POLYEVENT: A LARGE-SCALE BENCHMARK FOR INDUCTIVE RELATIONAL EVENT FORECASTING

Anonymous authors
Paper under double-blind review

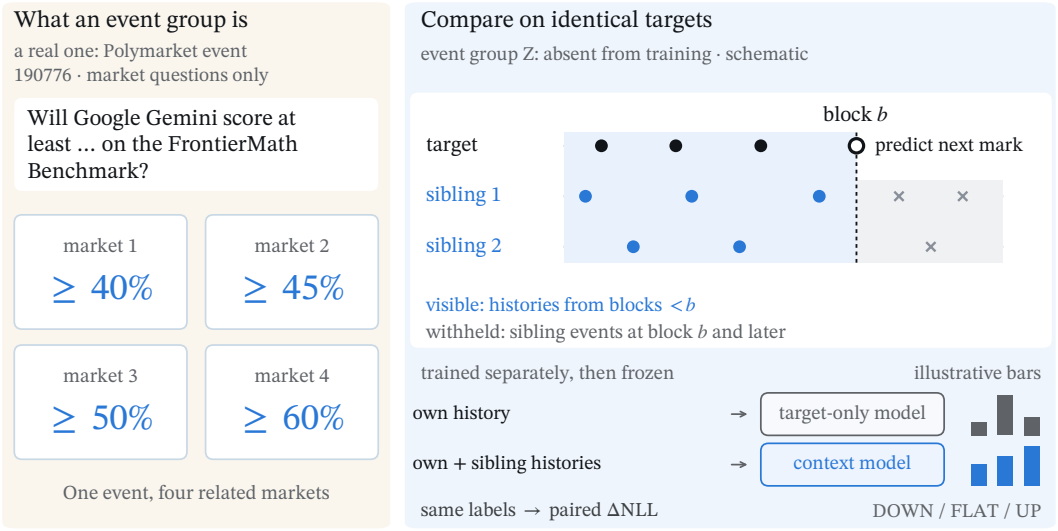

Figure 1: **What PolyEvent adds to event-sequence evaluation.** Left: a real event group illustrated by its market questions, selected without reading predictions. Right: a schematic group absent from training; models trained separately and frozen predict the same target rows using own history or additional strictly earlier same-group histories (Section 3.3); the models' predictive contrast does not by itself isolate information from architectural differences. All histories and output bars are schematic; the figure reports no estimates.

ABSTRACT

How should we evaluate forecasters on unseen event groups whose markets have trading histories? Reviewed benchmarks do not report this comparison. We introduce PolyEvent, an open event-sequence benchmark using Polymarket's official event groups. It aggregates 1.03B order-book fills into 281M (market, block) events across 652,016 groups; experiments use a fixed 10M-event subset. Tasks are binned block-gap prediction and three-way price-change prediction (UP, FLAT, DOWN); the relational track evaluates the latter. Separately trained own-history and context-augmented models are frozen and compared on identical targets from held-out multi-market groups, using strictly earlier-block information. Scores average over events (event-micro) or markets (market-macro). Construction choices—fill-level events, timestamp order, order-direction marks and calendar splitting—each change what is evaluated. As a use case, context lowers negative log-likelihood on a sealed whole-group holdout (primary market-macro $\Delta\text{NLL} = -0.0490$ nats), exceeding the 0.010-nat margin fixed before any sealed prediction. On development data, an audited placebo using other groups with matched sibling count yields $\Delta\text{NLL} = -0.0018$ nats, its interval containing zero; it does not isolate event-group identity. In a post-hoc study of 156 development threshold groups, features encoding threshold order improve over pooling and a capacity-matched shuffled-role control (market-macro $\Delta\text{NLL} = -0.0250$ nats against the control, simultaneous 95% CI $[-0.0377, -0.0122]$); event-micro gains are smaller. Data and code links will follow publication.

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.

1 INTRODUCTION

Markets in an event group absent from training may already have trading histories when a prediction is made. Can a forecaster use the earlier histories of related markets to predict the target market's next price change better than its own history alone? Polymarket's official *event groups* make this concrete: they link contracts about one real-world event, such as one per candidate or several thresholds, whose trades settle on-chain in block batches, often within a few blocks. Related benchmarks score next-event prediction, sequence classification, temporal-link prediction and order-book price direction, and relational stock forecasters report chronological splits. Table 4 (Appendix A) contrasts their published targets, holdout units and available histories. The reviewed evaluations do not directly report this comparison on groups excluded from training. Here, separately trained own-history and context-augmented models are compared on identical target rows, using strictly earlier blocks.

PolyEvent aggregates 1.03B order-book fills into 281M block-level events; experiments use a frozen 10M-event tier (Section 3.1). An event is a **(market, block) pair** ordered by block number, with two targets: the gap since the previous event and price-change direction (Section 3.2). The **independent-stream** track uses each market's own history; the **relational** track adds earlier same-group histories, withholding same-block siblings (Section 3.3). Platform grouping and temporal availability define siblings; we make no claim that siblings are causally related or driven by a particular news source. The main protocol is **cold-start by event** (Figure 1): no evaluated group is seen in training.

On a sealed whole-group holdout, same-group context gives market-macro $\Delta\text{NLL} = -0.0490$ nats (Section 4.1). On development data, context from a different event group with an identical number of siblings yields -0.0018, with an interval containing zero. History length, activity density, recency and absolute block position are not adequately matched, so this does not isolate event-group identity. Paired comparisons on identical rows reveal additional predictive structure on multi-market groups. In the 10M tier's training split, these account for 18.5% of groups, 66.2% of markets and 69.8% of events. Independent-stream reference baselines cover both single- and multi-market groups and score a separate population (Table 1).

The comparison also depends on what counts as an event, which histories are visible and how test groups are chosen. Four data-construction choices change what is evaluated: fill-level events leave most adjacent pairs with zero elapsed time; the snapshot's recorded timestamps are not monotone in block number; a contract-recording convention change mimics a behavioural shift in order-direction marks; and a calendar split selects a different test population. Section 4.4 quantifies each and gives the protocol that avoids it. These audits concern the construction of PolyEvent.

In summary, our main contributions are as follows:

- **(i) Benchmark and source data.** We release PolyEvent: 281M block-level events from 1.54M markets in 652,016 official event groups, linked to market questions, with split manifests, precomputed context, reference predictions and code. We also release the source corpus it is built from: five tables totalling 5.41B rows (raw fill logs, trades, a YES-normalised view, per-address records and market metadata).
- **(ii) An evaluation protocol.** Whole-group holdout, strictly earlier-block histories and paired scoring under both weightings measure the value of same-group history on unseen groups. Four construction audits and direct weighting tests quantify how construction and weighting choices change evaluation.
- **(iii) Predictive evidence.** A sealed evaluation finds a same-group gain on unseen groups. A post-hoc non-neural baseline (XGBoost), scored on development test, finds context gains both with and without the target's last price and elapsed blocks.
- **(iv) A structure-aware baseline.** In a post-hoc development-validation study on threshold groups, relative-threshold role features improve on pooling and a capacity-matched shuffled-role control.

2 RELATED WORK

Prediction-market data and modelling. The closest protocol we are aware of is EventX task T6 (EventX contributors, 2026), which also uses prediction-time sibling information but predicts *co-movement classes* over $1/3/7$-day horizons. PolyEvent instead evaluates next-event price marks

with whole official event groups held out. *Its split documentation and how its clusters relate to official event groups are quoted in Appendix A; we make no claim about the latter.* The closest transaction archive is the Polymarket-v1 Database: 1.20 billion trade records across 1.30 million markets, recovering ground-truth aggressor direction from the settlement layer (Qin & Yang, 2026). It covers the first-generation exchange to 2026-04-28, ours both generations to 2026-07-20; the counts are not comparable, and ours adds an evaluation corpus and protocol to the archive. Other work on these markets studies news, arbitrage, the market lifecycle and short-horizon pricing (Dubach, 2026; Tsang & Yang, 2026; Saguillo et al., 2025; Jia et al., 2026; Young, 2026; Yu et al., 2026); none targets held-out next-event prediction. PolyBench (Cheng et al., 2026) scores LLM forecasts on point-in-time snapshots with live news, and Prophet Arena (Yang et al., 2025) and Prediction Arena (Zhang et al., 2026) evaluate LLM and agent forecasts online; PolyEvent evaluates none of language models, news or agents and isolates a transaction-native sequence problem.

Relational event modelling. That economically linked assets predict one another is long established (Lo & MacKinlay, 1990; Hou, 2007; Cohen & Frazzini, 2008). *We do not present cross-market predictability as a new phenomenon.* Relational stock forecasters combine a stock's own history with industry or concept relations under chronological splits (Feng et al., 2019; Xu et al., 2021); holding out whole groups poses a different generalisation question, not a stricter one. Prior work estimates linkages from industry, supply chains or lead–lag dependence; here the platform supplies the grouping and evaluated groups are absent from training. Multivariate point processes represent dependence among event types (Butts, 2008; Xu et al., 2016; Zhang et al., 2021); inductive temporal-graph models embed nodes absent at training time (Xu et al., 2020); hierarchical forecasting reconciles predictions across an aggregation structure (Hyndman et al., 2011; Wickramasuriya et al., 2019). That is not the claim here: PolyEvent neither infers a causal graph nor assumes action between sibling markets.

Event-sequence and financial-sequence benchmarks. Neural temporal point processes model event streams through a learned intensity or density (Du et al., 2016; Mei & Eisner, 2017; Zuo et al., 2020; Zhang et al., 2020; Shchur et al., 2020; Yang et al., 2022), surveyed by Shchur et al. (2021). Table 4 contrasts the evaluation objects of EasyTPP (Xue et al., 2024), TGB-Seq, which shows that repeated edges can mask an inability to learn sequential dynamics (Yi et al., 2025), and FI-2010, roughly four million order-book events from five stocks over ten trading days, represented as 394,337 ten-event samples (Ntakaris et al., 2018). PolyEvent's time target is a discrete block gap, not a continuous-time intensity; event, edge and snapshot counts are not interchangeable, so we report corpus scale rather than claiming a rank across these suites. EBES (Osin et al., 2025) and TGB (Huang et al., 2023) are compared per cell in Appendix A.

3 THE POLYEVENT BENCHMARK

3.1 DATA AND BLOCK-BATCHED EVENT CONSTRUCTION

The corpus comprises Polygon CLOB fills from 2022-11-21 to 2026-07-20 across both CTF Exchange generations; the earlier automated-market-maker period and markets with no on-chain fill are absent. Complement-token fills are YES-normalised to $1 - p$, yielding one price series per market.

Fills settle in blocks, so treating each fill as an event creates large clusters of zero elapsed time, and the snapshot's timestamps are not monotone in block number (Section 4.4). We therefore define an event as a **(market, block) pair**, written U_B, and use block number as the ordering key. All fills of a market within one block form one event, priced at the last fill in the block by on-chain log index (Figure 8(a)). The event record carries no traded-size field; the two targets are functions only of block number and the block-close price sequence. The corpus contains $281{,}433{,}584$ events over $1{,}535{,}897$ markets and $652{,}016$ event groups.

Released corpus and reference-compute tier. The warm-start protocols keep 274M events after a minimum-length filter on market streams. Experiments use a deterministic 10M-event tier, drawn by hashing event groups and admitting all their markets and events from the cold-event manifest, which does not apply that length filter; tiers are therefore nested, and no event group is ever split by sampling. No model in this paper is trained on all 274M events (counts in Appendices C and Q).

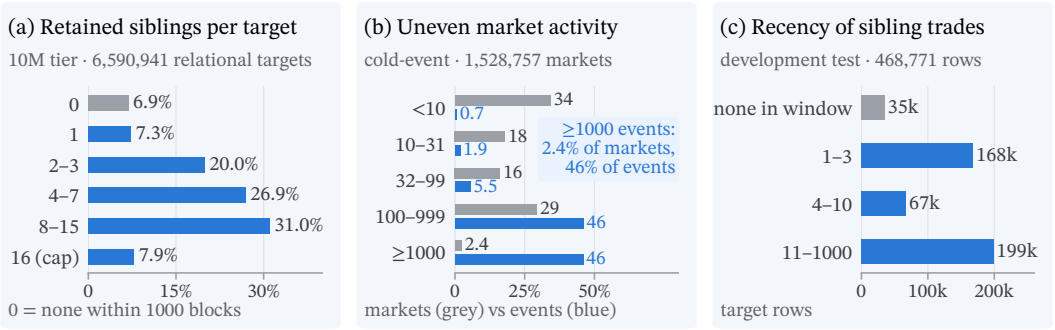

Figure 2: **Data and context profiles.** *Three different populations, one per panel; no cross-panel comparison is intended*, and the figure reports no predictive estimates. (a) The last bar is the retention cap $m_{\text{sib}}=16$, *not* a count of truncated groups. (b) Grey: share of retained markets; blue: share of retained event rows; both bucketed by the market's event count *in the full snapshot. This is why both event-micro and market-macro are reported* (Section 3.5); it says nothing about how far the two gains differ. (c) Blocks to the most recent visible sibling, on the development-test rows scored in Table 2.

3.2 Tasks and evaluation tracks

Each event carries two targets. **T1, the block gap**, is the number of blocks since the market's previous event: a discrete quantity of at least one, not a continuous inter-arrival time. Its canonical benchmark target is a categorical variable over eight bins. The relational track does not evaluate T1: given a target event at block b, it predicts the price mark from histories in blocks strictly earlier than b, not the event's timing. Earlier gaps are only an internal input to T2, and the current target's block gap is realised but *not* a reference-model input (Figure 8(c); bins: Appendix C).

T2, the price mark, is the sign of the change in the market's YES-normalised close from the previous event to the current one: UP, FLAT (the two recorded prices are exactly equal) or DOWN. A market's first event has no preceding price and hence no mark label; it is kept as context but not scored, leaving 279,897,687 labelled targets: 45.08% FLAT, 27.69% DOWN and 27.24% UP. The mark describes recorded trade prices, not a latent belief (Appendix Q).

Both tracks use the same corpus (Figure 8(c)). The **independent-stream track** treats every market as a separate sequence. The **relational track** also gives each target the histories of the other markets in its event group; such groups are a minority of groups but hold most markets and events (Section 1). Markets in multi-market groups appear in both tracks, so the two populations form a natural contrast rather than a matched one. Figure 2 profiles what a target sees.

3.3 Splits and online information sets

Splits are set from block coverage, counts and stream-length distributions alone, before any model is run and never from model performance. **Primary: cold-start by event.** Each group is assigned by the block of its first event, with boundaries B_1 and B_2 at the 0.8 and 0.9 event quantiles (Appendix C); events of a training group after B_1 are dropped. Three properties follow. **(i) Group-disjoint**: on the split manifest all three pairwise group intersections are empty, as are the market intersections; question templates and underlying entities can still recur. **(ii) Fitting-time ordering**: every training event precedes every test event. **(iii) Selection-time limitation**: 10.7% of validation events fall after B_2, so checkpoint selection overlaps test in block time. The benchmark therefore holds out start-block cohorts; it is not a strict online extrapolation. Secondary within-market prefix and cross-generation diagnostics are described in Appendix C; Section 4.4 reports a global calendar split only as a rejected design, not as a protocol.

Information set. Evaluation is one-step teacher-forced online: at every scored position the model is conditioned on the true history with fixed parameters, so no scored input contains a later observation (except in the permutation control; Section 3.4). Inputs come from strictly earlier blocks (Section 3.2), so sibling events that share block b with the target are not visible. On-chain logs order fills within a block; we use that order only to collapse fills into events and ignore it when forming information sets, so a prediction never sees sibling activity from the same compound on-chain action.

3.4 MODELS AND BLOCK-MASKED CONTEXT CONSTRUCTION

Independent-stream baselines. We report a marginal predictor, a per-market empirical predictor, a gradient-boosted model over hand-built history-derived features, a GRU and a causal Transformer, all seeing only the market's own $(\text{gap bin}, \text{mark})$ history: they differ in model family, not information.

Target-only (own-history) and event-context models. The two models compared in Section 4.1 share a target encoder, a GRU over the target's recent events, each represented by its gap bin and mark. The event-context model adds a sibling branch over the markets most recently active before block b within a fixed look-back window; sibling selection uses no information at or after b. Each retained sibling is encoded into one vector, and these vectors are combined by mean and max pooling over sibling markets, not over events. The two models share data, optimiser, batch size and update budget and differ only in the presence of the sibling branch (details, including an ablated co-movement feature, in Appendix C).

Permutation control. To separate the contribution of context from that of added parameters, we train the event-context model again with permuted context. The permutation is a **bijection within a split**: each target receives the sibling context of a different target from the same split, with exactly the same number of siblings, from a different event group, and never from itself. Beyond the sibling count, targets and donors are coarsened-matched on predefined bins of own-history and sibling-history length and, at the finest level only, of activity, sibling recency and target block; where cells cannot be matched, matching falls back to sibling count alone. Bin edges are fitted on the training split alone, and temporal position is not adequately controlled (Appendix C). Targets without an admissible donor are excluded from the comparison rather than paired with their own context; the excluded rows are reported and withheld from training in all arms, so the three arms train on identical rows.

Threshold-role arms. The pooling encoder receives sibling histories but no explicit relative-threshold features. In a threshold group the markets ask the same question at different levels, so each sibling has a rank relative to the target, parsed from the question texts. We test whether adding this relative order improves prediction on unseen groups (Figure 5). Three arms differ only in what is appended to each sibling's scalars: the pooling baseline B, the role-aware R, and a shuffled-role control $\mathrm{R}_{\mathrm{shuf}}$ whose market-to-order map is shuffled. R and $\mathrm{R}_{\mathrm{shuf}}$ match in parameter count and B does not, so $\Delta_{\mathrm{role}} = \mathrm{R} - \mathrm{R}_{\mathrm{shuf}}$ is the capacity-matched comparison. Permuting the map also moves the feature marginals, so neither contrast decomposes the gain into "role" and "capacity" parts (Appendix T).

3.5 METRICS AND INFERENCE

All results are negative log-likelihood (NLL) in nats on the held-out split. Model comparisons are NLL differences on identical evaluation rows, verified row-by-row.

Using the benchmark. The corpus is released whole; a submission is scored on the primary cold-event split at the 10M tier, the secondary splits being diagnostics. Independent-stream T1 and T2 report absolute NLL under both weightings, with no primary weighting; the relational T2 track serves three uses. For a predictive level, it reports absolute NLL on its fixed test rows under both weightings. To claim that context helps, a submission reports paired ΔNLL between explicitly named arms, a context model minus its own-history counterpart under the same training protocol. A claim to improve on existing methods needs no own-history arm. Methods are compared by group-clustered paired contrasts on identical rows, with the per-row predictions of the two frozen GRU arms as the recomputable reference. Sealed membership is released, so the sealed split is not a hidden test set. *We define no overall or composite PolyEvent score.*

We report two aggregations. **micro** is the mean of the per-event NLL over all scored events; **macro** is the mean over markets of the mean NLL within each market. Macro has exactly these two levels; event groups are not a third averaging level. Micro is dominated by long, heavily traded streams, whereas macro weights every market equally.

Uncertainty comes from a nested bootstrap that resamples event groups and training seeds jointly (2,000 replicates, 5 seeds per arm; micro, macro and their interaction share draws within a cell).

Table 1: **Independent-stream baselines under both protocols.** Test NLL under *event-micro* and *market-macro* weighting, lower is better; fitted rows average 5 seeds. *Cold-event* (813,347 test rows) is primary and *warm-prefix* diagnostic; they score different rows, so a cross-protocol difference is not a pure generalisation loss. *Point estimates*: the paired tests cover only XGBoost–GRU and GRU–Transformer (Appendix E). *Not comparable with Table 2*: the tracks score different rows.

	cold-event (primary)				**warm-prefix** (diagnostic)			
	T1 (block gap)		T2 (price mark)		T1 (block gap)		T2 (price mark)	
model	micro	macro	micro	macro	micro	macro	micro	macro
marginal	1.8795	2.1012	1.0724	1.1026	1.9148	1.9176	1.0138	1.0152
per-market	1.5480	1.8484	1.0121	1.0680	1.6907	1.5924	1.0003	1.0457
XGBoost	1.3128	1.5768	0.9471	1.0389	1.4062	1.4406	0.8626	0.9610
GRU	1.3014	1.5738	0.9034	1.0283	1.3770	1.4478	0.7315	0.8592
Transformer	1.3062	1.5820	0.9059	1.0329	1.3682	1.4318	0.7306	0.8574

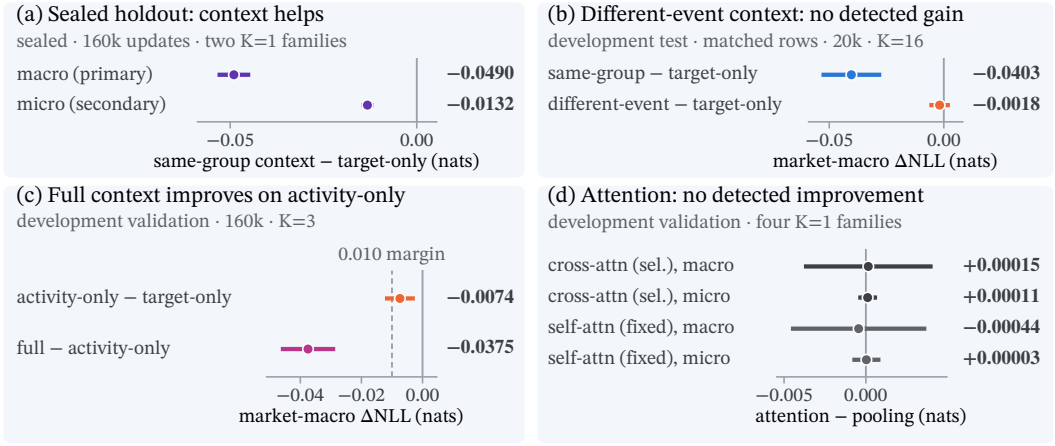

Figure 3: **Context gains and controls.** Every row states its own subtraction; bars are max-t simultaneous 95% intervals on identical rows. (a) Negative favours context. (b) Both arms are measured against target-only, *not* against each other. (c) Dashed: the 0.010-nat *point-estimate* margin, not a significance boundary. (d) Four $K{=}1$ families: cross-attention (validation-selected) minus frozen 160k pooling; self-attention (fixed 160k) minus a same-commit pooling refit. Neither difference is detected; this is not equivalence or an aggregator ranking (Appendices N, P). *Panels use distinct populations, budgets and inference families; no cross-panel magnitude comparison is intended.*

Event groups are the resampling cluster because their markets are related by construction; seeds are resampled because a row-only interval excludes training variance.

When several quantities are judged together, we report max-t simultaneous 95% intervals with Holm-adjusted p-values and state the family size with every set of intervals. Critical values from families of different sizes are not comparable, so we never compare interval widths across families.

Where a weighting is claimed to change a conclusion, we test the interaction contrast $D = \Delta_{\text{micro}} - \Delta_{\text{macro}}$ directly: significance under only one weighting does not establish that the two differ.

Checkpoint selection. Architecture, hyperparameters and early stopping are decided on validation data only. The two models compared on test were fixed, with their configuration and feature-construction fingerprints recorded, before the test split was read.

4 Results

We report relational gains, confirmed on sealed unseen groups (§4.1); their conditions and context controls, which do not identify a causal role of group identity (§4.2); method comparisons, in which a role-aware variant beats pooling and a capacity-matched control in a post-hoc study on development threshold groups (§4.3); and construction audits (§4.4). Each predictive comparison is a paired contrast between named arms under both weightings, neither a priori correct (§3.5).

Table 2: **Relational capability within model families.** Next-price-mark NLL on the same 468,771 development-test rows (3,704 markets, 535 held-out groups), 5 seeds per arm; Δ is context minus target-only on identical rows, with max-t simultaneous 95% intervals clustered on event groups. *The two model families are separate inference families* (GRU: 16 quantities, 20k updates; XGBoost: $K{=}3$, ≤ 400 rounds, *post-hoc*): interval widths are not comparable and no cross-family contrast is tested. (+P,+G) gives both arms the target's last price P and the blocks G elapsed since its last event; post-hoc, own $K{=}3$ family (Appendix R).

model	weighting	own-history	+ context	**paired ΔNLL** [95% CI]
GRU	event-micro	0.8236	0.8148	$\mathbf{-0.0088}$ $[-0.0111, -0.0065]$
GRU	market-macro	1.0238	0.9822	$\mathbf{-0.0416}$ $[-0.0550, -0.0282]$
XGBoost	event-micro	0.8715	0.8623	$\mathbf{-0.0092}$ $[-0.0167, -0.0017]$
XGBoost	market-macro	1.0350	1.0046	$\mathbf{-0.0304}$ $[-0.0390, -0.0218]$
XGBoost (+P,+G)	event-micro	0.8379	0.8318	$\mathbf{-0.0062}$ $[-0.0111, -0.0013]$
XGBoost (+P,+G)	market-macro	0.8480	0.8256	$\mathbf{-0.0224}$ $[-0.0296, -0.0152]$

4.1 What the benchmark measures on unseen event groups

Table 1 gives reference levels: the contract separates the GRU from XGBoost on independent-stream T2 in all four protocol–weighting cells; the GRU–Transformer order depends on protocol and task (Appendix E). Development relational results use the cold-event protocol (Figure 1).

For scale, on this development market-macro estimand a training-frequency marginal predictor scores 1.0795 (uniform: $\log 3 = 1.0986$). Same-group context lowers market-macro NLL by 4.07% relative to target-only, on this population and weighting only. The marginal is a same-population reference, not a task-difficulty estimate. Context lowers next-mark NLL in both model families (Table 2); the GRU market-macro -0.0416 comes from one max-t family of 16 quantities (c=3.02; Table 6).

A non-neural baseline with fixed summaries. In the post-hoc XGBoost pair of Table 2, 19 fixed summaries give market-macro ΔNLL -0.03039 $[-0.03901, -0.02176]$, past the 0.010 margin, while event-micro is detectable but sub-margin; we draw no cross-model inference. With the target's last price and elapsed blocks, the gain persists (+P,+G).

Confirmation on a sealed split. Given the known development weighting sensitivity, the 12,800-group sealed membership and the final-160k market-macro contrast as the primary estimand were fixed before scaling. After scaling, but before any sealed construction or prediction, we fixed the decision rule (ΔNLL ≤ -0.010 nats with a 95% interval excluding zero) and separate $K{=}1$ macro/micro families. One run generated predictions from ten frozen scaling checkpoints with no weight or preprocessing refit. Market-macro ΔNLL is $\mathbf{-0.0490}$ nats (95% CI $[-0.0534, -0.0446]$; 2,129 context-eligible groups, 15,562 markets, 3,462,333 target events; 5 seeds; Figure 3(a)), a 5% higher geometric-mean probability of the realised mark with markets weighted equally. The prespecified primary criterion was met. The secondary event-micro estimate, -0.0132 nats (CI $[-0.0148, -0.0116]$), also met both conditions but did not determine the verdict. The macro estimate is $3.7\times$ the micro one; their ratio is neither a market-size stratum nor a mechanism (Section 5). Membership is selected by group-ID hash without a future-period restriction, so the evaluation tests unseen groups rather than later periods, with a different population and budget (Appendix G).

4.2 When the gain appears, and what the controls allow

The relational gain grows with training-set size at a fixed update budget. Over nested prefixes of $64/256/1024/2561$ groups at a fixed 160k-update budget (Figure 4(a,b)), the preregistered market-macro endpoint trend is -0.0318 $[-0.0427, -0.0209]$. Size and repeat exposure are mechanically collinear and each tier uses a single deterministic subset, so this is not an isolated data-quantity effect and the seeds cover optimisation randomness only (Appendix L).

On development matched rows (20k updates) we compare same-group with different-event context (Section 3.4; Figure 3(b)). Capacity is unchanged and history length only coarsely matched (not at all where matching falls back to sibling count); density, recency and absolute block position are *not* equalised, so the control rules out capacity differences *but not residual history-length, temporal or activity-pattern differences* (Appendix O). The different-event arm shows no detected improvement over target-only, -0.0018 (Holm $p = 0.43$), while the arm-to-arm contrast is -0.0385 (Table 6).

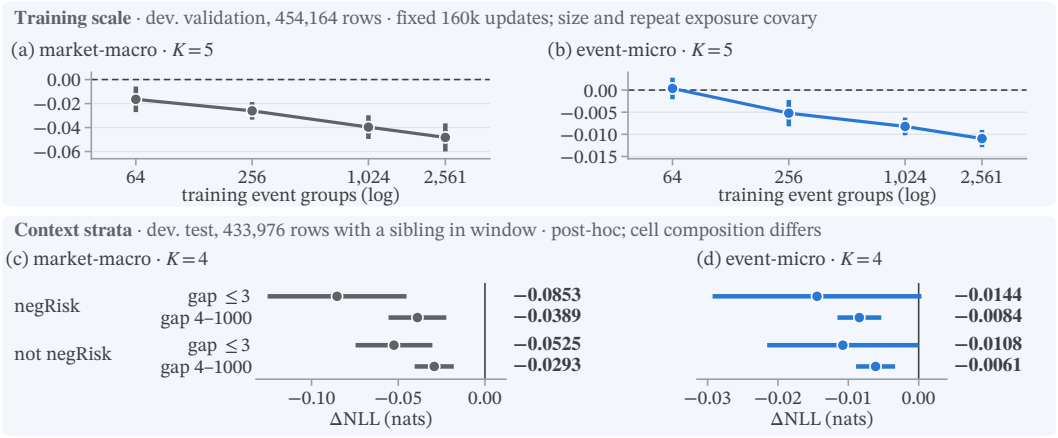

Figure 4: **Training scale and context strata.** ΔNLL: sibling context minus target-only on identical rows; negative favours context. Simultaneous 95% max-t intervals use separate market-macro and event-micro families. *Top*: four tiers and the endpoint trend (K=5); adjacent-tier differences are not tested (Appendix L). *Bottom*: platform `negRisk` × nearest-sibling block gap (K=4); 4–1000 blocks is roughly eight seconds to half an hour. Strata neither decompose the pooled gain nor identify causes (Appendix S). The two blocks have different populations and families; no cross-block comparison.

A second alternative is that siblings convey only recency and frequency. We test it in a separate, post-hoc 160k-update development-validation experiment, using new refits rather than the sealed checkpoints: under market-macro, a restricted activity-only arm (A) stays below the 0.010 margin against target-only, and the full one gives a joint increment beyond the 0.010-nat margin over that arm (Figure 3(c); Table 8). Sibling arms match nominal parameter count, but A's zeroed mark embedding is unreachable and untrained, so this is a joint predictive channel contrast, not a mechanism (Appendix K).

Post-hoc checks by realised mark and capped own-history length are descriptive (Appendix M); on development-test rows event-micro ΔNLL changes from -0.0438 at one earlier own event to a still-detected -0.0066 at the 32-event cap (87.8% of rows). Post-hoc strata show the gain in non-`negRisk` groups, and larger market-macro gains at sibling gaps ≤ 3 (Figure 4(c,d); Appendix S). We do not attribute the gain to event-group identity (Section 5), and the development event-micro gain (not the sealed estimate) does not reach 0.010 nats; on matched rows its placebo interval contains zero although Holm $p = 0.044$, a reading sensitive to multiplicity correction (Appendix D).

4.3 Which method changes are detected

The contract did not distinguish two attention aggregators from mean–max pooling at the evaluated budget (Figure 3(d); Appendices N, P). On the 156 parsed threshold groups (Figure 5; §3.4), a post-hoc development-validation study gives market-macro ΔNLL -0.0250 $[-0.0377, -0.0122]$ against the shuffled-role control and -0.0255 against pooling; event-micro gains are smaller (Appendix T). On identical rows the contract thus separates this role-aware variant from a capacity-matched control: an initial example of improving unseen-group prediction by exposing relative contract structure to the model. Attention and role studies differ in population, budget and reference and are not ranked.

4.4 Why the evaluation design matters

Weighting changes a model comparison. On identical independent-stream predictions at the warm-prefix T1 endpoint, the weightings put the GRU and XGBoost point estimates in the opposite order. Component intervals include zero; the directly tested weighting interaction excludes zero ($D = -0.0364$, Holm $p = 0.0011$; Figure 9): a point-ordering flip, not two established winners.

Four construction choices alter what is evaluated, each violating a requirement of the intended evaluation (not an exhaustive list); Figure 6 measures three of them, Figure 7 the fourth.

The event unit determines how often elapsed times are zero. Taking raw fills as events, 72.75% of adjacent within-market pairs have zero elapsed time; block aggregation reduces this to 0.000406%.

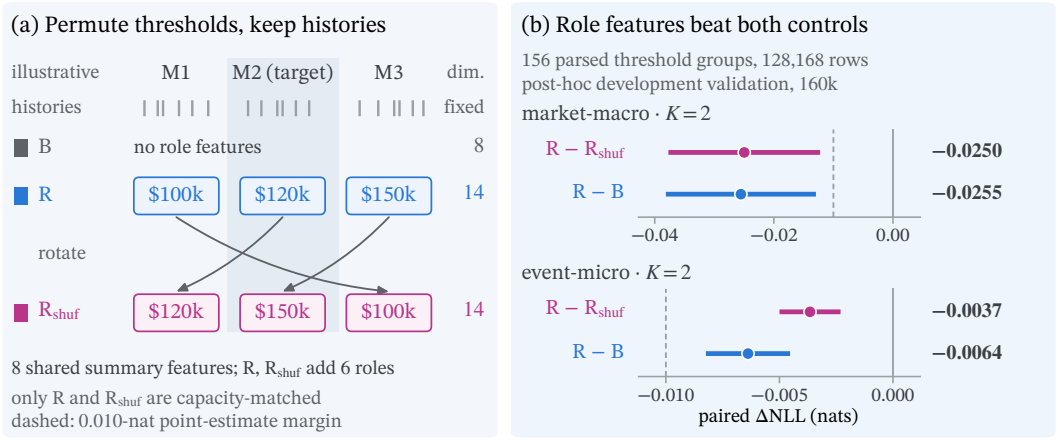

Figure 5: **Threshold-role features and predictive gains.** (a) A derangement moves every visible market's threshold, including the target's; histories stay put. It also moves feature marginals: neither contrast decomposes role and capacity effects (Appendix T). (b) Bars show simultaneous 95% max-t intervals; negative favours R. Market-macro and event-micro form *two separate* $K=2$ *families*, with no cross-family contrast. Micro estimates stay inside the margin. *Post-hoc development validation, not sealed.*

Timestamps do not always preserve canonical block order. Our timestamps mix fetched and interpolated block times; under them 62,060 adjacent pairs carry negative elapsed time (smallest -196 s), so we order by block and make no claim about later data or other chains (Appendix B).

A change in recording convention can be mistaken for drift. Between contract generations a volume-weighted buy share moves 39.9 percentage points, the price-change mark on the same events at most 0.80. Different denominators (volume vs. events) make them incomparable (Appendix B).

A global calendar split changes the population it evaluates. Splitting by calendar time leaves 5,288 distinct test markets against 268,180 under cold-start-by-event (market counts are non-disjoint across splits), inflating the FLAT test share — a selection effect, not temporal drift.

5 Limitations

The relational result applies only to multi-market event groups, not to single-market events, and sealed gains are not stratified by group structure. The sealed split (Section 4.1) tests unseen event groups under the observed regimes, not a future-calendar evaluation, and cannot separate same-group context from contemporaneous market-wide conditions. We do not attribute the gain to event-group identity; news, arbitrage or traders active in several markets cannot be separated here. Evidence comes from one platform and a 10M-event tier, not the full 274M eligible view. The primary relational comparison evaluates one frozen same-event sibling-pooling architecture; it does not establish optimality over alternative relational encoders. The original gap–mark baselines omit price levels and the blocks since the target's last event (Appendix R adds both to XGBoost; neural pairs with both inputs are untested). Only macro determines the sealed verdict; development event-micro does not reach 0.010 nats. The development-test evaluation includes an implementation-audit correction and is not an untouched one-shot evaluation. Full list and timeline: Appendices F and G.

6 Conclusion

PolyEvent makes relational forecasting on unseen event groups reproducibly measurable; its corpus, splits, protocol and reference baselines are released with this paper (Appendix Q). As a use case, same-event sibling context improves next-mark prediction, confirmed on a sealed holdout scored once (Section 4.1). In post-hoc development studies, context gains hold for a non-neural model, also given the target's last price and elapsed blocks (Appendix R), and on threshold groups role features improve over both pooling and a capacity-matched shuffled-role control (Appendix T). The audited construction choices change what is evaluated, and weighting changes a model comparison.

ETHICS STATEMENT

Data provenance and what is released. The corpus is derived from publicly readable Polygon transactions recording fills on Polymarket's central limit order book. The release has two layers. **The source corpus** (Appendix Q) reproduces the on-chain fill records: its raw fill logs, trades and per-address records carry the pseudonymous maker and taker addresses that every fill already carries on chain, and it adds no off-chain identity information. **The benchmark layer** consists of the event-level tables the benchmark is defined on — (`market_id, event_id, block_number, event_index, price_mark, delta_block, split`) — a market-level metadata table (question text, `negRisk` flag, creation time) linked by `market_id`, the split manifests, the training and evaluation code, and the frozen artefacts every reported number is extracted from. **The benchmark layer contains no account-level, address-level or order-level table, and no per-trader sequence**: its event tables carry no address field, so no benchmark task or reported model uses who traded.

Pseudonymity is not anonymity. On-chain addresses are pseudonymous, not anonymous: an address can be linked to an identity by information outside the chain, and the underlying transactions remain publicly readable whether or not we redistribute them. The source corpus copies addresses that are already public but makes them easier to analyse; *this risk is not removed, and we do not claim it is*. We link no address to an off-chain identity, and users of the corpus should not attempt to do so.

Terms of service and redistribution. The trade records are read from the public chain rather than from Polymarket's API; market metadata comes from the platform's public market listing. Licensing and redistribution terms accompany each layer of the release, and we will honour removal requests for any record a rights-holder identifies.

Foreseeable use and misuse. The benchmark targets next-event prediction on already-recorded data. It is a forecasting benchmark, not a trading system: nothing here is evaluated for profitability, execution cost, market impact or adversarial participants, and the cold-event protocol deliberately removes the temporal continuity a deployed strategy would rely on. *Reading any result in this paper as evidence about trading performance would be a misuse.* Prediction markets also carry regulatory constraints that differ by jurisdiction and are outside the scope of this work.

REPRODUCIBILITY STATEMENT

The supplementary material contains anonymised code for split construction, sibling-context construction, training, evaluation and figure generation, together with the frozen result files that every reported number is extracted from, their provenance records and content hashes, and the consistency checks that verify the manuscript against them. It supports auditing the reported results. **The event tables and split manifests are not included** in full: they are too large for the supplementary material, so the experiments cannot be re-run end to end from this bundle alone; a deterministic sample is included instead. The tables are released with their licensing and redistribution terms (Appendix Q); public links to the data release and code repository are withheld for anonymous review and will be added upon publication. Availability is tiered, by file. **In this bundle**: the construction, training, evaluation and audit scripts named in the appendices; the frozen JSON result files behind every reported number, with their attestations and content digests; the consistency checks, runnable in place; and a deterministic data sample (`benchmark/results/data_sample`): about 2% of the event groups of the 10M tier with their events, split assignment, market metadata, relational context and matching development prediction rows, plus the raw source-corpus rows of three of those groups, with a quick-start script that trains and scores simple baselines on it. **Not in this bundle**, beyond that sample: the event tables, the split manifests, the sealed membership table, the sibling role tables, and every per-row prediction file. **Already public**: the five-table source corpus with its collection pipeline (Appendix Q), whose address we withhold here to preserve anonymity and will add upon publication. The files in the second group accompany the data release; where the text says a resource is released, it refers to that release and not to this bundle. Section 3 fixes the event unit, the ordering key, the two targets and the split protocols; Appendix C gives the tier counts, the canonical T1 target, the relational gap-input tokenisation, the context parameters and the optimisation budget. Every quantity in this paper is extracted from a frozen result file by script rather than transcribed.

AI USE STATEMENT

All research questions, study design, experiments and analyses were conceived and directed by the authors. We used large language models to aid and polish the writing and for retrieval and discovery of related work. LLM-based coding assistants were also used, under the authors' direction, to help implement and debug code. All quantitative results were produced by executing code on the data, and the authors take full responsibility for the paper.

REFERENCES

Xavier Bouthillier, Pierre Delaunay, Mirko Bronzi, Assya Trofimov, Brennan Nichyporuk, Justin Szeto, et al. Accounting for variance in machine learning benchmarks. In *Proceedings of Machine Learning and Systems*, volume 3, 2021.

Carter T. Butts. A relational event framework for social action. *Sociological Methodology*, 38(1): 155–200, 2008. doi: 10.1111/j.1467-9531.2008.00203.x.

Pu Cheng, Juncheng Liu, and Yunshen Long. PolyBench: Benchmarking LLM forecasting and trading capabilities on live prediction market data. *arXiv preprint arXiv:2604.14199*, 2026. URL https://arxiv.org/abs/2604.14199.

Lauren Cohen and Andrea Frazzini. Economic links and predictable returns. *The Journal of Finance*, 63(4):1977–2011, 2008.

Mostafa Dehghani, Yi Tay, Alexey A. Gritsenko, Zhe Zhao, Neil Houlsby, Fernando Diaz, Donald Metzler, and Oriol Vinyals. The benchmark lottery. *arXiv preprint arXiv:2107.07002*, 2021. URL https://arxiv.org/abs/2107.07002.

Nan Du, Hanjun Dai, Rakshit Trivedi, Utkarsh Upadhyay, Manuel Gomez-Rodriguez, and Le Song. Recurrent marked temporal point processes: Embedding event history to vector. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1555–1564, 2016.

Philipp D. Dubach. The anatomy of a decentralized prediction market: Microstructure evidence from the Polymarket order book. *arXiv preprint arXiv:2604.24366*, 2026. URL https://arxiv.org/abs/2604.24366.

EventX contributors. EventX: A multimodal benchmark linking social media posts to prediction market dynamics, 2026. URL https://github.com/mlsys-io/EventXBench. Task T6 release t6.kdd.v2, July 2026.

Fuli Feng, Xiangnan He, Xiang Wang, Cheng Luo, Yiqun Liu, and Tat-Seng Chua. Temporal relational ranking for stock prediction. *ACM Transactions on Information Systems*, 37(2):Article 27, 2019. URL https://arxiv.org/abs/1809.09441.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64(12): 86–92, 2021.

Kyle Gorman and Steven Bedrick. We need to talk about standard splits. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2786–2791, 2019.

Kewei Hou. Industry information diffusion and the lead-lag effect in stock returns. *The Review of Financial Studies*, 20(4):1113–1138, 2007.

Shenyang Huang, Farimah Poursafaei, Jacob Danovitch, Matthias Fey, Weihua Hu, Emanuele Rossi, Jure Leskovec, Michael Bronstein, Guillaume Rabusseau, and Reihaneh Rabbany. Temporal graph benchmark for machine learning on temporal graphs. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2023. URL https://arxiv.org/abs/2307.01026.

Rob J. Hyndman, Roman A. Ahmed, George Athanasopoulos, and Han Lin Shang. Optimal combination forecasts for hierarchical time series. *Computational Statistics & Data Analysis*, 55(9): 2579–2589, 2011.

Huaiyu Jia, Luofeng Zhou, Wentao Zhang, Lin William Cong, Siguang Li, and Shuo Sun. Unlocking the forecasting economy: A suite of datasets for the full lifecycle of prediction market. *arXiv preprint arXiv:2604.20421*, 2026. URL https://arxiv.org/abs/2604.20421.

Sayash Kapoor and Arvind Narayanan. Leakage and the reproducibility crisis in machine-learning-based science. *Patterns*, 4(9), 2023.

Andrew W. Lo and A. Craig MacKinlay. When are contrarian profits due to stock market overreaction? *The Review of Financial Studies*, 3(2):175–205, 1990.

Hongyuan Mei and Jason Eisner. The neural Hawkes process: A neurally self-modulating multivariate point process. In *Advances in Neural Information Processing Systems*, 2017.

Adamantios Ntakaris, Martin Magris, Juho Kanniainen, Moncef Gabbouj, and Alexandros Iosifidis. Benchmark dataset for mid-price forecasting of limit order book data with machine learning methods. *Journal of Forecasting*, 37(8):852–866, 2018. doi: 10.1002/for.2543.

Dmitry Osin, Igor Udovichenko, Egor Shvetsov, Viktor Moskvoretskii, and Evgeny Burnaev. EBES: Easy benchmarking for event sequences. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 5730–5741, 2025. doi: 10.1145/3711896.3737428.

Boka Qin and Rui Yang. Polymarket-v1 Database. *arXiv preprint arXiv:2606.04217*, 2026. URL https://arxiv.org/abs/2606.04217.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

Oriol Saguillo, Vahid Ghafouri, Lucianna Kiffer, and Guillermo Suarez-Tangil. Unravelling the probabilistic forest: Arbitrage in prediction markets. *arXiv preprint arXiv:2508.03474*, 2025. URL https://arxiv.org/abs/2508.03474.

Oleksandr Shchur, Marin Biloš, and Stephan Günnemann. Intensity-free learning of temporal point processes. In *International Conference on Learning Representations*, 2020.

Oleksandr Shchur, Ali Caner Türkmen, Tim Januschowski, and Stephan Günnemann. Neural temporal point processes: A review. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, 2021.

Kwok Ping Tsang and Zichao Yang. Political shocks and price discovery in prediction markets: Evidence from the 2024 U.S. presidential election. *arXiv preprint arXiv:2603.03152*, 2026. URL https://arxiv.org/abs/2603.03152.

Shanika L. Wickramasuriya, George Athanasopoulos, and Rob J. Hyndman. Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization. *Journal of the American Statistical Association*, 114(526):804–819, 2019.

Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. Inductive representation learning on temporal graphs. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rJeW1yHYwH.

Hongteng Xu, Mehrdad Farajtabar, and Hongyuan Zha. Learning Granger causality for Hawkes processes. In *Proceedings of the 33rd International Conference on Machine Learning*, 2016.

Wentao Xu, Weiqing Liu, Lewen Wang, Yingce Xia, Jiang Bian, Jian Yin, and Tie-Yan Liu. HIST: A graph-based framework for stock trend forecasting via mining concept-oriented shared information, 2021. URL https://arxiv.org/abs/2110.13716.

Siqiao Xue, Xiaoming Shi, Zhixuan Chu, Yan Wang, Hongyan Hao, Fan Zhou, Caigao Jiang, Chen Pan, James Y. Zhang, Qingsong Wen, Jun Zhou, and Hongyuan Mei. EasyTPP: Towards open benchmarking temporal point processes. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/e66233a208ef32f56df6312263239fa0-Abstract-Conference.html.

Chenghao Yang, Hongyuan Mei, and Jason Eisner. Transformer embeddings of irregularly spaced events and their participants. In *International Conference on Learning Representations*, 2022.

Qingchuan Yang, Simon Mahns, Sida Li, Anri Gu, Jibang Wu, and Haifeng Xu. LLM-as-a-prophet: Understanding predictive intelligence with Prophet Arena. *arXiv preprint arXiv:2510.17638*, 2025. URL https://arxiv.org/abs/2510.17638.

Lu Yi, Jie Peng, Yanping Zheng, Fengran Mo, Zhewei Wei, Yuhang Ye, Zixuan Yue, and Zengfeng Huang. TGB-Seq benchmark: Challenging temporal GNNs with complex sequential dynamics. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/db5ca61dbc08cf5143c05ad2d1b0b2ca-Abstract-Conference.html.

Gregory Young. OpenMarket: A synchronized Polymarket-Binance dataset for high-frequency prediction-market research. *arXiv preprint arXiv:2607.26245*, 2026. URL https://arxiv.org/abs/2607.26245.

Haofei Yu, Yining Zhao, Guanyu Lin, and Jiaxuan You. Building social world models with large language models. *arXiv preprint arXiv:2606.11482*, 2026. URL https://arxiv.org/abs/2606.11482.

Jaden Zhang, Gardenia Liu, Oliver Johansson, Hileamlak Yitayew, Kamryn Ohly, and Grace Li. Prediction arena: Benchmarking AI models on real-world prediction markets. *arXiv preprint arXiv:2604.07355*, 2026. URL https://arxiv.org/abs/2604.07355.

Qiang Zhang, Aldo Lipani, Omer Kirnap, and Emine Yilmaz. Self-attentive Hawkes process. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 11183–11193, 2020.

Qiang Zhang, Aldo Lipani, and Emine Yilmaz. Learning neural point processes with latent graphs. In *Proceedings of the Web Conference 2021*, pp. 1495–1505, 2021. doi: 10.1145/3442381.3450135.

Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. Transformer Hawkes process. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 11692–11702, 2020.

APPENDIX ROADMAP

The appendix has four kinds of content, and the kind determines what a section can be used for. **(1) Benchmark and artefact** documents the resource and the protocol. **(2) Alternative explanations tested** reports controls that could have overturned the relational reading; each is development evidence. **(3) Null and negative results** reports comparisons that did not separate, including one construction we attempted and abandoned. **(4) Descriptive reanalyses** redescribes rows already scored and identifies nothing. *Only the sealed confirmation of Section 4.1 (*Confirmation on a sealed split*) is confirmatory; everything in (2), (3) and (4) is development or descriptive, and none of it enters a headline number.* Table 3 maps each result family to its evaluation population, budget and evidential role. Four close development-validation readings of the 160k context contrast differ by run, not by sampling: -0.0477 (Appendix J) and -0.0483 (Appendix L, 2,561 groups) select each checkpoint by validation event-micro NLL, whereas -0.0449 (Appendix K) and -0.0435 (Appendix O) are same-commit refits read at a fixed endpoint. Their intervals do not cover trainer-version or design differences.

Table 3: **Experiment map and scoring contract.** The reporting contract is fixed in Section 3.5; the headline and sealed contrast is sibling context minus target-only. The primary/secondary pair is only this paper's sealed confirmatory rule (market-macro primary, event-micro secondary), not a property of the benchmark; *we define no overall or composite PolyEvent score*, and no cross-row contrast is tested.

family (track → target)	evaluation population; budget	score / role
Rel. T2 · sealed confirmation (App. G)	sealed, 12,800 (2,129 elig.); 160k	paired ΔNLL; **confirmatory**
Rel. T2 · headline, placebo (§§4.1–4.2)	dev. test, 535 groups; 20k	paired ΔNLL; development
Rel. T2 · donor mapping (App. J)	dev. validation; 160k	paired ΔNLL; development
Rel. T2 · scaling (App. L)	dev. validation; 160k	paired ΔNLL; development
Rel. T2 · activity control (App. O)	dev. validation; 160k	paired ΔNLL; development
Rel. T2 · channel ablation (App. K)	dev. validation; 160k	paired ΔNLL; development
Rel. T2 · aggregators (App. N, P)	dev. validation; 160k	paired ΔNLL; development, null
Rel. T2 · reanalyses (App. M, S)	dev. test; none	paired ΔNLL; descriptive
Rel. T2 · non-neural pair (App. R)	dev. test; ≤400 rounds	paired ΔNLL; post-hoc
Rel. T2 · (+P) pair (App. R)	dev. test; ≤400 rounds	paired ΔNLL; post-hoc
Rel. T2 · (+P,+G) pair (App. R)	dev. test; ≤400 rounds	paired ΔNLL; post-hoc
Rel. T2 · role-aware arms (App. T)	dev. validation; 160k	paired ΔNLL; post-hoc
Ind. T1/T2 · baselines (Tab. 1)	cold-event test, 813,347 rows	reference NLL; descriptive

(1) Benchmark and artefact	
Sources for the benchmark comparison (Table 4)	App. A
Implementation details	App. C
Independent-stream baselines and block gaps	App. E
Construction audits: event unit, ordering, marks and splits	App. B
Evaluation governance and the audit timeline	App. G
Extended limitations	App. F
Data card	App. Q
(2) Alternative explanations tested	
Capacity — permutation under three donor assignments	App. J
Activity — cross-group activity summaries	App. O
Activity — three-arm channel-group ablation	App. K
Model-family specificity — non-neural XGBoost pair	App. R
Method separation — role-aware sibling features (threshold groups)	App. T
Contamination between siblings	App. H
Weighting — micro versus macro reverses a ranking	App. D
Scale — what the scaling curve does not separate	App. L
(3) Null and negative results	
Two sibling aggregators not distinguished	App. N
A third aggregator: attention among siblings	App. P
Absolute-time matching found infeasible	App. I
(4) Descriptive reanalyses *(identify nothing)*	
Per class; by own-history length; by realised class	App. M
By `negRisk` status and sibling recency	App. S

A SOURCES FOR THE BENCHMARK COMPARISON

Our audit follows work on leakage, splits, ranking stability, variance and documentation (Kapoor & Narayanan, 2023; Gorman & Bedrick, 2019; Recht et al., 2019; Dehghani et al., 2021; Bouthillier et al., 2021; Gebru et al., 2021).

Each cell of Table 4 was checked against the work's own paper (and, where stated, its released code) on 2026-09-15, and the later-added SWM-bench, PolyBench and EventX T6 rows on 2026-09-21; quotations are verbatim. *"None stated" and "none described" mean that no relation between sequences or series was found among the inputs and protocol the paper describes; they are not claims about what other work built on these benchmarks does.* Locations refer to the arXiv versions cited.

EasyTPP (arXiv:2307.08097v3). Event unit, §2: "each $k_i \in \{1, \ldots, K\}$ is a discrete event type" of a timestamped event. Holdout, §3: "we split the set of sequences into disjoint train, validation, and test set". Information set, §5.1: models "predict the next event time given only the preceding

Table 4: **Evaluation objects**: the published protocols this paper compares against; not a ranking of overall quality. Each protocol is as stated in its own paper, or in release documentation where a benchmark has none; per-cell sources are quoted in Appendix A.

	target (time scale)	relations / external context	holdout unit	information set
EasyTPP	next time, type; continuous	none stated	whole sequences	earlier events in sequence
TGB-Seq	destination rank; per edge	temporal graph	chronological edges	edges before t
FI-2010	mid-price direction; multiple event horizons	none described	trading day	book + past features
RSR	stock rank; next day	sector/firm links	calendar periods	own + linked stocks
HIST	stock trend; next day	stock concepts	calendar periods	own + shared concepts
SWM-bench	implied probability; daily	candidate news events	calendar date	16-day window + events
PolyBench	resolution; to settlement	order book + news	none; zero-shot	snapshot timestamp
EventX T6	co-movement class; $1/3/7$ days	siblings; event clusters	calendar; folds cluster-disjoint	prediction-time fields
PolyEvent	block gap; price mark; next event	official event groups	whole event groups	own + siblings, earlier blocks

events, as well as its type given both its true time and the preceding events". Targets, §5.1: "the log-probability they assign to the held-out data" and "time and type prediction by RMSE and error rate". Aggregation, App. B.2: "for each event in each held-out sequence", averaged. The model inputs described are the event times and types of a single sequence.

EBES (arXiv:2410.03399v2; discussed in Section 2, not a row of Table 4). Event unit, §1: "Every data point in an EvS is linked to a specific timestamp". Relations and target, §2: "datasets comprising pairs (S_i, y_i), where S_i represents a sequence and y_i is the classification target attributed to the entire sequence"; the benchmark covers classification only. Holdout, §3.1: an established test set where available, otherwise "the last 20% of sequences" by time, otherwise a random 20% of sequences. Information set, §4.2.3: the encoder output is aggregated "into a single vector". Aggregation, §3.3: the "mean and standard deviation of the target metric computed on the test set across these 20 runs", with a per-dataset metric.

TGB-Seq (arXiv:2502.02975). Event unit, §3: "a stream of timestamped edges". Relations, §4: "four bipartite datasets … and four non-bipartite datasets". Holdout, §4: "We split the datasets chronologically into training, validation, and test sets with a ratio of 70%/15%/15%", and "only nodes appearing in the training set are included in the validation and test sets"; the released preprocessing relaxes the node filter for source nodes of the Patent dataset. Information set and target, §3.1: given "a query edge (s, d, t), and all edges appeared before time t", the model ranks the true destination against 100 sampled negatives. Aggregation, §3.1: Mean Reciprocal Rank over test queries, averaged over three seeds (§5).

FI-2010 (arXiv:1705.03233v5). Scale, §1: "approximately four million events for ten consecutive trading days for five stocks"; §3.4: "each of our representations is a vector that contains information for 10 consecutive events", giving "394,337 representations". Relations, §3.1 and Fig. 1: data are extracted into "Day-by-day and stock-specific files"; no model input relating stocks is described. Holdout, §3.4: "anchored forward cross-validation" in which "the test set corresponds to one day of data". Target, §3.4: labels from "relative changes for the next 1, 2, 3, 5, and 10 events". Information set, §3.4: the "10-level limit order" book and "features describing the state of the LOB, exploiting past information". Aggregation, §3.4: "mean accuracy, recall, precision, and F1 score over all folds". *The ten-event window is an input representation, not a label horizon*: the paper states that "we use five different projection horizons for our labels", each "a different future projection interval of the mid-price movement" (§3.4).

RSR (arXiv:1809.09441v2). The unit is a stock on a trading day; §4.1 says the task is to "predict a ranking list of stocks for the following trading day". Relations are sector–industry and company

links (§4.2). The model combines each stock's price history with relation-aware embeddings (§3.1). Holdout, §4.1: "we chronologically separate the sequential data into three time periods" for training, validation and evaluation. Evaluation reports MSE, MRR and simulated investment return (§5.1).

HIST (arXiv:2110.13716v2). Each scored observation concerns a stock and date, using a 60-day history (§5.1). The model combines individual stock information with shared predefined and hidden concepts (Abstract and §3). Holdout, §5.1: "split them by time to obtain training set"; evaluation reports IC, Rank IC and Precision@N (§5.2). Its stock set and concept memberships can change; its chronological split does not imply that every test stock appeared in training.

SWM-bench (arXiv:2606.11482v1). The predicted quantity is a market-implied probability: "at time $t < T$, a social belief is denoted by the pair $b_t = (q, v_t)$" (§3), sampled at a daily close — "We sample v_t daily, using the closing price as the observed value" (§6). Inputs are a price window and candidate news events: "a look-back window of $w = 16$ days and cap the candidate event set at $m = 30$" (§6). Holdout, §6: "chronologically (training before Nov 1, 2025; testing thereafter)".

PolyBench (arXiv:2604.14199v1). The target is a market's eventual resolution: "A market is a specific, actively tradable proposition under an event that will ultimately resolve to a definitive Yes (True) or No (False)" (§3). Inputs couple book and news state to each snapshot: "synchronously coupling each snapshot with a Central Limit Order Book (CLOB) state and a real-time news stream" (Abstract). *It defines no train/test holdout*; models are scored zero-shot — "PolyBench therefore enforces genuine zero-shot temporal reasoning" (§2.2) — with the information set fixed by "a strict Bayesian baseline locked to the specific snapshot timestamp" (§4.1).

EventX T6 (repository `mlsys-io/EventXBench`, release `t6.kdd.v2`; read 2026-09-21). *This is the closest protocol to ours and the only one whose evidence comes from a project README rather than a paper.* Target, task list: "Forecast whether a primary market and its causally visible sibling markets co-move over a daily horizon", with unit "`(condition_id, bundle_day, horizon_days)` for 1, 3, and 7 days" and labels "`no_effect`, `primary_only`, `cross_market`". Information set: "Only prediction-time schema fields are model inputs". Holdout: "The July 2026 KDD releases use versioned, leakage-audited temporal splits"; the data contract adds "T4–T6 use `tier2.temporal.v2`; T6 is also event-cluster grouped", and the model-selection folds are separately described as "group-disjoint by `event_cluster_id`, with a row-wise `condition_id` fallback when a cluster ID is unavailable." *The release does not state the group intersection between its train and test partitions, so we do not infer that its group structure serves model selection alone. How `event_cluster_id` is constructed is not stated in the release we read, so we make no claim about whether it coincides with the platform's official event grouping.* The T6 release reports 766 train, 1,225 validation and 2,592 test rows.

TGB (arXiv:2307.01026v2; discussed in Section 2, not a row of Table 4). Node affinity prediction, §3.2: "given the observed evolution history of a temporal graph" up to the current time, the task "predicts the interaction frequency vector" of a node over candidate nodes "within a fixed future period", scored by "NDCG@10". Split, §4: "All datasets are split chronologically into the training, validation, and test sets, respectively containing 70%, 15%, and 15% of all edges". Inductive comparison, App. I: "We define inductive nodes as nodes that are unseen in the training set"; Table 7 reports it for link prediction on `tgbl-wiki`, where "inductive MRR is computed from all edges that has an inductive source node". The node-level task and this link-level comparison are separate protocols, and we do not combine them.

PolyEvent. Section 3.1 (event unit and groups), Section 3.3 (whole-group holdout and the strictly earlier-block information set) and Section 3.5 (event-micro and market-macro reporting).

B Construction audits: event unit, ordering, marks and splits

Ordering events by block number leaves $62{,}060$ adjacent within-market pairs with a negative elapsed time, out of $279{,}897{,}687$ pairs in total. Figure 7 gives the two distributions behind that count.

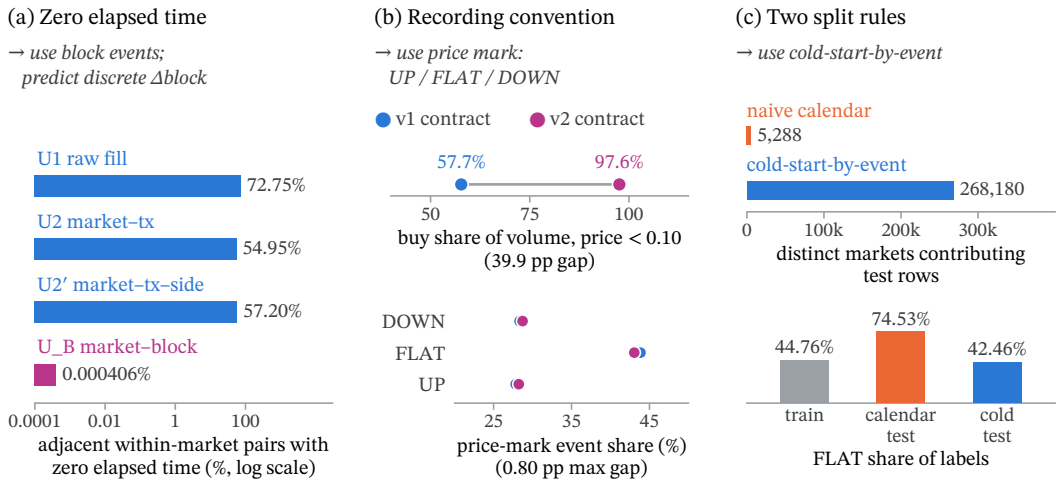

Figure 6: Three selected choices a standard pipeline would make — not an exhaustive list — and PolyEvent's response to each. (a) Adjacent within-market pairs with zero elapsed time, by event unit (log). (b) An order-direction statistic and the price-change mark across a contract-generation change, on the same events. *Different denominators* (traded volume against event counts) mean separate scales and non-comparable magnitudes, in this snapshot only. (c) Distinct markets contributing test rows and the FLAT label share, calendar split against cold-start-by-event; *market counts are non-disjoint across splits*. Panels are *not* ordered by severity; a fourth failure is in Figure 7.

The two remaining quantities behind Figure 6 are as follows. Across the contract-generation change the volume-weighted buy share in the lowest price bucket moves 39.9 percentage points, against at most 0.80 percentage points for the price-change mark on the same events; the two statistics have *different denominators* — traded volume against event counts — so their magnitudes are not comparable. A global calendar split leaves 5,288 distinct markets contributing test rows, against 268,180 under cold-start-by-event, with *market counts non-disjoint across splits*.

The negative gaps are not confined to sub-second clock jitter: 31,851 of them exceed one minute in magnitude and the largest is −196 seconds, while none exceeds one hour. No negative gap appears after November 2025 in this snapshot. That last statement is an inference from the absence of later months in the source table rather than a direct reading, and it is a statement about this snapshot only: we make no claim that the platform or the indexer has changed, and none about later data or about other chains.

The recorded timestamp field is not a pure canonical block-header reading. Block times were attached per block where the log returned them, fetched directly for small gaps, and linearly interpolated for larger ones to bound ingestion load; the field therefore mixes read and reconstructed values. We did not attempt to attribute the cause of individual reversals, and this audit does not isolate which construction step or upstream source produced any one of them. The protocol consequence does not depend on the cause: block number is the ordering key, so a non-monotone timestamp cannot reorder events.

C Implementation Details

Tier construction and exact counts. The tier is drawn by hashing `event_id` and taking a prefix in hash order; every market and every event of a selected group is admitted, so tiers are nested and no group is split by sampling. The tier used throughout contains 10,002,217 events. The multi-market subset used for the relational track contains 6,621,624 events, of which 6,590,941 carry a mark label and therefore serve as prediction targets. Of those targets, the cold-start split assigns 5,668,006 to training, 454,164 to validation and 468,771 to test, over 2,561, 535 and 535 event groups respectively. The coverage figures of Section 1 are computed on the tier's training split before the relational filter: multi-market groups are 2,604 of 14,093 groups (18.5%), 22,464 of 33,953 markets (66.2%) and 5,848,416 of 8,379,958 events (69.8%), counting the 24 groups with more than 64 markets that the relational track later excludes.

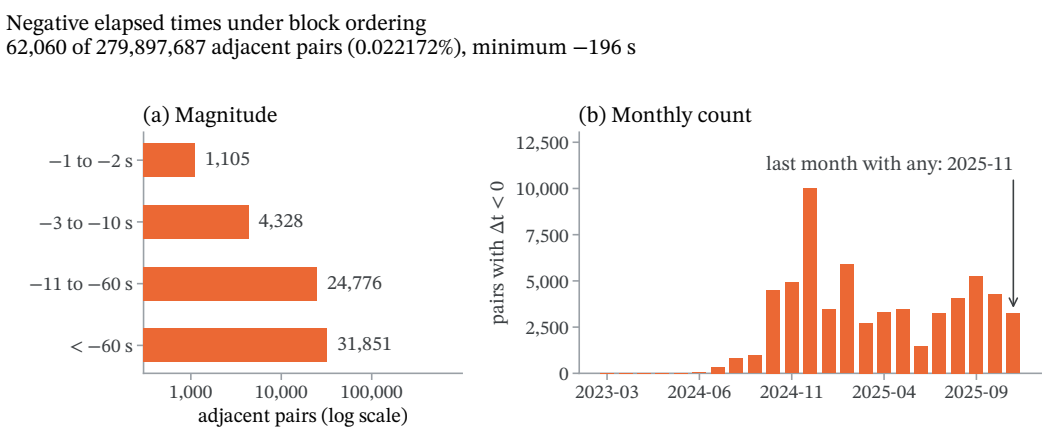

Figure 7: Negative elapsed times under block ordering, over all 279,897,687 adjacent within-market pairs of the corpus. (a) Magnitude of the 62,060 negative gaps, in seconds (log scale); the four bins partition the negative gaps only. (b) The same gaps by calendar month; months with no negative gap do not appear in the source table, and none appears after November 2025 in this snapshot. This figure describes one snapshot of one chain and does not establish a cause.

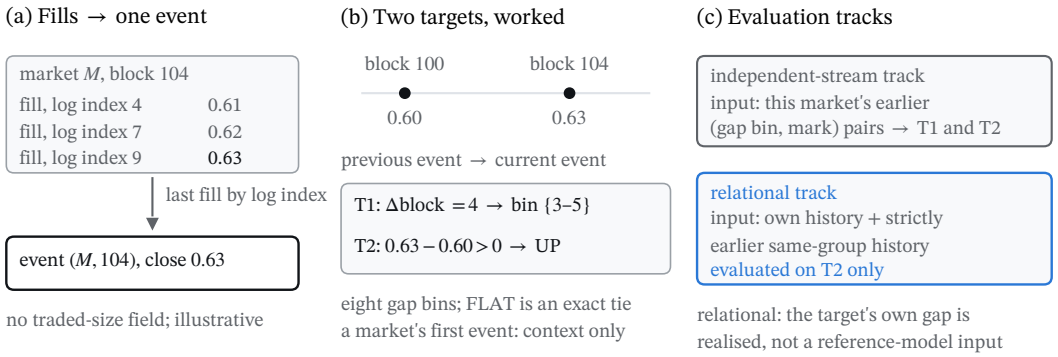

Figure 8: **From raw fills to benchmark targets** (one illustrative example; *not* a sample of the corpus). (a) All fills of a market inside one block collapse into one event, priced at the last fill by on-chain log index. (b) The same event, scored against its predecessor: the block gap falls in a canonical bin and the close change gives the mark. This is how targets are constructed, *not* information a model receives in advance. (c) The two tracks read different inputs and are evaluated on different populations (Section 3.2); on the relational track the target's own gap is realised but is not an input to the reference models (Appendix R adds it to XGBoost).

Canonical T1 target and relational input encoding. The canonical T1 endpoint is raw `delta_block` under the independent-stream track, with eight bins whose labels include the upper endpoint: {1}, {2}, {3–5}, {6–10}, {11–50}, {51–500}, {501–5000}, {>5000}. The relational track evaluates T2 only and uses observed historical gaps as an internal input: seven half-open bins, {1}, {2–4}, {5–9}, {10–49}, {50–499}, {500–4999}, {≥ 5000}, plus a start-of-stream symbol. This lossy tokenisation is not an alternative T1 endpoint and does not change T2. We retain it for reproducibility of the frozen T2 models; both relational arms share it, and **per-bin quantities from the canonical target and internal input representation must not be compared**.

Context construction parameters. The target encoder is a GRU over the target's own last $L_{\text{own}} = 32$ events. The sibling branch takes at most $m_{\text{sib}} = 16$ sibling markets active within a look-back window of $W = 1000$ blocks, each encoded by a GRU over its own last $L = 8$ events. Each sibling also carries *eight summary features*: the block gap from its latest event to b, its event count in the window, its UP, FLAT and DOWN proportions, and three co-movement values — the rate described below, its support, and an indicator that the rate is undefined. Event groups with more than 64 markets are excluded from the relational track for tractability; they account for 0.17% of groups and 1.91% of events in the tier's training split. The target-only model has 116,931 parameters and the event-context model 167,587.

Ablated co-movement feature. For each (target, sibling) pair we computed the historical rate at which their directional marks agreed, restricted to sibling events strictly earlier than b and within the look-back window, shrunk toward 0.5 and reported with its support. Only directional marks enter the rate: including FLAT lets FLAT/FLAT pairs dominate the "agreement" count. The feature is present in the main model and ablated on the same matched population as the permutation control. Removing it costs 0.0078 nats of market-macro ΔNLL ($[-0.0151, -0.0005]$, Holm $p = 0.010$), so *the evaluated co-movement statistics provide a detectable additional macro-NLL improvement*. This is the increment from adding these particular statistics to this architecture. It is *not* a decomposition of the sibling-context effect, and it does not identify a mechanism such as common information arrival: the ablation removes one input, not one cause. No claim in the main text rests on it.

Optimisation. Training uses an update budget rather than an epoch budget, so that arms with different per-epoch costs receive matched update budgets. Both arms of every comparison share the optimiser, batch size, learning-rate schedule and update budget, and differ only in the component under test. The main 20k development event-context runs use AdamW at learning rate 10^{-3} with weight decay 0.01, batch size 512, a budget of $20{,}000$ updates with evaluation every $2{,}000$ and early stopping after 3 evaluations without improvement on validation, target embedding width 128, sibling embedding width 64, and at most 16 siblings per target. Five seeds (1–5) are trained per arm. *Each seed is aggregated independently and the seed values are then averaged; predictions from different seeds are never pooled before aggregation.* Early stopping binds *asymmetrically*, and this matters for reading the result. The five target-only seeds stopped at $20{,}000$, $12{,}000$, $18{,}000$, $20{,}000$ and $14{,}000$ updates. All five context seeds ran to the cap, and in **four of the five the selected checkpoint was the final evaluation**, with validation loss still improving at the budget boundary; only one seed selected earlier ($18{,}000$). **The context arm is therefore budget-limited rather than converged.** The reported result characterises the frozen capped-training protocol rather than either architecture at convergence. Additional optimisation could change both the absolute losses and their contrast; *the direction is not identified by the observed learning curves*.

These diagnostics concern the original 20k development protocol only. Before the scaling study and the sealed evaluation, both arms were extended to a prespecified 160k-update cap, and the checkpoints frozen there are the ones used for the sealed confirmation and the scaling study (Section 4.1). *The two protocols select checkpoints differently, and the distinction matters for every statement below.* The 20k protocol used early stopping with a patience of three, so a run could halt before its cap. The 160k runs pass the same patience value but disable the rule (`--ignore-patience`), so **early stopping never fires there**: each run trains to the full budget and the reported checkpoint is the argmin over all 80 validation evaluations (Appendix L). Under that cap the selected checkpoint is interior to the budget for every seed: 20k, 64k, 76k, 92k and 134k updates for target-only and 78k, 92k, 104k, 120k and 132k for sibling-context, so even the latest of the ten leaves $26{,}000$ updates — thirteen further evaluations — none of which improved on it. *An interior argmin shows only that validation loss stopped improving on this grid; it is not by itself evidence that the arm is unconstrained by the budget, and it is not convergence.* The shared cap equalises the maximum number of parameter updates and batch exposures, not realised wall-clock time or FLOPs — a context update is more expensive. *The asymmetry noted above — several target-only seeds stopping early, leaving the two arms unequally close to their own optima — belongs to the 20k protocol, where early stopping was active, and does not carry over to the 160k runs.* A data-scaling curve run at a fixed update budget would still confound added data with residual under-training; Appendix L states the collinearity that applies to the curve actually reported here. Per-seed training under the 20k protocol took 1.3–2.1 minutes for the target-only arm and 3.2–3.3 minutes for the context arm on a single GPU. *The device model was not recorded in the run logs, and we do not infer it.*

Absolute levels behind the reported differences. On the full relational test population ($468{,}771$ rows, $3{,}704$ markets) the target-only model reaches market-macro NLL 1.0238 and the context model 0.9822. On the matched subset ($395{,}669$ rows, $3{,}689$ markets) the three arms are 1.0254 (target-only), 0.9851 (context) and 1.0236 (permuted context); the co-movement ablation is at 0.9928. Under micro weighting the corresponding target-only levels are 0.8236 on the full population and 0.8633 on the matched subset. *These are not comparable to Table 1*, whose independent-stream models are evaluated on a different population ($813{,}347$ rows over $7{,}264$ markets) with a different architecture; the denominator for the relational result is the target-only arm above.

Table 5: Target–donor balance for the permutation control, pooled over the 5,658,945 rows with an admissible donor and at least one sibling. *Sibling count is matched exactly and is not shown.* $|\Delta|$ is the absolute target–donor difference; *spread* is the variable's own scale on the same rows. A ratio near zero indicates tight matching; above one indicates the variable is not controlled by this design.

matched variable	exact	median $\lvert\Delta\rvert$	$p95\ \lvert\Delta\rvert$	spread	$p95$/spread
target history length	93.1%	0	5	4.9	1.03
effective sibling history	80.1%	0	12	35.4	**0.34**
sibling event count	3.3%	58	642	258.2	2.49
recency of latest sibling	5.9%	22	459	154.7	2.97
target block	0.0%	5,473,456	22,343,210	9,166,280	**2.44**

Table 6: **Development-test context family: all 16 quantities.** Paired ΔNLL in nats on identical rows, negative favouring the first-named arm; 5 seeds, nested event-group-by-seed bootstrap clustered on event groups, *one* max-t simultaneous family (c=3.02). $D = \Delta_{\text{micro}} - \Delta_{\text{macro}}$. *The two populations differ*: rows are compared only within a population. $p < 0.0005$ is below the bootstrap resolution, not an exact zero.

contrast	point	95% CI	Holm p
all 468,771 **rows**			
same-group context − target-only			
market-macro	−0.0416	[−0.0550, −0.0282]	< 0.0005
event-micro	−0.0088	[−0.0111, −0.0065]	< 0.0005
interaction D	+0.0328	[+0.0199, +0.0457]	< 0.0005
matched rows (395,669**, with an admissible donor)**			
same-group context − target-only			
market-macro	−0.0403	[−0.0534, −0.0273]	< 0.0005
event-micro	−0.0088	[−0.0109, −0.0066]	< 0.0005
interaction D	+0.0316	[+0.0189, +0.0443]	< 0.0005
different-event context − target-only (placebo)			
market-macro	−0.0018	[−0.0063, +0.0027]	0.426
event-micro	−0.0010	[−0.0021, +0.0001]	0.044
interaction D	+0.0008	[−0.0034, +0.0050]	0.549
same-group − different-event (arm-to-arm)			
market-macro	−0.0385	[−0.0512, −0.0258]	< 0.0005
event-micro	−0.0077	[−0.0099, −0.0055]	< 0.0005
context without co-movement − target-only			
market-macro	−0.0326	[−0.0442, −0.0210]	< 0.0005
event-micro	−0.0068	[−0.0098, −0.0037]	< 0.0005
interaction D	+0.0258	[+0.0141, +0.0375]	< 0.0005
full context − context without co-movement			
market-macro	−0.0078	[−0.0151, −0.0005]	0.010
event-micro	−0.0020	[−0.0045, +0.0005]	0.061

Matching quality of the permutation control. Table 5 reports, for each matched variable, the share of target–donor pairs agreeing exactly and the distribution of the absolute difference. Sibling count is matched exactly by construction and is omitted. *The last column is the $p95$ difference divided by the variable's own spread*, which is the quantity to read: near zero means tightly matched, above one means not. Target history length is identical for 93.1% of pairs and median-matched, but its upper tail is not controlled by the $p95$/spread criterion; effective sibling history is matched more closely. Activity density and recency enter only the finest level and are not controlled (ratios above one), and **target block not at all**. The matcher relaxes in three levels: all five bins (743,961 rows, 452,915 of them with no sibling), then the own-history and sibling-history bins only (4,115,706), then sibling count only (1,252,193); the target-block bins are fitted on training blocks, all of which precede validation and test, so they do not separate those rows. These diagnostics are pooled over the 5,658,945 rows that have a donor and at least one sibling, *not* computed per split. Rows without an admissible donor are excluded from the comparison and withheld from training in every arm: 7.16% of training rows, 0.00% of validation rows and 15.59% of test rows.

Why the relational track does not evaluate T1. Its rows are anchored at the current target event, whose block gap is realised but *not* an input to the reference models (Figure 8(c); the XGBoost

extension of Appendix R adds it). Evaluating T1 there would require moving the forecast origin to the preceding event and rebuilding the sibling information set, i.e. a different contract.

Split construction and integrity. The boundaries are B_1=86,985,535 (2026-05-16 20:46:35 UTC) and B_2=88,740,292 (2026-06-18 20:43:53 UTC), at the 0.8 and 0.9 quantiles of the event distribution. The minimum stream length of 32 events is the warm-start protocols' filter and is not applied to the cold-event split. Groups are assigned by first block, and events of a training group after B_1 are dropped: $4{,}800$ training groups are truncated this way, contributing the $9{,}546{,}874$ dropped events. The three splits hold $423{,}169$, $116{,}940$ and $111{,}907$ groups and $996{,}422$, $264{,}155$ and $268{,}180$ markets, summing to the $652{,}016$ groups; the three splits plus the dropped tail sum to the $281{,}433{,}584$ events. The three market counts sum to $1{,}528{,}757$, which is $7{,}140$ short of the $1{,}535{,}897$ markets of the corpus. Those $7{,}140$ markets are a consequence of the truncation rule rather than a gap in the manifest: every one of their events is labelled *excluded*, their earliest event is at block $86{,}985{,}539$ — after B_1 — and all $1{,}411$ groups holding them also hold training events, so each is a market whose events fall entirely inside the dropped tail of a training group and therefore appears in no split. This was verified on the released manifest by a read-only audit script whose digest, and the manifest digest it binds, are released with the artefact. Computed directly on the manifest, all three pairwise group intersections and all three pairwise market intersections are empty. Block ranges are $[35{,}896{,}869,\ 86{,}985{,}535]$ for training, $[86{,}985{,}617,\ 90{,}568{,}738]$ for validation and $[88{,}740{,}305,\ 90{,}568{,}741]$ for test, so no training event follows any test event, while 10.73% of validation events fall after B_2. The manifest and its construction script are released with content hashes.

Secondary and diagnostic split construction. The within-market prefix protocol splits each market $80/10/10$ along its own event index; every market with enough history participates, so the label distribution is drawn from the same population in all three parts. Unlike the primary protocol, it does not guarantee that every training event precedes every test event. The cross-generation diagnostic places markets that traded only under the older contract family on the training side and markets that traded only under the newer one on the evaluation side, excluding markets that span both.

D EVALUATION WEIGHTING CHANGES WHICH MODEL APPEARS BETTER

Development headline in levels. The market-macro levels behind the development contrast — target-only 1.0238 against sibling-context 0.9822, a 4.07% relative reduction — are stated in Section 4.1. They are *specific to the full relational evaluation population under market-macro weighting*, not to micro weighting, the matched subset, or the corpus; the $\log 3 = 1.0986$ reference is a per-row constant, hence the same value under either weighting and on any population, and is *not* an upper bound on attainable signal.

Development permutation coverage and micro-weighted readings. *These are development-split quantities, not part of the sealed confirmation.* A permuted context exists for 395,669 of the 468,771 test rows (84.4%, over 3,689 markets and the same 535 groups), and every permutation quantity is measured on that subset, so those comparisons stay row-for-row paired. Restricting the headline to the same subset gives -0.0403 $[-0.0534, -0.0273]$, so the excluded rows do not carry the effect. *The permuted arm is defined only on this subset*, so a full-population permutation quantity would mix permuted and unpermuted rows and we report none (levels in Appendix C).

On the *full* development-test population the micro-weighted headline is -0.0088 $[-0.0111, -0.0065]$ — roughly a fifth of the macro effect and, unlike it, not reaching 0.010 nats. *The matched subset gives the same two displayed digits but a different interval*, $[-0.0109, -0.0066]$, so the two populations' readings are not interchangeable. On that subset the permutation leaves a small residue, -0.0010 $[-0.0021, +0.0001]$: the simultaneous interval contains zero while the Holm-adjusted p is 0.044, and we report both readings; the arm-to-arm contrast is -0.0077 $[-0.0099, -0.0055]$. The frozen criterion was set on macro weighting, where the control shows *no detected gain* under either reading (simultaneous interval containing zero, Holm $p = 0.43$); *an interval containing zero is not equivalence*.

We measure the difference directly with the interaction contrast $D = \Delta_{\text{micro}} - \Delta_{\text{macro}}$ (Section 3.5). Appendix Figure 9 reports these interaction contrasts with their simultaneous intervals, and Table 6 lists all 16 development-test quantities.

For XGBoost against a GRU on independent streams, D excludes zero in all four protocol $\times$ task cells (family of 24, c=3.051). Under cold-event T1 the GRU–XGBoost improvement is larger under event-micro than market-macro averaging; the preregistered direct interaction is -0.0084 with Holm-adjusted $p = 0.0085$. Under warm-prefix T1 the two point estimates favour opposite models: their direct weighting interaction is detectable ($D = -0.0364$, Holm-adjusted $p = 0.0011$), although the max-t simultaneous intervals for the two component contrasts individually include zero (the micro component has Holm $p = 0.021$). *The component-level conclusion is sensitive to the multiplicity correction; the interaction itself is stable.* For the target-only against context comparison of Section 4.1, D also excludes zero but with the *opposite* sign: $+0.0328$ on the full evaluation population and $+0.0316$ on the matched subset, so there macro weighting reports the larger advantage. For a GRU against a causal Transformer, D contains zero in all four cells.

The direction of the change depends on which kind of stream the gain is concentrated in.

E INDEPENDENT-STREAM BASELINES AND BLOCK GAPS

Table 1 gives test NLL for five models on both tasks and protocols, under both weightings. PER-MARKET does not require the evaluated market to appear in training, so it applies to unseen markets: each event is scored from the N strictly earlier outcomes of the same market (marks for T2, gap bins for T1) as $p_k = (n_k + \alpha\pi_k)/(N + \alpha)$, shrunk toward the training marginal π with $\alpha \in \{1, 3, 10, 30, 100\}$ selected on validation; with no earlier outcome it returns π. Seeds are aggregated independently and then averaged over 5 seeds, except MARGINAL and PER-MARKET, which are deterministic. The largest across-seed range of any fitted cell is 0.0081 under cold-event (T1 market-macro, Transformer) and 0.0211 under warm-prefix (T1 market-macro, GRU) — a spread, and *not an interval estimate*.

For the price mark, the GRU beats XGBoost in all four protocol $\times$ weighting cells, every max-t simultaneous interval excluding zero; for the block gap only the cold-event micro interval does so. *The complete point-estimate ranking differs between event-micro and market-macro averaging in three of the four protocol–task cells*; the reported family tests only the two adjacent pairs XGBoost–GRU and GRU–Transformer, so the *full five-model ordering* of this table is descriptive and the marginal and per-market rows carry no paired test.

Under event-micro weighting, the GRU and Transformer point estimates differ by at most 0.009 nats. Their market-macro differences are larger, and they are covered by the same reported 24-quantity simultaneous family. Writing Δ for Transformer minus GRU, the market-macro interval excludes zero in three of the four protocol–task cells: cold-event T1 $+0.0082$ $[+0.0029, +0.0136]$ and T2 $+0.0046$ $[+0.0014, +0.0078]$, where the GRU is lower, and warm-prefix T1 -0.0159 $[-0.0293, -0.0026]$, where the Transformer is lower; warm-prefix T2 contains zero, -0.0018 $[-0.0102, +0.0066]$. *Which of the two is preferred therefore depends on the protocol and the task*, so these cells support no single ranking across settings. We report them within settings and do not rank the two models overall.

F EXTENDED LIMITATIONS

This is the full limitations statement summarised in Section 5.

Population. The relational result applies only to multi-market event groups (group-, market- and event-level coverage, Section 1; these characterise different aspects of scope). Nothing here extends to single-market events. The sealed gain is not stratified by group structure. A post-hoc stratification of the development-test rows by `negRisk` status and sibling recency is reported in Appendix S; it is descriptive, and because `negRisk` flags one exclusivity mechanism only, it cannot say whether the gain concentrates in groups whose outcomes are logically constrained. Repeating it on the sealed split would be a new analysis and could not reuse that split as a confirmatory test. Sealed membership is a *deterministic hash* of the group identifier, so it tests **unseen event groups under the observed regimes** — *not a future-calendar evaluation* — and **cannot separate same-group context from contemporaneous market-wide conditions**. A three-arm ablation gives the mark-bearing and mark-derived channels a material joint increment (Appendix K); *this is observational and does not exclude broader activity-based explanations*. Only 16.6% of sealed groups yield a context target, *fixed in advance and neither enlarged nor replaced*.

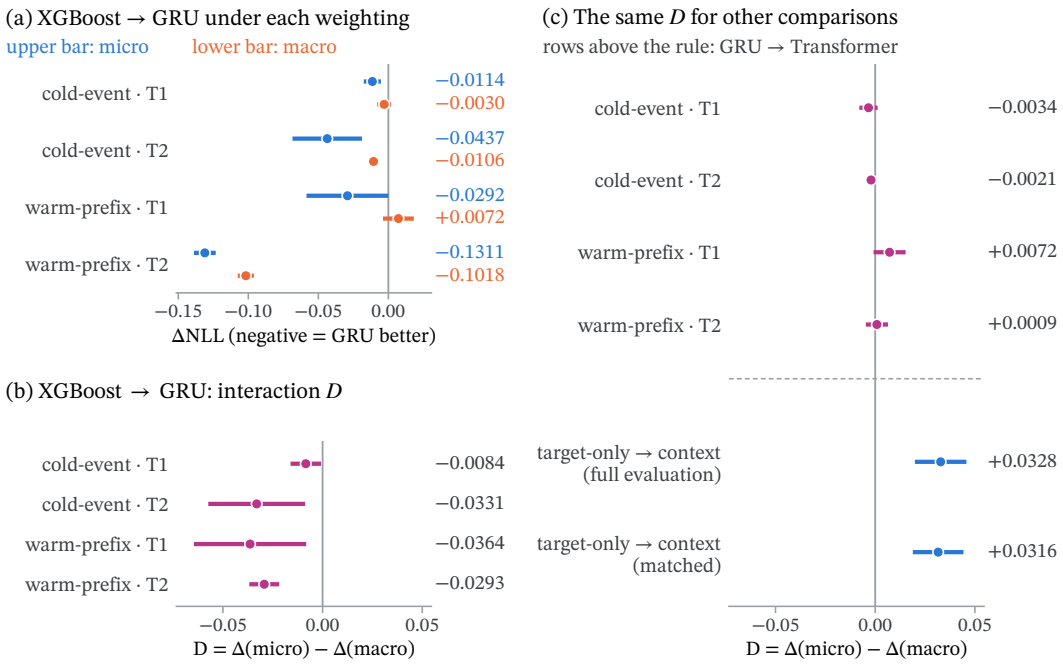

Figure 9: Evaluation weighting. (a) The XGBoost → GRU improvement under each weighting, for both tasks and both protocols. (b) Its interaction contrast $D = \Delta_{\text{micro}} - \Delta_{\text{macro}}$, tested directly rather than inferred from which weighting is significant; D excludes zero in all four cells, so micro reports the larger advantage. (c) The same D for GRU → Transformer, where it contains zero in all four cells, and for target-only → context on test — both the full and the matched population — where it excludes zero with the *opposite* sign. Panels (b) and (c) share one x-axis because they are the same estimand; (a) is a different estimand and has its own. All intervals are max-t simultaneous 95% intervals from a nested event-×-seed bootstrap clustered on event groups, 5 seeds. *Critical values from families of different sizes are not comparable* — family 24 with c=3.051 above the dashed rule, family 16 with c=3.02 below — so interval widths must not be compared across the rule.

Evaluation weighting. Only macro determines the verdict; development event-micro does not reach the prespecified 0.010-nat threshold. These are different estimands over different populations, reported side by side; we designate neither as correct.

Model and data scale. All models train on a deterministic 10M-event tier, not the full 274M eligible view. *The primary relational comparison evaluates one frozen same-event sibling-pooling architecture; it does not establish optimality over alternative relational encoders.* No architecture search was run. At the same development budget, cross-attention (Appendix N) and attention among siblings (Appendix P) **were each not distinguished** from mean–max pooling; an interval containing zero is not equivalence, and three aggregators do not cover relational-baseline breadth.

Scaling. Figure 11 and Appendix L state the fixed-update-budget and single-subset limitations.

Mechanism and implementation. We do not attribute the gain to event-group identity. The corpus records transactions, not causes; news, arbitrage or traders active in several markets cannot be separated here. The co-movement ablation gives a detectable additional macro-NLL improvement, but removing an input is not identifying a cause (Appendix C).

Experimental governance. The development-test split had three forward evaluations; the channel ablation separately used development *validation* post hoc, and a later post-hoc non-neural baseline (Appendix R) used development test thrice more, health gates verified before test inference. Reported numbers correct scored-split checkpoint selection and two defects: padded positions carried category tokens, and the control was not a within-split bijection. **That run is an implementation-audit correction, not an untouched one-shot evaluation**; Appendix G gives the timeline. No permutation endpoint was prespecified for sealed (Appendix G).

G EVALUATION GOVERNANCE AND THE AUDIT TIMELINE

The disclosure in Section 5 is expanded here. The *development* held-out *test* split underwent three forward evaluations, listed below; the three-arm channel ablation (Appendix K) is a **separate, post-hoc forward experiment on development *validation*** and is *not* one of the three, and the later non-neural baseline (Appendix R) is a separate post-hoc experiment on development *test* that likewise is not one of the three (Stages 5–7 below). Table 7 lists every read of these splits in date order, with the split it touched and whether it trained models or generated predictions; the paragraphs below give what was run, what was wrong with it, and which artefact it produced. The first two are superseded; the third is the development result reported in the main text. Stage 4 is the separate sealed split: its membership was fixed before scaling, and it had one scoring run that generated predictions.

Table 7: Every read of the relational development-test and sealed splits, in date order. *Trains*: new models were fitted. *New predictions*: a model was run forward on the split; rows without them re-aggregate the Stage 3 per-row predictions. Post-hoc experiments on development *validation* — including the channel ablation (Appendix K) and the role-aware arms (Appendix T, 17–24 September) — trained new models and scored validation rows only, with no development-test or sealed predictions; the parsing audit of Appendix T separately read development-test question texts.

date	occasion	split	trains	new predictions
28 Jul	Stage 1	dev. test	yes	yes
28 Jul	Stage 2	dev. test	yes	yes
30 Jul	Stage 3, reported (§4.1)	dev. test	yes	yes
1–4 Aug	Stage 4, sealed confirmation	sealed	no	yes (one run)
5 Aug	per-class decomposition (App. M.1)	dev. test	no	no
13 Aug	own-history strata (App. M.2)	dev. test	no	no
16 Aug	conditional-class analysis (App. M.3)	dev. test	no	no
26 Aug	Stage 5, XGBoost pair (App. R)	dev. test	yes	yes
17 Sep	`negRisk` $\times$ recency strata (App. S)	dev. test	no	no
24 Sep	Stage 6, price-augmented pair (App. R)	dev. test	yes	yes
24 Sep	Stage 7, price + elapsed-blocks pair (App. R)	dev. test	yes	yes

Stage 1 (28 July). The first event-context evaluation of the held-out split passed the scored split to both the early-stopping criterion and the final scoring step, so when the scored split was the test split the checkpoint was the one that minimised test loss. No hyperparameter or architecture decision was taken from it, but the checkpoint was selected on the scored split. It gave market-macro $\Delta\mathrm{NLL} = -0.0404$.

Stage 2 (28 July). We separated the two splits: the evaluation script now takes an explicit early-stopping split that defaults to validation independently of the scored split, and the prediction artefacts record both, so downstream inference cannot silently misreport which was used. We committed the correction, and the pre-commitment that the corrected run would be designated primary *regardless of its outcome*, before that run was executed. It gave -0.0410. At this point the held-out split had been read twice.

Stage 3 (30 July), the run reported here. A subsequent audit of the implementation found two defects that affect every number produced before it. *(i)* Padded positions in the recurrent encoders carried the token pair (gap bin 1, DOWN) — both are valid categories, not filler — and the number of padded positions equals the sequence length subtracted from the window, so history length entered the representation directly. *(ii)* The permutation control was not a bijection, was drawn across splits rather than within one, and matched only on sibling count. Both were repaired: padding uses reserved symbols with masked recurrence, and the control is the split-local, different-event bijection described in Section 3.4. All three arms were then retrained on an identical set of training rows and the held-out split was scored once more. This is an **implementation-audit correction**, not an untouched one-shot evaluation, and we do not present it as one.

What changed and what did not. The point estimates moved little (market-macro $-0.0404 \rightarrow -0.0410 \rightarrow -0.0416$). We draw no conclusion from that closeness: the superseded control was not a valid permutation, so whether it agreed by chance or through compensating errors cannot be

determined from these runs. The defensible statement is that the corrected implementation preserves the same conclusion.

Development evidence. Before the corrected held-out run, the same corrected implementation was exercised on the validation split, giving market-macro -0.0418 with the permuted control at -0.0005. That run guided the decision to proceed and is reported here as *development evidence* only. It used a different comparison family and a split that also drove early stopping, so *its intervals must not be compared with the test intervals in the main text*.

Stage 4 (1–4 August), the sealed split. Development analyses had already established weighting sensitivity. On 1 August, before scaling outcomes, we selected the final-160k market-macro context-minus-target quantity as the high-level fresh-holdout primary. On 2 August, still before scaling outcomes, we fixed the membership — $12{,}800$ event groups chosen by a deterministic hash of the group identifier, restricted to groups outside the hash prefix that every development tier is drawn from — and recorded its digest without building context features or inspecting any statistic of it. Scaling outcomes were recorded on 3 August at 10:58 EDT. At 12:09 EDT, after those outcomes but before any sealed construction or prediction, we froze the complete split-specific rule: the -0.010 threshold, 95% interval condition, market-macro primary, event-micro secondary and two separate $K{=}1$ families. On 4 August the split was built and scored in one run that generated predictions, with the ten already-frozen checkpoints. The context layer of the built split accounts for all $12{,}800$ groups: $2{,}129$ context-eligible, $10{,}648$ whose market count lies outside the relational track's frozen 2–64 range (Appendix C), and 23 with no labelled target — the conservation ledger recorded in the post-build attestation whose digest is listed below. The driver refuses to proceed unless, among other conditions, the recomputed membership digest matches the recorded one over all 64 hexadecimal characters, each checkpoint file matches both its run manifest and a table of digests fixed in advance, no weight changes across inference, and the number of preprocessing fits performed on the sealed data is zero.

Disjointness from development is exact rather than approximate: the sealed groups are restricted to those outside the hash prefix from which every development tier is drawn. The primary family holds a single quantity ($K{=}1$), so its interval needs no multiplicity adjustment; no bootstrap replicate was as extreme in $2{,}000$ draws, which is the empirical resolution $1/2000$ of that resampling.

Absolute levels on the sealed population. The published result file stores only the two differences. Recomputed from the ten prediction artefacts it lists — digests verified, rows identical across files, each seed aggregated separately and then averaged — the target-only arm reaches market-macro NLL 1.0038 and event-micro 0.8644, and the context arm 0.9548 and 0.8512; their differences reproduce the reported contrasts to within 10^{-15}. These are the two components of the frozen contrast, not a new endpoint, and they carry no interval. They describe a different population from the development levels of Appendix C and are not compared with them.

Stage 5 (26 August), post-hoc non-neural baseline. A separate, later experiment (Appendix R) trained an XGBoost pair — target-only versus target+19 fixed sibling-window summaries, five seeds, frozen configuration — on the frozen relational rows and scored the development-test split once. Its health gates (finite losses, non-trivial best iteration, both arms beating the train-frequency marginal, training completeness against frozen constants) were verified *before* that single test inference, and its operating budget was raised from 8 to 16 CPU-hours by a dated amendment after a recorded first resource-smoke rejection — before any formal training or test scoring. It is not one of the three Stage 1–3 evaluations, is not part of the sealed decision, and its outcome was bound by a placement rule fixed before execution: the pair is reported in the main text regardless of direction.

Stage 6 (24 September), price-augmented pair. The same post-hoc chain with the target's last price appended to both arms (Appendix R), under a protocol frozen before any price was recovered. Price recovery reproduced every frozen label, all gates held before the single test inference, and the outcome is reported regardless of direction. It is not one of the three Stage 1–3 evaluations and not part of the sealed decision.

Stage 7 (24 September), price and elapsed-blocks pair. A frozen amendment to the Stage 6 protocol appended the blocks since the target's last trade to both arms, reusing the Stage 6 features

unchanged; recovered prices matched the frozen ones exactly, all gates held before the single test inference, and the outcome is reported regardless of direction.

Checkpoint selection and the primary estimand. The scaling checkpoints — including the ten 160k-cap states later scored on the sealed split — are the best validation *event-micro* NLL states among updates within the cap, whereas the sealed primary estimand is *market-macro*. We record the mismatch rather than reconciling it afterwards: selection read the validation split only, never test or sealed rows, and the rule belonged to the training protocol frozen before the sealed decision rule was fixed. The fixed-endpoint reads of Appendices O and K use no validation checkpoint selection at all.

What the sealed plan did not contain, and why we do not add it. The prespecified rule fixed one primary quantity — market-macro ΔNLL of sibling context against target-only — and *no* permuted-context endpoint. None was computed on the sealed split. Separate 160k permutation checkpoints existed for development mapping sensitivity (Appendix J), but neither those checkpoints nor a mapping-specific permutation endpoint was prespecified for this sealed split. Scoring one or more of them now would require choosing a new endpoint and donor mapping after observing PASS, constructing the corresponding sealed inputs, modifying the sealed evaluation plan, and using the split again. **We decline to do so: computing a new endpoint on a split after its criterion has been observed to pass is a second use of a population whose evidential value rests on a single scoring**, and the result would no longer be separable from the decision to look again. The different-event placebo therefore remains *development* evidence (Section 4.2), and the asymmetry between a sealed confirmation and a development-only control is a stated limitation rather than one this split can close. Any such control would have to be prespecified for a new sealed split.

The command was invoked twice. The first invocation was refused at the *first* checkpoint by a structural check and therefore produced no prediction on any sealed row; the cause is the metadata defect described below. The second invocation produced the numbers reported in Section 4.1 (*Confirmation on a sealed split*). *One invocation produced predictions.*

A metadata defect in the frozen checkpoints. In the training script, a top-level diagnostic loop reused the identifier that holds the sibling-history width, overwriting it after the model and its inputs had already been constructed. Consequently all forty scaling checkpoints record that width as a diagnostic level rather than its true value, while the run manifest written earlier in the same execution records it correctly. A source-level scan confirms the identifier is not read again by the model, the input packing or the optimiser after the overwrite, so weights and every reported estimate are unaffected; only the recorded field is wrong. Because that field can no longer be trusted, the structural check now binds the *entire* training-context digest instead, which is strictly stronger than the single-field comparison it replaces.

Subsequent descriptive reanalysis (13 August). This reanalysed the Stage 3 development-test predictions after model fitting and endpoint selection (Table 7). It involved no new forward evaluation, checkpoint selection, retraining, or use of sealed data. The stratification of Appendix M.2 is its only product; its bins, weighting, families and reporting rules were fixed before any of its quantities were computed.

A further reanalysis (16 August). The conditional analysis of Appendix M.3 reanalyses the same Stage 3 development-test predictions. It involved *no new forward evaluation, no retraining, no checkpoint selection, and no use of sealed data*. Its class sets, estimands, families, resampling and reporting rules — including the admissible wording for each outcome branch — were fixed in a protocol frozen *before any of these conditional quantities were computed*, and the identity of every input, of the implementation and of the protocol itself is bound by digest in the released attestations.

Artefact digests. The sealed run is reproducible from four SHA-256 digests released with the benchmark: the split membership, `2512c3a5a247da53298fdf742fc7f0b5680f4efad4e59c8c0fb14a739fba3c2a`, recomputable from the released membership table; the pre-build manifest of inputs, code and parameters, `80a1139939452f7eb9e806b68390347d17d9d31b407dccbeb9d2d0aac8604788`; the post-build attestation recording the materialised split and both conservation ledgers, `036a6f33e52d0e02978382f63527a24f`

af4cc4eab6afc5b4ae340f63dad605a6; and the aggregate digest of the ten prediction artefacts, c5fcf971e8d85325f81d5bb02ef617cc4126700c9db56af39a18a751b2008375. The published result file references the first three by full digest, so the chain from the pre-registered plan through the built split to the actual predictions is checkable without trusting any intermediate summary.

H SIBLING CONTAMINATION

Before the permutation control was available, we measured contamination directly: whether training on markets that are siblings of an evaluation market inflates that market's score, even for a model with no cross-market channel. Adding sibling markets to training improved held-out NLL by -0.0088 on T1 and -0.0176 on T2, while an equal quantity of non-sibling data had no detectable effect. The model in that experiment cannot read sibling context, so the effect is not mediated by an explicit cross-market input: *training-set sibling overlap can move scores even without such a channel. It is not a bound on what a relational model would gain from the same overlap*, in either direction.

This leakage path does not exist under the cold-event protocol. That protocol holds out *whole event groups*, so no market that shares a group with an evaluated market can appear in training (Section 3.3); the contamination measured here requires exactly that co-occurrence. The number is reported to quantify what a group-splitting protocol would cost, *not* as a caveat on the development and sealed results of Section 4.1.

This analysis rests on a **single matched draw** of non-sibling markets; we did not repeat the matching. It is reported here rather than in the main text because the headline result does not depend on it. *It asks a different question from the matched permutation control of Section 4.2*: that control varies which group's context a model receives *at prediction time*, whereas this measurement varies which markets appear *in training*. Neither substitutes for the other.

I ABSOLUTE-TIME MATCHING WAS EVALUATED AND FOUND INFEASIBLE

Section 4.2 states that absolute block position is not controlled by the permutation control. We evaluated whether it could be, by adding an absolute-block caliper to the matcher and measuring the resulting matched population on the $5{,}668{,}006$ eligible training rows against the same representativeness criterion used for the reported control (coverage $\geq 80\%$ of eligible rows and maximum standardised mean difference < 0.1 on every matching covariate).

The standardised mean difference here is not the quantity in Table 5: Table 5 measures paired target–donor differences, whereas this table measures representativeness of the matched target population relative to all eligible targets. A design can match pairs loosely on a variable and still yield a representative population, or the reverse.

caliper	row coverage	max |SMD|	criterion
none (reported control)	92.84%	0.0778	met
$\pm$3M blocks	64.38%	0.6185	not met
$\pm$10M blocks	72.37%	0.4296	not met

Both finite calipers fail on *both* conditions. The failure is also not confined to the time dimension: at $\pm$3M the caliper-selected population is unrepresentative in sibling effective history length (0.4495) and sibling count (0.4175) as well, the two covariates the reported control matches closely. A caliper would therefore trade an unmatched covariate for several newly unmatched ones, on a population that excludes a third of eligible rows.

Under the evaluated matcher, tighter absolute-block calipers substantially reduced coverage and changed the evaluated population; therefore this audit did not identify event-group identity separately from absolute time. We make no claim that no better matcher exists, and this audit does not resolve the confounding — it establishes only that the version we evaluated cannot, and that a result obtained on the caliper-selected population would not be interpretable either way: retention of the effect would speak only to a heavily selected subpopulation, and its disappearance could not be separated from that selection.

A matching-free contemporaneous control was also evaluated. A control that keeps the same lookback window but draws its context from markets outside the target's event group needs no matching step, so we specified its gates in advance — coverage $\geq 80\%$ and $|\mathrm{SMD}| \leq 0.10$ on the listed covariates, the non-strict bound written in that control's own frozen protocol (the caliper audit above uses a strict < 0.1) — and ran a training-free feasibility check on $50{,}000$ training targets. Selecting non-group donors by the *same* recency rule used for siblings, coverage is 84.51%, but the two arms' contexts are not comparable: the standardised mean difference in recency is -0.5217 (95% CI $[-0.5301, -0.5128]$) against a threshold of 0.10, with $p95\,|\Delta\text{block}|$ of 66 against a same-group window of $1{,}000$ blocks, and the $12{,}105$ distinct donors carry an effective count of 544.7. *Within the frozen rule and on this data, enlarging the candidate pool made donors systematically more recent*, so the arms differ in more than group membership and we did not train it. This is a negative result for the construction we evaluated; it does not show that no symmetric matched construction exists, and it does not resolve the temporal confounding.

The symmetric matched construction was then evaluated at full scale. The preceding paragraph leaves open whether a construction that *matches* the two arms, rather than avoiding matching, is feasible. We specified one in advance and ran it on the full population. Each target's k same-group siblings (arm B) are paired with k distinct donors (arm C) drawn from a different event group in the same split, strictly earlier, inside the same $1{,}000$-block window, and inside 0.20 training standard deviations on recency, activity density and effective history; the assignment is minimum-cost bipartite matching, and a target is retained only if *every* one of its k siblings is matched. Four gates were frozen before the run: retained-row coverage $\geq 80\%$, event-group coverage $\geq 80\%$, $|\mathrm{SMD}| < 0.10$ for representativeness of the retained targets on six covariates, and $|\mathrm{SMD}| < 0.10$ for balance between the two arms. The run is CPU-only and trains nothing; it decides constructibility alone.

Of $5{,}704{,}050$ eligible targets, $1{,}560{,}345$ were retained. Row coverage is 27.88% on training and 20.55% on validation against the 80% gate, and event-group coverage is 87.55% on training but 73.77% on validation. The retained population is also unrepresentative on training, with standardised mean differences of $+0.5745$ (target block), -0.5171 (sibling count k), -0.3700 (activity density), $+0.2468$ (effective history) and $+0.2060$ (recency) against a threshold of 0.10. Arm balance is the one gate that is met — every $|\mathrm{SMD}|$ is at most 0.011 under all three donor assignments — so the pairs that do form are balanced, and what fails is how few targets form them and which targets those are. Descriptively, the all-siblings requirement is the binding constraint: retained targets carry fewer siblings (4.72 against 7.00) and lower activity density (6.25 against 22.11) than the eligible population, both directions that make a complete assignment easier. We do not claim this as a demonstrated mechanism.

Under the prespecified matcher and thresholds, this matched contemporaneous construction failed at least one feasibility criterion, so no B/C training was run. This conclusion applies only to this construction and does not imply that every alternative matching design is infeasible. We report it as a registered negative feasibility result: it improves transparency about what was attempted, but it also confirms that the confounding of event-group identity with contemporaneous state remains unresolved, and it is not evidence for either direction of the reported contrast.

J THE PERMUTATION CONTROL UNDER THREE DONOR ASSIGNMENTS

The control of Section 4.2 pairs each target with a donor from a different event group, and that pairing is one particular assignment. To check that the null is not an artefact of a single arbitrary pairing, we re-ran the control under three deterministic donor assignments — the original construction and two block-sorted alternatives differing in construction and random seed — on the *validation* population ($454{,}164$ rows, $4{,}051$ markets, 535 event groups), at the $160{,}000$-update budget, with five seeds. Macro and micro form two *separate* max-t families of ten quantities each (c=2.83 and c=2.78); the widths are not comparable between them.

Across three donor assignments the permutation control stays within $[-0.0020, -0.0006]$ nats macro, and all pairwise differences between assignments have simultaneous intervals containing zero: under each evaluated assignment the placebo gain is not detected, and no difference between assignments is detected. The null therefore does not rest on one particular evaluated permutation; it does *not* address the temporal confounding. The structural contrasts under the three assignments are -0.0457,

-0.0460 and -0.0471 nats macro, each with a simultaneous interval excluding zero, against a real-versus-target estimate of -0.0477. Under the micro family the real-versus-target estimate is -0.0108 and the three placebo estimates have magnitude at most 0.0001.

Four qualifications bind. First, *evaluated* is doing work: three assignments were run, so nothing follows about donor assignments in general. Second, an interval containing zero is not equality: the pairwise intervals have half-widths of 0.0021 to 0.0024 nats and reach 0.0038 nats, so the assignments are not shown to agree. Third, this analysis was run at a different training budget and on a different population from the control reported in Section 4.2, so it does not certify that specific estimate; it addresses only whether the choice of pairing drives the null. Fourth, the real-versus-target estimate quoted above is *not* an interchangeable replicate of the D_R reported in Appendix O, even though the two are computed on the same population at the same budget: the runs in this appendix use an earlier trainer version that ran to the full budget but still selected each checkpoint by best validation micro NLL, whereas Appendix O reports a same-version refit read at a fixed endpoint with no validation checkpoint selection. The two estimates therefore differ in the checkpoint rule, the trainer version and the refit itself, and the gap between them is not an estimate of any one of these.

K THREE-ARM CHANNEL-GROUP ABLATION

Three arms were trained under one protocol, one commit and one input digest, five seeds each, to a fixed 160k-update endpoint evaluated once: **target-only**; **restricted activity-only**, whose sibling mark embeddings and mark-derived scalars are zeroed while its gap bins and its recency and count features are kept; and the **full** sibling encoder (Figure 10; all six quantities in Table 8). Parameter counts are $116{,}931$ for target-only and $167{,}587$ for both the activity-only and the full arm, and every arm was trained to the same 160k-update endpoint. *The two sibling arms therefore match on **nominal** parameter count only*: zeroing the mark embeddings with `zeros_like` keeps them out of the computation graph, so they receive no gradient and are never trained. *Effective capacity is consequently not identical, and the contrast should not be read as holding capacity exactly fixed.*

Table 8 reports all six quantities. Macro and micro are *separate* $K{=}3$ simultaneous-inference families, corrected independently ($c = 2.26$ and $c = 2.32$); *their critical values and interval widths are not comparable, and no cross-family contrast is tested*. The five result states are the prespecified mutually exclusive classification against an unsigned materiality margin $m = 0.010$ applied to the point estimate. That margin is an *operational reporting threshold* fixed in advance, not an economic or trading-utility threshold: 0.010 nats corresponds to raising the geometric-mean probability assigned to the realised class by $e^{0.010} - 1 \approx 1.0\%$. The identity $I_C = D_F - D_A$ holds by construction within each bootstrap replicate; the largest absolute deviation over all replicates and the point estimate was 2.1×10^{-17}.

Table 8: Three-arm ablation: all six quantities. ΔNLL in nats; negative favours the first named arm. Intervals are max-t simultaneous 95% intervals from a nested event-×-seed bootstrap clustered on event groups, 2,000 replicates, 535 event groups, 5 seeds. $p < 0.0005$ *means below the* $1/2000$ *bootstrap resolution, not an exact zero. This table supports no statement about mechanism, about news, about event-group identity, or about any single mark channel, and it is not a decomposition: the two channel groups do not partition the encoder's inputs.*

contrast	point	SE	95% CI	Holm p	state
market-macro — primary family, $c = 2.26$					
full − target-only	-0.04490	0.00442	$[-0.05489, -0.03490]$	< 0.0005	material negative
activity-only − target-only	-0.00741	0.00217	$[-0.01230, -0.00251]$	0.001	sub-margin negative
full − activity-only	-0.03749	0.00393	$[-0.04638, -0.02859]$	< 0.0005	material negative
event-micro — secondary family, $c = 2.32$					
full − target-only	-0.01112	0.00102	$[-0.01348, -0.00877]$	< 0.0005	material negative
activity-only − target-only	-0.00305	0.00096	$[-0.00528, -0.00083]$	0.002	sub-margin negative
full − activity-only	-0.00807	0.00084	$[-0.01003, -0.00612]$	< 0.0005	sub-margin negative

What this does and does not establish. The primary market-macro contrast between the full and the restricted activity-only arm meets the prespecified materiality margin, so the evaluated mark-bearing and mark-derived channels carry a material *joint* increment over the evaluated timing and count channels. *This is a joint predictive channel contrast, not a mechanism.* It does not attribute the

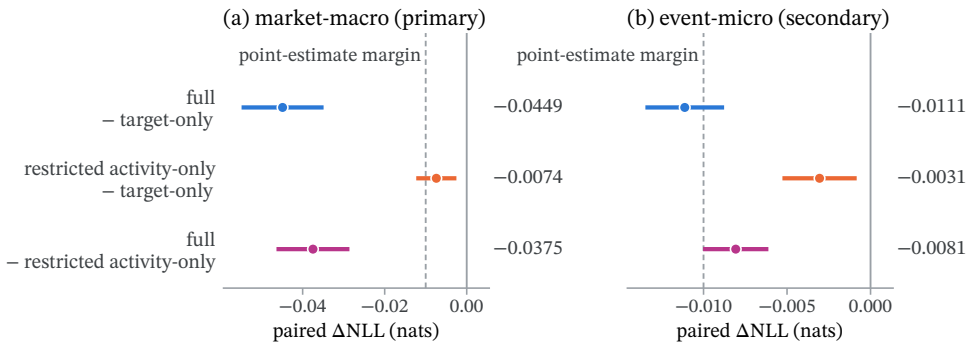

Figure 10: Three-arm channel-group ablation at a fixed 160k-update endpoint; ΔNLL is paired on identical evaluation rows and *negative values favour the first named arm*. *Macro* averages within a market then across markets; *micro* weights every target event equally. The *restricted activity-only* arm zeroes the sibling mark embeddings and mark-derived scalars while keeping the gap bins and the recency and count features — one restricted representation of activity, *not* a complete one. Bars are **max-t simultaneous 95% intervals** from a nested event-×-seed bootstrap **clustered on event groups**, 5 seeds. *Macro and micro form separate K=3 families (c=2.26 and 2.32) and support no cross-panel contrast; the two panels use different horizontal scales.* The dashed line is the 0.010-nat materiality margin, a criterion on the *point estimate* only. *The full-minus-activity contrast measures the joint increment of the evaluated mark-bearing and mark-derived channels; it identifies neither a mechanism nor a specific mark channel and does not exclude broader activity-based explanations.* Table 8 has all six quantities.

increment to news, to event-group identity, or to any individual mark channel, and *it does not exclude broader activity-based explanations*: the activity-only arm is one restricted representation of activity, not a complete one. The secondary event-micro contrast excludes zero but falls below the margin; *both families are reported and neither is used to infer the other*.

Provenance. The fifteen trajectories completed, but the first publication attempt was refused by a structural check on the prediction key names and that output directory was permanently terminated. The artefacts were recovered by byte-for-byte copy into a new directory under a recorded amendment, with every file's digest verified against a manifest taken before the copy; nothing was retrained and no prediction array was read before the decision to recover was taken. The fixed-endpoint record was materialised after the off-site backup rather than before it, and its attestation records that ordering. Digests for the protocol, the driver, the three prediction archives and every attestation are in the released result file.

L What the scaling curve does and does not separate

Across the four nested prefixes the market-macro gain strengthens from -0.0165 nats at 64 groups to -0.0483 at 2561, with the preregistered endpoint trend contrast -0.0318 $[-0.0427, -0.0209]$.

With the update budget held fixed, training-set size and repeat exposure per row are mechanically collinear — a $40.1\times$ range in eligible training rows against a $40.1\times$ range in expected repeat exposure — so Figure 11 is a *fixed-update-budget* result and not an isolated data-quantity effect. The relational advantage strengthens even as each training row is revisited about forty times less often, but no statement about training on more data at matched exposure follows from it.

Each tier draws a *single* deterministic nested subset of groups, fixed by the same hash ordering at every size. The five seeds therefore cover optimisation randomness only and carry *no* information about how much the curve would move under a different draw of groups at the same size. This binds hardest at the 64-group tier, where a single subset is the smallest fraction of the eligible population.

Checkpoint selection at the two smallest tiers is left-censored. Every run trains to the full 160,000-update budget with validation evaluated every 2,000 updates and early stopping disabled, so the selected checkpoint is the argmin over all 80 evaluated points and not an early stop. At the 64-group tier all 10 trajectories (two arms × five seeds) select the first evaluated point, and at 256 groups 9 of

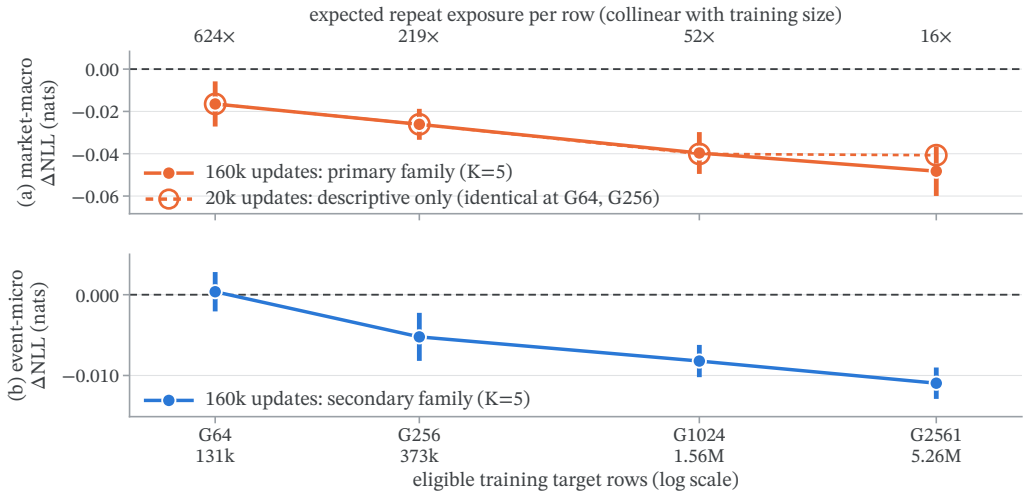

Figure 11: Sibling-context benefit against training-set size at a *fixed* 160k-update budget. Each point is $\Delta\text{NLL} =$ (sibling context) $-$ (target only); negative denotes improvement. Bars are **max-t simultaneous 95% intervals** from a nested event-×-seed bootstrap **clustered on event groups**, 5 seeds, within a *preregistered* family of five whose fifth member is the endpoint trend contrast. (a) market-macro, primary family (c=2.46). (b) event-micro, a separate secondary family (c=2.51); the smallest tier is *not* separately distinguishable from zero. *Open circles in (a) are a 20k-update budget in a separate diagnostic family used for no claim*; no 20k–160k contrast is tested. The top axis reports expected repeat exposure (tick labels rounded), *mechanically collinear* with training size because the budget is fixed. *Adjacent-tier differences are not in the family, so this figure does not establish that each successive increase is separately significant*, and the single deterministic subset per tier means the seeds cover optimisation randomness only.

10 do; at 1024 and 2561 groups none do. Performance before the first evaluated point is therefore unobserved at the two smallest tiers, and the preregistered endpoint contrast has one endpoint, the 64-group tier, inside that regime. Both arms are selected on the same grid, which limits the scope for differential selection but does not guarantee that it cancels, since the arms may differ in how they behave before the first evaluated point. The 64-group estimate is accordingly a difference of performance at a fixed checkpoint grid, not a difference of each arm's own best attainable performance. The ten checkpoints scored in Section 4.1 (*Confirmation on a sealed split*) are those of the 2561-group tier, where no trajectory is left-censored.

The four tier estimates and the endpoint trend form the family fixed before any NLL was read; adjacent-tier differences are *not* in it, so no single step is tested, and we make no functional-form claim and do not extrapolate beyond the largest tier evaluated.

M DESCRIPTIVE REANALYSES OF THE SIBLING GAIN

The three reanalyses below share one status: they *describe* how the development gain distributes across rows already scored, and none of them identifies a mechanism or licenses a causal reading. *None was pre-registered, none is a decomposition of the sealed estimate, and none changes any headline number.*

M.1 PER-CLASS ADDITIVE DECOMPOSITION (DESCRIPTIVE)

45–53% of price marks are FLAT, so it is reasonable to ask which class has the most-negative additive contribution. We answer it with an *additive* decomposition whose three parts sum exactly to the reported difference, rather than with conditional means.

Write $d_{s,i} = \ell^{\text{context}}_{s,i} - \ell^{\text{target}}_{s,i}$ for seed s and row i, so negative denotes improvement. For class k,

$$C^{\text{micro}}_{s,k} = \frac{1}{N}\sum_i \mathbf{1}[y_i = k]\, d_{s,i}, \qquad C^{\text{macro}}_{s,k} = \frac{1}{M}\sum_m \frac{1}{n_m}\sum_{i\in m} \mathbf{1}[y_i = k]\, d_{s,i},$$

with N the number of evaluation rows and n_m *all* rows of market m — *not* the class-specific counts, so a market with no row of class k contributes zero to that class and the three parts sum to the total. Each seed is decomposed independently and the five seeds are then averaged, matching the aggregation of Section 3.5.

class	rows	share	C^{macro}	C^{micro}
FLAT	249,568	53.24%	−0.02382	−0.00595
DOWN	110,967	23.67%	−0.00572	+0.00398
UP	108,236	23.09%	−0.01209	−0.00683
sum	468,771	100%	−0.04162	−0.00879

The sums reproduce the reported market-macro and event-micro differences to within 7×10^{-18}, and the recomputed totals match the frozen result file to the same order. *The class shares are those of this evaluation population*, which is not the corpus marginal of Section 3.2.

The most-negative contribution is FLAT under market-macro weighting and UP under event-micro weighting. **This is a descriptive decomposition of the corrected 20k development difference.** It carries no significance statement, it is not a decomposition of the sealed confirmation, and *an additive contribution is not evidence for a mechanism* — in particular it does not establish that the improvement is an activity or liquidity signal, and it does not show that directional information is absent.

M.2 Stratifying the sibling gain by own-history length (descriptive)

The sibling gain is 4.7 times larger under market-macro than event-micro weighting on the development held-out predictions; the corresponding sealed ratio is 3.7, but sealed data are not stratified here. This appendix describes the development-split gap only. It involved no retraining, checkpoint selection, or new forward evaluation.

Write δ_{is} for the per-row difference between the two arms (sibling context minus target-only, so negative denotes improvement) and stratify targets by S, their **own-history length capped at** 32 — the cap is the context builder's own-history window ($L_{\text{own}} = 32$, Appendix C), so in the scored inputs lengths beyond 32 are not observed. Bins, weighting, families and reporting rules were fixed before any of these quantities were computed. Two objects are reported, each as its own max-t family of seven: the conditional event-micro mean $H_b = N_b^{-1} \sum_{i \in b} \delta_i$, and an exact additive decomposition C_b satisfying $\sum_b C_b = \Delta_{\text{macro}} - \Delta_{\text{micro}}$; its sum agrees with the frozen full-precision development contrast to 1.4×10^{-17}.

S	rows	markets	groups	H_b (SE)	simultaneous 95% CI	Holm p
$\{0,1\}^\dagger$	3,704	3,704	535	−0.04376 (0.00763)	[−0.06452, −0.02299]	< 0.0005
$\{2\}$	3,081	3,081	510	−0.03939 (0.00590)	[−0.05545, −0.02333]	< 0.0005
$\{3\text{–}4\}$	5,307	2,760	499	−0.03670 (0.00590)	[−0.05276, −0.02064]	< 0.0005
$\{5\text{–}8\}$	8,863	2,386	463	−0.02810 (0.00432)	[−0.03987, −0.01634]	< 0.0005
$\{9\text{–}16\}$	14,626	1,992	434	−0.02159 (0.00273)	[−0.02901, −0.01417]	< 0.0005
$\{17\text{–}31\}$	21,542	1,640	389	−0.01692 (0.00231)	[−0.02321, −0.01064]	< 0.0005
$\{32\}$	411,648	1,252	343	−0.00659 (0.00057)	[−0.00814, −0.00505]	< 0.0005

	C_b (SE)	simultaneous 95% CI	Holm p
$\{0,1\}^\dagger$	−0.01850 (0.00322)	[−0.02725, −0.00974]	< 0.0005
$\{2\}$	−0.00540 (0.00121)	[−0.00868, −0.00212]	0.0010
$\{3\text{–}4\}$	−0.00554 (0.00101)	[−0.00829, −0.00278]	< 0.0005
$\{5\text{–}8\}$	−0.00351 (0.00055)	[−0.00500, −0.00203]	< 0.0005
$\{9\text{–}16\}$	−0.00223 (0.00035)	[−0.00317, −0.00128]	< 0.0005
$\{17\text{–}31\}$	−0.00115 (0.00029)	[−0.00194, −0.00035]	0.0010
$\{32\}$	+0.00349 (0.00046)	[+0.00225, +0.00473]	< 0.0005

† No $S = 0$ rows occurred in this population; the frozen $\{0,1\}$ bin therefore contains $S = 1$ rows here.

The seven H_b point estimates **increased monotonically**, from −0.04376 at $S \in \{0,1\}$ to −0.00659 at $S = 32$, and all seven simultaneous intervals exclude zero. **This is not a significance test of a**

monotone trend; no trend contrast was pre-registered. The pattern is a descriptive association with capped own-history length. Holm-adjusted p values are reported against the empirical resolution $1/2000$ of the resampling.

In the decomposition, the largest single contribution is -0.01850 in the shortest bin, which holds exactly one row per market, and the only positive contribution is $+0.00349$ at $S = 32$, which holds 87.8% of the rows. The C_b terms add exactly to the macro–micro gap, but their mixed signs preclude interpreting them as non-negative shares.

An additive contribution is not evidence for a mechanism: this decomposition is an exact re-expression of the two weightings, and it does not establish that sibling context substitutes for absent own history. The inference was run as three gated stages — identity and alignment, array integrity, then the scientific computation — and the three consistency attestations are released with the benchmark.

M.3 Conditioning the sibling gain on the realised mark class (descriptive)

Because FLAT is the largest class, it is reasonable to ask whether the sibling gain is confined to it. We answer with a *conditional* estimand fixed in advance: the same per-row difference restricted to FLAT rows and to directional (UP ∪ DOWN) rows. This is a development-split analysis; the sealed split is neither stratified nor reused here, and no forward evaluation, retraining or checkpoint selection was involved.

Write δ_{is} for the per-row difference between the two arms (sibling context minus target-only, so negative denotes improvement). For a class set c, the **event-micro** quantity is the mean of $\bar{\delta}_i$ over the rows of c, and the **market-macro** quantity averages within each market over *that market's rows of* c and then over the markets that have at least one such row. *Those denominators and market sets differ between the two class sets and from the unconditional ones*, so the two conditional quantities are not a decomposition and do not sum to the reported difference; the object that does sum exactly is the decomposition of Appendix M.1. Market-macro and event-micro form two *separate* max-t families of three, each comprising the two class sets and the contrast between them, which is tested directly rather than inferred from which class set is significant.

weighting	quantity	point	simultaneous 95% CI	Holm p
market-macro	FLAT	-0.04429	$[-0.07873, -0.00986]$	0.013
(K=3, c=2.17)	directional	-0.03162	$[-0.05527, -0.00798]$	0.007
	contrast	$+0.01267$	$[-0.04254, +0.06788]$	0.614
event-micro	FLAT	-0.01117	$[-0.02945, +0.00711]$	0.631
(K=3, c=2.04)	directional	-0.00608	$[-0.02704, +0.01487]$	1.000
	contrast	$+0.00509$	$[-0.03393, +0.04411]$	1.000

The FLAT set has $249{,}568$ rows over $3{,}086$ markets and 516 event groups, the directional set $219{,}203$ rows over $3{,}312$ markets and 519 groups; both market counts are below the $3{,}704$ of the full population, which is why the two are not a partition of it.

Under market-macro weighting, on directional rows the sibling gain remains detectable at this resolution. The contrast between the two class sets *was not detected at this resolution*; an interval containing zero is not equality, and no equivalence test was pre-registered, so nothing here shows the two class sets to be alike. Under event-micro weighting all three event-micro intervals contain zero, and we report them unconditionally.

Three limits bind. Conditioning on a realised class is *descriptive, not a decomposition*, and it is available only after the label is known; it is *not* a gate a deployed model could apply. The result shows that the gain is **not confined to FLAT rows**; it does *not* refute an activity-related explanation of the gain, and the FLAT point estimate under market-macro weighting remains the more negative of the two while the contrast contains zero. And these are conditional means on the development split, not a sealed confirmation.

Absolute levels on the same population. On the full development relational test population, market-macro NLL is 1.0795 for a marginal predictor fitted on the relational training split's class frequencies, 1.0238 for target-only and 0.9822 with sibling context. *The* FLAT *share is* 46.57% *in*

that training split and 53.24% *in this evaluation population*, so part of the marginal predictor's loss reflects that difference between populations; it is a same-population reference point, not an estimate of how hard the task is.

The analysis ran as three gated stages — identity and metadata, array integrity, then the scientific computation — whose attestations are released with the benchmark; the collapse of the class sets to all three classes reproduces the reported unconditional difference, and every bootstrap replicate reproduces it as well.

N TWO SIBLING AGGREGATORS ARE NOT DISTINGUISHED AT A FIXED UPDATE BUDGET

The reported context arm pools sibling representations by concatenating their mean and their maximum. To ask whether the benchmark separates relational architectures, we replaced only that aggregation with a single-layer target-conditioned cross-attention — four heads of sixteen, with no feed-forward block, normalisation, residual or dropout — and kept the maximum half unchanged. Encoders, features, training rows, the full 2561-group tier, the $160{,}000$-update budget and the five seeds are all unchanged, and a controlled parameter setting makes the attention arm reproduce the pooling arm exactly, so the two are nested by construction. The comparison, its direction and its three outcome branches were fixed before the run.

Under the frozen configuration and compute budget, we did not detect a difference between target-conditioned attention and mean–max sibling pooling. The market-macro estimate is $+0.00015$ nats (95% CI $[-0.00378, +0.00408]$, Holm $p = 0.94$, family of one) and the event-micro estimate, a separate family of one, is $+0.00011$ $[-0.00047, +0.00068]$ ($p = 0.70$), both on the development-validation population of $454{,}164$ rows, $4{,}051$ markets and 535 event groups. *An interval containing zero is not equivalence*, and this experiment covers two aggregators only, so it does not close the gap in relational-baseline breadth.

The attention arm has $188{,}323$ parameters against $167{,}587$, and on identical batches without evaluation it costs 12.59 against 11.04 seconds per thousand updates, at peaks of 188.3 against 187.7 MiB. Its selected checkpoints are at 154k, 130k, 90k, 92k and 124k updates, so no seed selects the first evaluated point or reaches the cap. *All selected checkpoints lie inside the evaluated grid; this does not establish convergence or budget independence*, and we do not claim the null would survive a larger budget. The pooling arm reuses the frozen predictions of the 2561-group tier while the attention arm was trained with a later revision of the training script; a deterministic regression test on a CPU fixture reproduces the pooling arm's logits, loss, gradients and post-update parameters bitwise, and both arms carry the same input digest. This is a development-split result at a fixed budget with a single attention configuration, not a sealed confirmation and not a comparison between architecture families. Appendix P reports a third aggregator against a baseline retrained from the same commit.

O CROSS-GROUP ACTIVITY SUMMARIES DO NOT ELIMINATE THE SIBLING GAIN

A natural alternative to a cross-market information channel is contemporaneous market activity: if busy periods are busy everywhere, a sibling context could be a proxy for a common activity level rather than for cross-market information. We therefore built four pre-specified cross-group activity summaries — computed from the relational track's training and development-validation events in the $1{,}000$ blocks strictly before the target block, excluding the target's own event group and every development-test event, with the FLAT-share prior and the standardisation fitted on training rows only — and appended them to the target representation. Every training event precedes B_1, so training events alone would leave almost every validation window empty; with this panel 98.49% of validation rows have a non-empty window. Four arms were retrained from the same commit at the frozen $160{,}000$-update budget with five seeds each, and read at a fixed endpoint with no validation checkpoint selection: a sibling and a target-only arm without the summaries, and the same two arms with them. A controlled parameter setting makes each activity arm reproduce its counterpart exactly, so the arms are nested by construction; all four share bitwise-identical digests on every common input, and the two activity arms add the same summary array.

Two contrasts and their interaction were fixed before the run: D_R, the sibling contrast without the summaries; D_G, the same contrast with them; and $I = D_G - D_R$. On the development-validation population of $454{,}164$ rows, $4{,}051$ markets and 535 event groups, the market-macro family (three quantities, simultaneous max-t critical value 2.35) gives $D_R = -0.04353$ $[-0.05436, -0.03270]$, $D_G = -0.04528$ $[-0.05560, -0.03497]$ and $I = -0.00175$ $[-0.00668, +0.00317]$; the event-micro family, reported as a secondary family of the same three quantities, gives -0.01123 $[-0.01354, -0.00892]$, -0.01149 $[-0.01359, -0.00939]$ and -0.00026 $[-0.00203, +0.00151]$.

The evaluated cross-group activity summaries do not eliminate the sibling gain; a change in the contrast was not detected at this resolution. We report I with its interval and leave the comparison with any threshold to the reader. Three limits are worth stating plainly. This is an observational control, *not* causal identification. D_R is a same-version refit that reproduces a material-negative contrast on the development split; it is *not* an independent replication of the sealed headline, and it is *not* an interchangeable replicate of the real-versus-target estimate in Appendix J, which was read under a different checkpoint rule and an earlier trainer version. The experiment addresses the activity-summary alternative only: it does not resolve group identity, contemporaneous donors, or common temporal shocks, since event groups share the same history and the resampling is conditional on the realised features.

P A THIRD SIBLING AGGREGATOR: ATTENTION AMONG SIBLINGS

Appendix N replaces mean–max pooling with target-conditioned cross-attention. A second axis of relational structure is left untouched there: siblings never see each other. We therefore added a third arm in which a single layer of self-attention — four heads of sixteen, with no feed-forward block, normalisation or dropout, and a residual connection — runs over the siblings of each target before the *same* mean–max pooling. Attention is restricted to siblings of the same target, and zeroing the output projection makes the arm reproduce the pooling arm bitwise, so the pooling arm is a special case of it. The third arm adds $16{,}640$ parameters ($184{,}227$ against $167{,}587$), fewer than the cross-attention arm adds.

Unlike Appendix N, both arms here were trained from the same commit: five seeds each, the frozen $160{,}000$-update budget, no intermediate evaluation and no checkpoint selection, with a bitwise-identical input digest. The estimand, the aggregation, the resampling seed and five signed outcome states were fixed in a protocol frozen before any prediction file was opened.

Under this configuration and compute budget we did not detect a difference between sibling self-attention and mean–max pooling. The signed market-macro difference is -0.00044 nats (95% simultaneous CI $[-0.00458, +0.00370]$, Holm $p = 0.83$, family of one); the event-micro family, reported unconditionally as a secondary family of one, gives $+0.00003$ $[-0.00083, +0.00089]$ ($p = 0.93$). On identical batches the arms cost 9.22 against 8.02 seconds per thousand updates.

An interval containing zero is not equivalence, and no equivalence test was pre-registered. The self-attention arm carries $16{,}640$ more parameters, so a null cannot be attributed to the aggregator alone; only the nesting direction was established, not that the third arm is strictly more expressive; and three aggregators do not cover the space of relational architectures. This is a development-split result at a fixed budget with a single configuration, not a sealed confirmation.

Q DATA CARD

Source corpus. The release includes the source corpus PolyEvent is built from: $5{,}412{,}260{,}320$ rows in five tables, namely $1{,}713{,}705{,}587$ raw order-fill logs (every on-chain fill event, including fills not linked to a market), $1{,}028{,}646{,}108$ market-linked trades, a YES-normalised view of the same trades ($1{,}028{,}646{,}107$ rows), $1{,}639{,}420{,}835$ per-address records (keyed by address, with a maker or taker role flag) and metadata for $1{,}841{,}683$ markets, together with the collection pipeline. It covers the same period and exchange contracts as PolyEvent. Its raw logs, trades and per-address records carry the pseudonymous on-chain addresses (Ethics statement).

Composition. $281{,}433{,}584$ `(market, block)` events over $1{,}535{,}897$ markets and $652{,}016$ platform event groups, derived from $1{,}028{,}646{,}108$ CLOB-period fills recorded on Polygon between

2022-11-21 and 2026-07-20. After a minimum-length filter, 733,857 market streams and 274,477,061 events remain. Experiments use a frozen 10M-event tier (Appendix C).

Fields. Each event record carries `market_id`, `event_id`, `block_number`, `event_index`, `price_mark` $\in \{$UP, FLAT, DOWN$\}$, `delta_block`, and the split assignment; `event_id` is what makes the relational track reconstructible by a third party. A separate market-level table carries each market's question text, `negRisk` flag and creation time, linked to events by `market_id`. The released event manifest retains unbinned `delta_block`; derived relational context artefacts retain only its seven-bin-plus-start input tokenisation and cannot reconstruct the exact gap. Event groups are the platform's own `event_id` grouping, not inferred by us. **No address, account, order or trader-level field is included** in the benchmark tables.

Collection and processing. Fills were read from public chain transactions, aggregated to the `(market, block)` unit (Section 3.1), ordered by block number, and marked by the sign of the YES-normalised price change. Section 4.4 documents the four construction choices this differs from and why.

Splits. The cold-event protocol holds out whole event groups. The sealed split is a deterministic hash of the group identifier, fixed before any scaling result existed; its membership digest and the artefact digests are in Appendix G.

Known limitations. The mark is a three-way discretisation of price change and discards magnitude. The original reference models read gap–mark histories without absolute price levels, and the released event records carry marks rather than prices. A post-hoc XGBoost extension adds the target's last price and the blocks since its last trade (Appendix R); the data release carries the price as a separate file aligned by `tid` for the development relational targets, and the elapsed blocks likewise; full price histories, volume and order-book inputs remain untested. The relational track applies only to multi-market groups. One snapshot of one chain: we make no claim about later data, other chains, or other prediction-market platforms. Negative elapsed times *under block ordering* (Appendix B) are a property of this snapshot's recorded timestamp field, which mixes read and interpolated block times; they are negative only when the timestamp field is read against the canonical block order, and we did not attribute their cause. The mark is taken from the block's last recorded trade price, so it can carry execution-side effects such as bid–ask bounce: the target is the predictability of *recorded trade-price marks*, not of a latent belief process.

Distribution and licence. The source corpus, the derived event tables, split manifests, code and frozen artefacts are released with their licensing and redistribution terms; public links are withheld during anonymous review and will be added upon publication; the underlying chain transactions are public records not owned by us. Maintenance is by the authors; the release is versioned by content digest so that a reported number can always be traced to the artefact it came from.

R A NON-NEURAL BASELINE: XGBOOST WITH FIXED SIBLING SUMMARIES

This experiment asks one question: does the sibling-context increment survive outside our model family, when a standard non-neural learner is given the same rows and *fixed* low-dimensional summaries of the same sibling windows?

Contract. Both arms are XGBoost models (multi-class softprob, eta 0.1, depth 8, subsample and column subsample 0.8, histogram method, CPU, 16 threads, at most 400 boosting rounds with validation early stopping after 25; seeds 1–5) trained on the frozen relational rows (5,668,006 train / 454,164 validation). X-OWN uses the 27-dimensional own-history block; X-SIB appends 19 fixed sibling-window summaries (presence and count, event-count sums/max/mean per window, per-mark sums, minimum and mean block distance, nearest-sibling last mark, and last-mark counts over $\{0, 1, 2, 3\}$). The two arms share rows, splits, seeds and configuration; zero-sibling rows are retained with explicit zeros. Row identity against the frozen relational artefacts, feature-block parity between arms, leakage and domain gates (strictly-earlier siblings, mark/gap domains, window bound), and a per-row identity attestation over the parquet inputs all passed before training.

Health gates (before any test scoring). All losses finite; best iteration between 121 and 147 across the ten fits; both arms beat the train-frequency marginal (validation log-loss 1.0429) on every seed (X-OWN 0.9232–0.9233, per-seed mean 0.9232; X-SIB 0.9169–0.9172, mean 0.9171); training completeness matched the frozen row counts, seed set, configuration and feature dimensions exactly. The single test inference ran only after these gates were green.

Result. Estimands and inference follow the shared contract: paired ΔNLL of X-SIB minus X-OWN on identical development-test rows (468,771 rows, 535 event groups, 3,704 markets), nested event-group-by-seed bootstrap with one shared seed draw per replicate, 2,000 replicates, one $K{=}3$ family (event-micro, market-macro and their difference) with max-t critical value $c{=}2.322$ and Holm adjustment. Negative favours X-SIB.

quantity	point	SE	simultaneous 95% CI	Holm p	five-state
market-macro	-0.03039	0.00371	$[-0.03901, -0.02176]$	< 0.0005	**material negative**
event-micro	-0.00920	0.00322	$[-0.01668, -0.00173]$	0.0015	sub-margin negative
difference	$+0.02118$	0.00451	$[+0.01071, +0.03166]$	< 0.0005	material positive

The margin is the project-wide 0.010 nats. p-values below 0.0005 are zero exceedances in 2,000 draws (empirical resolution $1/2000$). The positive difference says the two weightings are distinguishable — the macro-weighted increment is larger — and nothing else.

Absolute levels and the reporting contract. Against the frozen official reference levels (Table 2; reporting contract in Section 3.5):

model	event-micro	market-macro
X-OWN (XGBoost, own block)	0.87154	1.03499
X-SIB (XGBoost, + fixed summaries)	0.86233	1.00461
GRU target-only (official reference)	0.82357	1.02379
GRU sibling-pool	0.81478	0.98217

Comparing each arm against the *same-information* GRU reference — the appropriate descriptive pairing — both XGBoost arms remain above it: X-OWN minus GRU target-only is $+0.04797/+0.01120$ micro/macro, and X-SIB minus GRU sibling-pool is $+0.04756/+0.02244$. These differences are **descriptive only**: no cross-model significance is tested and no cross-model inference is drawn, X-OWN is a within-family control and not a reference, and a null here would not have shown sequence encoding necessary — these fixed summaries are one aggregation, not the space of them.

Governance and budget. The experiment ran as one post-hoc development-test chain (Stage 5 in Appendix G): identity, feature construction and leakage gates; a resource smoke on a 5% stride subsample with same-seed bitwise-determinism verification; then formal training and the single gated test inference. The first smoke measured a conservative upper-bound projection of 12.54 CPU-hours against a pre-registered 8 CPU-hour operating budget and was *rejected and recorded*; the budget was raised to 16 CPU-hours by a dated amendment before any formal training, and the re-run chain passed at a projection of 13.98 CPU-hours (measured feature-build peak 51.5 GiB). No rerun after results.

Provenance. All numbers above are read from the frozen inference artefact (sha256 `3f9b8761...`) and re-derived from its stored bootstrap matrix; the paired predictions (`ee6813cc...`), health record (`ad12cd37...`) and test attestation (`0be93d7c...`) are byte-pinned by a 23-file manifest, together with the executed chain's identity, build and resource records and the recorded first-smoke rejection. The official reference is pinned by the shared reporting contract (`6995394f...`).

Adding the target's last price (post-hoc). The original reference models read no price levels (Section 5). A separate chain, frozen before any price was recovered, asks whether the sibling summaries still help once the target's last price is known. Both arms append one feature, the YES-normalised close of the target market's most recent event strictly before the target block, recovered

with the split manifest's own price rule; recomputing every mark from the recovered prices reproduces all 6,590,941 frozen labels. O+P is the 27-dimensional own block plus that price and C+P adds the 19 sibling summaries; configuration, rows, seeds and inference are those above, and the development-test split was scored once more (Stage 6 in Appendix G).

model	event-micro	market-macro
X-OWN (above)	0.87154	1.03499
X-SIB (above)	0.86233	1.00461
O+P (own block + last price)	0.85105	0.89024
C+P (O+P + sibling summaries)	0.84445	0.86756
quantity	point	simultaneous 95% CI
C+P − O+P, market-macro	−0.02268	[−0.02933, −0.01603]
C+P − O+P, event-micro	−0.00660	[−0.01190, −0.00130]
interaction, market-macro	+0.00771	[+0.00149, +0.01392]
interaction, event-micro	+0.00261	[+0.00018, +0.00503]

The primary family (K=3, c=2.35) is C+P minus O+P: market-macro is material negative (Holm $p < 0.0005$), event-micro sub-margin negative (Holm $p = 0.0015$), and their difference $+0.01608$ $[+0.00830, +0.02387]$. The secondary family (K=3, c=2.28) is the interaction $(\text{C+P} - \text{O+P}) - (\text{X-SIB} - \text{X-OWN})$ on the same draws (Holm $p = 0.012$ macro, 0.024 micro; the micro-minus-macro difference -0.00510 $[-0.01159, +0.00139]$ contains zero). Descriptively, the last price lowers market-macro NLL by 0.145 for the own-only arm and 0.137 with sibling summaries (event-micro 0.020 and 0.018), far more than the sibling summaries do. Given the price, the summaries still give a material market-macro gain, smaller than without it by 0.0077 nats, about a quarter. Both arms share the frozen 400-round cap; two of five O+P seeds reached it (best round 396) and the other three stopped early, as did every C+P seed (best rounds 232–324), so the contrast holds for the frozen budget, not for either arm at convergence. This covers one price level in one model family: it does not control for the full price history, volume or the order book, and it says nothing about the GRU arms. The protocol, the recovery, build, resource, health and test records and the inference file are released with their digests.

Adding the time since the target's last trade (post-hoc). The target arm sees the gaps between its past events but not how many blocks have passed since its last trade, whereas the sibling summaries include each sibling's distance to the target block. A frozen amendment appends to both arms $b - b_{\text{last}}$, the blocks since the target market's previous event (median 18), recomputed with the same price rule; the recovered prices equal the frozen ones exactly, the 28 and 47 earlier columns are reused unchanged, and the development-test split was scored once more (Stage 7 in Appendix G). This is an input extension for T2, not a T1 input.

model	event-micro	market-macro
O+P+G (O+P + blocks since last trade)	0.83793	0.84803
C+P+G (C+P + blocks since last trade)	0.83176	0.82563
quantity	point	simultaneous 95% CI
C+P+G − O+P+G, market-macro	−0.02240	[−0.02957, −0.01523]
C+P+G − O+P+G, event-micro	−0.00617	[−0.01108, −0.00126]
interaction vs. the price pair, market-macro	+0.00028	[−0.00567, +0.00624]
interaction vs. the price pair, event-micro	+0.00043	[−0.00058, +0.00143]

The primary family (K=3, c=2.28) is C+P+G minus O+P+G: market-macro is material negative (Holm $p < 0.0005$) and event-micro sub-margin negative (Holm $p = 0.0020$). The secondary family (K=3, c=2.28) is the interaction $(\text{C+P+G} - \text{O+P+G}) - (\text{C+P} - \text{O+P})$ on the same draws; all three of its intervals contain zero, so no change in the context gain was detected once the elapsed blocks are added, which is not equivalence. The elapsed blocks themselves lower market-macro NLL by 0.042 in both arms (descriptive). All five O+P+G seeds reached the 400-round cap with their best round among the last seven, while every C+P+G seed stopped early; over its last 25 rounds O+P+G's validation log-loss fell by at most 0.00006, small beside the validation gap of about 0.004 between the arms; this row-weighted validation reading shows neither market-macro convergence nor that

longer training would leave the gap unchanged. The contrast therefore holds for the frozen budget, not for either arm at convergence, and it covers one model family and two own inputs, not the full price history.

S PREDICTIVE GAINS BY NEGRISK STATUS AND SIBLING RECENCY

Post-hoc descriptive stratification of the already-reported development-test rows, re-using the frozen per-row predictions; nothing is retrained. The driver reproduces the pooled reading (-0.0416 market-macro, -0.0088 event-micro) to 10^{-15} before any stratum is formed. Three families, each with max-t simultaneous 95% intervals computed separately for the two weightings: `negRisk` status ($K = 3$), sibling recency ($K = 4$), and their crossing ($K = 4$). Negative favours same-group context.

stratum	rows	groups	market-macro	event-micro
`negRisk`	247,306	160	-0.0535 $[-0.0758, -0.0311]$	-0.0107 $[-0.0144, -0.0069]$
not `negRisk`	221,465	375	-0.0359 $[-0.0475, -0.0244]$	-0.0067 $[-0.0086, -0.0048]$
difference			-0.0175 $[-0.0422, +0.0072]$	-0.0039 $[-0.0082, +0.0004]$
gap ≤ 3 blocks	168,192	360	-0.0659 $[-0.0877, -0.0442]$	-0.0130 $[-0.0204, -0.0057]$
gap 4–1000	265,784	486	-0.0326 $[-0.0418, -0.0233]$	-0.0073 $[-0.0093, -0.0053]$
no sibling in window	34,795	497	0.0032 $[-0.0012, +0.0075]$	0.0004 $[-0.0021, +0.0029]$
difference (near − far)			-0.0334 $[-0.0558, -0.0109]$	-0.0057 $[-0.0129, +0.0015]$
`negRisk` × near	102,836	129	-0.0853 $[-0.1254, -0.0452]$	-0.0144 $[-0.0293, +0.0004]$
`negRisk` × far	136,692	145	-0.0389 $[-0.0556, -0.0223]$	-0.0084 $[-0.0115, -0.0053]$
not `negRisk` × near	65,356	231	-0.0525 $[-0.0746, -0.0303]$	-0.0108 $[-0.0215, -2.4 \times 10^{-5}]$
not `negRisk` × far	129,092	341	-0.0293 $[-0.0405, -0.0180]$	-0.0061 $[-0.0089, -0.0034]$

What the table says. Gains appear in `negRisk` and non-`negRisk` groups alike, and the difference between those two populations is **not detected** under either weighting. Gains are larger at sibling gaps of at most three blocks than at 4–1000 blocks; that near-minus-far difference **excludes zero under market-macro and contains zero under event-micro**, so it is a one-weighting reading. The crossing cell that is neither `negRisk` nor recent shows a gain under both weightings. Targets with no sibling activity in the window show no detected gain.

What the table does not say. `negRisk` is the platform's flag for one exclusivity mechanism; its absence does *not* establish that the markets are non-exclusive, so these rows cannot rule out an exclusivity explanation. The strata differ in composition — group counts, market counts and activity all vary across them — so a difference between two strata is **not** a statement about how much proximity causes. The far bin spans 4–1000 blocks, roughly eight seconds to half an hour, so it does not exclude arbitrage on a minute scale. These estimates are **not** a decomposition of the pooled gain, and none of this is evidence about semantics.

T A ROLE-AWARE METHOD ON THRESHOLD GROUPS

Post-hoc exploratory development evidence. This study was specified after the main results were known; it is evaluated on **development validation** at a fixed 160,000-update endpoint with **no checkpoint selection**, so those rows enter no training and no selection here. It is **not** a sealed confirmation, and it tests the predictive value of **one** structure — the relative order of numeric thresholds — so it is **not** evidence about market semantics. The submission carries the protocol and its four amendments, the parser, the group-classification rules, the role-table builder, the result files of the current and the three superseded versions, and the build attestations, which record the content digests of the two role tables used here; the tables themselves are rebuilt by that builder and are released with the event tables (Appendix Q), so this appendix claims no bitwise reproduction of them.

Why this structure. The reported sibling encoder receives eight summary features per sibling (Appendix C) and no threshold-derived input, so the fact that a sibling's threshold lies above or below the target's is absent from its input, not discarded by its architecture. R keeps the same mean–max pooling and only attaches the role values to each sibling, which a permutation-invariant encoder can

use; what changes is the available feature, not the invariance. Threshold groups ("will BTC be above $100k", "will the high temperature be 26°C") are common, and the relative order of two thresholds is set by the question texts, not by trading; we check recorded creation times, not the text at each prediction time (below).

Role features and what they may not use. A group is labelled `threshold` by a deliberately conservative, high-precision rule on the question texts alone. The questions must share a common token prefix and suffix totalling at least three tokens; the differing spans must be non-empty, pairwise distinct, and at most six tokens each; every span must contain a digit; and either the shared template or every span must carry a comparison or unit cue. Groups whose questions match a sports side-bet pattern are labelled separately and never enter this population, and a group that fails any condition is labelled `other` rather than guessed at. The rule is `classify_group` in the released preflight audit and the threshold parser described next is `parse_group`; both ship with the submission, so the labelling can be re-derived rather than taken on trust. Within a group labelled `threshold`, every market — target markets *and* markets that only ever appear as siblings — is parsed into a sort key and a group type (`ladder_up`, `ladder_down`, `bins`). A group parses only if all of its markets parse, keys are unique, directions are not mixed, and (since the correction below) no varying span is a date, the spans lie on one numeric axis, and no span is a percent range whose hyphen would be read as a minus sign; otherwise its role features are exactly zero. Each (target, sibling) pair receives six values: a parsed flag, the sign of the key difference, the normalised rank difference, and the group type as a one-hot. **Raw threshold values never enter the model** (they are not comparable across events), and the rank is computed only among group markets whose first *observed* trade block precedes the target block, so a later-listed rung cannot influence an earlier prediction.

Coverage and manual check. Of the 165 development-validation threshold groups, 156 parse (94.5%; 128,168 of 130,547 rows, 1,405 markets); 114 of 130 parse on the development test split, one fewer after the correction below, which leaves validation unchanged. In 25 parsed groups drawn from train and validation, the group type and the induced order were correct in $25/25$ by manual inspection. The nine unparsed validation groups are direction-mixed in every sampled case ("reach $X" together with "dip to $Y"; "hit (HIGH)" together with "hit (LOW)"), which the rules deliberately refuse. The original parsing rules were frozen before any loss was read; the correction below was fixed before any loss of the run it governs.

The information set of the role features. Group type, parse success, keys and rank are all computed from the markets *visible at the prediction block*: a market counts only once it has an observed trade at a strictly earlier block, where "observed" means recorded in the frozen context artefacts, which see a market no earlier than its first trade. *This took two corrections, and we report both.*

The first version decided group type and parse success over every market the group ever contains in the split, so a market joining later could change an earlier prediction's features. Two gates bound that replacement: run in whole-split mode the builder reproduces the first version's tables *bit for bit*, so only the visibility rule changed; and re-auditing the replacement under the visibility rule, with an independent reimplementation, reproduced it value by value (580,319 validation and 7,133,459 training pairs per arm) and found **no row whose features would change**.

That audit was blind to a second dependence, and the blindness was structural: its reimplementation decided eligibility the same way the builder did, from the set of markets appearing as a prediction target *anywhere in the split*. A market enters that set only if it has a labelled event, which depends on whether it trades again later — so whether a market would later be scored could switch the role features of an earlier row on or off. Reproducing the builder cannot detect a property the builder itself lacks. The current version decides eligibility and group type from the visible markets alone, and is checked by a test aimed at the property rather than at agreement: holding every present and past record fixed and appending a *later* target row must leave the current features bit-for-bit unchanged. The same test is run against the superseded rule, where it must fail, so that a passing result cannot come from a test that measures nothing.

Removing the dependence *adds* role features to rows the previous rule had zeroed and changes no existing row: across both arms and all splits, 162 target–sibling pairs appear that were absent before, none disappear, and no shared row takes a different value. Those pairs sit in 3 training, 5 validation and 33 test target rows, and touch 3 of the 156 evaluated groups. The evaluated population

is unchanged. The arms were retrained on the corrected tables; that run was in turn superseded by the parsing correction below, and its numbers are kept as a corrected record.

A second rule, in which a market counts once its recorded contract-creation time precedes the block's timestamp, is *more permissive*: it would admit markets that exist but have not yet traded. Rebuilding under it changes 4,330 of the 128,168 evaluated rows (3.38%) and 15,134 training rows for R, and slightly more for R_shuf, almost entirely through rank. We did not adopt it: block numbers are this benchmark's ordering key, the timestamp field mixes recorded and interpolated values (Appendix B), and the stricter rule needs only the released context artefacts. *Those counts were measured against the second version's construction and are reported as that audit found them; they were not recomputed against the current one.* The disagreement between the two rules is reported, not resolved.

The audit also flags 9 validation and 29 training groups as mixed-direction candidates by movement verb; the parser refuses all of them, so none contributes features. That is evidence this audit found no parsed group affected, **not** that the parsing rules are complete.

A parsing correction. A later audit read the question texts behind every parse the role tables used (4,950 distinct parses of visible member sets) and found parses that were wrong rather than missing, none of them in the evaluated groups. In six training groups the parser had read a date as a threshold ("will … tweet again on March 9, …, 15, 2026"), placed two candidates' leads on one axis without their sign ("Trump lead by …" with "Harris lead by …"), placed a point spread and a points total on one axis (three games), or ranked counts of different words; on the development test it had also read deadline dates as thresholds in two groups and the hyphen of a percent range as a minus sign in one. Three rules were added, each of which only refuses a parse: a varying span that is a date, spans that do not lie on one numeric axis, and a percent range written with a hyphen; no kept parse changed type or key. The correction removes 14,570 of 7,133,466 training target–sibling pairs (0.20%), so 3,261 training target rows lose their role features while remaining training targets, and 205 development-test pairs; it removes **no validation pair**, and the 156 evaluated groups are unchanged. The rules, the expected change for every affected group and the reporting plan were fixed in a fourth amendment before the rebuild; the rebuilt tables equal the previous ones minus exactly those pairs, in both arms, and R and R_shuf were retrained with B reused. Every number below is from that run; the previous one is reported as superseded.

Three arms. All three share the context artefact, the training rows, seeds 1–5, the 160k fixed endpoint, AdamW at 10^{-3} with the default weight decay, batch 512, and the encoder widths, and differ only in the sibling scalar block. They are not training runs of one commit: B was trained for the first version and is reused unchanged, while R and R_shuf were retrained for the current role construction; the shared inputs are verified by the per-key input digests below. **B** is the reported pooling arm (8 summary features; its input fingerprint equals the frozen fingerprint of Appendix K's arms bit-for-bit). **R** appends the six role values (14 features). **R_shuf** is architecturally identical to R but, inside each parsed group, the map from markets to threshold keys is replaced by a fixed-seed derangement before the six values are recomputed — same architecture, same parameter count, same feature definitions and the same number of role values per pair, with the correspondence broken. It is **not** a distribution-preserving intervention: recomputing the features after the derangement shifts their marginals, from 51.81% to 49.50% of pairs with a positive key difference and from 0.3391 to 0.4057 in mean absolute normalised rank distance (build attestations). R_shuf therefore controls capacity and feature definition, *not* feature distribution. Every non-`sf` per-key input digest of R and R_shuf equals B's; only `sf` and the added `sib_role` array differ.

Reproducing the pooling arm. B reads no role table — its sibling block is the eight sibling summary features — so neither the visibility rules nor the parsing correction can alter its inputs, and the arm trained for the first version is reused unchanged; the inference driver re-checks its input fingerprint against the frozen value before any estimate is formed. On the same rows its absolute NLL is 0.874045 event-micro and 0.972410 market-macro against 0.874138 and 0.973828 for the corresponding arm of Appendix K, i.e. within 0.000093 and 0.001418 — inside the tolerances (0.002, 0.005) fixed in advance from the measured training non-determinism, and **not** a bitwise match.

Results. Paired ΔNLL on identical rows, negative favouring the first arm; nested event-group-by-seed bootstrap, 2,000 replicates, max-t simultaneous 95% intervals with Holm adjustment; macro and

micro are separate families. The materiality margin is the project-wide 0.010 nats. *The rows below come from different inference families and the family size differs by row*: for each weighting, $K=2$ for the two primary contrasts, for the two contrasts outside the parsed groups and for the strict-population contrasts, and $K=1$ for R_shuf − B and for the all-validation contrast. Intervals are therefore not comparable across families, and no cross-row contrast is tested.

population	contrast	market-macro	event-micro	
parsed threshold groups	R − R_shuf	-0.0250	-0.0037	point
(156 groups, 128,168 rows)		$[-0.0377, -0.0122]$	$[-0.0050, -0.0023]$	CI
	R − B	-0.0255	-0.0064	point
		$[-0.0381, -0.0129]$	$[-0.0082, -0.0045]$	CI
	R_shuf − B	-0.0006	-0.0027	point
		$[-0.0068, +0.0057]$	$[-0.0043, -0.0012]$	CI
all validation rows	R − B	-0.0077	-0.0020	point
(535 groups, 454,164 rows)		$[-0.0130, -0.0024]$	$[-0.0033, -0.0006]$	CI
rows outside the parsed groups	R − B	$+0.0017$	-0.0002	point
(379 groups, 325,996 rows)		$[-0.0033, +0.0068]$	$[-0.0019, +0.0015]$	CI
	R − R_shuf	$+0.0001$	$+0.0001$	point
		$[-0.0048, +0.0049]$	$[-0.0018, +0.0020]$	CI

Under market-macro weighting the correct-role arm beats the capacity-matched control by 0.0250 nats, past the margin, while **R_shuf − B is not detected**; that pattern is what one expects if the correspondence between markets and thresholds is what helps, and it is the evidence we claim. It is **not** a decomposition into "role" and "capacity" parts: shuffling also moves the feature marginals, so the shuffled arm differs from R in more than correspondence alone. Under event-micro weighting the arithmetic is $(\mathrm{R}-\mathrm{B}) = (\mathrm{R}-\mathrm{R_shuf}) + (\mathrm{R_shuf}-\mathrm{B}) = -0.0037 + (-0.0027) = -0.0064$: **about two-fifths of the micro improvement is already present in the shuffled-role arm**, which is a reason to read the micro numbers cautiously, not a measurement of how much capacity contributes. Both weightings are reported; neither is designated correct. An interval containing zero is not equivalence, and **we claim no equivalence between R_shuf and B**.

The protocol also requires a stricter population: the groups in which *every* evaluated row with at least one sibling parses under the markets visible at that row's own block. That is 144 of the 156 groups and $104{,}450$ of the $128{,}168$ rows, and on it the comparison is if anything larger — market-macro -0.0266 $[-0.0403, -0.0128]$ against the shuffled-role control and -0.0268 against pooling, with event-micro -0.0037 and -0.0065. Rows whose target has no sibling in the window carry no role features either way and cannot decide whether a group qualifies, so they are excluded from that test; the restriction is recorded with the result. This is a different question from the creation-time sensitivity below and neither substitutes for the other.

Outside the parsed groups all four intervals contain zero — **no change was detected**, which is not a demonstration of no effect. That population is *not* "non-threshold groups": it also contains the threshold groups whose thresholds did not parse.

Availability sensitivity. Repeating the two primary contrasts after excluding any group containing a market whose recorded creation time is later than its first observed trade changes nothing, because no validation threshold-group market is of that kind (0 of the keyed markets); the check therefore has no rows to remove and is **not** independent evidence. It is a check on recorded creation times, not a per-prediction-time audit of the question text.

Budget and provenance. *Four* amendments were recorded before the runs they govern. The first raised the ceiling from 5.4 to 8.0 GPU-hours after a short run measured 11.6 s per thousand updates on the device used; the first version then took 8.50 GPU-hours, 0.50 GPU-hours above that ceiling, which was reported rather than hidden. The second, written before any retraining, fixed the visibility rule, the reuse of B, the unchanged decision rules and a 7.0 GPU-hour ceiling. The third, written before the second retraining and before any of its predictions existed, removed the remaining dependence on later records, kept every decision rule, population and bootstrap parameter unchanged, and fixed in advance that the superseded numbers would be reported alongside the new ones and that a changed verdict would be decided by the new run. Under it R ($10{,}424$ s) and R_shuf ($10{,}439$ s)

were retrained *concurrently on one GPU*, the longer arm running 2.90 hours. The fourth, written after the parsing audit and before the rebuild, fixed the three refusal rules, the expected change for every affected group, the reuse of B and the same reporting commitments. Under it R ($11{,}577$ s) and R_shuf ($11{,}403$ s) were again retrained concurrently on one GPU. The longer arm ran 3.22 hours; their sum, 6.38 GPU-hours, overstates the occupancy, because per-arm seconds overlap and cannot be added. *We report the longer arm's own wall clock and make no claim about total card time: the launcher records the session start but not its end, so occupancy including start-up is not separately attested.* B was not retrained. The fixed-endpoint sidecars of the arms are byte-identical because that file records only architecture, seeds, budget and training-row mask; the arms are distinguished by their input fingerprints and by the role-table digests recorded in the run records.

The superseded versions. All three earlier constructions are reported, and only the current one is read. The first, before any visibility rule, gave market-macro $\Delta\text{NLL} = -0.0230$ $[-0.0359, -0.0100]$ against the shuffled-role control and -0.0236 against pooling, with event-micro -0.0025 and -0.0052. The second, after group type and parse success were restricted to markets visible at the block but while eligibility still consulted the split-wide target set, gave -0.0226 $[-0.0361, -0.0091]$ and -0.0239, with event-micro -0.0030 and -0.0053. The third, which removed that dependence but still used the uncorrected parser, gave -0.0237 $[-0.0353, -0.0120]$ and -0.0243, with event-micro -0.0032 and -0.0057. The current version gives -0.0250 $[-0.0377, -0.0122]$ and -0.0255, with event-micro -0.0037 and -0.0064. *All four agree in verdict*: the macro comparison against the shuffled-role control is material under the 0.010-nat margin and the micro comparisons are detectable but below it. The point estimates move by less than 0.0025 nats across the four, which we report as agreement in sign and verdict and **not** as evidence that any one of them is more accurate. Rebuilt under the visibility rule, the first version's role features differ on 157 of the $128{,}168$ evaluated rows (0.12%) for R and $4{,}106$ (3.20%) for R_shuf, and on 126 and $14{,}881$ training rows; the shuffled arm is the more sensitive because its derangement is a function of the member set.