MOREVA: WHEN DOES ATTENTION HELP MAMBA FORECAST EVENT STREAMS?

Anonymous authors
Paper under double-blind review

ABSTRACT

Event streams often comprise irregularly timed events with categorical types, motivating joint prediction of the next event's time and type. State-space models such as Mamba efficiently summarize event histories in a fixed-size recurrent state, but may lose information about specific distant events. This limitation motivates hybrids that augment recurrent updates with attention-based retrieval. Although such hybrids have shown promise across sequence-modeling tasks such as Large Language Model and Time Series Forecasting, when attention improves temporal event forecasting and whether standard benchmarks contain dependencies that require retrieval beyond a recurrent summary remains unclear. We investigate these questions with Moreva, a shared event-history model that adds parallel causal-attention branches to selected layers of a pretrained Mamba backbone. On a synthetic process with known dependency length, attention lets Moreva recover type information placed hundreds to thousands of events back, which the Mamba-only model and the baselines miss. On EasyTPP benchmark datasets, Moreva achieves the state-of-the-art compared to time-point-process, regular time series and language-model-based baselines. Our code is available at `https://anonymous.4open.science/r/Moreva-7A91`.

1 INTRODUCTION

Time-stamped event streams arise in online advertising, where clicks precede purchases (Xu et al., 2014), and in clinical records, where visits carry diagnoses and treatments (McDermott et al., 2023). Social and financial platforms produce them as well (Farajtabar et al., 2017; Yang et al., 2018; Bacry & Muzy, 2014), as do earthquake catalogs (Ogata, 1988). Across these domains we study the next-event question of predicting when an event occurs and its type from an irregular history, as in temporal point process (TPP) benchmarks (Xue et al., 2024; Kong et al., 2026).

Sequence models commonly receive discrete tokens or numerical samples on a regular grid, so adapting irregular events to either interface requires an explicit representation choice (Ansari et al., 2025; Kong et al., 2026). Two properties matter here. **First**, forecasting is multi-task. The next time is a continuous gap and the next type a discrete category, and the two answers can depend on each other. **Second**, useful information may lie far back in an event history. Recommender systems process lifelong histories containing tens of thousands of actions per user, and longer histories can improve click-through prediction (Pi et al., 2019; 2020; Chang et al., 2023; Zhai et al., 2024). Aftershock rates can also decay as a power law (Ogata, 1988), and market-order signs exhibit long memory (Lillo & Farmer, 2004). These examples motivate studying how forecasting models use distant history and when attention adds value beyond a recurrent summary.

These two properties pull architectures in different directions, and each family has a characteristic weakness. Transformer point processes attend over every past event (Zuo et al., 2020; Rezaei et al., 2026). Attention can retrieve a specific event from any distance, but its cost grows quadratically and its key–value cache grows with every event. State-space point processes, including Mamba-based ones, instead fold the history into a fixed-size recurrent state (Gao et al., 2024; Chang et al., 2025; Gu & Dao, 2023; Lahoti et al., 2026). The state updates in linear time and keeps inference memory constant. Its selective decay can implement recency-weighted summaries analogous to decaying point-process excitation. Under finite precision, however, a fixed-size state cannot preserve every arbitrary past event exactly as the history grows (Jelassi et al., 2024). Hybrid architectures combine these complementary mechanisms. In language modeling, Jamba interleaves Mamba and attention layers,

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.

while Hymba runs state-space and attention heads in parallel, achieving competitive performance with reduced inference memory (Lieber et al., 2024; Dong et al., 2025). In computer vision, MambaVision combines Mamba-based mixers with self-attention blocks to achieve a favorable accuracy–throughput tradeoff in image classification (Hatamizadeh & Kautz, 2025). In regular time-series forecasting, SST combines Mamba and Transformer experts at different temporal scales forecasting (Xu et al., 2025). These results motivate similar designs for event forecasting. Although Mamba–Transformer event models have been evaluated (Gao et al., 2024), when attention contributes predictive information beyond a recurrent summary, and whether its benefits justify its computational cost, remain less understood. This leads to our first question, *when does attention improve a recurrent event forecaster, and what information does it recover?*

The answer depends on a second, less examined question, *how far back does useful information lie in event streams, and can current models reach it?* Standard benchmarks rarely contain such histories. EasyTPP sequences hold at most about a hundred events, and natural streams hide which past event matters, so crediting long-range retrieval requires known lags.

We study both questions with Moreva, a shared event-history model that adds causal attention in parallel with Mamba in a few layers of a pretrained Mamba backbone and predicts the next time and type with separate heads. Swapping the mixers yields Mamba-only and Transformer-only counterparts with the same interface and recipe, which we compare on a synthetic process with known lags and on benchmark event data. Our contributions are as follows.

- We introduce Moreva for joint time and type forecasting. It leads the compared baselines in dataset-averaged next-event time error and type accuracy on EasyTPP, and we also evaluate complete-horizon forecasting.
- We derive an exact decomposition of retrieval's optimal risk reduction into timing and type-information gains.
- We show on a controlled synthetic process that attention recovers distant-event information missed by the Mamba-only model and evaluated point-process baselines.

2 Related Work

Event-process models. RMTPP (Du et al., 2016) and the Neural Hawkes Process (NHP) (Mei & Eisner, 2017) use recurrent histories for time and type prediction. Attention-based models include the Transformer Hawkes Process (THP) (Zuo et al., 2020), Self-Attentive Hawkes Process (SAHP) (Zhang et al., 2020), and AttNHP (Yang et al., 2022). Decomposable Transformer Point Processes (DTPP) separate time and type components (Panos, 2024). Other approaches use direct inter-event distributions (Shchur et al., 2020), or flow matching (Kerrigan et al., 2026). HYPRO reweights proposed trajectories for long-horizon prediction (Xue et al., 2022). EasyTPP provides shared data and evaluation infrastructure (Xue et al., 2024). Moreva shares one recurrent event interface across sources and predicts time and type with direct heads. SurF, our primary comparator, uses a learned time-rescaling transformation and a Transformer encoder for heterogeneous event streams, pooling EasyTPP datasets before per-source fine-tuning (Rezaei et al., 2026).

Heterogeneous and pretrained models. Event Stream GPT (McDermott et al., 2023) handles complex continuous-time events and FlexTPP (Draxler et al., 2025) models mixed attributes. FIM-PP (Berghaus et al., 2026) amortizes point-process inference. Chronos-2 pretrains on numeric series (Ansari et al., 2025), and Qwen3 is a general-purpose language-model family (Yang et al., 2025). Text-aware methods bring in language models through abductive reasoning (LAMP) or textual descriptions of types and intervals (TPP-LLM, Language-TPP) (Shi et al., 2023; Liu & Quan, 2024; Kong et al., 2026). Moreva keeps time and type as separate numerical fields at every recurrent position.

State space backbones. Mamba provides input-selective recurrence with linear sequence scaling, and Mamba-3 extends its recurrence and complex state update (Gu & Dao, 2023; Lahoti et al., 2026). Mamba Hawkes Process (MHP) uses observed inter-event gaps as discretization steps in its Mamba recurrence and trains a point-process likelihood jointly with next-event losses, and MHP-E feeds the Mamba representations through Transformer blocks (Gao et al., 2024). S2P2 models continuous-time state evolution and event-driven jumps with a parallel scan (Chang et al., 2025). In language modeling,

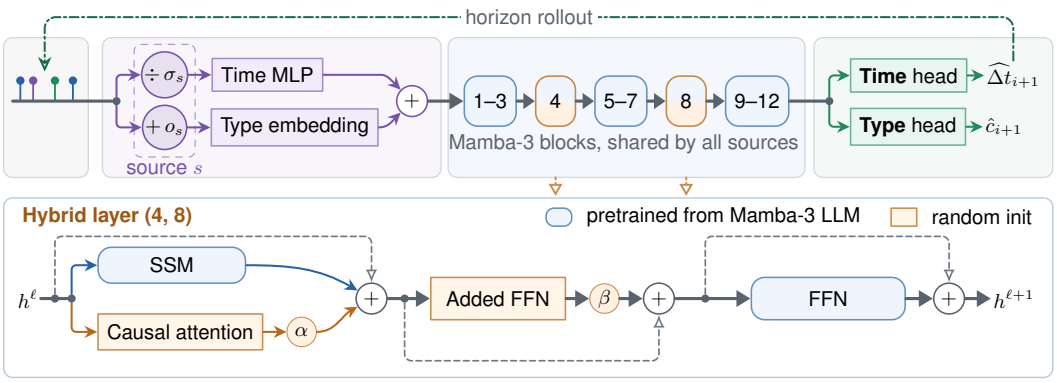

Figure 1: **Unified Moreva architecture.** In the top row, source-aware operations (dashed box) scale gaps by the source's σ_s and offset types by o_s into one shared embedding table (Eqs. 2 and 3). One 12-layer Mamba-3 backbone serves all sources, and separate heads predict the next time and type. Horizon rollout feeds each predicted event back until the first event past H. In the bottom row, layers 4 and 8 add causal attention beside the pretrained SSM and an added FFN before the pretrained FFN (Eq. 10), and dashed arrows are residual paths. Rounded boxes are pretrained from the Mamba-3 large language model (LLM), and square boxes are randomly initialized. Setting $\alpha = \beta = 0$ recovers Mamba-only. MLP denotes a multilayer perceptron, SSM a state-space model and FFN a feed-forward network.

Jamba interleaves attention and Mamba layers, and Hymba runs attention and state-space model (SSM) heads in parallel within a layer (Lieber et al., 2024; Dong et al., 2025). SSMs alone are weaker at copying exact tokens from far back (Jelassi et al., 2024). Moreva adds parallel attention to pretrained Mamba-3 blocks behind a shared event interface (Section 3.4), and our study tests how attention's benefit varies with dependency length and computational cost.

3 Method

Moreva represents an observed event history once and uses it to predict the next time and type separately. Figure 1 shows the architecture. We define the prediction contract and event representation, explain why and how we mix Mamba with attention, and then describe training and the extension to complete-horizon queries.

3.1 Problem Definition and Notation

Event streams. Each source s has K_s types $[K_s] = \{0, \dots, K_s - 1\}$ and a disjoint offset o_s into one shared type table. A length-n sequence records times $t_{s,i}$, stored gaps $\Delta t_{s,i} \geq 0$ and types $c_{s,i} \in [K_s]$, with history $\mathcal{H}_{s,i} = \big((\Delta t_{s,j}, c_{s,j})\big)_{j=1}^{i}$. With σ_s the standard deviation of training gap targets, floored at 10^{-6}, $x_{s,i} = \Delta t_{s,i}/\sigma_s$ is the normalized gap and $\bar{x}_{s,i} = i^{-1} \sum_{j=1}^{i} x_{s,j}$ its causal mean. We omit the source subscript when considering a single source.

Next-event prediction. From $\mathcal{H}_{s,i}$, the model predicts the next gap $\widehat{\Delta t}_{s,i+1}$ and type probabilities $p_{s,i}(k)$, $k \in [K_s]$, with $\hat{c}_{s,i+1} = \arg\max_k p_{s,i}(k)$. Every event after the first is a target, so inputs and targets are shifted by one position:

$$\underbrace{[(\Delta t_{s,1}, c_{s,1}), \dots, (\Delta t_{s,n-1}, c_{s,n-1})]}_{\text{observed inputs}} \longrightarrow \underbrace{[(\Delta t_{s,2}, c_{s,2}), \dots, (\Delta t_{s,n}, c_{s,n})]}_{\text{prediction targets}}. \tag{1}$$

Causal computation restricts output i to the first i inputs, so one batched forward pass supervises every valid transition ($\mathcal{V}$ in a minibatch).

Complete-horizon prediction. A query (t_0, H) asks for every event in $(t_0, t_0 + H]$, possibly none, given the history up to t_0. It has n_k events of type k and count $\sum_k n_k$. Hats mark predictions, so Type MAE, the mean absolute error of type counts, is $\sum_k |\hat{n}_k - n_k|$, and D is the observation duration of the stream. Horizon training gives a prefix q a window $H_q = r_q m_q$ (multiplier r_q, recent

mean gap m_q) with in-window targets I_q and first crossing event j_q^+, and Q is a minibatch of queries (Section 3.6).

Model and training. Features $\phi_{s,i}$ (Eq. 2), type embedding E and time MLP f_t form the input $z_{s,i}$. The backbone B_θ passes a hidden state h^ℓ into each layer ℓ, starting from $h^1 = z_{s,i}$, and gives final states $h_{s,i}$ for heads g_t, g_c. Scalars α_ℓ, β_ℓ scale the added branches of hybrid layer ℓ (Eq. 10). An SSM layer maps inputs u_i to states ξ_i via input-dependent $\bar{A}_i, \bar{B}_i$. Selector J scores a source set $\mathcal{D}$ by development root mean squared error RMSE_s and type accuracy Acc_s against training-fitted $\mathrm{RMSE}_s^0, \mathrm{Acc}_s^0$.

Analysis. For the population analysis, fix one source s with $K_s \geq 2$ and write

$$K = K_s, \qquad \sigma = \sigma_s.$$

Let $(\mathcal{H}, Y, C)$ denote a generic valid history–target triple drawn from this source, where Y is the next gap and C the next type. Let S be a recurrent representation of $\mathcal{H}$ and let A be information computed from the same history by attention. For a representation Z, $\mathcal{R}^*(Z) = \inf_{f,p} \mathbb{E}[(Y - f(Z))^2/\sigma^2 - \log p_C(Z)/\log K]$ is the optimal population risk over measurable regressors f and class-probability maps p, the population form of Eq. 6. Here $I(C; A \mid S)$ is conditional mutual information. For the theoretical point-process model below, t_i denotes the event time and $\Delta t_i = t_i - t_{i-1}$. A K-type exponential-kernel process has intensities $\lambda_k(t) = \mu_k + \sum_{j:t_j<t} \kappa_{k,c_j} e^{-\omega_k(t-t_j)}$ and excitation state $S_i \in \mathbb{R}^K$, $S_{k,i} = \sum_{j \leq i} \kappa_{k,c_j} e^{-\omega_k(t_i - t_j)}$, with excitation columns $\kappa_{:,c}$ and decay vector ω. The exact lag-L task outputs c_{i-L} at every eligible position after the first L events for every K-ary sequence, and b counts the bits of a recurrent model's complete persistent state.

3.2 Source-aware causal event representation

Gaps are normalized by the training-only scale σ_s. The input has six features:

$$\phi_{s,i} = [\log(1 + x_{s,i}),\ x_{s,i},\ \log(1 + \bar{x}_{s,i}),\ \log(1 + i),\ \mathbf{1}[x_{s,i} = 0],\ \mathbf{1}[i = 1]]\,. \tag{2}$$

The current and running gaps separate local timing from the history's typical scale.

Source qualification through the offsets o_s prevents type-identifier collisions in the shared embedding table. The six vocabularies contain 75 types in total, plus one padding entry. A time MLP maps the six features through 128 hidden units to width 768. The event input is

$$z_{s,i} = \mathrm{Dropout}(\mathrm{LayerNorm}[E(o_s + c_{s,i}) + f_t(\phi_{s,i})])\,. \tag{3}$$

3.3 Backbone and Heads

A 12-layer, width-768 single-input single-output (SISO) Mamba-3 backbone maps the event inputs to causal states $h_{s,1:i} = B_\theta(z_{s,1:i})$ (Lahoti et al., 2026). We retain all recurrent blocks, their pretrained FFNs, and the final normalization, replacing the language embedding and output vocabulary with event modules. The final architecture adds attention and FFN branches at layers 4 and 8 (Section 3.4), and without them it is Mamba-only.

Separate 128-hidden-unit MLPs read the same history state. The time head produces a positive normalized gap, while the type head is restricted to the known source vocabulary:

$$\begin{aligned} \widehat{\Delta t}_{s,i+1} &= \sigma_s\, \mathrm{softplus}(g_t(h_{s,i})), \qquad \hat{c}_{s,i+1} = \arg\max_k p_{s,i}(k), \\ p_{s,i}(k) &= \frac{\exp(g_c(h_{s,i})_{o_s+k})}{\sum_{u=0}^{K_s-1} \exp(g_c(h_{s,i})_{o_s+u})}. \end{aligned} \tag{4}$$

3.4 Mixing Mamba and attention

What recurrence and attention model. A selective state-space layer compresses the observed history into a fixed-size state through the input-dependent recurrence $\xi_i = \bar{A}_i \xi_{i-1} + \bar{B}_i u_i$ (Gu & Dao, 2023). Past events therefore influence the current state through products of input-dependent transitions. This structure is well suited to summaries that can be updated recursively. For example, the exponential-kernel excitation state obeys $S_i = \mathrm{diag}(e^{-\omega \Delta t_i}) S_{i-1} + \kappa_{:,c_i}$. This state determines the conditional law of the next gap and type. A selective state-space cell that realizes these transition

and input maps can maintain S_i with state size independent of the history length (Appendix B). MHP explicitly uses the observed gap Δt_i as the discretization interval used to form its recurrent transition (Gao et al., 2024). In Moreva, normalized gap features enter the event embedding, and the retained Mamba-3 layers compute their own data-dependent transitions and discretization intervals from the layer inputs (Lahoti et al., 2026). The Hawkes construction above therefore shows that the selective recurrent form can represent Hawkes excitation. Under finite precision, however, a fixed-size recurrent state has bounded exact-recall capacity. A deterministic recurrent predictor with a b-bit state must satisfy $b \geq L \log_2 K$ to solve the exact lag-L task (Jelassi et al., 2024).

Attention has a complementary profile. Global causal attention compares the current query with all retained past keys. With an appropriate rotary position embedding (RoPE) parameterization, it can place arbitrarily high weight on the event exactly L positions back within a bounded context (Su et al., 2024), under assumptions formalized in Appendix B. This direct access comes at the cost of per-step computation and a key–value cache that grow with the retained history.

Why mix. Event histories support prediction through two complementary dependency structures. Some predictions depend on a compact summary of many past events, such as the decaying excitation of an exponential-kernel process. Others depend on retrieving the identity of a particular distant event. Recurrence naturally maintains the former, and attention gives direct access to the latter. Both the next gap and the next type may depend on either structure. Our hybrid therefore combines Mamba for recurrent summarization with attention for selective retrieval.

The following result formalizes this motivation under the same regression and classification losses used to train Moreva.

Theorem 1 (Retrieval Gain). With S, A, Y, C and $\mathcal{R}^*$ as in Section 3.1, if $\mathbb{E}[Y^2] < \infty$, then

$$\mathcal{R}^*(S) - \mathcal{R}^*(S, A) = \frac{\mathbb{E}\left[\left(\mathbb{E}[Y \mid S, A] - \mathbb{E}[Y \mid S]\right)^2\right]}{\sigma^2} + \frac{I(C; A \mid S)}{\log K}. \tag{5}$$

The proof is provided in Appendix B.1.

Architectural implications. Theorem 1 measures the predictive information that attention adds beyond a recurrent state. Appendix B gives two representative cases. For exponential-kernel event dynamics, the relevant history can be compressed into a finite-dimensional state that follows a gap-dependent recurrence. If Mamba represents this sufficient state exactly, attention cannot further reduce the optimal population risk. By contrast, when prediction depends on an event exactly L positions back, attention can retrieve that event directly from a bounded context, whereas exact recurrent storage under finite precision requires memory that grows with L.

These cases motivate combining the two architectures. Mamba supplies an efficient recurrent summary, and attention recovers information that the summary may lose. The result describes representational capacity under ideal prediction heads, and training determines how much of the potential gain appears in practice. Section 4.2 evaluates this prediction using controlled lagged-dependency experiments.

How we mix. MHP-E composes Mamba and Transformer blocks sequentially (Gao et al., 2024). We use parallel fusion, also explored in Hymba (Dong et al., 2025), retaining every pretrained Mamba block. At layers 4 and 8 of the twelve, the native SSM and causal RoPE attention read the same layer input with separate normalizations. Their residual sum passes through an added SwiGLU FFN (Shazeer, 2020) and then the retained pretrained FFN (Figure 1). Learned scales on the added branches start at 0.01, so training begins near the pretrained Mamba computation, and zero scales recover Mamba-only exactly. The other ten blocks are unchanged (Appendix A), and global attention adds a quadratic sequence-length term.

3.5 Objective and training regimes

Each minibatch contains sequences from one source. For its valid transitions $\mathcal{V}$, the loss is

$$\mathcal{L}_s = \frac{1}{|\mathcal{V}|} \sum_{i \in \mathcal{V}} \left[\left(\frac{\widehat{\Delta t}_{s,i+1} - \Delta t_{s,i+1}}{\sigma_s} \right)^2 - \frac{\log p_{s,i}(c_{s,i+1})}{\log \max(K_s, 2)} \right]. \tag{6}$$

Normalizing the time error and dividing the type cross-entropy by $\log \max(K_s, 2)$ keep both terms on a comparable scale across sources.

Each Joint epoch visits every training sequence once in shuffled source-homogeneous minibatches, and one validation-selected checkpoint serves all sources. Fine-tune (FT) starts from that checkpoint and trains on one source, selecting a checkpoint for each source. All parameters are trainable in both regimes.

Validation-only checkpoint selection. To select one checkpoint for both outputs, we compare development scores with a constant-gap and majority-type reference fitted only on training data and minimize

$$J = \frac{1}{|\mathcal{D}|} \sum_{s \in \mathcal{D}} \left[\frac{\mathrm{RMSE}_s}{\max(\mathrm{RMSE}_s^0, 10^{-6})} + \frac{1 - \mathrm{Acc}_s}{\max(1 - \mathrm{Acc}_s^0, 0.01)} \right]. \tag{7}$$

Here $\mathcal{D}$ spans the benchmark sources for Joint and the adapted source alone for Fine-tune, and one checkpoint supplies both predictions.

3.6 History-scaled multi-window supervision

We evaluate horizon rollout without further training and after supervision on future windows at several scales relative to the observed event rate. Compared with a single fixed multiplier, this exposes a history to many relative durations with the same backbone and heads. For a sampled training prefix q, let m_q be the mean of the positive non-initial gaps among its last 16 observed gap positions, requiring at least two positive observations. The training window has duration

$$H_q = r_q m_q, \qquad r_q \in \{0.25, 0.50, 1, 2, 4, 8, 16, 32, 64, 128\}. \tag{8}$$

Multipliers are cycled over sampled prefixes, one window per exposure, and windows without a recorded event beyond them are skipped (Appendix E).

Window supervision distinguishes the in-window targets I_q from the first crossing event j_q^+. For a minibatch of queries Q, we use

$$\mathcal{L}_{\mathrm{win}} = \frac{1}{|Q|} \sum_{q \in Q} \left[\frac{\sum_{j \in I_q \cup \{j_q^+\}} ((\widehat{\Delta t}_{s,j} - \Delta t_{s,j})/\sigma_s)^2}{|I_q| + 1} - \frac{\sum_{j \in I_q} \log p_{s,j-1}(c_{s,j})}{\max(|I_q|, 1) \log \max(K_s, 2)} \right]. \tag{9}$$

An observed empty window still supplies its crossing-gap target, and training is teacher-forced under causal masks.

At inference, the decoder repeatedly predicts a gap and type, feeds back its own event, and stops at the first predicted event beyond H, so the number of emitted events is the predicted count. Rollouts are capped at 128 events.

4 Experiments

4.1 Setup and aggregation

Mamba-only, Transformer-only and Moreva share the event interface, heads and training recipe, with different mixer and augmentation configurations (Section 3.4), and have 88.72M, 87.68M and 103.46M parameters. Transformer-only replaces each Mamba-3 mixer with parameter-matched causal RoPE attention, and Moreva adds parallel attention and FFN branches. RQ1 tests retrieval of known synthetic dependencies. RQ2 and RQ3 cover next-event and multi-event forecasting on five EasyTPP datasets, and RQ4 weighs retrieval against cost.

The datasets are Taxi, StackOverflow, Taobao, Amazon and Retweet with their released splits (Appendix A). Time RMSE is in each dataset's released unit and type accuracy is the fraction of correct next types. We average the training seeds of each dataset for trained models, then weight datasets equally, while frozen models and deterministic controls are evaluated once.

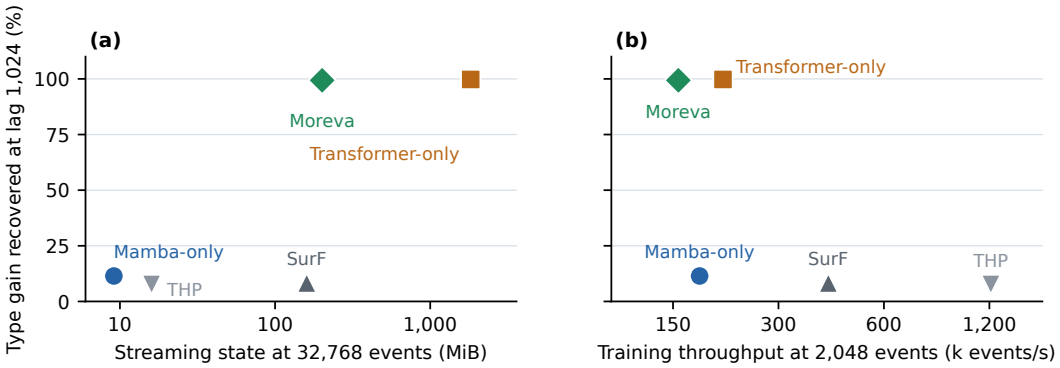

Figure 2: Long-distance retrieval and cost for the three event-interface- matched architectures and, in gray, the SurF and THP baselines at their actual sizes. Vertical position is the fraction of the attainable next-type gain recovered at lag 1,024 on the synthetic Lagged-Dependency Process (three-seed test means). The left panel shows counted per-sequence streaming state at a 32,768-event context in mebibytes (MiB, 2^{20} bytes, log scale, lower is better), and the right panel measured training throughput at 2,048 events (log scale, higher is better). Appendices I and D.1 give full scores and the measurement protocol.

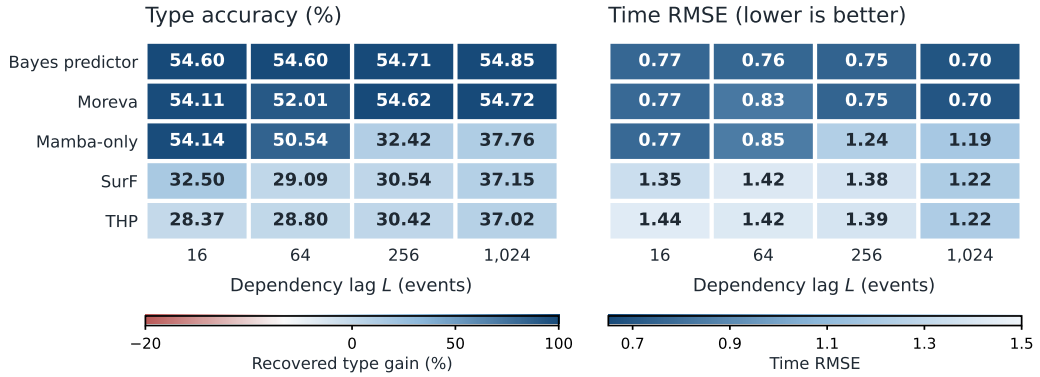

Figure 3: Four-lag Lagged-Dependency test heatmap (1,228,500 targets, with three-seed means for trained rows). Cells show type accuracy and Time RMSE. Bayes uses the population-optimal accuracy and squared-error predictors, evaluated on the same test set. Type colors encode gain relative to the short-history reference and Bayes ceiling. Time colors encode lower error. SurF time is conditional on valid outputs. Appendix Figure 9 adds the Transformer-only, SAHP and short-history reference rows. Appendix I reports coverage and comparison limits.

We adapt the regular-series forecaster Chronos-2 (Ansari et al., 2025) using event index as the sequence axis, with train-normalized gap and one-hot type channels. This keeps irregular timing and simultaneous events without resampling or interpolation. Each query encodes only its observed prefix, and Appendix D gives the Joint training and native-quantile decoder.

4.2 RQ1: Does attention retrieve dependencies that Mamba misses?

Natural benchmarks hide how far back predictive information lies. Our Lagged-Dependency Process uses 20 types and 4,096-event sequences, in which the next type and log-gap depend on the previous type and on the type $L \in \{16, 64, 256, 1024\}$ events earlier (Appendix I). The short-history reference knows only the previous type and position, and the Bayes predictor uses the true conditional. Figure 2 plots recovered type gain at the longest lag against streaming state and training throughput, and Figure 3 shows type accuracy and time RMSE, including the Bayes predictor, at every lag.

The Mamba-only model reaches most of the ceiling up to $L = 64$ but stays near the reference at $L = 256$ and 1,024. Transformer-only approaches the Bayes predictor at every lag (Appendix Figure 9), and Moreva comes within 0.2 accuracy points at $L = 256$ and 1,024, while recovering 90% of the attainable type gain at $L = 64$. Both also approach Bayes time RMSE at the long lags.

Table 1: Five EasyTPP datasets, averaged equally over datasets and over training seeds. Single-event columns give next-event test scores after Joint training and after per-source Fine-tune (FT) or one-source training. Multi-event columns score complete-horizon forecasts on development queries, and Moreva adds multi-scale horizon FT (Section 3.6). Joint models other than FIM-PP and Chronos-2 also trained on Earthquake. Bold marks the best value.

		Single-event				Multi-event	
		Joint		Per-source			
Model	Training	RMSE↓	Type↑	RMSE↓	Type↑	Type MAE↓	W1↓
Moreva	Joint, FT	**3.51**$_{\pm 0.00}$	**58.80**$_{\pm 0.04}$	**3.51**$_{\pm 0.00}$	**58.83**$_{\pm 0.04}$	**3.53**$_{\pm 0.07}$	**19.94**$_{\pm 0.17}$
SurF	Joint, FT	3.54$_{\pm 0.01}$	58.43$_{\pm 0.19}$	3.53$_{\pm 0.00}$	58.51$_{\pm 0.11}$	4.50$_{\pm 0.04}$	20.68$_{\pm 0.07}$
THP	Joint, FT	3.74$_{\pm 0.13}$	41.01$_{\pm 2.30}$	3.62$_{\pm 0.00}$	55.74$_{\pm 0.33}$	4.70$^{*}_{\pm 0.09}$	20.71$^{*}_{\pm 0.11}$
AttNHP	Joint, FT	3.55$_{\pm 0.00}$	56.54$_{\pm 1.22}$	3.54$_{\pm 0.01}$	57.52$_{\pm 0.56}$	4.19$^{*}_{\pm 0.08}$	20.85$^{*}_{\pm 0.50}$
FIM-PP	Joint, FT	3.51$_{\pm 0.00}$	57.34$_{\pm 0.05}$	3.52$_{\pm 0.00}$	57.27$_{\pm 0.02}$	4.22$_{\pm 0.01}$	20.54$_{\pm 0.10}$
Language-TPP	Joint	3.99$^{*}_{\pm 0.11}$	56.22$_{\pm 0.64}$	–	–	–	–
Chronos-2	Joint	3.71$_{\pm 0.00}$	53.89$_{\pm 0.10}$	–	–	19.83$^{*}_{\pm 0.95}$	20.73$^{*}_{\pm 0.11}$
NHP	One source	–	–	3.53$_{\pm 0.00}$	58.49$_{\pm 0.04}$	3.96$_{\pm 0.04}$	20.63$_{\pm 0.16}$
THP	One source	–	–	3.55$_{\pm 0.00}$	58.11$_{\pm 0.09}$	4.84$_{\pm 0.10}$	20.56$_{\pm 0.04}$
SAHP	One source	–	–	3.57$_{\pm 0.02}$	56.16$_{\pm 1.05}$	15.82$_{\pm 6.75}$	21.14$_{\pm 0.06}$
Train mean / majority		–	–	3.97	42.88	5.55	23.20

RMSE is in each dataset's time unit and Type is accuracy in percent. Type MAE is the mean absolute error of per-type counts, and W1 is the first Wasserstein distance of horizon-normalized event times $\times 100$. Subscripts give the sample standard deviation over seeds. * marks valid outputs or completed paths only, unranked.

SurF, THP, and SAHP remain near or below the short-context type reference at the two long lags, and Section 4.5 weighs the cost of attention. At $L = 256, 1024$, Moreva also reduces both normalized MSE and type CE relative to Mamba-only (Appendix Table 8). This agrees with the predictive value of additional history information in Theorem 1, although the experiment compares trained models and the theorem concerns optimal population risks.

4.3 RQ2: Does Moreva outperform baselines on next-event prediction?

In Table 1, Joint pools five sources for FIM-PP and Chronos-2 and also Earthquake for other models. Fine-tune (FT) specializes each Joint model, and one-source models train separately. Joint training asks whether one model serves heterogeneous streams, and per-source training tests whether its advantage survives specialization. We reimplement the Gauss–Legendre quadrature (GLQ) variant of SurF (Rezaei et al., 2026), our closest pooled-training baseline (Appendix C).

Moreva leads every compared method in both regimes. Relative to SurF, it improves macro type accuracy by 0.37 percentage points in Joint and 0.32 in Fine-tune and lowers macro time RMSE by 0.77% and 0.44%. Relative to the values printed by Rezaei et al. (2026) for these datasets, its type accuracy improves by 3.12 and 1.69 points in Joint and Fine-tune. FIM-PP (Berghaus et al., 2026) is retrained for Table 1; frozen in-context inference is appendix-only (Appendix D). Section 4.5 compares the matched arms.

4.4 RQ3: Does Moreva's lead extend to multi-event forecasting?

On 3,258 development queries per horizon, Table 1 scores complete-horizon forecasts produced by feeding back predicted events. Baselines use their checkpoints without horizon training, and Moreva adds multi-scale horizon FT with 128 window updates (Section 3.6). We show $H/D = 10\%$, and Appendix Table 6 gives every duration.

At $H/D = 10\%$, Moreva with multi-scale FT has the lowest Type MAE, 21.7% below SurF and 10.9% below one-source NHP, the next lowest, and the lowest Time-shape W1. Relative to its no-FT parent, which has the highest Type-time F1 (Table 2 and Appendix Figure 6), multi-scale FT trades 1.01 points of Type-time F1 for lower Type MAE and W1 (Table 2). Every ablation in Table 2 lowers Type-time F1, by 0.17 to 1.01 points, and after horizon FT raises W1$\times$100, by 0.08 to 0.41. Type MAE moves by at most 0.19 in either direction.

4.5 RQ4: Does a hybrid structure achieve the best trade-off?

The natural benchmarks barely separate the matched arms. On the five datasets the three arms are within 0.08 points of macro type accuracy and 0.16% of macro time RMSE in both regimes, the

Table 2: Ablations of Moreva at $H/D = 10\%$, with the best value per column in bold. Mamba-only removes the added attention and FFN branches, and Transformer-only uses attention in all twelve mixers. Fixed r=8 replaces multi-scale horizon FT, so its no-FT cells (–) equal Moreva's. F1 is Type-time F1, the harmonic mean of event-matching precision and recall at one history-gap tolerance.

	No horizon FT			Horizon FT		
Variant	Type MAE↓	W1×100↓	F1 (%)↑	Type MAE↓	W1×100↓	F1 (%)↑
Moreva	4.44	**20.45**	**30.96**	3.53	**19.94**	**29.95**
− multi-scale (fixed r=8)	–	–	–	3.54	20.35	28.95
Mamba-only	4.46	20.53	30.79	**3.51**	20.02	29.64
Transformer-only	**4.25**	20.60	30.71	3.58	20.16	29.53

Transformer-only arm last on both. All three exceed SurF in both regimes, so the margin comes from the shared event interface and recipe. Multi-event scores also rank the architectures inconsistently (Table 2).

The arms separate on long-lag retrieval (RQ1) and on cost (Figure 2). Full-prefix throughput is mixed. Transformer-only is faster at 2,048 events, and Mamba-only trains 7% faster at 32,768 (Appendix D.1). The arms differ mainly in the state carried during streaming rollout. Mamba-only keeps 9.1 MiB per sequence at any context, whereas Transformer-only's key–value cache grows to 1.8 gibibytes (GiB) at 32,768 events and Moreva's two attention layers need 0.2 GiB. SurF and THP, far smaller models, train faster and keep smaller key–value caches than the Transformer-only arm, but recover almost none of the lagged information and trail all three arms in next-event type accuracy and multi-event Type MAE. Among the displayed models, only Moreva pairs near-ceiling retrieval at the longest lags with a streaming state far below Transformer-only's, at a cost of 12–14% of Mamba-only's training throughput.

5 Conclusion

We find that a Mamba–attention hybrid is effective for event prediction. Mamba keeps a fixed-size streaming state, and attention in two of twelve layers retrieves distant events. Moreva leads the compared methods on five EasyTPP datasets and, on a synthetic process with known lags, recovers events hundreds to thousands of positions back that Mamba-only misses. Its attention layers cost 12–14% of training throughput and keep a streaming state far smaller than Transformer-only's. These results clarify how Mamba and its hybrids serve temporal point processes. Because standard benchmarks rarely require long-range retrieval, we will seek real event data with long-range dependencies. As such data grow, we also plan cross-architecture distillation from attention-heavy teachers into Mamba-heavy students to keep TPP inference efficient.

Artificial intelligence use statement

Generative artificial intelligence (AI) tools assisted with methodology and experiment-design feedback, software implementation and debugging, quantitative result analysis and interpretation, literature search, and manuscript drafting and editing. The authors reviewed all AI-assisted work and tested the resulting code. They reproduced reported metrics from stored artifacts. They also checked claims and citations against the underlying evidence and primary sources. The authors take responsibility for the final text, claims, code, and other artifacts produced with the aid of generative AI.

Reproducibility statement

The appendix documents event representation, model sizes, released splits, training and checkpoint selection, baseline protocols and seed variability. Every reported number is regenerated from stored per-source, per-seed predictions by the scripts that produce the tables and figures.

References

Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, Mononito Goswami, Shubham Kapoor, Danielle C. Maddix, Pablo Guerron, Tony Hu, Junming Yin, Nick Erickson, Prateek Mutalik Desai, Hao Wang, Huzefa Rangwala, George Karypis, Yuyang Wang, and

Michael Bohlke-Schneider. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv:2510.15821*, 2025. URL https://arxiv.org/abs/2510.15821.

Emmanuel Bacry and Jean-François Muzy. Hawkes model for price and trades high-frequency dynamics. *Quantitative Finance*, 14(7):1147–1166, 2014. URL https://arxiv.org/abs/1301.1135.

David Berghaus, Patrick Seifner, Kostadin Cvejoski, César Ojeda, and Ramsés J. Sánchez. In-context learning of temporal point processes with foundation inference models. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=h9HwUAODFP.

Jianxin Chang, Chenbin Zhang, Zhiyi Fu, Xiaoxue Zang, Lin Guan, Jing Lu, Yiqun Hui, Dewei Leng, Yanan Niu, Yang Song, and Kun Gai. TWIN: TWo-stage interest network for lifelong user behavior modeling in CTR prediction at Kuaishou. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3785–3794, 2023. doi: 10.1145/3580305.3599922.

Yuxin Chang, Alex Boyd, Cao Xiao, Taha Kass-Hout, Parminder Bhatia, Padhraic Smyth, and Andrew Warrington. Deep continuous-time state-space models for marked event sequences. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2412.19634.

Xin Dong, Yonggan Fu, Shizhe Diao, Wonmin Byeon, Zijia Chen, Ameya Mahabaleshwarkar, Shih-Yang Liu, Min-Hung Chen, Yoshi Suhara, Yingyan Celine Lin, et al. Hymba: A hybrid-head architecture for small language models. In *International Conference on Learning Representations*, volume 2025, pp. 97729–97755, 2025.

Felix Draxler, Yang Meng, Kai Nelson, Lukas Laskowski, Yibo Yang, Theofanis Karaletsos, and Stephan Mandt. Transformers for mixed-type event sequences. In *Advances in Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=MtwsRjPZhf.

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. doi: 10.1145/2939672.2939875.

Mehrdad Farajtabar, Yichen Wang, Manuel Gomez-Rodriguez, Shuang Li, Hongyuan Zha, and Le Song. Coevolve: A joint point process model for information diffusion and network evolution. *Journal of Machine Learning Research*, 18(41):1–49, 2017.

Anningzhe Gao, Shan Dai, and Yan Hu. Mamba hawkes process. *arXiv preprint arXiv:2407.05302*, 2024. URL https://arxiv.org/abs/2407.05302.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. URL https://arxiv.org/abs/2312.00752.

Ali Hatamizadeh and Jan Kautz. Mambavision: A hybrid mamba-transformer vision backbone. In *2025 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 25261–25270. IEEE, 2025.

Samy Jelassi, David Brandfonbrener, Sham M. Kakade, and Eran Malach. Repeat after me: Transformers are better than state space models at copying. *arXiv preprint arXiv:2402.01032*, 2024. URL https://arxiv.org/abs/2402.01032.

Gavin Kerrigan, Kai Nelson, and Padhraic Smyth. EventFlow: Forecasting temporal point processes with flow matching. In *Proceedings of The 29th International Conference on Artificial Intelligence and Statistics*, 2026. URL https://arxiv.org/abs/2410.07430.

Quyu Kong, Yixuan Zhang, Yang Liu, Panrong Tong, Enqi Liu, and Feng Zhou. Byte-token enhanced language models for temporal point processes analysis. In *Proceedings of the ACM Web Conference 2026*, 2026. doi: 10.1145/3774904.3792197. URL https://arxiv.org/abs/2502.07139v2.

Aakash Sunil Lahoti, Kevin Li, Berlin Chen, Caitlin Wang, Aviv Bick, Zico Kolter, Tri Dao, and Albert Gu. Mamba-3: Improved sequence modeling using state space principles. In *International Conference on Learning Representations*, volume 2026, pp. 86173–86202, 2026.

Opher Lieber, Barak Lenz, Hofit Bata, Gal Cohen, Jhonathan Osin, Itay Dalmedigos, Erez Safahi, Shaked Meirom, Yonatan Belinkov, Shai Shalev-Shwartz, et al. Jamba: A hybrid transformer-mamba language model. *arXiv preprint arXiv:2403.19887*, 2024. URL https://arxiv.org/abs/2403.19887.

Fabrizio Lillo and J. Doyne Farmer. The long memory of the efficient market. *Studies in Nonlinear Dynamics & Econometrics*, 8(3), 2004. doi: 10.2202/1558-3708.1226.

Zefang Liu and Yinzhu Quan. TPP-LLM: Modeling temporal point processes by efficiently fine-tuning large language models. *arXiv preprint arXiv:2410.02062*, 2024. URL https://arxiv.org/abs/2410.02062.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Matthew B. A. McDermott, Bret Nestor, Peniel Argaw, and Isaac Kohane. Event stream GPT: A data pre-processing and modeling library for generative, pre-trained transformers over continuous-time sequences of complex events. In *Advances in Neural Information Processing Systems*, volume 36, pp. 24322–24334, 2023. URL https://openreview.net/forum?id=hiO0735tmc.

Hongyuan Mei and Jason Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://arxiv.org/abs/1612.09328.

Yosihiko Ogata. Statistical models for earthquake occurrences and residual analysis for point processes. *Journal of the American Statistical Association*, 83(401):9–27, 1988. doi: 10.1080/01621459.1988.10478560.

Aristeidis Panos. Decomposable transformer point processes. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://papers.nips.cc/paper_files/paper/2024/hash/a1f0d96171c8e79019ae35ee439c2938-Abstract-Conference.html.

Qi Pi, Weijie Bian, Guorui Zhou, Xiaoqiang Zhu, and Kun Gai. Practice on long sequential user behavior modeling for click-through rate prediction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2671–2679, 2019. doi: 10.1145/3292500.3330666.

Qi Pi, Guorui Zhou, Yujing Zhang, Zhe Wang, Lejian Ren, Ying Fan, Xiaoqiang Zhu, and Kun Gai. Search-based user interest modeling with lifelong sequential behavior data for click-through rate prediction. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pp. 2685–2692, 2020. doi: 10.1145/3340531.3412744.

Mohammad R. Rezaei, Tejas Balaji, and Rahul G. Krishnan. SurF: A generative model for multivariate irregular time series forecasting. *arXiv preprint arXiv:2605.14069*, 2026. URL https://arxiv.org/abs/2605.14069.

Noam Shazeer. GLU variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020. URL https://arxiv.org/abs/2002.05202.

Oleksandr Shchur, Marin Biloš, and Stephan Günnemann. Intensity-free learning of temporal point processes. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HygOjhEYDH.

Xiaoming Shi, Siqiao Xue, Kangrui Wang, Fan Zhou, James Y. Zhang, Jun Zhou, Chenhao Tan, and Hongyuan Mei. Language models can improve event prediction by few-shot abductive reasoning. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/5e5fd18f863cbe6d8ae392a93fd271c9-Abstract-Conference.html.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Lizhen Xu, Jason A. Duan, and Andrew B. Whinston. Path to purchase: A mutually exciting point process model for online advertising and conversion. *Management Science*, 60(6):1392–1412, 2014. doi: 10.1287/mnsc.2014.1952.

Xiongxiao Xu, Canyu Chen, Yueqing Liang, Baixiang Huang, Guangji Bai, Liang Zhao, and Kai Shu. Sst: Multi-scale hybrid mamba-transformer experts for time series forecasting. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 3655–3665, 2025.

Siqiao Xue, Xiaoming Shi, James Y. Zhang, and Hongyuan Mei. HYPRO: A hybridly normalized probabilistic model for long-horizon prediction of event sequences. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/dfbb3d1807b21dadee735eb75069ada4-Abstract-Conference.html.

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://arxiv.org/abs/2307.08097.

An Yang et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Chenghao Yang, Hongyuan Mei, and Jason Eisner. Transformer embeddings of irregularly spaced events and their participants. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2201.00044.

Guolei Yang, Ying Cai, and Chandan K. Reddy. Recurrent spatio-temporal point process for check-in time prediction. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 2203–2211, 2018. doi: 10.1145/3269206.3272003.

Jiaqi Zhai, Lucy Liao, Xing Liu, Yueming Wang, Rui Li, Xuan Cao, Leon Gao, Zhaojie Gong, Fangda Gu, Michael He, Yinghai Lu, and Yu Shi. Actions speak louder than words: Trillion-parameter sequential transducers for generative recommendations. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 58484–58509, 2024. URL https://proceedings.mlr.press/v235/zhai24a.html.

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. URL https://proceedings.mlr.press/v119/zhang20q.html.

Shuai Zhang, Chuan Zhou, Yang Liu, Peng Zhang, Xixun Lin, and Zhi-Ming Ma. Neural jump-diffusion temporal point processes. In *International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=d1P6GtRzuV.

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. URL https://proceedings.mlr.press/v119/zuo20a.html.

A ARCHITECTURE, DATA, AND SELECTION

Parameter accounting. Moreva has 103,464,208 parameters, with 88,715,532 in the event backbone and heads and 14,748,676 in the added branches at layers 4 and 8. All twelve pretrained layers and their native FFNs are retained. Removing the language vocabulary leaves 88.35M backbone parameters. The event embedding and prediction heads add 0.37M.

Moreva computation. At each hybrid layer $\ell \in \{4, 8\}$, let h^ℓ denote the hidden state entering the layer and the N operators its separate root-mean-square (RMS) normalizations:

$$\begin{aligned} q^\ell &= h^\ell + \mathrm{SSM}_\ell(N_s h^\ell) + \alpha_\ell \,\mathrm{Attn}_\ell(N_a h^\ell), \\ v^\ell &= q^\ell + \beta_\ell F_{\mathrm{add},\ell}(N_f q^\ell), \\ h^{\ell+1} &= v^\ell + F_{\mathrm{pre},\ell}(N_p v^\ell). \end{aligned} \tag{10}$$

Global causal RoPE attention uses 12 heads at width 768 with no attention dropout. The added SwiGLU FFN has hidden width 2,176. All twelve native pretrained FFNs retain width 1,536. Learnable scalars α_ℓ, β_ℓ start at 0.01. New branch sums are accumulated in 32-bit floating point (float32) and cast to the native mixer data type before the native fused add/norm. Disabling both added branches exactly recovers native outputs in graphics processing unit (GPU) checks. The architecture was chosen on three-source development data (Appendix F), and Table 1 reports its test scores.

The union type embedding has 76 entries including padding and width 768. The time encoder is a $6 \to 128 \to 768$ MLP. The time and type heads are $768 \to 128 \to 1$ and $768 \to 128 \to 75$ MLPs. All three use the sigmoid linear unit (SiLU) between their linear layers. Input LayerNorm and dropout 0.10 precede the backbone. Softplus makes the normalized time prediction positive. Its final-layer bias is initialized so the initial prediction is near the equal-source mean of training gap means divided by training gap scales. Source slicing is applied to the type logits before cross-entropy and argmax.

Event interface. Removing the language vocabulary changes the model's interface and keeps its recurrent depth. The retained backbone accounts for 88.35M parameters and the new event modules for 0.37M. Every event occupies one input position regardless of the printed length of its gap or type identifier.

The shared state can encode associations between previous gaps and types, while the separate heads map that state to different output spaces. There is no additional recurrent pass for the type task and no autoregressive text decoding for the time task.

Effect of changing time units. If every gap of a source is rescaled by a positive constant a, its training standard deviation also scales by a. Provided the numerical scale floor is inactive, $(a\Delta t_{s,i})/(a\sigma_s) = x_{s,i}$, so all six features and the normalized time loss are unchanged. Multiplication by σ_s in the output head restores the requested unit.

Type-loss scale. For a uniform type prediction and $K_s \geq 2$, the unscaled cross-entropy is $\log K_s$, so the scaled term equals one for every source. This provides a common reference magnitude across vocabularies.

Released data and target coverage. The next-event comparison uses the pinned releases named in Section 4, with the numerical 16-type Amazon from `easytpp/amazon` and the remaining corpora from the Neural Jump-Diffusion Temporal Point Processes (NJDTPP) archive (Zhang et al., 2024). We retain released splits, type schemas, event order, first gaps and same-time events. Normalization uses training targets only. Every event after the first in each test sequence is evaluated. Padding and sequences without transitions add no targets. Each evaluation covers 210,323 test transitions across the five datasets.

Optimizer and schedule. Training uses AdamW (Loshchilov & Hutter, 2017) with a backbone learning rate of 10^{-5}. The event modules use 3×10^{-4} in Joint and 10^{-4} in Fine-tune. Weight decay is 0.01 and gradient-norm clipping is 1.00. Batch sizes are 64 training sequences and 128 evaluation sequences. Batches are dynamically padded, source-homogeneous, and retain each sequence's available history. Parameters are stored in float32 with bfloat16 (16-bit floating point with an eight-bit exponent) autocast during model computation. Seeds 1–3 are 20260921, 20260922, and 20260923, respectively.

Each stage has at most 20 epochs, at least five, and early-stopping patience four. One epoch of linear warmup precedes cosine decay with a 0.10 relative learning-rate floor. For update index u, warmup length W, and maximum length U, the multiplier is $(u+1)/W$ during warmup and $0.10+0.45[1+\cos(\pi(u-W)/(U-W))]$ afterward. Fine-tuning starts a new optimizer and schedule. It does not resume the Joint optimizer.

One validation selector for both outputs. Equation 7 defines the checkpoint selector. Its constant gap predictor uses the training gap mean and its constant type predictor uses the training majority category. Development evaluation follows each epoch. A strict improvement updates the selected checkpoint. The same checkpoint predicts both outputs. Test targets do not enter checkpoint selection.

B Recurrent summaries and distant retrieval

This section proves Theorem 1, formalizes the architectural roles of recurrence and attention, and specializes the result to the Lagged-Dependency Process. These are population and representability results. They do not claim that training necessarily finds the stated constructions.

B.1 Proof of the main theorem

Proof of Theorem 1. The regression function f and class-probability function p are optimized independently in the definition of $\mathcal{R}^*$. We therefore consider the squared-loss and log-loss terms separately.

Let

$$m_S = \mathbb{E}[Y \mid S], \qquad m_{SA} = \mathbb{E}[Y \mid S, A].$$

The Bayes predictor under squared loss is the conditional mean. Conditional-mean orthogonality gives

$$\mathbb{E}[(Y - m_S)^2] - \mathbb{E}[(Y - m_{SA})^2] = \mathbb{E}[(m_{SA} - m_S)^2]. \tag{11}$$

After division by σ^2, this is the regression term in Equation 5.

Under log loss, the Bayes prediction is the true conditional class distribution. The optimal log-loss risks based on S and (S, A) are, respectively,

$$H(C \mid S) \qquad \text{and} \qquad H(C \mid S, A).$$

Their difference is

$$H(C \mid S) - H(C \mid S, A) = I(C; A \mid S).$$

After division by $\log K$, this is the classification term in Equation 5. Adding the two reductions proves the theorem.

Relation to type accuracy. Cross-entropy and accuracy are different quantities. Let

$$p_k(S, A) = \Pr(C = k \mid S, A), \qquad q_k(S) = \Pr(C = k \mid S).$$

The improvement in Bayes accuracy satisfies

$$0 \le \mathbb{E}[\max_k p_k(S, A)] - \mathbb{E}[\max_k q_k(S)] \le \min\left\{1, \sqrt{\frac{I(C; A \mid S)}{2}}\right\}. \tag{12}$$

The upper bound follows from total variation, Pinsker's inequality, and Jensen's inequality. No positive lower bound follows from mutual information alone, because attention may improve the predicted probabilities without changing their argmax.

B.2 Architectural cases

Proposition 2 (Recurrent sufficiency and distant retrieval)**.** *The following two cases separate the roles of recurrence and attention.*

1. *Consider a well-defined, non-explosive K-type exponential-kernel point process. The conditional law of the next gap and type is determined by a K-dimensional recurrent state. This state obeys a diagonal, gap-dependent recurrence and can be maintained exactly by*

a selective recurrence whose transition and input maps realize the corresponding Hawkes update. Its state size is independent of the history length.

2. *For exact lag-L retrieval on a bounded context, assume that the attention head can produce position-independent query and key components. Then there exists a parameterization of a causal RoPE attention head whose retrieval error can be made arbitrarily small. In contrast, a deterministic recurrent predictor whose complete persistent, input-dependent state contains at most b bits that solves the exact delay task at every position for every sequence over K types must satisfy*

$$b \geq L \log_2 K.$$

The same lower bound applies to exact regression of an injective function of the lagged type.

Proof. For the first case, consider the conditional intensities

$$\lambda_k(t \mid \mathcal{H}_t) = \mu_k + \sum_{j:t_j<t} \kappa_{k,c_j} e^{-\omega_k(t-t_j)}, \qquad \mu_k, \omega_k > 0, \quad \kappa_{k,c} \geq 0. \tag{13}$$

Immediately after event i, define

$$S_{k,i} = \sum_{j \leq i} \kappa_{k,c_j} e^{-\omega_k(t_i - t_j)}.$$

Splitting the most recent event from this sum gives

$$S_i = \mathrm{diag}(e^{-\omega \Delta t_i}) S_{i-1} + \kappa_{:,c_i}. \tag{14}$$

This is a diagonal, gap-dependent recurrence with a type-dependent input.

At elapsed time $\tau > 0$ after event i,

$$\lambda_k(t_i + \tau \mid \mathcal{H}_i) = \mu_k + e^{-\omega_k \tau} S_{k,i}.$$

Writing $\lambda = \sum_k \lambda_k$, the joint density of the next gap and type is

$$\begin{aligned} &p(\Delta t_{i+1} = \tau, c_{i+1} = k \mid \mathcal{H}_i) \\ &\quad = \lambda_k(t_i + \tau \mid \mathcal{H}_i) \exp\left\{ - \int_0^\tau \lambda(t_i + v \mid \mathcal{H}_i)\, dv \right\}. \end{aligned} \tag{15}$$

Every term on the right is a function of S_i. Therefore,

$$(\Delta t_{i+1}, c_{i+1}) \perp \mathcal{H}_i \mid S_i.$$

The state is sufficient for both next-gap regression and next-type classification. The Hawkes state therefore follows a diagonal, gap-dependent recurrence with a type-dependent input. A selective state-space cell has the generic update

$$\xi_i = \bar{A}_i \xi_{i-1} + \bar{B}_i u_i.$$

If its transition and input maps realize

$$\bar{A}_i = \mathrm{diag}(e^{-\omega \Delta t_i}), \qquad \bar{B}_i u_i = \kappa_{:,c_i},$$

then, with $\xi_i = S_i$, the cell maintains the Hawkes sufficient state exactly. Thus the Hawkes update is compatible with a Mamba-style selective recurrence. This representability statement does not claim that an arbitrary parameterization or trained Mamba instance necessarily realizes these maps. Suitable output heads can approximate the corresponding Bayes predictors on a compact state domain.

Let A be any measurable function of the observed history $\mathcal{H}_i$. Since

$$(\Delta t_{i+1}, c_{i+1}) \perp\!\!\!\perp \mathcal{H}_i \mid S_i,$$

if $S = S_i$ is represented exactly, then

$$\mathbb{E}[Y \mid S, A] = \mathbb{E}[Y \mid S], \qquad I(C; A \mid S) = 0.$$

Thus attention computed from the same observed history adds no additional population-level information beyond the sufficient state.

For the second case, consider a query position $i \geq L$ and let $\mathcal{D}_i$ denote the finite set of admissible relative offsets. Assume that the attention head can produce position-independent query and key

components, for example through projection biases or a constant feature channel. Let the RoPE frequency planes be $\theta_1, \ldots, \theta_M$ (Su et al., 2024). Choose these components so that

$$q_m \overline{k_m} = e^{-\mathrm{i}\theta_m L}.$$

At relative offset $d = i - j$, the positional score is

$$a(d) = \sum_{m=1}^{M} \cos(\theta_m (d - L)). \tag{16}$$

The score is maximized at $d = L$. Let $\theta_{\min}$ be the smallest positive RoPE frequency and assume

$$\max_{d \in \mathcal{D}_i} |d - L| < \frac{2\pi}{\theta_{\min}}.$$

Then the maximum is unique on the admissible context. Because $\mathcal{D}_i$ is finite, the score margin

$$\delta = \min_{d \in \mathcal{D}_i,\ d \neq L} \big(a(L) - a(d)\big)$$

is strictly positive.

Let $N_i = |\mathcal{D}_i|$ be the number of admissible key positions. After scaling the scores by $\gamma > 0$, the attention weight on position $i - L$ satisfies

$$w_{i,i-L} \geq \frac{1}{1 + (N_i - 1)e^{-\gamma\delta}}. \tag{17}$$

If $\|v(c)\| \leq V$, then

$$\left\| \sum_j w_{ij} v(c_j) - v(c_{i-L}) \right\| \leq 2V(N_i - 1)e^{-\gamma\delta}. \tag{18}$$

For a fixed bounded context, this error approaches zero as $\gamma \to \infty$. Thus the construction gives arbitrarily accurate retrieval, although finite-scale softmax attention need not retrieve the event exactly.

For the recurrent lower bound, consider a deterministic predictor whose complete persistent, input-dependent recurrent state belongs to a set $\mathcal{S}$ satisfying

$$|\mathcal{S}| \leq 2^b.$$

This state includes any recurrent buffers carried between positions. Fixed model parameters are not counted because they do not depend on the input history.

Fix a common prefix and append each of the K^L possible length-L type suffixes. If two different suffixes lead to the same recurrent state, append the same sequence of L additional types to both histories. Because both the future inputs and the initial recurrent states are identical, deterministic updates produce identical future outputs. However, the next L required lag-L outputs reveal the two different old suffixes, which is a contradiction.

Consequently, all K^L suffixes must produce distinct recurrent states:

$$|\mathcal{S}| \geq K^L.$$

Together with $|\mathcal{S}| \leq 2^b$, this gives

$$2^b \geq K^L, \qquad b \geq L \log_2 K.$$

If the regression target $r(c_{i-L})$ is injective in c_{i-L}, exact regression identifies the lagged type and therefore obeys the same lower bound. □

B.3 Application to the Lagged-Dependency Process

Corollary 3 (Population gains on the synthetic diagnostic)**.** *Let*

$$C = c_t, \qquad U = (c_{t-1}, t), \qquad V = \begin{cases} 0, & t < L, \\ c_{t-L}, & t \geq L. \end{cases}$$

The synthetic process satisfies

$$\Pr(C = k \mid U, V) = \mathrm{softmax}\left(A_{c_{t-1}} + B_V\right)_k$$

and

$$\log \Delta t_t = a_{c_{t-1}} + b_V + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, \nu^2),$$

where $\nu = 0.35$ *in the diagnostic.*

The Bayes predictor of the raw gap is

$$m(U, V) = \mathbb{E}[\Delta t_t \mid U, V] = \exp\left(a_{c_{t-1}} + b_V + \frac{\nu^2}{2}\right). \tag{19}$$

Let $\mathrm{MSE}^*(Z)$ *and* $\mathrm{CE}^*(Z)$ *denote the optimal population squared error and cross-entropy using representation* Z*. Revealing the lagged type gives*

$$\mathrm{MSE}^*(U) - \mathrm{MSE}^*(U, V) = \mathbb{E}\left[e^{2a_{c_{t-1}} + \nu^2} \mathrm{Var}(e^{b_V} \mid U)\right], \tag{20}$$

$$\mathrm{CE}^*(U) - \mathrm{CE}^*(U, V) = I(C; V \mid U). \tag{21}$$

Thus retrieval may benefit time regression, type classification, both tasks, or neither task, depending on which conditional relationship the distant type changes.

These identities concern population MSE and cross-entropy. They are not differences in the reported RMSE and accuracy, and the empirical short-history reference need not equal the population-optimal predictor based on U.

Proof. Because Δt_t is conditionally log-normal,

$$\mathbb{E}[\Delta t_t \mid U, V] = \exp\left(a_{c_{t-1}} + b_V + \frac{\nu^2}{2}\right) = m(U, V).$$

For $t < L$, the position contained in U determines that $V = 0$, so the same expression applies without requiring retrieval of an earlier event.

Applying Theorem 1 with $S = U$ and $A = V$ gives

$$\mathrm{CE}^*(U) - \mathrm{CE}^*(U, V) = I(C; V \mid U).$$

For the regression term,

$$\mathrm{MSE}^*(U) - \mathrm{MSE}^*(U, V) = \mathbb{E}\left[\left(m(U, V) - \mathbb{E}[m(U, V) \mid U]\right)^2\right].$$

Because c_{t-1} is measurable with respect to U,

$$\begin{aligned}\mathrm{Var}(m(U, V) \mid U) &= \mathrm{Var}\left(e^{a_{c_{t-1}} + \nu^2/2} e^{b_V} \mid U\right) \\ &= e^{2a_{c_{t-1}} + \nu^2} \mathrm{Var}(e^{b_V} \mid U).\end{aligned}$$

Taking expectations proves Equation 20.

Relation to the reported diagnostic. These identities concern population MSE and cross-entropy. Figure 3 reports RMSE and accuracy, which are different quantities. Moreover, the short-history reference uses an empirical training distribution and is therefore a finite-sample plug-in predictor. It need not attain the population-optimal risks based on U.

Indexing. Under the shifted prediction contract in Equation 1, the hidden state at position $t - 1$ predicts event t. For $t \geq L$, the key for $V = c_{t-L}$ is therefore at attention offset $L - 1$. For $t < L$, the default value $V = 0$ is determined by the position included in U, so no distant retrieval is required. The process lag remains L, and an exact recurrent solution to the delay task must distinguish K^L possible length-L suffixes.

What the theory establishes. Together, these results distinguish two ways in which an event history may support prediction. When the next-event distribution depends on a compact recurrent summary, a Mamba-style recurrence can maintain the required information without increasing its state size with the history. When prediction instead depends on the identity of a particular distant event, attention

provides direct retrieval, while, under finite precision, exact recurrent storage requires memory that grows with the lag.

This distinction applies to both outputs, since time and type prediction may each depend on either a recurrent summary or a retrieved event. The results concern population risk and representational capacity. In particular, the finite-state lower bound describes worst-case exact retrieval. It does not predict the average error of the trained Mamba model.

C SurF comparison protocol

Reconstruction. No SurF code, checkpoints or configurations are available, so the SurF rows in Table 1 reconstruct the method from the paper.

Reconstructed model. The displayed reconstruction implements SurF with eight fixed Gauss–Legendre nodes. The encoder follows the paper's appendix, with concatenated type and time embeddings, two causal Transformer history layers, three causal type-refinement layers, width 128, four attention heads and dropout 0.10, without the optional conformer. Time inputs are cumulative released gaps divided by the source's maximum training-sequence duration, a causal substitute for the per-sequence maximum in the paper. Training minimizes the conditional next-event negative log-likelihood, conditioned on the first event, plus type cross-entropy with equal weight. Point forecasts use the predictive mean obtained by integrating the learned survival function. The type prediction is a history-only argmax. Type vocabularies occupy the same disjoint source namespaces as in Moreva.

Recipe, selection, and budgets. SurF is trained jointly on the same sources as Moreva's Joint model and then fine-tuned on each source from its selected joint checkpoint, for three seeds and 21 training stages. We use AdamW with learning rate 3×10^{-4} for joint training and 10^{-4} for fine-tuning, weight decay 10^{-5}, batch size 128, cosine warm restarts every ten epochs, at most 50 epochs, at least ten, and patience ten. Checkpoints are selected on development data only, by the equal-source mean of time RMSE divided by the training gap scale, with a penalty for numerical failures. SurF has 4.08M parameters and receives 38,134 updates over all stages and seeds. Its test predictions cover every target transition, with no invalid time or type output.

Selector. The SurF selector uses development time error only, whereas Moreva's selector combines time and type error.

D Baseline protocols and uncertainty

Baseline roster and distinct regimes. Table 1 combines Moreva, TPP, language-model, and frozen-inference results while keeping their training regimes separate. The NHP, THP, and SAHP specialists each train a fresh source-local model. The separate Joint/Fine-tune THP and AttNHP campaign instead pools the same sources as Moreva's Joint model and then adapts each selected checkpoint. The specialist recipe allows 100 epochs, at least ten epochs, and patience ten, with AdamW and learning rate 10^{-3}.

Frozen FIM-PP protocol. FIM-PP's released checkpoint supports up to 22 event types, covering every source-local vocabulary here (Berghaus et al., 2026). For each source, one fixed draw of up to 2,000 train-partition sequences supplies the in-context examples. No parameters are updated and neither development nor test targets enter this context. After each observed history event, the frozen model yields type-specific conditional intensities. Our deterministic local decoder integrates the survival function with 64-node Gauss–Laguerre quadrature for the expected next gap and chooses the type with highest intensity at that predicted time. The single evaluation covers all 210,323 test transitions of the five datasets with finite time outputs and legal type outputs.

Retrained FIM-PP protocol. The Joint, FT FIM-PP row starts from the same released checkpoint and updates all of its weights jointly on the five train partitions, one shared model per seed and three seeds. Each source keeps eight fixed train-partition sequences as in-context examples and its own source-local type vocabulary. Training minimizes the conditional negative log-likelihood with AdamW at learning rate 5×10^{-5} for at most 20 epochs (at least five, patience four), and the

checkpoint is selected on the equal-source development likelihood without test targets. Fine-tune then restores each same-seed Joint selection, reinitializes the optimizer and trains on one source at learning rate 1.5×10^{-5}, keeping the Joint-to-Fine-tune ratio of the THP and AttNHP recipe, with selection on that source's development likelihood. Prediction uses the frozen protocol's decoder.

Joint Chronos-2. Our event-index adaptation represents event i from source s by $[x_{s,i}, \mathrm{onehot}_{K_s}(c_{s,i})]$, where $x_{s,i} = \Delta t_{s,i}/\sigma_s$ uses the stored gap and frozen training-only scale from Section 3.1. One series position denotes one event, so physical intervals may differ between positions. All observed events, including the first-event gap and simultaneous events, are retained without binning, interpolation or truncation. Because the Chronos-2 encoder is bidirectional, every query receives a separate observed prefix and an independent attention group. Only its $K_s + 1$ channels share that group. Even prefixes of the same sequence cannot attend to one another.

We initialize the released 27.93M-parameter Chronos-2 checkpoint and update all weights jointly on the five sources, with no per-source Fine-tune stage. The native quantile loss supervises the next normalized gap and one-hot type. Future values enter only as loss targets, never as input covariates. At the first forecast position, the median gap forecast is projected to nonnegative support and multiplied by σ_s, while the largest type-channel median determines the predicted type. These channel scores are not class probabilities. Training uses AdamW at 10^{-6}, weight decay 0.01, gradient clipping at 1, batches of eight sequences and at most three epochs, with one warmup epoch and early-stopping patience one. Checkpoints minimize the equal-source sum of development RMSE and type error, each divided by its train-mean/majority baseline error, plus a time-validity penalty.

Multi-event inference feeds this decoder's predictions back into the Joint checkpoints without further training, on the same development queries and 128-event cap as the other rollouts. At $H/D = 10\%$, 12 of 9,774 query–seed paths fail and 79 reach the cap. Starred scores use the 9,683 complete paths.

Time-only and frozen baselines. DTPP uses the pinned official time-component fitter, separately on each source and seed, and reports time only, because its type component conditions on the true next time. Frozen Chronos-2, an appendix baseline distinct from the Joint Chronos-2 of Table 1, uses the released 120M weights with event-gap and one-hot type channels and is evaluated once. FIM-PP likewise uses frozen released weights, one fixed train-only context draw per source, and one deterministic evaluation. Unlike Chronos-2, its type prediction uses the intensity at its predicted time. Statistical controls use only the observed prefix or training data.

Scoring. Table 1 scores every finite nonnegative time output and every legal source-local type within the reported seeds, and Table 3 reports strict numerical flags within those cohorts. Invalid Language-TPP types count as errors, and its starred RMSE averages valid outputs.

THP's per-source single-event scores use seeds 20260922 and 20260923 for both metrics on all five datasets (420,646 target–seed pairs). This post-hoc subset excludes seed 20260921 after its Retweet numerical-integration failures were inspected and is not ranked. The original three-seed Time RMSE is 86.78 ± 144.04 and type accuracy is $55.99 \pm 0.50\%$. THP Joint and multi-event results retain all three seeds; the original predictions are preserved. Neither retained THP Fine-tune seed has strict numerical flags.

Table 3: Strict numerical-integration and parser diagnostics on the five datasets (630,969 targets per three-seed model and setting). DTPP has no type prediction.

Model	Setting	Targets	Flagged time	Strict accept (%)	Flagged type
AttNHP	joint	630,969	1,475	99.766	1,475
AttNHP	finetune	630,969	2,677	99.576	2,677
Language-TPP	joint	630,969	6	99.999	9

D.1 Architecture cost and streaming state

Figure 4 compares 88.72M Mamba-only, 103.46M Moreva and 87.68M Transformer-only. All three use the same event interface, 75-type table, heads, twelve blocks and native FFNs. The Transformer-only replaces every Mamba-3 mixer with causal RoPE attention of inner width 1,216 (19 heads of width 64), chosen so that each attention mixer has about the parameter count of the Mamba-3 mixer it replaces (3.74M versus 3.82M), giving 87.68M parameters. Weights are random because cost

depends only on shapes. Sequences have random gaps and types at a fixed length, and every batch holds 32,768 events, so lengths from 128 to 32,768 use 256 to one sequence per batch.

Timing uses one idle NVIDIA RTX PRO 6000 Blackwell GPU under the project lock, float32 parameters with bfloat16 autocast, PyTorch scaled-dot-product attention and the native Mamba-3 kernels, without compilation. Each setting runs three warmup steps and ten timed steps with device synchronization around each. We report the median, and every setting's range is within 1% of its median. A training step comprises the forward pass, time and type losses, backward pass, gradient clipping and an AdamW update. Inference is a no-gradient full-prefix forward pass. All three architectures lose throughput at 32,768 events, where a batch holds a single sequence. Moreva keeps 86–88% of Mamba-only's training throughput at every length from 512 events.

Streaming state is counted from cache sizes. Each Mamba-3 layer's recurrent state comes from its own inference-cache allocator, and each attention layer stores a bfloat16 key and value of its inner width per past event.

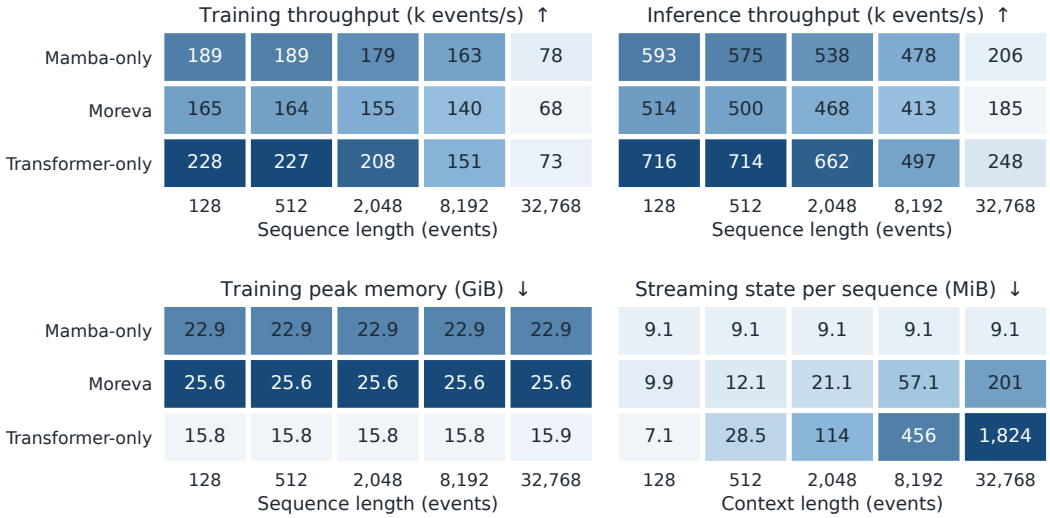

Figure 4: Architecture cost at every length of the synthetic stress test. Panels show throughput and training peak memory at each sequence length and streaming state at each context length. Each panel has its own color scale, logarithmic for streaming state. Darker cells are larger. Median of ten timed steps. Display rounding only.

External baselines in the trade-off table. SurF is the fixed GLQ reconstruction of Table 1 and THP the aligned Joint/Fine-tune THP, each at its actual size (4.08M and 51,500 parameters). Their costs are timed with the same shape rule, in their own training precision (float32) and objectives. Their streaming state counts a bfloat16 key and value per attention layer and past event (five layers of width 256 for SurF, two of width 64 for THP). SurF's multi-event scores are those of Table 1. THP's come from its frozen source-specific Fine-tune checkpoints, rolled out on the same queries with the same decoder cap and scoring. Its point time is the predictive mean and its type the time-marginalized argmax, as in Table 1.

Transformer-only multi-event results. Figure 5 gives the Transformer-only arm's complete-horizon development scores under the protocol of Appendix E: its source-specific Fine-tune checkpoints are rolled out without horizon FT or after 128 fixed-multiplier or multi-scale window updates, with the attention mixers at the added-parameter learning rate. Appendix Figure 6 gives Moreva under the same protocol and queries.

E Multi-event protocol and training-horizon ablation

Physical-time queries. The target for a query (t_0, H) is every event in $(t_0, t_0 + H]$. The predictor receives no future event count. We use all 3,258 released development sequences of the five datasets, with t_0 at the last event at or before each sequence's temporal midpoint and $H \in \{0.05D, 0.10D, 0.20D\}$, where D is that sequence's released duration. Queries, histories and

Table 4: Retrieval, prediction and cost of the matched arms and of SurF and THP under their own architectures. Long-range columns give recovered type gain on the synthetic process. Next-event results are test macros on the five EasyTPP datasets. Multi-event results use full-development queries at $H/D = 10\%$ without horizon FT. Throughput is measured at 2,048 events and streaming state counted at a 32,768-event context. *THP multi-event scores cover completed paths only and are unranked. Bold marks the best eligible entry per column.

	Long range (%)↑		Next-event type (%)↑		Multi-event		Cost	
Model	L=256	L=1,024	Joint	Fine-tune	F1↑	Type MAE↓	k ev./s↑	MiB↓
Mamba-only	10	11	**58.81**	**58.84**	30.79	4.46	179	**9.1**
Transformer-only	**100**	**100**	58.73	58.77	30.71	**4.25**	208	1,824.0
Moreva	**100**	99	58.80	58.83	**30.96**	4.44	155	201.1
SurF	2	8	58.43	58.51	30.86	4.50	416	160.0
THP	2	8	41.01	55.74	28.82*	4.70*	**1,212**	16.0

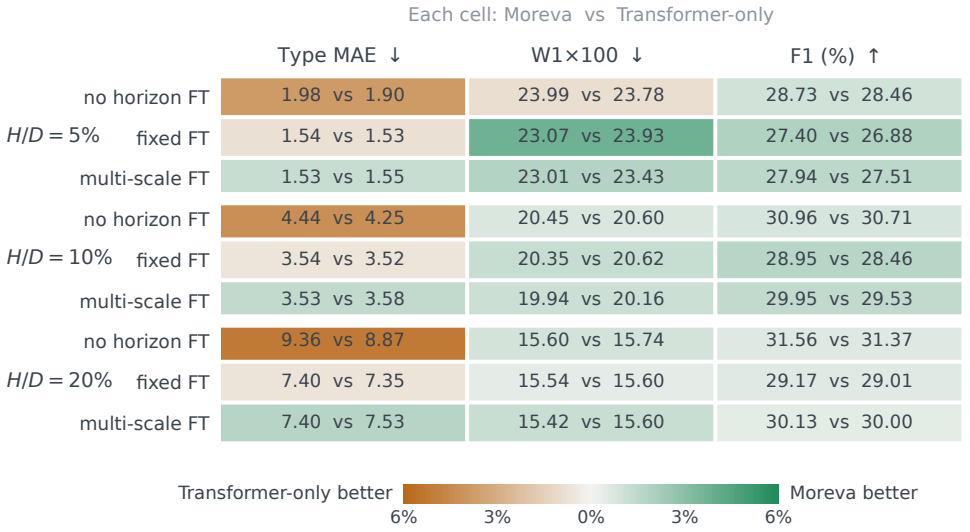

Figure 5: Transformer-only against Moreva on the multi-event development queries of the five EasyTPP datasets (macro over datasets, three seeds), without horizon FT and after fixed-multiplier or multi-scale horizon FT. Each cell gives both scores. Its color is the relative gap, green where Moreva is better and orange where Transformer-only is. W1 is horizon-normalized Time-shape W1×100, and F1 is Type-time F1.

targets match across methods. Histories also match across horizons. Empty target windows are retained. Rollouts that reach the 128-event cap are censored.

Metrics and denominators. Type MAE first sums absolute count errors over types within a query, $\sum_k |\hat{n}_k - n_k|$, then averages queries within each dataset and datasets equally. Time-shape W1 compares unit-mass event-time distributions on the horizon-normalized axis when both sequences are nonempty, with target coverage reported alongside it. Type-time F1 matches same-type events within a tolerance equal to the median of the positive gaps among the last 16 observed non-initial gap positions, requiring at least one positive gap. This tolerance is held constant across horizons. Pool matched, predicted, and true event counts within each dataset, regime, horizon, and seed before computing F1.

Ablating our multi-window supervision. Multi-window supervision defines the optional horizon-FT setting in Section 3.6. The fixed-scale ablation changes only the duration of the supervised future window. Let $\bar{\Delta}_{\text{hist}}$ be the arithmetic mean of the observed positive gaps in the declared recent-history context. The fixed condition uses $H = 8\bar{\Delta}_{\text{hist}}$. The multi-scale condition uses

$$H = r\bar{\Delta}_{\text{hist}}, \qquad r \in \{0.25, 0.50, 1, 2, 4, 8, 16, 32, 64, 128\}. \tag{22}$$

Multipliers are cycled across candidate training queries, and each prefix exposure supplies one horizon in either condition. Both retain observed empty targets. Table 2 lists it at $H/D = 10\%$ and Figure 6 at every duration.

Paired training conditions. Each horizon-FT condition starts from the same source-specific Fine-tune checkpoint and receives 128 updates with a fresh optimizer. The paired conditions share input batches and eligible prefixes. The recent-history mean first takes the last 16 observed non-initial gap positions, then keeps only positive gaps, requiring at least two. A training prefix is eligible only when both its assigned multiplier and the fixed control have a recorded crossing event, which restricts the training population. Time loss covers in-window targets and the crossing gap. Type loss covers in-window targets only.

Coverage and aggregation. Each horizon contains 9,774 query-seed pairs per neural method, all covered by Type-time F1. Scores average seeds within source and then the five datasets equally, without combining durations. Except for THP, AttNHP, one-source SAHP, Chronos-2 and frozen FIM-PP (Table 6), every forecast completes without failure or cap censoring. THP's predictive-mean integrals fail on 72, 101 and 164 pairs at $H/D = 5\%, 10\%, 20\%$, all on Retweet. Its completed-path scores remain unranked. SurF and THP use frozen source-specific Fine-tune checkpoints, whereas the horizon-FT variants receive additional updates. Prefix rate/type frequency uses observed history, and train mean/majority uses training statistics. These deterministic controls each have 3,258 forecasts per duration, all with defined F1.

Time-shape W1 is conditional on both paths being nonempty, with its coverage reported in Figure 6.

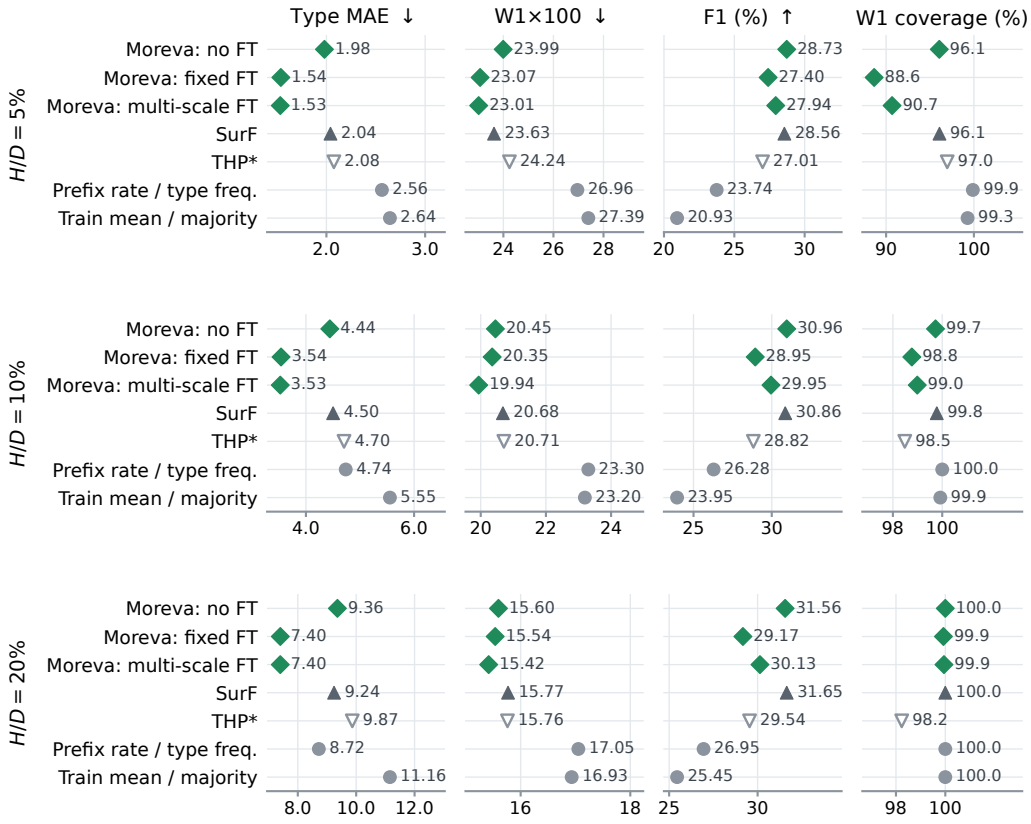

Figure 6: Multi-event forecasting on the full development partitions of the five EasyTPP datasets (Earthquake excluded). Moreva uses source-specific Fine-tune checkpoints, with no updates for no FT and 128 window updates for fixed and multi-scale FT. Neural scores average three seeds within each source, deterministic controls have one evaluation, and the five sources are weighted equally. W1 is horizon-normalized Time-shape W1$\times$100 and F1 is Type-time F1 at one history-gap tolerance. W1 coverage counts nonempty prediction/target pairs over all nonempty targets. *THP fails numerical integration on 72, 101 and 164 of 9,774 query-seed pairs at $H/D = 5\%, 10\%, 20\%$, all on Retweet. Its scores cover completed paths only and are drawn hollow. All other paths complete without failures or cap censoring, and their F1 covers every query-seed pair. D is the released sequence duration, and H sets the rollout boundary.

Table 5: Seed variability of the multi-event results in Tables 1 and 2 at $H/D = 10\%$. Each entry is the sample standard deviation across three seed-level, equal-weight macros over the five EasyTPP-5 datasets (Earthquake excluded). W1 is multiplied by 100 and F1 is in percentage points.

Method	Type MAE SD	W1×100 SD	F1 SD (pp)
Moreva: no FT	0.098	0.065	0.297
Moreva: fixed FT	0.043	0.060	0.553
Moreva: multi-scale FT	0.072	0.171	0.882
SurF	0.040	0.065	0.294
THP*	0.090	0.114	0.368
Prefix rate / type freq.	–	–	–
Train mean / majority	–	–	–
Transformer-only: no FT	0.078	0.019	0.257
Transformer-only: multi-scale FT	0.202	0.300	0.843

Added point-process rollouts. Table 6 extends the comparison to AttNHP, rolled out from its frozen source-specific Fine-tune checkpoints, and to the one-source NHP, THP and SAHP specialists of Table 1, each with its source-local type vocabulary. All use THP's decoder, whose point time is the predictive mean and whose type is the time-marginalized argmax. None receives horizon training. AttNHP's predictive-mean integrals fail on 151, 175 and 223 pairs at $H/D = 5\%, 10\%, 20\%$, on Taxi and Amazon. As for THP, its completed-path scores are unranked. Retrained FIM-PP rolls out its source-specific Fine-tune checkpoints, as in Table 1, and its Joint checkpoints, shown here only, with its own decoder (Appendix D). Both complete every path. Frozen Chronos-2 is one deterministic evaluation of the 3,258 queries per duration. Its median gap forecast is projected to nonnegative support, and a zero first gap is an invalid event time, which fails 5 queries on two sources. It overpredicts event counts, and its rollouts reach the 128-event cap on 23, 67 and 146 queries, mostly on Taobao. Its completed-path scores are unranked. Frozen FIM-PP is likewise one evaluation, with the released weights, the fixed train-only context and the decoder of its next-event protocol. No path fails, but it also overpredicts event counts and reaches the cap on 48, 112 and 232 queries, mostly on Amazon, so its completed-path scores are unranked. One-source NHP has lower Type MAE than every model without horizon FT at each duration, and its Type-time F1 is within 0.1 points of the highest at $H/D = 10\%$ and 20%. One-source SAHP overpredicts event counts, so its Type MAE is several times larger and its rollouts reach the 128-event cap on 301 query-seed pairs at $H/D = 20\%$.

Table 6: Multi-event development macros of the baseline rollouts in Table 1 at every duration on the five EasyTPP datasets (Earthquake excluded), with three seeds, equal source weights and the queries, 128-event cap and scoring of Figure 6. Rows roll out source-specific Fine-tune checkpoints, Joint checkpoints for FIM-PP Joint and Chronos-2 Joint, or source-local specialists, all without horizon training. Frozen Chronos-2 and FIM-PP are evaluated once. W1 is Time-shape W1×100 and F1 is Type-time F1 in percent. The last column of each duration counts failed and capped query–seed pairs out of 9,774, or queries out of 3,258 for the frozen rows.

	$H/D = 5\%$				$H/D = 10\%$				$H/D = 20\%$			
Rollout	Type MAE	W1	F1	Fail/cens.	Type MAE	W1	F1	Fail/cens.	Type MAE	W1	F1	Fail/cens.
AttNHP Fine-tune	1.87	23.19	27.93	151/151	4.19	20.85	30.19	175/175	8.81	15.99	30.73	223/223
NHP one source	1.76	23.96	28.07	0/0	3.96	20.63	30.92	0/0	8.30	15.66	31.56	0/0
THP one source	2.14	23.86	27.81	0/0	4.84	20.56	29.77	0/0	10.27	15.69	30.18	0/0
SAHP one source	7.42	24.08	20.33	0/0	15.82	21.14	23.05	0/0	31.81	16.01	23.92	0/301
FIM-PP Fine-tune	1.87	24.41	27.92	0/0	4.22	20.54	30.18	0/0	9.00	15.71	30.91	0/0
FIM-PP Joint	1.87	24.21	28.13	0/0	4.21	20.51	30.42	0/0	8.94	15.74	31.08	0/0
Chronos-2 Joint	10.81	21.74	21.10	12/18	19.83	20.73	24.33	12/91	29.24	16.33	26.34	12/515
Chronos-2 frozen	11.81	22.46	19.39	5/28	19.50	20.62	22.81	5/72	26.88	15.79	24.21	5/151
FIM-PP frozen	5.19	24.32	16.42	0/48	12.06	20.76	18.20	0/112	28.40	16.55	16.97	0/232

Training-scale effect and comparison scope. For Moreva, multi-scale supervision improves macro Type MAE over fixed $r = 8$ by 0.39%, 0.41%, and 0.01%, and F1 by 0.54, 1.01, and 0.96 percentage points across the three durations.

F HYBRID ARCHITECTURE ABLATION

On the same frozen small-data subsets of Taxi, Taobao and Retweet, we compare Mamba-only (88.72M), Mamba + attention (93.44M, parallel causal attention only) and Moreva (103.46M, parallel attention and an additional FFN). All arms retain all twelve native pretrained FFNs. The latter two add attention at layers 4 and 8. Only the final arm adds the extra FFNs shown in Figure 1.

Each arm receives 600 next-event updates and then 128 identical multi-scale window updates. The second stage starts from each arm's selected first-stage checkpoint (update 600 in every run) with a fresh optimizer. Its terminal checkpoint is evaluated without a horizon-based selector. The per-source caps are 1,024 training and 256 development sequences. Horizon evaluation uses 16 fixed development sequences per source at each of $H/D = 5\%, 10\%, 20\%$. All 864 generated paths complete without failure or censoring, and Type-time F1 covers every query. Source/seed metrics are averaged equally.

The next-event macro type accuracies are 69.98%, 70.02%, and 70.02%. All time RMSEs round to 5.49. Thus the added components yield near parity, not a substantial next-event gain. At $H/D = 20\%$, Moreva reduces Type MAE from 6.50 to 6.43 (1.12%) and improves Type-time F1 from 25.80% to 27.04% (1.24 percentage points).

The Mamba-only, Mamba + attention and Moreva runs use 1.00, 1.16 and 1.17 times the measured optimizer time, respectively, across the two stages. The full three-seed evaluation on the five EasyTPP datasets confirms the near parity, with Moreva and Mamba-only within 0.02 points of macro type accuracy in either regime.

G Fraction of augmented layers

We vary the fraction of the twelve retained pretrained Mamba layers receiving an added causal-attention branch. The four nested settings add branches at no layers, layer 8, layers 4 and 8, or layers 2, 4, 8 and 10, corresponding to $0/12$, $1/12$, $2/12$ and $4/12$. Each augmented layer also gets the same additional FFN as Moreva, while all twelve native pretrained FFNs remain in every setting. Thus this sweep varies the number and placement of complete augmentation sites. The $2/12$ setting is the selected 103.46M architecture, and the $0/12$ setting is 88.72M Mamba-only.

The quality experiment reuses the frozen Taxi, Taobao and Retweet small-data train/development subsets, train-only normalization and seeds of Appendix F. Every setting receives the same 600 next-event updates and 128 paired multi-window updates. The latter start from each setting's next-event development-selected checkpoint and use a fixed terminal checkpoint, without horizon-based selection. Next-event time and type are scored on identical development targets. Autonomous Type-time F1 and Type MAE use the same physical-time horizon queries at $H/D = 5\%, 10\%, 20\%$. Figure 7 shows mean paired differences from Mamba-only ($0/12$).

The efficiency experiment uses the same twelve-block event interface but randomly initialized weights, because it is a shape-controlled synthetic stress test. On one idle GPU, each setting uses the same 32,768 events per batch, three warmup steps and ten synchronized timed steps. Training timing includes forward and backward passes, time/type losses, clipping and an AdamW update. Inference is a no-gradient full-prefix forward pass. Streaming state counts the native Mamba recurrent caches and one bfloat16 key/value cache per added attention layer. Figure 8 reports throughput relative to $0/12$ and counted per-streaming state.

H Event streams with text

This benchmark pools Retweet, StackOverflow, Taobao and Taxi with their released splits and the 24-type Amazon release, whose events carry review summaries. Amazon gaps are raw days, split by date so that test targets follow all training targets. Each query keeps its full observed history. Only targets of its split are scored. Moreva and locally retrained Language-TPP start from language-pretrained weights, train jointly on the five sources with three seeds, and select checkpoints on development data. Time and type are predicted from the same observed histories and scored on identical targets.

A frozen 0.5B-parameter language-model encoder embeds each observed event's source, type label and summary. A trainable projection maps this embedding into Moreva's event input. With that projection, Moreva has 103.68M trainable parameters. Its added attention and FFN parameters use a learning rate of 10^{-4}. Language-TPP (Kong et al., 2026) is retrained locally from its released model and tokenizer because its training and evaluation code is not released. It serializes events with byte-token time intervals and uses greedy decoding without repair.

Table 7 reports three-seed means and sample standard deviations. Moreva has higher mean type accuracy and lower mean time RMSE on every source, with type accuracy higher by 5.27 points

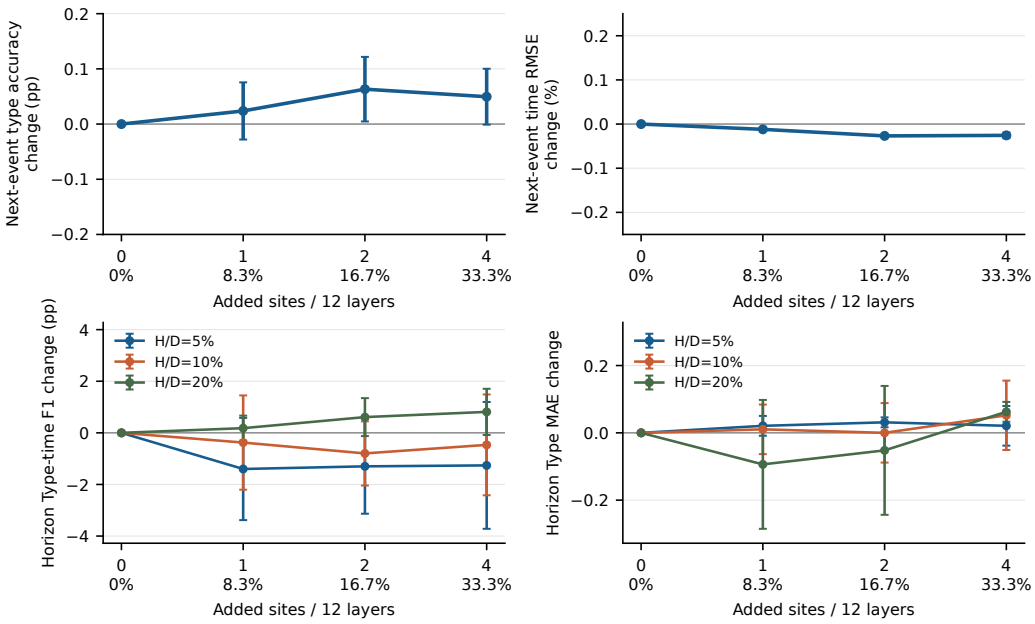

Figure 7: Development-only quality against Mamba-only $(0/12)$. In the top row, blue traces are seed means. Time RMSE is the mean within-source relative change, avoiding an average over unlike time units. In the bottom row, colored points are seed means at each physical-time horizon. Whiskers show sample standard deviations. Positive type accuracy and Type-time F1 changes are better. Negative Time RMSE and Type MAE changes are better.

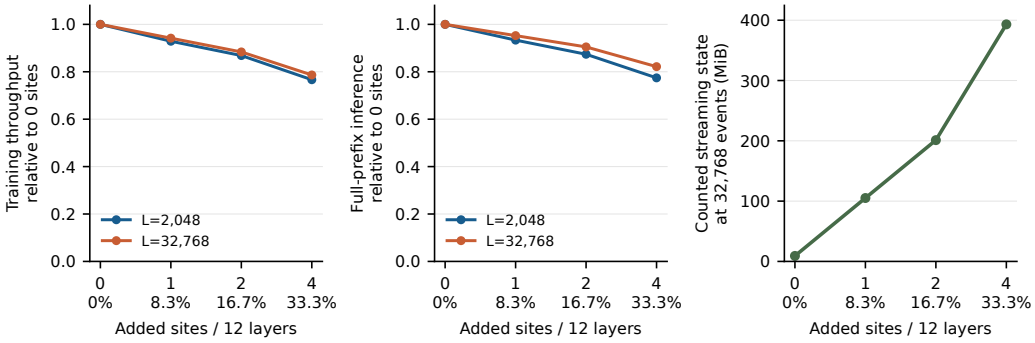

Figure 8: Synthetic cost of the four augmentation fractions. Each throughput point is the median of ten synchronized steps on the same GPU, divided by the $0/12$ throughput at the same length and mode. The right panel counts streaming state at a 32,768-event context.

on Amazon. Time variability is large for Language-TPP on Taobao. Its invalid type outputs, 24 on Amazon and 78 on Taobao, count as errors.

Table 7: Five-source text benchmark, test split, three-seed mean ± sample SD. Time RMSE uses each source's released unit, which is days for Amazon. Bold marks the best listed model per source and metric.

	Time RMSE↓		Type accuracy (%)↑	
Source	Moreva	Language-TPP	Moreva	Language-TPP
Retweet	**15.76 ± 0.01**	17.77 ± 0.75	**59.92 ± 0.04**	58.70 ± 0.75
StackOverflow	**1.01 ± 0.00**	1.16 ± 0.08	**45.43 ± 0.11**	40.91 ± 6.33
Taobao	**0.17 ± 0.00**	0.33 ± 0.33	**61.25 ± 0.08**	55.99 ± 2.23
Taxi	**0.28 ± 0.00**	0.31 ± 0.01	**91.71 ± 0.01**	91.05 ± 0.44
Amazon (text)	**162.61 ± 0.63**	175.76 ± 1.19	**35.25 ± 0.29**	29.98 ± 3.35

I DEPENDENCY-LENGTH DIAGNOSTICS

The Lagged-Dependency Process places predictive information at a known lag and compares the matched architectures with the fixed SurF reconstruction and two likelihood-based TPP baselines. Figure 9 reports their test scores.

Lagged-Dependency Process. With $K = 20$ types, the type and log-gap at position t follow

$$P(k_t \mid k_{t-1}, k_{t-L}) \propto \exp\big(A_{k_{t-1}} + B_{k_{t-L}}\big), \qquad \log \Delta t_t \sim \mathcal{N}\big(a_{k_{t-1}} + b_{k_{t-L}}, 0.35^2\big), \tag{23}$$

where $A, B \in \mathbb{R}^{K\times K}$ and $a, b \in \mathbb{R}^K$ are fixed random matrices shared by all lags (entries $\mathcal{N}(0, 1.6^2)$, $\mathcal{N}(0, 2.2^2)$, $\mathcal{N}(0, 0.3^2)$, $\mathcal{N}(0, 0.8^2)$). For $t < L$ the long context is the fixed default type 0, so every target is predictable from the emitted sequence. Each lag has 1,000 / 100 / 300 training / development / test sequences of 4,096 events, generated once with a fixed seed. The Bayes ceiling uses the true conditional. The reference predictor knows k_{t-1} and the position, hence the default context for $t < L$, and marginalizes k_{t-L} with its empirical training distribution otherwise. It is the value a model with no memory beyond one event can reach. Information recovery is $(\text{model} - \text{reference})/(\text{ceiling} - \text{reference})$. Figure 9 shows the underlying scores and recovered type gain. Mamba-only uses three 4,096-event sequences per update (12,288 events), at most 20 epochs, and three seeds. Both configurations use backbone learning rate 3×10^{-4} at every lag.

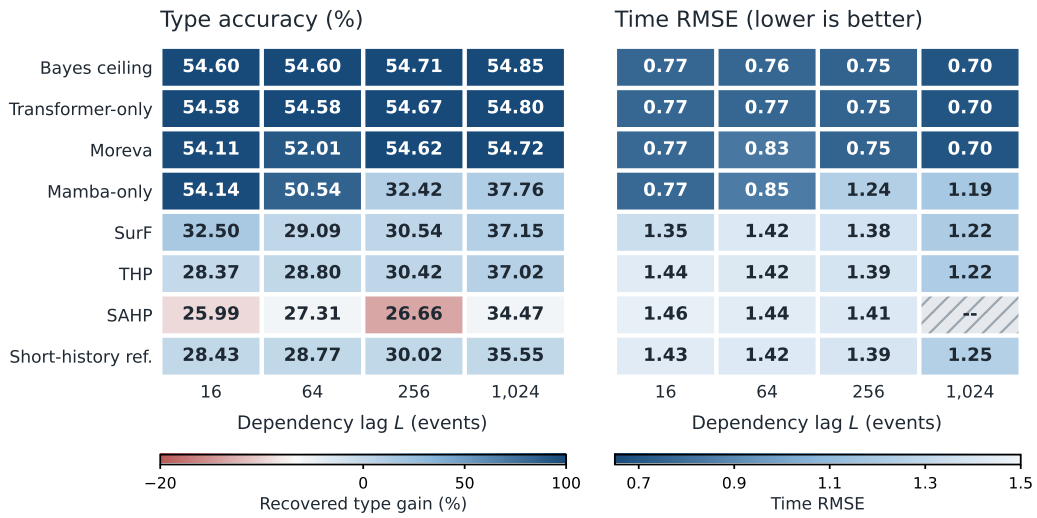

Figure 9: All rows of the Lagged-Dependency test heatmap (three-seed means over 1,228,500 targets per seed). Figure 3 shows the Bayes predictor, Moreva, Mamba-only, SurF and THP rows. Colors are as in Figure 3. SurF time is conditional on valid outputs, and SAHP has no all-target time score at $L = 1024$ (hatched).

Theorem-aligned empirical risks. Table 8 supplements RMSE and accuracy, which remain the primary interpretable metrics. On the same frozen targets, we compute $\mathrm{NMSE} = N^{-1}\sum_i((\widehat{\Delta t}_i - \Delta t_i)/\sigma_L)^2$ and $\mathrm{CE} = -N^{-1}\sum_i \log p_i(k_i)$, using natural logarithms, all $K = 20$ types, and the model's training-only gap standard deviation $\sigma_L = \max(\mathrm{Std}_{\mathrm{train}}(\Delta t), 10^{-6})$. NMSE and $\mathrm{CE}/\log K$ directly match the two components of the paper loss. Their sum is the empirical counterpart of the normalized objective in Theorem 1. These finite-sample evaluations of trained models and deterministic predictors are not exact population Bayes risks or an empirical proof of the population theorem. Lower joint risk indicates better performance under this theorem-aligned objective.

Moreva has 103.46M parameters with the same event modules and recipe. With the 20-type vocabulary it has 103.41M parameters, and its added attention and FFN parameters use their own learning rate of 10^{-4}.

SurF uses the paper-equation reconstruction of Table 1 with its own loss, predictive-mean decoder and optimizer, on the same full 4,096-event sequences, three sequences per update and three seeds. As in the SurF paper, it has no positional encoding, and order enters through the causal mask and a cumulative-time embedding. It trains for 20 epochs and selects on conditional time RMSE plus type

Table 8: Theorem-aligned empirical LDP risks ($K = 20$, lower is better). Every row uses all 1,228,500 frozen test targets. Trained models report mean ± sample standard deviation over three seeds, and the reference and Bayes predictor are deterministic. Output coverage is 100% for every row.

Model	Loss-normalized MSE ↓	Type CE (nats) ↓	CE / $\log K$ ↓	Joint normalized risk ↓
		Lag $L = 16$		
Short-history reference	0.82	2.24	0.75	1.57
Bayes predictor	0.24	1.38	0.46	0.70
Mamba-only	0.24 ± 0.00	1.40 ± 0.00	0.47 ± 0.00	0.71 ± 0.00
Transformer-only	0.24 ± 0.00	1.38 ± 0.00	0.46 ± 0.00	0.70 ± 0.00
Moreva	0.24 ± 0.00	1.40 ± 0.01	0.47 ± 0.00	0.71 ± 0.00
		Lag $L = 64$		
Short-history reference	0.81	2.23	0.74	1.56
Bayes predictor	0.24	1.38	0.46	0.69
Mamba-only	0.29 ± 0.01	1.54 ± 0.06	0.51 ± 0.02	0.80 ± 0.02
Transformer-only	0.24 ± 0.00	1.38 ± 0.00	0.46 ± 0.00	0.70 ± 0.00
Moreva	0.28 ± 0.04	1.48 ± 0.09	0.49 ± 0.03	0.77 ± 0.07
		Lag $L = 256$		
Short-history reference	0.80	2.19	0.73	1.53
Bayes predictor	0.23	1.37	0.46	0.69
Mamba-only	0.64 ± 0.03	2.12 ± 0.01	0.71 ± 0.00	1.34 ± 0.03
Transformer-only	0.24 ± 0.00	1.37 ± 0.00	0.46 ± 0.00	0.69 ± 0.00
Moreva	0.24 ± 0.00	1.38 ± 0.00	0.46 ± 0.00	0.70 ± 0.00
		Lag $L = 1024$		
Short-history reference	0.75	1.98	0.66	1.41
Bayes predictor	0.23	1.35	0.45	0.68
Mamba-only	0.69 ± 0.00	1.92 ± 0.00	0.64 ± 0.00	1.33 ± 0.00
Transformer-only	0.24 ± 0.00	1.35 ± 0.00	0.45 ± 0.00	0.69 ± 0.00
Moreva	0.24 ± 0.00	1.35 ± 0.00	0.45 ± 0.00	0.69 ± 0.00

accuracy, with every target kept in the type denominator. Invalid type outputs count as errors. SurF returns a valid time for at least 99.97% of targets in every run on this process. Both Mamba-only and Moreva have higher type accuracy on all targets and lower time RMSE on the matched SurF-valid targets at every lag and seed.

Likelihood-based TPP baselines. We additionally train THP (Zuo et al., 2020) and SAHP (Zhang et al., 2020) on the same frozen full-length sequences, three seeds per lag. These pinned EasyTPP models have 64-dimensional hidden states and their respective likelihood objectives, using three sequences per update and at most 20 epochs. Checkpoints are selected on development time and type scores before testing. THP has 44,240 parameters and SAHP has 31,624, far fewer than Mamba-only or Moreva. Both are scored on the same 1,228,500 test targets per lag and seed, with target identities matching Mamba-only. At $L = 256$ and $L = 1024$, THP recovers at most 8% of the attainable type gain. SAHP's mean type accuracy is below the short-context reference at both long lags. At $L = 1024$, it has 554 invalid time/type outputs across the three seeds, so no all-target Time RMSE is reported.