summary adds beyond the text.

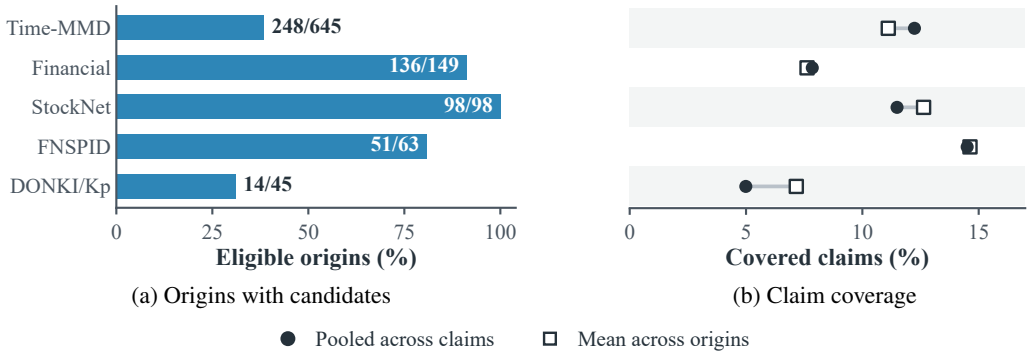
F.1 Summary Definition and Interface Controls

Executable summary. Residual adapters on a frozen `sentence-transformers/all-MiniLM-L6-v2` encoder (Wang et al., 2020) augment the backbone forecast with numerical inputs, original text, and optionally an ECA summary. For each atomic claim, ECA contributes seven candidate-weighted coordinates: Supported, Refuted, Unresolved, executable (Supported plus Refuted), grounded (structural series binding), numeric-atom, and temporal-atom mass. A claim without a covered candidate contributes $(0, 0, 1, 0, 0, 0, 0)$. Claim vectors are averaged with Historical-role weights, and a zero coordinate completes the eight-dimensional auxiliary branch. The shuffled condition gives each recipient the ECA summary of a different recipient with an earlier origin in the same task and split. One fixed assignment is shared by all fitted seeds, and no summary is recomputed. The assignment is many-to-one, and a donor summary serves at most nine recipients.

With the summary construction and learned parameters fixed, $\phi(X, S)$ is a deterministic function of the raw inputs, so $\mathbb{E}[Y \mid X, S, \phi(X, S)] = \mathbb{E}[Y \mid X, S]$ and the summary has no predictive innovation. It can still change fitted performance, because the adapter sees the inputs only after they are encoded. We report $\Delta_{\text{add}} = (R_{\text{text}} - R_{\text{text+ECA}})/R_{\text{text}}$ and $\Delta_{\text{pair}} = (R_{\text{text+shuffled ECA}} - R_{\text{text+ECA}})/R_{\text{text+shuffled ECA}}$ in percent.

Table 12: **Executable-summary utility across fusion interfaces.** Effects are median relative-MSE percentages across the 30 task–backbone cells. "Band" counts 90% intervals lying wholly inside the retrospective $\pm 1\%$ working band.

Interface	Text dims.	Add paired ECA		Paired vs. shuffled ECA	
		Median (%)	Band	Median (%)	Band
Small	32	2.699×10^{-4}	29/30	-1.235×10^{-4}	29/30
Base	32	2.395×10^{-4}	30/30	1.885×10^{-4}	30/30
Large	32	-4.991×10^{-4}	30/30	-3.237×10^{-6}	29/30
Full text	384	2.555×10^{-3}	30/30	2.450×10^{-4}	30/30
FiLM	384	-2.274×10^{-3}	30/30	-1.684×10^{-4}	30/30
Total	—	—	**149/150**	—	**148/150**



(a) Origins with candidates (b) Claim coverage

Figure 12: **Candidate availability and claim coverage use different denominators.** Counts in (a) refer to the fixed test populations.

Fusion controls. Small, Base, and Large fusion MLPs use widths $(64, 32)$, $(128, 64)$, and $(256, 128)$ over a 32-dimensional text projection; Full text keeps all 384 text coordinates, and FiLM modulates the numerical branch with text (Perez et al., 2017). Each interface covers five tasks, six backbones, and three adapter seeds.

All five interfaces give small median effects (Table 12). Of the three intervals not wholly inside the band, only Large/Time-MMD/TimesFM pairing (-3.229%, 90% interval $[-5.399, -1.483]\%$) lies wholly outside it. The $\pm 1\%$ band is a working criterion set retrospectively, not a preregistered equivalence test.

F.2 Candidate Coverage and Binding Diagnostics

A small effect could also arise because only some recipients or claims have candidates. Figure 12 reports the origins with at least one candidate separately from claim coverage, whose pooled and origin-mean versions weight claims and recipients differently.

For Time-MMD, 248 of 645 test origins contain candidates, and for Financial 136 of 149; the 11.1% and 7.6% values in Figure 12(b) are mean within-origin claim coverage. When we re-evaluate the fitted Base interface only on candidate-bearing recipients, with models, seeds, and resampling clusters unchanged, all 30 cells for both addition and pairing keep their 90% intervals inside the working band, as on the full populations. Recipients without candidates therefore do not account for the small effects.

Binding agreement. DeepSeek V4 Flash and V4 Pro independently receive each claim, its provenance, series definition, forecast origin, and pre-origin evidence, without ECA outputs or each other's responses; consensus requires matching role and verdict with confidence at least 0.80 from both. On a label-agnostic set of 101 claims, 98 reach consensus ($\kappa = 0.981$), and the typed interface agrees on all 98 labels at executable coverage 0.408. The labels are generated by models, not by

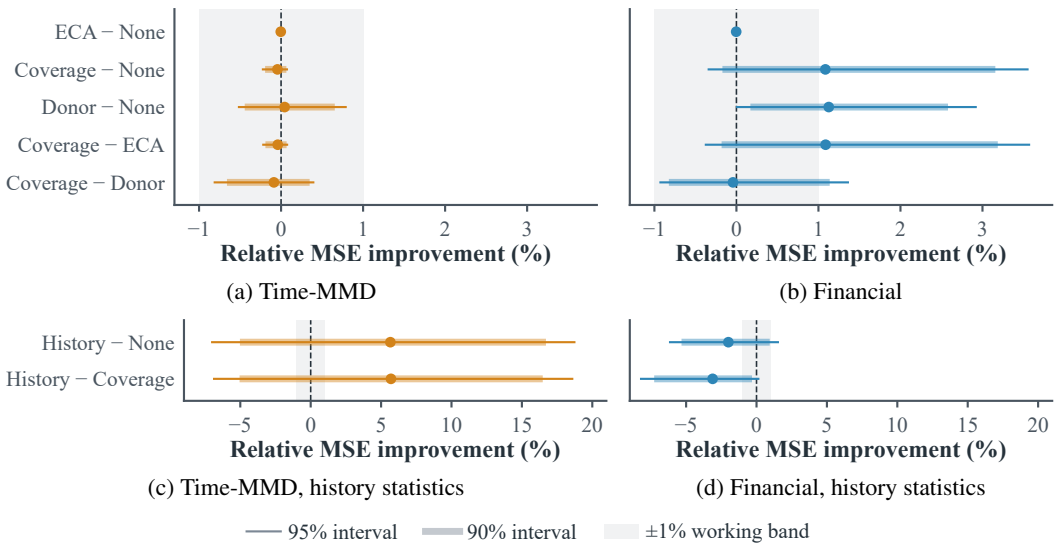


Figure 13: **Sensitivity to summary formation.** Each row A − B is the seed-averaged relative-MSE improvement of condition A over condition B (None: no summary; ECA: deployed ECA; Coverage: coverage-aware; Donor: donor summary; History: history statistics). Thin and thick bars are paired 95% and 90% intervals; shading marks the retrospective $\pm 1\%$ working band. Panels (c, d) use a wider scale. None of the 14 contrasts has $q < 0.05$.

humans, and we use them only for evaluation. Entity swaps and out-of-range times are rejected by the typed interface, whereas value edits remain executable and can change the verdict.

F.3 Summary Formation and Positive Control

We next fix PatchTST, numerical inputs, original text, and recipients on Time-MMD and Financial, and change only the auxiliary summary. We compare five conditions: no summary, the deployed ECA summary, a coverage-aware summary, a donor summary, and history statistics. The coverage-aware summary $\phi^{\text{cov}}_{1:7} = \rho\mu + (1-\rho)(0,0,1,0,0,0,0)$ drops role weighting and uses the candidate coverage ρ and the mean summary μ of candidate-bearing claims. The donor summary is the coverage-aware summary of the time-legal donor. History statistics are eight training-standardized statistics of the numerical history: mean, standard deviation, first, last, minimum, maximum, last minus first, and least-squares slope. The no-summary and deployed-ECA conditions reuse the Base-interface fits, and all conditions use seeds 101, 211, and 307.

On all 149 Financial origins, the coverage-aware summary improves over the deployed ECA summary by 1.0873%, and none of the 14 contrasts passes joint correction (Figure 13). Restricted to candidate-bearing recipients (248 Time-MMD origins in 74 clusters, 136 Financial origins in 122 clusters), the coverage-aware versus deployed-ECA effect is -0.129036% $[-0.964169, 0.148311]$ on Time-MMD and 1.257529% $[-0.241790, 3.973848]$ on Financial; against the donor summary it is -0.224821% and 0.138248%, with intervals that also include zero. Neither the coverage-aware summary nor correct pairing gives a significant improvement.

Same-interface positive control. To test whether the same eight-dimensional auxiliary branch can use a strong signal, we run a non-deployable control. We retain the PatchTST residual adapter but replace the summary with future targets: Time-MMD supplies its four targets and four zeros, while Financial supplies one target and seven zeros. Targets are standardized using training-target statistics; numerical inputs, original text, base forecasts, and eligible recipients are unchanged. The no-summary fits are reused, and test targets are introduced only as the oracle auxiliary input after model fitting and preprocessing. The oracle reduces Time-MMD MSE by 85.35% (95% interval $[79.39, 90.45]\%$), from 0.255035398 to 0.037359901, and Financial MSE by 85.71% ($[79.97, 92.94]\%$), from 0.000948207 to 0.000135451. The auxiliary branch can therefore use a strong signal when one is supplied.

Interpretation reweighting. This experiment fixes candidate sets, binding, execution, and aggregation and changes only the interpretation weights: the three-scorer ECA average at temperature 8 is reweighted by the selector of Appendix C.3. Reweighted and donor-reweighted summaries are fitted with the same three seeds and compared with the deployed and no-summary fits. Reweighting changes the summary at 135 of 645 Time-MMD and 33 of 149 Financial test origins. All six relative-MSE contrasts have magnitude below 0.006%, their 90% intervals lie inside the working band, and none passes joint BH correction (smallest $q = 0.0708$).

G Limitations and Future Work

Several questions remain open. The constructive separation in Section 3.2 is an identification result. It shows that historical support and the paired history–text distribution do not in general determine predictive innovation, but it does not tell how often such separations occur in naturally occurring language. The factual-reversal experiments cover numerical and temporal statements that can be checked directly against the observed series, and other semantic relations expressed by textual descriptions remain untested.

Native TSVer verification is a retrospective known-source system comparison. ECA uses historically reused calibration that is not nested within the scorer folds, and the compared systems do not use identical evidence preprocessing; a fully nested protocol would give a cleaner estimate. The forecasting bridge uses one frozen time-series/text encoder pair across five tasks and six backbones, and repeating it with other encoder pairs would be a natural extension. Several cell-level intervals are wide, and DONKI/Kp contains only six test clusters. The cross-cell synthesis summarizes the fixed task–backbone grid studied here, not a population of forecasting tasks or model families.

The executable-summary experiments keep the original text and fixed candidate sets. They measure the fitted utility that the supplied summaries add, and do not evaluate an end-to-end system that discovers evidence, builds interpretations, and forecasts from verified language. An important direction for future work is to train representations with factual contrasts or verified claims in the objective, as hard negatives do for vision–language compositionality (Yuksekgonul et al., 2023), and to test whether correspondence then becomes more informative about functional success. Naturally occurring factual contrasts and broader representation families would make such tests more general.