FORECASTING TIME SERIES RESPONSES TO COUNTERFACTUAL EXOGENOUS EVENT DESCRIPTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Real-world time series respond to external events. Prior work shows that exogenous-event text can improve factual forecasts, while counterfactual modeling has been studied for structured interventions and target-derived or target-descriptive text. Yet whether a model produces the appropriate numerical response when the same history is paired with a different free-form exogenous event remains unclear. We propose `MEETS`, a conditional diffusion model with factorized history – event conditioning and matched conditioning-swap objectives that encourage use of both sources. Across traffic, energy, and treatment-response domains, `MEETS` lowers factual forecasting error and improves counterfactual event-specific responses, including direction and magnitude, while generalizing to unseen event configurations.[1]

1 INTRODUCTION

Many time series depend on external events: traffic volume responds to weather (Datla & Sharma, 2008), and tumor growth to chemotherapy and radiotherapy (Geng et al., 2017). For a lung-cancer patient, a clinician choosing among chemotherapy next week, radiotherapy in two weeks, or both tomorrow needs the tumor trajectory each choice would produce from the same history (Figure 1). We therefore ask: given a system's observed state and a specified exogenous event, what trajectories could follow?

Most methods that pair time series with text use the text as context about the series (Williams et al., 2025; Liu et al., 2024; Jia et al., 2024; Zhong et al., 2025; Su et al., 2025; Zhang et al., 2025a; Wu et al., 2026b; Gu et al., 2026), or as a specification of the desired output (Woo et al., 2025; Ge et al., 2025; Qiu et al., 2026; Narasimhan et al., 2024). Methods that condition forecasts on external events (Xu et al., 2025; Ge et al., 2026; Zhang et al., 2025b; Wang et al., 2026; 2024) are evaluated mainly on observed history–event pairs or do not score same-history event swaps against a numerical reference. TADiff (Gu et al., 2026) comes closest, forecasting the same history under swapped textual conditions, but its real-data alternate conditions are largely target-derived, and unobserved alternates are scored semantically rather than numerically. Counterfactual treatment-response methods compare alternate interventions directly (Lim, 2018; Bica et al., 2020; Li et al., 2021; Melnychuk et al., 2022; Ma et al., 2024; Li et al., 2026), but only over structured spaces of categorical treatments, doses, or treatment sequences. Benchmarks obtain exact alternate outcomes from simulators, and evaluation on observational data remains difficult (Mu et al., 2026; Saito & Yasui, 2020; Curth & Van Der Schaar, 2023; Gentzel et al., 2019). Existing work therefore does not establish whether a forecaster gives the right numerical response when the same history is paired with a different free-form event.

We study *text-conditioned time-series modeling under exogenous events*: modeling the distribution of future trajectories given an observed history and a natural-language event description. Because the event is an input rather than inferred from the history, we can hold the history fixed, vary the event, and ask whether the forecast changes appropriately. This raises two challenges. First, low error on observed history–event pairs does not by itself constrain predictions for recombined pairs (Shalit et al., 2017). Second, on real data the future under an alternate event is unobserved, so evaluation needs a reference for that missing trajectory.

[1] All code and data needed to reproduce all findings will be made public. It is currently available as a zip in supplementary materials

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

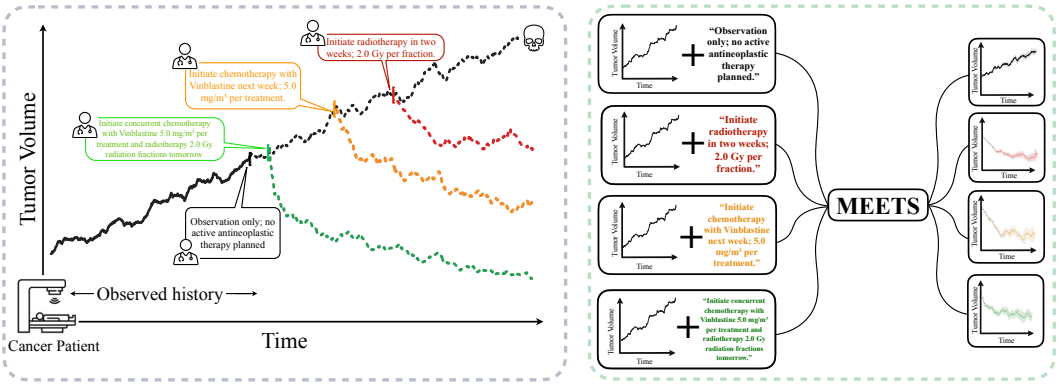

Figure 1: The described event, not the history alone, determines how the system evolves. From the same history, different exogenous events lead to different futures, so a forecaster must change its output correctly when the event changes. MEETS forecasts under any described event, and its response is scored even when the alternate future was never observed, against the response of a similar history that did experience that event.

We introduce MEETS (**M**ultimodal **E**xogenous **E**vent-Conditioned **T**ime **S**eries), a probabilistic forecaster for this setting. The history says where the system is and the events say what acts on it next, so MEETS encodes them separately: one token per history step and one per event, with events attending to one another so concurrent events can interact. A diffusion denoiser generates the whole horizon jointly, reading history and events through separate cross-attentions. Because the diffusion loss scores denoising only under the matched condition, it does not require the forecast to depend on either input; we therefore also penalize the model when a history or event set borrowed from another example denoises as well as the example's own. The penalty saturates, so it does not reward arbitrarily large responses.

Across traffic, energy, and tumor-volume data, MEETS is best on every factual metric and has the lowest response error under swapped events. Four further findings emerge. First, factual accuracy does not guarantee a correct event response: strong probabilistic forecasters with competitive factual error overestimate the response to an alternate event by 26–46% on the tumor-volume simulator, where the reference is exact, while MEETS is within 2%. Second, how the future is generated matters: models that probabilistically sample the whole horizon align far better under event swaps than autoregressive, pointwise, or deterministic ones, and even a zero-shot joint-horizon sampler beats every trained model of those kinds. Third, events are used best when they have both their own conditioning pathway and an explicit training signal: removing the conditional-utility objective cuts directional alignment to 6–9% of the full model, and merging history and events into one attention memory cuts the directional response to about a third. Finally, among unseen event configurations, timing is the only attribute on which MEETS's error depends on transfer direction: it degrades when treatment begins earlier than any onset seen in training, but not later.

Our study makes three main contributions. First, we study both factual and counterfactual forecasting under true exogenous events and construct three datasets for it. Second, we develop MEETS, which keeps history and events on separate attention paths and is trained to use both. Third, we extend nearest-neighbor PEHE (Shalit et al., 2017; Schwab et al., 2018) to matched time-series histories, free-form events, and full trajectories, scoring the forecast shift a swapped event induces by its direction and scale, not only its error, on two real-world datasets where the alternate future never occurred and on a simulator where it is known.

2 Related Work

Text-conditioned forecasting. Text-conditioned forecasting is an active area where the goal is to improve forecasts with information that the observed history alone does not carry. Most methods condition on text *about* the series: its domain, background knowledge, or descriptions of its behavior, sometimes derived from the target itself (Williams et al., 2025; Liu et al., 2024; Jia et al., 2024; Zhong et al., 2025; Su et al., 2025; Zhang et al., 2025a; Wu et al., 2026b; Gu et al., 2026). Such text

differs in kind from an exogenous event: it describes the system or its outcome, whereas an event states a cause whose effect must be inferred (Xu et al., 2025). A smaller line of work conditions on exogenous events (Xu et al., 2025; Ge et al., 2026; Zhang et al., 2025b; Wang et al., 2026; 2024), but trains and evaluates primarily under the events that occurred, even though factual accuracy does not constrain predictions for rarely observed history–event pairs (Shalit et al., 2017). Unfortunately, none of these scores the response to a substituted event against a numerical reference on real data: IATSF swaps interventions with no observed future to score against (Xu et al., 2025), and CAMEF uses alternative announcements as training augmentation (Zhang et al., 2025b). Most similar to our work, Gu et al. (2026) do forecast under substituted textual conditions and fine-tune toward agreement with them. However, on most real benchmarks these conditions are derived from the target, so a swap changes the stated outcome rather than what acts on the system, and unobserved alternates are scored semantically rather than numerically. We condition on exogenous events, train the model to rely on them without rewarding ever-larger responses, and score counterfactual responses against reference futures.

Counterfactual and treatment-response modeling. Counterfactual treatment-response methods are a natural approach to our problem, since comparing outcomes under alternative inputs is their central object (Lim, 2018; Bica et al., 2020; Li et al., 2021; Melnychuk et al., 2022; Ma et al., 2024). However, their interventions are *structured*: categorical treatments, doses, or treatment sequences drawn from a fixed space, and even the closest method, CLEF, conditions on categorical tokens (Li et al., 2026). Exogenous events have no such space: rainfall varies in type, intensity, onset, duration, and overlap with other events, and a structured encoding must fix these attributes in advance and represents compound events only at combinatorial cost (Xu et al., 2025). Causal generative methods further require a causal graph or adjustment set over the intervention variables (Wu et al., 2026a; Xia et al., 2026); since events such as weather cannot be assigned, we instead model $p_\theta(X_{T+1:T+L} \mid X_{1:T}, E)$ predictively. This literature also offers a promising avenue for evaluating outcomes that are never observed: when the counterfactual outcome is missing, NN-PEHE substitutes the observed outcome of a nearest neighbor under the alternative treatment (Hill, 2011; Shalit et al., 2017; Schwab et al., 2018). We extend this construction to matched time-series histories, free-form events, and full future trajectories. Response error alone, however, cannot separate a wrong response from its absence, so we also measure the direction and scale of the response.

Multimodal time series and language models. Many recent works train models that simultaneously embed time series with language (Sun et al., 2024; Jin et al., 2024; Wang et al., 2025; Xie et al., 2025) or generate and edit time series using natural language or metadata (Woo et al., 2025; Ge et al., 2025; Qiu et al., 2026; Narasimhan et al., 2024). These models are feasible backbones for our work, but not built for event-conditioned response: their text describes the series or specifies the output. For example, these works train models that can take in instructions like "make the series decrease" rathen than event descriptions like "rain increases for 3 hours."

3 Problem Definition

Let $\mathbf{X}_{1:T} = (\mathbf{x}_1, \ldots, \mathbf{x}_T) \in \mathbb{R}^{T \times d}$ denote the target history of a time series with d variables, observed up to the present time T, and let $\mathbf{X}_{T+1:T+L} \in \mathbb{R}^{L \times d}$ denote the future target trajectory over a horizon of length L. An *exogenous event* e is a free-form text description of an occurrence that originates outside the target series, such as weather or a treatment intervention; it is never derived by describing, summarizing, or transforming future values of the target itself. Every event lies in the future: it is active over an interval $\mathcal{I} \subseteq \{T+1, \ldots, T+L\}$, and its timing is conveyed to the model only through its text.

Because several events can fall in the same horizon, with possibly overlapping intervals, the model is conditioned on the *event set* $E = \{e_1, \ldots, e_j\}, j \geq 0$, containing every event recorded for the horizon. When nothing occurs during the horizon, E is either empty or contains a single *null event* whose text says so, depending on how the source records the absence. E is specified at time T: it may be known in advance, supplied by an external forecasting source, or posed as a hypothetical. Events that occurred during the history are not provided to the model; their effect is visible only through $\mathbf{X}_{1:T}$. Appendix D describes how event sets, including null events for event-free horizons, are constructed for each dataset.

Given the observed target history and the event set, our objective is to model the conditional distribution of the future target trajectory, $p_\theta(\mathbf{X}_{T+1:T+L} \mid \mathbf{X}_{1:T}, E)$, where θ denotes the trainable parameters. Writing $p^\star$ for the data-generating distribution, success means $p_\theta(\cdot \mid \mathbf{X}_{1:T}, E) \approx p^\star(\cdot \mid \mathbf{X}_{1:T}, E)$ for the event set E recorded with a history and, holding that history fixed, for an alternate event set E'. Futures under E' are generally unobserved in real-world data, so evaluating the latter requires constructed references, which we describe in Section 5.3.

4 METHODOLOGY

To model how a series evolves under described events, we propose MEETS (Multimodal Exogenous Event-Conditioned Time Series), a conditional diffusion model that generates future trajectories from $p_\theta(\mathbf{X}_{T+1:T+L} \mid \mathbf{X}_{1:T}, E)$. The history and the event set play different roles: the history tells us where the series is, and the event set tells us what acts on it next. MEETS therefore encodes them separately into two distinct encodings (Figure 2a,b). Each denoising block consults the two encodings through separate cross-attentions (Figure 2c). Keeping the encodings apart does not guarantee that the model uses both, so we also train MEETS to denoise better with each example's own history and event set than with ones borrowed from other examples.

Encoding the History and the Events. We encode each input separately into a sequence of tokens that the denoiser later queries. To keep the history at its native resolution, we give every time step its own token instead of patching or aggregating. We linearly embed each observation $\mathbf{x}_t \in \mathbb{R}^d$, which holds all d variables at time t, and add a learned position embedding to obtain $\mathbf{h}_t$. A Transformer encoder f_H then lets each time step attend to every other, producing the history encoding $\mathbf{M}_H = f_H(\mathbf{h}_1, \ldots, \mathbf{h}_T)$, with one token per time step (Figure 2b).

We then encode the event set $E = \{e_1, \ldots, e_j\}$ the same way. To keep events distinct, we give each event its own token instead of joining all descriptions into one prompt. A pretrained text encoder f^{text}, fine-tuned with the rest of the model, reads e_r; we average its outputs over the text and project the result with g_E to obtain $\mathbf{v}_r$. Because events in the same horizon may overlap and interact, such as chemotherapy given alongside radiotherapy, an event-set Transformer f_{event} lets the event tokens attend to one another, producing the event encoding $\mathbf{M}_E = f_{\text{event}}(\mathbf{v}_1, \ldots, \mathbf{v}_j)$, with one token per event (Figure 2a). We add no position embedding, because E is a set: concurrent events have no natural order, and each event's text already states when it is active. When E is empty, $\mathbf{M}_E$ is empty and contributes nothing; a null event is encoded like any other event.

The two encoders never exchange information, because the history and the events cover different stretches of time: the history ends at the present time T, and every event lies in the horizon that follows. MEETS therefore enters generation with two separate encodings, $\mathbf{M}_H$ and $\mathbf{M}_E$.

Generating Forecasts. To model a distribution over futures rather than a single forecast, we generate the future with a denoising diffusion model (Ho et al., 2020), which can draw many plausible trajectories for the same history and event set. During training, we add Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, I)$ to the clean future $\mathbf{Y}_0 = \mathbf{X}_{T+1:T+L}$ at a random step $n \in \{1, \ldots, N\}$, giving $\mathbf{Y}_n = \sqrt{\bar{\alpha}_n}\,\mathbf{Y}_0 + \sqrt{1-\bar{\alpha}_n}\,\boldsymbol{\epsilon}$ for a fixed noise schedule $\bar{\alpha}_n$, and train a denoiser $\epsilon_\theta(\mathbf{Y}_n, n, \mathbf{X}_{1:T}, E)$ to recover $\boldsymbol{\epsilon}$.

The denoiser treats the noisy future the way we treated the history: each future time step becomes a token, and we add embeddings of its position and of n to obtain $\mathbf{Z}^{(0)}$. These tokens pass through a stack of blocks (Figure 2c), each performing four steps in order: future self-attention, cross-attention to the history encoding $\mathbf{M}_H$, cross-attention to the event encoding $\mathbf{M}_E$, and a feed-forward network. The future tokens first attend to one another, which keeps the trajectory coherent across the horizon. They then query the history encoding, which gives the state the series has reached at time T, and then the event encoding, which describes how the events alter that state. We query the history first because it is the baseline: when E is empty, the event query contributes nothing and the history alone conditions the forecast.

In each cross-attention, the future tokens form the queries and the encoding forms the keys and values, so the two encodings reach the future through separate paths and are never merged. Every block repeats both queries, so later blocks can retrieve different information as the estimate of the future becomes more precise. After the last block, a normalization and a linear layer produce the noise estimate $\hat{\boldsymbol{\epsilon}}_\theta$.

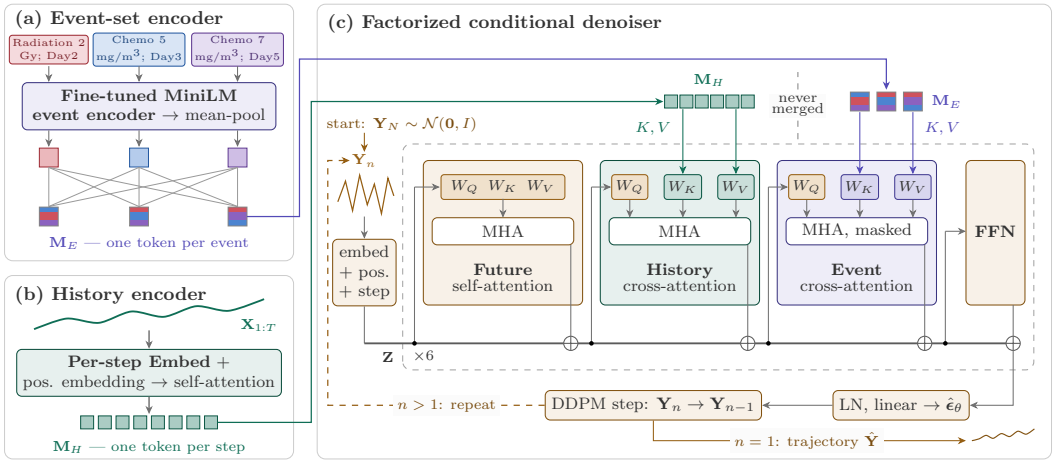

Figure 2: **Overview of MEETS. (a)** Each event is encoded into its own token; self-attention across events gives the event encoding $\mathbf{M}_E$. **(b)** Each time step becomes its own token; self-attention across time gives the history encoding $\mathbf{M}_H$. **(c)** In each denoising block, the noisy future attends to itself, then to $\mathbf{M}_H$, then to $\mathbf{M}_E$ (masked to ignore padded event slots); the two encodings are never merged. To forecast, DDPM steps are repeated from Gaussian noise (dashed loop).

To forecast, we start from $\mathbf{Y}_N \sim \mathcal{N}(\mathbf{0}, I)$ and repeat the standard DDPM step $\mathbf{Y}_n \to \mathbf{Y}_{n-1}$ down to $n = 1$, holding $\mathbf{X}_{1:T}$ and E fixed (dashed loop in Figure 2c). Each run yields one trajectory $\hat{\mathbf{Y}}$; we draw 100 of them to approximate $p_\theta(\mathbf{X}_{T+1:T+L} \mid \mathbf{X}_{1:T}, E)$ (Appendix C.4).

Training. We train the denoiser with the standard diffusion loss. For each training example i, we draw one corruption $(\mathbf{Y}_n^{(i)}, n_i, \boldsymbol{\epsilon}_i)$ and measure how well the model recovers the noise from the example's own history and event set, averaged over a minibatch of B examples:

$$\delta_i^+ = \frac{1}{Ld} \left\| \epsilon_\theta\left(\mathbf{Y}_n^{(i)}, n_i, \mathbf{X}_{1:T}^{(i)}, E_i\right) - \boldsymbol{\epsilon}_i \right\|_F^2, \qquad \mathcal{L}_{\text{diff}} = \frac{1}{B} \sum_{i=1}^{B} \delta_i^+. \tag{1}$$

This loss alone does not ensure that the model uses both encodings. If the history already predicts the future well, the model can ignore E and still reach a low loss, and it will then barely respond to an alternate event set E'. A highly informative event set can likewise let the model ignore the history.

We therefore test each input directly. Keeping the corruption fixed, we replace the example's event set with that of another example, obtained by rotating the minibatch rather than by searching for a similar one, and recompute the error, giving δ_i^{E-}; replacing the history instead gives δ_i^{H-}. A model that uses an input should denoise worse with a borrowed one, and because the corruption is unchanged, any difference in error comes from the replaced input alone. We penalize the model when the borrowed input does as well as the example's own input, and call this penalty the *utility loss*:

$$\mathcal{L}_E = \operatorname*{mean}_i \operatorname{softplus}\left(\delta_i^+ - \delta_i^{E-}\right), \qquad \mathcal{L}_H = \operatorname*{mean}_i \operatorname{softplus}\left(\delta_i^+ - \delta_i^{H-}\right), \tag{2}$$

where each average skips examples whose borrowed input is identical to their own. The borrowed input serves only as a comparison; we never treat the observed future as its outcome.

The complete objective is then $\mathcal{L} = \mathcal{L}_{\text{diff}} + \lambda_E \mathcal{L}_E + \lambda_H \mathcal{L}_H$, with $\lambda_E = \lambda_H = 0.1$ so that the diffusion loss remains the main objective. We ablate both utility losses in our experiments; construction details are in Appendix C.3 (Figure 3).

5 Experiments

We evaluate MEETS under factual, observed events (§5.2), counterfactual events (§5.3), and unseen event configurations (§5.3.1), then ablate its components (§5.3.2).

Dataset	Traffic Volume			Energy Usage			Tumor Volume		
	Train	Validation	Test	Train	Validation	Test	Train	Validation	Test
Window Count	4,702	1,009	1,003	221,751	45,857	44,115	38,504	8,650	8,370
Episode Count	8,010	1,932	2,075	69,993	12,082	14,505	180,140	43,810	40,576

Table 1: Dataset statistics across data splits.

5.1 Experimental Setup

Datasets. We use **three** datasets from distinct domains. Table 1 gives split statistics; Appendix D details pre-processing, event construction, windowing, and splits.

(1) **Traffic Volume** (Hogue, 2019): hourly traffic volume measurements, with corresponding weather observations from which we construct exogenous event descriptions. We use 24 hours of history to forecast the next 12.

(2) **Energy Usage** (Liu et al., 2023): industrial electricity consumption measurements sampled at 15-minute intervals, with exogenous event descriptions representing extreme-weather events. We use 96 observations spanning 24 hours as history and predict the subsequent 48 observations spanning 12 hours.

(3) **Tumor Volume Simulator** (Geng et al., 2017): simulated non-small cell lung cancer tumor-volume trajectories under chemotherapy, radiotherapy, and their combinations, with the future treatments serving as events. Each simulated patient has a 30-timestep untreated history branched into multiple 15-timestep futures under different treatment sequences. Unlike the real-world datasets, the simulator gives the future under alternate events, so they can be evaluated directly.

Baselines. We compare against **five** recent methods: **TADiff**, which forecasts under counterfactual text conditions; **Aurora**, a pretrained multimodal time-series foundation model evaluated zero-shot; **IATSF**, which conditions on future-aligned textual interventions; **EventTSF**, which aligns textual events with temporal segments during probabilistic forecasting; and **CLEF**, which predicts condition-dependent outcomes at arbitrary future times.

Each method receives our future-event text through its native conditioning input, and any other text input it expects is left empty. CLEF predicts one future step per query, so we train and query each step separately with the same history and events. For the deterministic CLEF and IATSF, we sample trajectories by adding validation residuals from the same forecast lead (Appendix E). Every method produces 100 trajectories per test example, and we report mean $\pm$ std over 10 inference repetitions.

5.2 Forecasting factual trajectories

We first consider *factual* forecasting: each model receives the observed history and the events that actually occurred, and its forecast is compared to the realized future on the held-out test split.

Metrics. We use the 100 trajectories each model produces (Section 5.1). We score point accuracy with mean squared error (MSE) and mean absolute error (MAE) on the mean trajectory, and probabilistic accuracy with the continuous ranked probability score (CRPS) and weighted quantile loss (WQL) on the full set. Lower is better for all four; definitions are in Appendix F.

Findings. `MEETS` is best on every dataset and metric (Table 2), and relative to persistence baseline,[2] it reduces MSE by 98.5–99.0% and MAE by 92.9–98.5%. The remaining gaps sort by how each baseline generates the horizon. IATSF and CLEF, which add validation residuals to a deterministic forecast, have the worst CRPS and WQL everywhere, and on the real datasets CLEF's CRPS exceeds its own MAE: a residual pool fixes its spread on the validation period, so under a chronological split it imports the wrong error scale, whereas on Tumor Volume, split by patient, it transfers. A residual bootstrap can therefore score worse than its own point forecast on non-stationary data, and

[2]The persistence baseline predicts the last observed value at every forecast step, gives MSE 4.68, 8.29, 4.06 and MAE 2.12, 2.62, 1.97 on Traffic Volume, Energy Usage, and Tumor Volume.

Dataset	Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
Traffic Volume	TADiff	<u>0.12</u>±0.01	<u>0.20</u>±0.01	<u>0.13</u>±0.01	<u>0.17</u>±0.01
	Aurora	0.52±0.02	0.58±0.01	0.45±0.01	0.59±0.02
	IATSF	1.25±0.04	0.91±0.02	0.92±0.02	1.21±0.03
	EventTSF	0.70±0.02	0.54±0.02	0.42±0.02	0.55±0.03
	CLEF	0.99±0.01	0.87±0.01	1.22±0.03	1.55±0.03
	MEETS	**0.07**±0.00	**0.15**±0.00	**0.11**±0.00	**0.14**±0.00
Energy Usage	TADiff	<u>0.11</u>±0.00	<u>0.09</u>±0.00	<u>0.07</u>±0.00	<u>0.10</u>±0.01
	Aurora	0.19±0.00	0.11±0.00	0.09±0.00	0.13±0.00
	IATSF	1.55±0.03	1.05±0.03	0.91±0.03	1.38±0.04
	EventTSF	0.74±0.01	0.16±0.01	0.13±0.01	0.20±0.01
	CLEF	1.41±0.02	1.08±0.02	1.22±0.04	1.80±0.06
	MEETS	**0.08**±0.00	**0.04**±0.00	**0.03**±0.00	**0.05**±0.01
Tumor Volume	TADiff	<u>0.09</u>±0.00	0.06±0.00	0.04±0.00	0.05±0.00
	Aurora	0.11±0.00	<u>0.04</u>±0.00	<u>0.03</u>±0.00	<u>0.03</u>±0.00
	IATSF	0.73±0.03	0.22±0.03	0.18±0.05	0.19±0.05
	EventTSF	0.10±0.00	0.12±0.00	0.10±0.00	0.10±0.00
	CLEF	0.10±0.00	0.14±0.00	0.11±0.01	0.11±0.01
	MEETS	**0.06**±0.00	**0.03**±0.00	**0.02**±0.00	**0.02**±0.00

Table 2: Factual setting results (mean ± std over 10 inference repetitions). **Bold**: best; <u>underline</u>: second best. MEETS is best on every cell. Every natively probabilistic model has CRPS below its MAE on every dataset; the residual-bootstrapped CLEF exceeds it on both real datasets and IATSF on Traffic Volume, so the samples are no better than their point forecasts.

CRPS above MAE is the diagnostic. Among native samplers, joint-horizon generation dominates: the MSE order is identical on both real datasets, and zero-shot Aurora beats the trained IATSF everywhere and EventTSF and CLEF on every metric on Energy Usage and all but MSE on Tumor Volume. Sampling the whole horizon at once matters more than having seen the dataset. MEETS and TADiff, the two trained joint-horizon diffusion models, are the top two on MSE everywhere and differ in how events meet the history, thereby showing once the generation family is fixed, the conditioning interface is the remaining lever. On Tumor Volume, MSE no longer separates four of the five baselines while MAE, CRPS, and WQL still rank the three joint-horizon samplers first; on the simulator the comparison must rest on distributional metrics.

5.3 Forecasting Counterfactual Trajectories

We next ask how the forecast changes when the event does. In real data, the future under an event that did not occur is unobserved, so we take a reference future from a test instance with a similar history but a different event set.

For each test instance with history $\mathbf{X}_{1:T}$, observed event set E_{real}, and realized future $\mathbf{X}^{\text{real}}_{T+1:T+L}$, the donor is the DTW-nearest other test instance whose event set differs from E_{real}. Its event set becomes E_{swap}, and its realized future $\mathbf{X}^{\text{ref}}_{T+1:T+L}$ stands in for the future $\mathbf{X}_{1:T}$ would have followed under E_{swap}, as in nearest-neighbor approximations to PEHE (Shalit et al., 2017; Schwab et al., 2018). With no distance threshold, every test instance yields exactly one evaluation pair.

On Tumor Volume, no proxy is needed: the simulator gives each history several realized futures, and every pair with distinct event sets forms an evaluation pair, one treated as $(E_{\text{real}}, \mathbf{X}^{\text{real}}_{T+1:T+L})$ and the other as $(E_{\text{swap}}, \mathbf{X}^{\text{ref}}_{T+1:T+L})$, so the counterfactual reference is exact.

Metrics. For each pair, we forecast the same history $\mathbf{X}_{1:T}$ twice, under E_{real} and under E_{swap}, taking the mean of the 100 samples as $\hat{\mathbf{X}}^{\text{real}}_{T+1:T+L}$ and $\hat{\mathbf{X}}^{\text{swap}}_{T+1:T+L}$.We compare the predicted shift $\Delta\hat{\mathbf{X}} = \hat{\mathbf{X}}^{\text{swap}}_{T+1:T+L} - \hat{\mathbf{X}}^{\text{real}}_{T+1:T+L}$, with the reference shift $\Delta\mathbf{X}^{\text{ref}} = \mathbf{X}^{\text{ref}}_{T+1:T+L} - \mathbf{X}^{\text{real}}_{T+1:T+L}$. We score these shifts with four metrics averaged over pairs (Appendix G). ResponseMSE and ResponseMAE measure the error between the two shifts. Directional Alignment (DA) measures whether they point the same way with a similar shape. Directional Response Score (DRS) measures how far the predicted shift moves along the reference, in units of the reference: 1 is the correct scale, 0 is no movement along it, and negative values are movement against it.

Dataset	Model	RespMSE ↓	RespMAE ↓	DA ↑	DRS → 1
Traffic Volume	TADiff	0.15±0.00	0.21±0.00	0.67±0.00	0.78±0.00
	Aurora	0.24±0.00	0.29±0.00	0.43±0.00	0.49±0.00
	IATSF	1.24±0.03	1.10±0.03	0.15±0.03	-0.45±0.05
	EventTSF	0.87±0.01	0.46±0.01	-0.08±0.01	0.02±0.00
	CLEF	0.45±0.01	0.25±0.01	0.23±0.02	0.44±0.01
	MEETS	**0.14**±0.00	**0.19**±0.00	**0.91**±0.00	**1.20**±0.00
Energy Usage	TADiff	0.22±0.00	0.25±0.00	0.67±0.00	0.43±0.01
	Aurora	0.32±0.00	0.27±0.00	0.66±0.00	0.22±0.01
	IATSF	1.49±0.03	0.74±0.03	0.05±0.05	-0.47±0.05
	EventTSF	0.52±0.00	0.37±0.00	-0.04±0.00	0.06±0.02
	CLEF	0.45±0.01	0.61±0.00	0.43±0.01	0.04±0.01
	MEETS	**0.12**±0.00	**0.06**±0.00	**0.89**±0.00	**0.76**±0.01
Tumor Volume	TADiff	0.21±0.00	0.14±0.00	**0.97**±0.01	1.46±0.00
	Aurora	0.25±0.00	0.12±0.00	0.87±0.01	1.26±0.01
	IATSF	0.95±0.03	0.35±0.03	0.43±0.03	-0.42±0.05
	EventTSF	0.94±0.00	0.14±0.00	-0.01±0.01	0.06±0.01
	CLEF	0.25±0.00	0.35±0.00	0.74±0.00	0.89±0.00
	MEETS	**0.20**±0.00	**0.06**±0.00	0.96 ±0.00	**0.98**±0.01

Table 3: Response results (mean ± std over 10 inference repetitions). **Bold**: best; underline: second best. DA and DRS are ranked by absolute distance from 1. MEETS has the lowest response error and the DRS nearest 1 on every dataset. Error cannot separate a response from its absence: EventTSF, with DA near 0 everywhere, matches TADiff's ResponseMAE on Tumor Volume. DA cannot see scale: TADiff has the best DA there but overshoots the reference by 46%.

On Traffic Volume and Energy Usage, imperfect donor matching adds model-independent noise to the reference shift, which mainly biases DRS toward 0 and affects DA less. There, a DRS below 1 need not mean an under-sized response, whereas one above 1 is conservative evidence of an over-sized one. On Tumor Volume, where the reference is exact, DRS reads directly.

Findings. MEETS forecasts the counterfactual response best (Table 3): lowest response error on every dataset, highest DA on the real datasets and within 0.01 of TADiff's on Tumor Volume, and DRS nearest 1 everywhere. Error is the weakest view: a model with no response scores the mean reference shift, so EventTSF, which does not respond, matches TADiff's ResponseMAE on Tumor Volume. Response is judged on DA and DRS. On DA the split is by generation: every joint-horizon sampler, zero-shot Aurora included, has higher DA than every autoregressive or deterministic model on every dataset; conditioning a joint-horizon sampler on the event produces a response, training on the target does not. DA is scale-blind: on the exact reference TADiff, with the highest DA, and Aurora overshoot while MEETS is within 2%. TADiff's fine-tuning maximizes an unbounded inner product with the text embedding, so a larger move is always rewarded, whereas MEETS's utility term saturates; a response objective must saturate, and scale must be scored separately from direction. IATSF alone has negative DRS with positive DA on every dataset, its large moves opposing the reference, which a mean cosine cannot show. On the real references donor noise attenuates DRS toward 0, so MEETS's DRS above 1 on Traffic Volume is conservative evidence of over-response: the saturating term stops rewarding larger responses but does not calibrate their magnitude.

5.3.1 Generalizing to Unseen Events

We test events unseen in training on Tumor Volume, holding out one attribute at a time; all other settings follow Section 5.1. *Intensity:* we train on some treatment doses and test on a held-out higher or lower dose, for both chemotherapy and radiotherapy (Tables 7–10). *Interaction:* we train on chemotherapy or radiotherapy alone and test on both given at once (Table 11). *Timing:* we split the horizon into early, middle, and late windows, train with treatment in one window and test in another, in both directions (Tables 12–17). Across the eleven splits, MEETS is best on all four metrics on nine; both exceptions are chemotherapy dose splits, and in both the winner is TADiff.

Unseen intensity. TADiff beats MEETS on chemotherapy MSE in both directions (0.66–0.94×) but not on radiotherapy (1.4–2.1×); extrapolating upward the chemotherapy win is on MSE and MAE only, with MEETS keeping the lower CRPS and WQL, whereas extrapolating downward TADiff

leads on all four. In the simulator chemotherapy acts through a decaying concentration and radiotherapy as a single fraction at the treated step, so dose extrapolation on the persistent treatment is the one regime where MEETS is not best on every metric, and either treatment alone would have hidden it: dose generalization must be reported per treatment type. Toward lower doses, the future stays close to untreated continuation, so Aurora's MSE drops 3–6× while MEETS moves by at most 1.2× and TADiff, whose fine-tuning rewards a larger move without bound, gets worse for both treatments exactly where the true response shrinks; dose generalization is therefore tested in both directions, since the easy direction differs by model.

Unseen interaction. On the combined treatment, only TADiff and CLEF stay within 1.4× of MEETS on MSE; the rest are 3.6× or worse. Aurora matches MEETS to within 1.2× on MAE, CRPS, and WQL but is 4× worse on MSE (MSE/MAE^2 of 37 against 13): it handles typical windows but badly misses a few, which MAE or CRPS alone would hide.

Unseen timing. Later onset leaves more of the horizon untreated, which again favors continuation: Aurora is second on MAE in all three later-onset splits, while TADiff shows its largest deficit anywhere (5.7× MEETS on MSE). Earlier onset breaks continuation from the first step, making Aurora 2.2–4× and MEETS up to 2.6× worse than the matching later-onset split; timing is the only attribute on which MEETS's error depends on direction. Because timing concerns placement rather than magnitude, EventTSF, gives onset a segment-aligned slot, stays flat at 2–2.5× MEETS across timing splits, whereas TADiff's pooled vector degrades as the shift grows, from 1.6–2.7× on one-window shifts to 2.3–5.7× on two-window shifts.

5.3.2 Ablation Study

Event processing. Encoding events separately cuts MSE 2.2–2.3× relative to concatenating them into one text, and adding event-set self-attention cuts it a further 12–21× (Table 18). Without self-attention the event signal is an attention-weighted average of independently encoded events, so a joint effect can be no larger than the strongest single event's; the interaction between events is where most of the accuracy lies. On response the two steps differ: separating events roughly doubles DA and halves ResponseMSE relative to concatenation, but self-attention accounts for the remaining 8–11× in ResponseMSE. Separating events is enough to respond in the right direction; only their interaction makes the response accurate, so an event encoding that pools the set into one vector caps accuracy at the architecture.

Conditional-utility loss. We train with the diffusion loss alone, then add only $\mathcal{L}_H$, only $\mathcal{L}_E$, and both. The diffusion loss alone leaves DA at 6–9% of MEETS's (Table 20): it can be minimized from history alone, so ignoring the event costs nothing. $\mathcal{L}_E$, the only term that rewards the factual event over a mismatched one, alone recovers 83–84% of the DA but leaves MSE 5–7× higher and DRS below 0.75; $\mathcal{L}_H$, which penalizes denoising as well from borrowed history, anchors that response to the observed series. Matched-condition likelihood does not teach a response; it must be trained with explicit mismatch penalty, such terms must be ablated jointly as each understates its effect alone.

History–event conditioning. Fusing the history and event encodings into a single cross-attention memory raises MSE 14–20× and cuts DRS to a third of MEETS's on every dataset (Table 19). In a shared softmax the 24–96 history tokens explain more of the factual future than the event tokens and absorb the attention. The shrinkage is identical across domains, so it is architectural rather than a data effect: history and events need separate cross-attention.

6 Conclusion

We introduced MEETS, a probabilistic forecaster that models how a system evolves from its observed history under natural-language descriptions of exogenous events, together with an evaluation framework that scores responses to alternate events even when the alternate future is unobserved. Across traffic, energy, and tumor-volume data, MEETS forecasts more accurately than existing text-conditioned methods under both observed and alternate events. Our results show that factual accuracy does not certify a correct event response, so event-conditioned forecasters must be evaluated on the direction and scale of their response, not only on the futures they were trained to reproduce. Future work should address the temporal placement of events, where generalization remains hardest, and should validate matched real-data references so that alternate-event accuracy can be measured in absolute rather than comparative terms.

AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic data sets. We have not used generative AI tools for other tasks with required disclosure, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools to edit software code and assist with literature searches to identify relevant papers within specific research domains. We have reviewed all AI-assisted work. AI-assisted code edits were reviewed and tested by the authors, and papers identified through AI-assisted literature searches were independently verified and reviewed by the authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

We have reviewed the ICLR Code of Ethics and are not aware of any ethical concerns or violations arising from this work. The study does not involve the collection of human-subject data or other procedures requiring additional ethical approval. We take responsibility for ensuring that the research, experiments, and reporting comply with the ICLR Code of Ethics.

REPRODUCIBILITY STATEMENT

We will release all code and data required to reproduce the results of this work. The complete codebase used for our experiments is included as a zipped archive in the supplementary materials. Upon publication, we will additionally release the model weights required to replicate our experiments and reported results. The main paper and appendices provide the corresponding methodological, implementation, dataset-processing, experimental, and evaluation details needed for reproducibility.

REFERENCES

Alberto Abadie, Alexis Diamond, and Jens Hainmueller. Synthetic control methods for comparative case studies: Estimating the effect of california's tobacco control program. *Journal of the American Statistical Association*, 105(490):493–505, 2010. doi: 10.1198/jasa.2009.ap08746. URL https://doi.org/10.1198/jasa.2009.ap08746.

Ioana Bica, Ahmed M Alaa, James Jordon, and Mihaela van der Schaar. Estimating counterfactual treatment outcomes over time through adversarially balanced representations. *International Conference on Learning Representations*, 2020.

D. R. COX. Two further applications of a model for binary regression. *Biometrika*, 45(3-4):562–565, 12 1958. ISSN 0006-3444. doi: 10.1093/biomet/45.3-4.562. URL https://doi.org/10.1093/biomet/45.3-4.562.

Alicia Curth and Mihaela Van Der Schaar. In search of insights, not magic bullets: Towards demystification of the model selection dilemma in heterogeneous treatment effect estimation. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 6623–6642. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/curth23b.html.

Sandeep Datla and Satish Sharma. Impact of cold and snow on temporal and spatial variations of highway traffic volumes. *Journal of Transport Geography*, 16(5):358–372, 2008. ISSN 0966-6923. doi: https://doi.org/10.1016/j.jtrangeo.2007.12.003. URL https://www.sciencedirect.com/science/article/pii/S0966692307001299.

Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators, 2021.

Yunfeng Ge, Jiawei Li, Yiji Zhao, Haomin Wen, Zhao Li, Meikang Qiu, Hongyan Li, Ming Jin, and Shirui Pan. T2s: High-resolution time series generation with text-to-series diffusion models. In *International Joint Conference on Artificial Intelligence*, 2025.

Yunfeng Ge, Ming Jin, Yiji Zhao, Hongyan Li, Bo Du, Chang Xu, and Shirui Pan. Eventtsf: Event-aware non-stationary time series forecasting, 2026. URL https://arxiv.org/abs/2508.13434.

Changran Geng, Harald Paganetti, and Clemens Grassberger. Prediction of treatment response for combined chemo- and radiation therapy for non-small cell lung cancer patients using a biomathematical model. *Scientific Reports*, 7:13542, 2017. doi: 10.1038/s41598-017-13646-z. URL https://doi.org/10.1038/s41598-017-13646-z.

Amanda Gentzel, Dan Garant, and David Jensen. The case for evaluating causal models using interventional measures and empirical data. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/a87c11b9100c608b7f8e98cfa316ff7b-Paper.pdf.

Shuqi Gu, Yongxiang Zhao, Baoyu Jing, and Kan Ren. What if tomorrow is the world cup final? counterfactual time series forecasting with textual conditions. *arXiv preprint arXiv:2605.14422*, 2026.

Sungwon Han, Seungeon Lee, Meeyoung Cha, Sercan O Arik, and Jinsung Yoon. Retrieval augmented time series forecasting. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 21774–21797. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/han25d.html.

Jennifer L. Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 2011. doi: 10.1198/jcgs.2010.08162. URL https://doi.org/10.1198/jcgs.2010.08162.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

John Hogue. Metro Interstate Traffic Volume. UCI Machine Learning Repository, 2019. DOI: https://doi.org/10.24432/C5X60B.

Furong Jia, Kevin Wang, Yixiang Zheng, Defu Cao, and Yan Liu. GPT4MTS: Prompt-based large language model for multimodal time-series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(21):23343–23351, March 2024. doi: 10.1609/aaai.v38i21.30383. URL https://doi.org/10.1609/aaai.v38i21.30383.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-LLM: Time series forecasting by reprogramming large language models. In *International Conference on Learning Representations (ICLR)*, 2024.

Michelle M Li, Kevin Li, Yasha Ektefaie, Ying Jin, Yepeng Huang, Shvat Messica, Tianxi Cai, and Marinka Zitnik. Controllable sequence editing for biological and clinical trajectories. *https://pmc.ncbi.nlm.nih.gov/articles/PMC11838772/*, 2026.

Rui Li, Stephanie Hu, Mingyu Lu, Yuria Utsumi, Prithwish Chakraborty, Daby M. Sow, Piyush Madan, Jun Li, Mohamed Ghalwash, Zach Shahn, and Li-wei Lehman. G-net: a recurrent network approach to g-computation for counterfactual prediction under a dynamic treatment regime. In Subhrajit Roy, Stephen Pfohl, Emma Rocheteau, Girmaw Abebe Tadesse, Luis Oala, Fabian Falck, Yuyin Zhou, Liyue Shen, Ghada Zamzmi, Purity Mugambi, Ayah Zirikly, Matthew B. A. McDermott, and Emily Alsentzer (eds.), *Proceedings of Machine Learning for Health*, volume 158 of *Proceedings of Machine Learning Research*, pp. 282–299. PMLR, 04 Dec 2021. URL https://proceedings.mlr.press/v158/li21a.html.

Bryan Lim. Forecasting treatment responses over time using recurrent marginal structural networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/56e6a93212e4482d99c84a639d254b67-Paper.pdf.

G. Liu, J. Liu, Y. Bai, et al. EWELD: A large-scale industrial and commercial load dataset in extreme weather events. *Scientific Data*, 10:615, 2023. doi: 10.1038/s41597-023-02503-6. URL https://doi.org/10.1038/s41597-023-02503-6.

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

Yuchen Ma, Valentyn Melnychuk, Jonas Schweisthal, and Stefan Feuerriegel. Diffpo: A causal diffusion model for learning distributions of potential outcomes. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 43663–43692. Curran Associates, Inc., 2024. doi: 10.52202/079017-1384. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/4d254c96c5ff500dda2ac0a58987aeba-Paper-Conference.pdf.

Valentyn Melnychuk, Dennis Frauen, and Stefan Feuerriegel. Causal transformer for estimating counterfactual outcomes. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 15293–15329. PMLR,

17–23 Jul 2022. URL https://proceedings.mlr.press/v162/melnychuk22a.html.

Jacob A. Mincer. Front matter. In Jacob A. Mincer (ed.), *Economic Forecasts and Expectations: Analysis of Forecasting Behavior and Performance*, pp. –20–0. National Bureau of Economic Research, 1969. ISBN 0-870-14202-X. URL https://www.nber.org/chapters/c1213.

Wenhao Mu, Facundo Yan, Anik Mumssen, Marisa Eisenberg, and Alexander Rodriguez. Benchmarking counterfactual prediction in epidemic time series with time-varying interventions. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2026.

Sai Shankar Narasimhan, Shubhankar Agarwal, Oguzhan Akcin, Sujay Sanghavi, and Sandeep P. Chinchali. Time weaver: A conditional time series generation model. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 37293–37320. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/narasimhan24a.html.

Jiaxing Qiu, Dongliang Guo, Brynne Sullivan, Teague R Henry, and Thomas Hartvigsen. Instruction-based time series editing. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pp. 1216–1227, 2026.

Yuta Saito and Shota Yasui. Counterfactual cross-validation: Stable model selection procedure for causal inference models. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8398–8407. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/saito20a.html.

Patrick Schwab, Lorenz Linhardt, and Walter Karlen. Perfect Match: A Simple Method for Learning Representations For Counterfactual Inference With Neural Networks. *arXiv preprint arXiv:1810.00656*, 2018.

Uri Shalit, Fredrik D. Johansson, and David Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3076–3085. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/shalit17a.html.

Chen Su, Yuanhe Tian, and Yan Song. Multimodal conditioned diffusive time series forecasting, 2025. URL https://arxiv.org/abs/2504.19669.

Chenxi Sun, Hongyan Li, Yaliang Li, and Shenda Hong. TEST: Text prototype aligned embedding to activate LLM's ability for time series. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/a4352e2c9d93582898a2a20e1f514e8f-Paper-Conference.pdf.

Ben Van Calster, Daan Nieboer, Yvonne Vergouwe, Bavo De Cock, Michael J. Pencina, and Ewout W. Steyerberg. A calibration hierarchy for risk models was defined: from utopia to empirical data. *Journal of Clinical Epidemiology*, 74:167–176, 2016. doi: 10.1016/j.jclinepi.2015.12.005.

Chengsen Wang, Qi Qi, Jingyu Wang, Haifeng Sun, Zirui Zhuang, Jinming Wu, Lei Zhang, and Jianxin Liao. Chattime: A unified multimodal time series foundation model bridging numerical and textual data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 12694–12702, 2025.

Siyuan Wang, Peng Chen, Yihang Wang, Wanghui Qiu, Guo Guo, Bin Yang, and Yang Shu. Unlocking the value of text: Event-driven reasoning and multi-level alignment for time series forecasting. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 31182–31210, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/34a1fc7890141f1ada3d8bc6199cce07-Paper-Conference.pdf.

Xinlei Wang, Maike Feng, Jing Qiu, Jinjin Gu, and Junhua Zhao. From news to forecast: Integrating event analysis in llm-based time series forecasting with reflection. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 58118–58153. Curran Associates, Inc., 2024. doi: 10.52202/079017-1853. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/6aef8bffb372096ee73d98da30119f89-Paper-Conference.pdf.

Andrew Robert Williams, Arjun Ashok, Étienne Marcotte, Valentina Zantedeschi, Jithendaraa Subramanian, Roland Riachi, James Requeima, Alexandre Lacoste, Irina Rish, Nicolas Chapados, and Alexandre Drouin. Context is key: A benchmark for forecasting with essential textual information. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=ih2WuBT1Fn.

Jaeyun Woo, Jiseok Lee, and Brian Kenji Iwana. Towards time series generation conditioned on unstructured natural language, 2025. URL https://arxiv.org/abs/2506.22927.

Dongze Wu, Feng Qiu, and Yao Xie. Doflow: Flow-based generative models for interventional and counterfactual forecasting on time series. In *International Conference on Learning Representations*, 2026a.

Xingjian Wu, Jianxin Jin, Wanghui Qiu, Peng Chen, Yang Shu, Bin Yang, and Chenjuan Guo. Aurora: Towards universal generative multimodal time series forecasting. In *The Fourteenth International Conference on Learning Representations*, 2026b.

Yutong Xia, Chang Xu, Yuxuan Liang, Li Zhao, Qingsong Wen, Roger Zimmermann, and Jiang Bian. Causal time series generation via diffusion models, 2026. URL https://arxiv.org/abs/2509.20846.

Zhe Xie, Zeyan Li, Xiao He, Longlong Xu, Xidao Wen, Tieying Zhang, Jianjun Chen, Rui Shi, and Dan Pei. Chatts: Aligning time series with llms via synthetic data for enhanced understanding and reasoning. In *Proceedings of the VLDB Endowment, 2025*, 2025.

Zhijian Xu, Hao Wang, and Qiang Xu. Intervention-aware forecasting: Breaking historical limits from a system perspective, 2025. URL https://arxiv.org/abs/2405.13522.

Da Zhang, Bingyu Li, Zhuyuan Zhao, Junyu Gao, Feiping Nie, and Xuelong Li. Unidiff: A unified diffusion framework for multimodal time series forecasting, 2025a. URL https://arxiv.org/abs/2512.07184.

Yang Zhang, Wenbo Yang, Jun Wang, Qiang Ma, and Jie Xiong. Camef: Causal-augmented multimodality event-driven financial forecasting by integrating time series patterns and salient macroeconomic announcements. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD '25, pp. 3867–3878, New York, NY, USA, 2025b. Association for Computing Machinery. ISBN 9798400714542. doi: 10.1145/3711896.3736872. URL https://doi.org/10.1145/3711896.3736872.

Siru Zhong, Weilin Ruan, Ming Jin, Huan Li, Qingsong Wen, and Yuxuan Liang. Time-vlm: Exploring multimodal vision-language models for augmented time series forecasting, 2025. URL https://arxiv.org/abs/2502.04395.

APPENDIX

A LIMITATIONS

Our study has several limitations that suggest natural extensions. First, our experiments focus on univariate target series; extending MEETS to high-dimensional multivariate systems is an important direction for future work. Second, datasets pairing time series with sufficiently detailed future exogenous-event descriptions remain scarce, limiting the range of domains we can evaluate. Third, real-world alternate-event outcomes are inherently unobserved, motivating our matched-reference evaluation; future benchmarks with richer intervention data could provide stronger empirical references. Finally, our controlled evaluation of unseen event configurations is concentrated in the Tumor Volume simulator, and extending such tests to additional real-world domains is a promising direction.

B RELATED WORKS: EXPANDED

B.1 TEXT-CONDITIONED FORECASTING

Text about the series. Most text-conditioned forecasters use text as context about the series rather than as a cause acting on it. CiK evaluates tasks in which textual context—constraints, background knowledge, or future information—is necessary for accurate prediction, but it does not isolate the response to a replaceable future event (Williams et al., 2025). GPT4MTS adds related textual information through a prompt-based LLM framework (Jia et al., 2024); MCD-TSF uses text as supplementary description of the observed history (Su et al., 2025); UniDiff combines text and temporal information through unified cross-attention with separate guidance strengths (Zhang et al., 2025a); and Aurora uses multimodal domain knowledge to guide a probabilistic foundation model (Wu et al., 2026b). Time-VLM generates its textual and visual inputs from the input series itself (Zhong et al., 2025). Time-MMD's textual streams do include events and policies (Liu et al., 2024), but its benchmark is organized around exploiting aligned text to improve prediction rather than holding a history fixed and replacing one future event with another.

Exogenous events. IATSF formulates forecasting with temporally synchronized external interventions expressed in natural language (Xu et al., 2025). It states that conditioning text must describe external factors that influence the system but are not themselves outcomes of it, since text that describes or summarizes the trajectory would leak future state. Its own benchmark meets this only in part: the Electricity holiday captions are obtained by thresholding one of the target channels, and the Atmospheric Physics captions describe the trends and levels of forecast channels such as temperature, pressure, and humidity. Its FIATS model swaps future interventions on a single test sample and shows that the forecast changes, but because no observed future exists for the same history under the alternate intervention, the numerical accuracy of that trajectory is not established. EventTSF conditions non-stationary forecasts on natural-language events through event-aware diffusion and temporal alignment (Ge et al., 2026), but its factual objective does not constrain history–event combinations that were weakly represented or absent during training. CAMEF adds LLM-generated alternative announcements as augmentation for financial forecasting (Zhang et al., 2025b), but these supply no observed market trajectory under the alternative condition. VoT and From News to Forecast reason over exogenous events and news, but evaluate factual forecasts rather than same-history event substitution (Wang et al., 2026; 2024).

Factual versus alternate-event performance. Training and evaluating on factual examples $(X_{1:T}, E, X_{T+1:T+L})$ constrains prediction where the joint history–event distribution places mass; it need not constrain predictions for a recombined pair $(X_{1:T}, E')$ that is rare or unsupported. Shalit et al. (2017) make the analogous point in counterfactual learning: factual prediction error alone is insufficient when treated and control distributions differ. Strong factual performance is therefore no guarantee of strong alternate-event performance, which must be tested explicitly. TADiff is the closest prior work: it forecasts the same history under multiple future textual conditions with a diffusion model, and fine-tunes the forecaster toward agreement with the text condition by maximizing an unbounded inner product with the text embedding (Gu et al., 2026). We therefore do

not view alternate textual conditioning itself as a gap. Three aspects make direct transfer to our setting non-trivial. First, its text-attribution mechanism conditions on both a historical and a future textual condition, whereas our setting provides an observed history and future event text with no aligned historical description. Second, it constructs real-data alternate conditions by textual compatibility with the historical condition, whereas the compatibility of an external driver with a history is determined by the system's response rather than text–text similarity. Third, its real benchmarks do not cleanly instantiate external drivers: the ETTm1, Traffic, and Exchange captions are derived from the target series, and the Weather captions and target both describe the weather system. Most importantly, when real alternate futures are unobserved, TADiff evaluates text-semantic consistency through DTTC rather than numerical accuracy, which does not establish whether a trajectory has the appropriate direction, magnitude, timing, and range for the target system.

B.2 Counterfactual and Treatment-Response Modeling

Structured interventions. RMSN, CRN, G-Net, and Causal Transformer predict longitudinal outcomes under alternative treatment sequences or strategies (Lim, 2018; Bica et al., 2020; Li et al., 2021; Melnychuk et al., 2022), and DiffPO targets full potential-outcome distributions rather than point estimates (Ma et al., 2024). Their intervention spaces, however, are explicitly structured: categorical treatments, doses, or treatment sequences. Extending them to natural-language events requires an additional grounding problem, since event type, intensity, timing, duration, and compositional semantics must first be inferred from text and mapped to the parts of the system that respond. IATSF makes a related argument: numeric or one-hot exogenous inputs must be sampled at the target's rate even when interventions are sparse, and represent compound events only with combinatorial overhead (Xu et al., 2025).

Predictive conditioning versus causal identification. DoFlow generates interventional and counterfactual trajectories through a causal DAG and continuous normalizing flows (Wu et al., 2026a), and CaTSG uses backdoor-adjusted diffusion guidance (Xia et al., 2026). Applying them to our setting would require specifying the intervention variables and the causal structure or adjustment set, and many exogenous events, such as weather, cannot be assigned. We instead target predictive conditioning, $p_\theta(X_{T+1:T+L} \mid X_{1:T}, E)$, without requiring it to identify an interventional causal estimand; causal generative methods solve a stronger but different problem. CLEF learns when a condition takes effect and which variables it should modify, enabling immediate and delayed edits while preserving unaffected portions of a longitudinal sequence (Li et al., 2026). Its control mechanisms motivate desiderata such as temporal localization and selective response, but it operates on categorical condition tokens and does not interpret a natural-language external event acting on a distinct target process.

When exact alternate outcomes are required, simulators provide them: EpiCF-Bench uses a calibrated epidemic simulator to generate factual and alternate-policy trajectories (Mu et al., 2026). In observational settings, evaluation without counterfactual labels is difficult (Saito & Yasui, 2020; Curth & Van Der Schaar, 2023; Gentzel et al., 2019). The closest precedent for our references is NN-PEHE, which substitutes the observed outcome of a nearest neighbor from the alternative treatment group for the missing counterfactual outcome (Shalit et al., 2017; Schwab et al., 2018). We extend this construction from static covariates and structured treatments to matched time-series histories, natural-language events, and full future trajectories. Synthetic Control instead builds a weighted donor combination whose post-intervention outcomes serve as the missing comparison path (Abadie et al., 2010), and RAFT uses the realized futures of retrieved histories as predictive features rather than as reference outcomes (Han et al., 2025).

B.3 Time Series and Language

Alignment, generation, and editing. TEST, Time-LLM, ChatTime, and ChatTS align time series with language models for forecasting, classification, question answering, or reasoning over observed series (Sun et al., 2024; Jin et al., 2024; Wang et al., 2025; Xie et al., 2025); successful cross-modal alignment does not by itself determine how a future external event should change a target trajectory. Text-to-series generation and editing condition outputs on natural language (Woo et al., 2025; Ge et al., 2025); InstructTime, for example, edits an existing series from natural-language instructions with controllable strength (Qiu et al., 2026), and Time Weaver generates series from heterogeneous

metadata (Narasimhan et al., 2024). In these tasks the text or metadata specifies the desired output, whereas in our setting the text describes an external occurrence whose effect must be inferred: "make the series decrease" specifies an output property, but "heavy rain for three hours" does not say whether a particular target should decrease, increase, lag, or change variance.

C MEETS: IMPLEMENTATION DETAILS

C.1 EVENT AND HISTORY ENCODERS

Event-text encoder. Each event $e_r \in E$ is tokenized independently and passed through a pre-trained bidirectional language encoder f^{text}, which is fine-tuned jointly with the rest of the model. We use `sentence-transformers/all-MiniLM-L6-v2`, whose token-level hidden dimension is $D_{\text{text}} = 384$, with a maximum tokenized length of 96. Let $\mathbf{U}_r$ denote the final token-level hidden states of $f^{\text{text}}(e_r)$, with rows $\mathbf{U}_{r,m} \in \mathbb{R}^{D_{\text{text}}}$, and let $\mathcal{P}_r$ be the set of non-padding token positions of e_r. We average over these positions,

$$\bar{\mathbf{u}}_r = \frac{1}{|\mathcal{P}_r|} \sum_{m \in \mathcal{P}_r} \mathbf{U}_{r,m},$$

and map the result into the model dimension D with two projections,

$$\tilde{\mathbf{u}}_r = \mathrm{LN}(\mathrm{GELU}(W_{\text{text}} \bar{\mathbf{u}}_r + \mathbf{b}_{\text{text}})), \qquad \mathbf{v}_r = \mathrm{LN}(\mathrm{GELU}(W_E \tilde{\mathbf{u}}_r + \mathbf{b}_E)) \in \mathbb{R}^D.$$

Together, the two projections form the map g_E of Section 4.

Event-set Transformer. The event tokens of one horizon attend to one another through

$$\mathbf{M}_E = f_{\text{event}}(\mathbf{v}_1, \ldots, \mathbf{v}_j) \in \mathbb{R}^{j \times D},$$

where f_{event} is a pre-normalized Transformer encoder with two layers and eight attention heads. No position embeddings are added across event tokens, so the forecast does not depend on the order in which the events are stored. This matters when events are concurrent and have no natural order; the timing of each event is contained in its text, as specified in Section 3.

Event slots are padded only for batching, up to the largest number of events in the minibatch. Padded slots are excluded through a key-padding mask in the event-set Transformer and through the same validity mask in event cross-attention. When $E = \varnothing$, all event slots are invalid, the event-set Transformer is skipped, and the event cross-attention output is set to zero, so the event encoding contributes nothing; no learned null-event token is introduced. A null event, where the source provides one, is tokenized and encoded exactly like any other event.

History encoder. For each time step t, the observation $\mathbf{x}_t \in \mathbb{R}^d$ is projected into the model dimension and combined with a learned position embedding,

$$\mathbf{h}_t = W_H \mathbf{x}_t + \mathbf{b}_H + \mathbf{p}_t^H.$$

The history encoding is

$$\mathbf{M}_H = f_H(\mathbf{h}_1, \ldots, \mathbf{h}_T) \in \mathbb{R}^{T \times D},$$

where f_H is a pre-normalized Transformer encoder with three layers and eight attention heads, followed by a final layer normalization. f_H uses no attention mask, so every time step attends to every other. All datasets in our experiments are univariate ($d = 1$). The history encoder receives no event representations, and the event encoder receives no time-series tokens.

C.2 DIFFUSION PROCESS AND DENOISER ARCHITECTURE

Forward process. We use a DDPM with $N = 1000$ diffusion steps and a cosine noise schedule $\beta_1, \ldots, \beta_N$. Define

$$\alpha_n = 1 - \beta_n, \qquad \bar{\alpha}_n = \prod_{n'=1}^{n} \alpha_{n'}.$$

During training, n is drawn uniformly from $\{1, \ldots, N\}$, and the clean future $\mathbf{Y}_0 = \mathbf{X}_{T+1:T+L} \in \mathbb{R}^{L \times d}$ is corrupted as

$$\mathbf{Y}_n = \sqrt{\bar{\alpha}_n}\, \mathbf{Y}_0 + \sqrt{1 - \bar{\alpha}_n}\, \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, I).$$

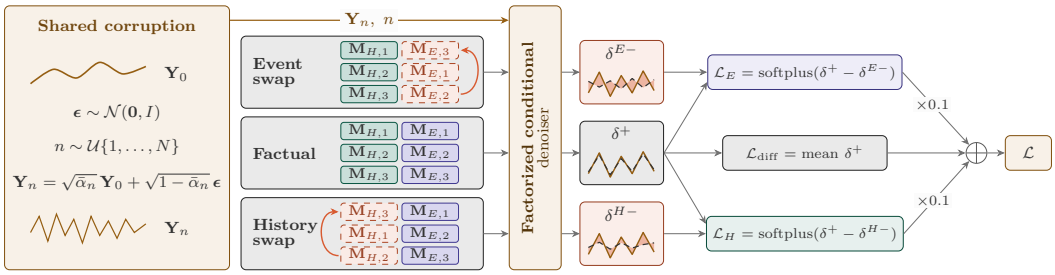

Figure 3: **Training with the utility loss**, shown for a minibatch of three examples; $\mathbf{M}_{H,i}$ and $\mathbf{M}_{E,i}$ are the history and event encodings of example i. **Left:** one corruption $(\mathbf{Y}_n, n, \boldsymbol{\epsilon})$ is drawn per example and shared by all three passes. **Middle:** the own-input pass pairs each history with its own event set; the event-replacement pass borrows event sets by rotating the minibatch, and the history-replacement pass borrows histories instead (borrowed encodings in red). All passes use the same denoiser. **Right:** the resulting errors define $\mathcal{L}_{\text{diff}}$ and the utility losses $\mathcal{L}_E$ and $\mathcal{L}_H$ (Eqs. 1–2), which are combined with weight 0.1 into $\mathcal{L}$

Future tokens. Each noisy future time step becomes its own token. Let $\boldsymbol{\gamma}(n)$ denote a learned transformation of a sinusoidal diffusion-step embedding,

$$\boldsymbol{\gamma}(n) = W_2\,\mathrm{SiLU}(W_1\,\mathrm{SinEmb}(n) + \mathbf{b}_1) + \mathbf{b}_2,$$

with hidden width $4D$. For future position $\tau = 1, \ldots, L$,

$$\mathbf{z}_\tau^{(0)} = W_Y \mathbf{Y}_{n,\tau} + \mathbf{b}_Y + \mathbf{p}_\tau^F + \boldsymbol{\gamma}(n),$$

where $\mathbf{p}_\tau^F$ is a learned position embedding, and $\mathbf{Z}^{(0)} = [\mathbf{z}_1^{(0)}; \ldots; \mathbf{z}_L^{(0)}]$.

Denoising block. Each block consists of pre-normalized residual sublayers. Let $\mathrm{CA}(\mathbf{A}, \mathbf{M})$ denote multi-head cross-attention with queries computed from $\mathbf{A}$ and keys and values computed from the encoding $\mathbf{M}$. Block ℓ computes

$$\begin{aligned}
\mathbf{S}^{(\ell)} &= \mathbf{Z}^{(\ell)} + \mathrm{SA}^{(\ell)}\left(\mathrm{LN}_{\text{self}}^{(\ell)}(\mathbf{Z}^{(\ell)})\right), \\
\mathbf{R}^{(\ell)} &= \mathbf{S}^{(\ell)} + \mathrm{CA}_H^{(\ell)}\left(\mathrm{LN}_H^{(\ell)}(\mathbf{S}^{(\ell)}),\, \mathbf{M}_H\right), \\
\mathbf{C}^{(\ell)} &= \mathbf{R}^{(\ell)} + \mathrm{CA}_E^{(\ell)}\left(\mathrm{LN}_E^{(\ell)}(\mathbf{R}^{(\ell)}),\, \mathbf{M}_E\right), \\
\mathbf{Z}^{(\ell+1)} &= \mathbf{C}^{(\ell)} + \mathrm{FFN}^{(\ell)}\left(\mathrm{LN}_{\text{FFN}}^{(\ell)}(\mathbf{C}^{(\ell)})\right),
\end{aligned}$$

Future self-attention and history cross-attention use no mask. Event cross-attention masks padded event slots, and its output is zero when $E = \varnothing$ (Appendix C.1). We use six blocks, $\ell = 0, \ldots, 5$, and the noise estimate is

$$\hat{\boldsymbol{\epsilon}}_\theta = W_O\,\mathrm{LN}\left(\mathbf{Z}^{(6)}\right) + \mathbf{b}_O.$$

Sizes. The model width is $D = 256$. All attention modules in the denoiser use eight heads, feed-forward layers have width $4D$ with GELU activation, and dropout is 0.1 throughout.

C.3 Utility Loss Construction

Shared corruption. For each example i, the own-input pass, the event-replacement pass, and the history-replacement pass use the same diffusion step n_i, Gaussian noise $\boldsymbol{\epsilon}_i$, and corrupted future $\mathbf{Y}_n^{(i)}$; only the input under test changes. The event text of the minibatch is encoded once, and the resulting event representations are reused by all three passes. Because the passes run in training mode, they may use different dropout realizations.

Borrowed event sets. We obtain borrowed event sets through a nonzero cyclic shift π of the minibatch, so that no example receives its own event set. The complete event set of an example and its validity mask move together; individual events within an example are never shuffled. Starting from a random nonzero shift, we examine all nonzero cyclic shifts and select the one with the fewest examples whose borrowed event set is identical in text to their own. Let $\mathcal{K}_E$ denote the examples whose borrowed event set differs from their own. An empty event set and a null-event set are both valid inputs and may be borrowed like any other. Differing refers to the event text only; it does not imply that two event sets are semantically unrelated. For $i \in \mathcal{K}_E$,

$$\delta_i^{E-} = \frac{1}{Ld} \left\| \epsilon_\theta\Big(\mathbf{Y}_n^{(i)}, n_i, \mathbf{X}_{1:T}^{(i)}, E_{\pi(i)}\Big) - \boldsymbol{\epsilon}_i \right\|_F^2 .$$

Borrowed histories. We obtain borrowed histories through a random nonzero cyclic shift ρ of the minibatch, so that no example receives its own history and each history is borrowed exactly once. Let $\mathcal{K}_H$ denote the examples whose borrowed history is not an exact duplicate of their own. For $i \in \mathcal{K}_H$,

$$\delta_i^{H-} = \frac{1}{Ld} \left\| \epsilon_\theta\Big(\mathbf{Y}_n^{(i)}, n_i, \mathbf{X}_{1:T}^{(\rho(i))}, E_i\Big) - \boldsymbol{\epsilon}_i \right\|_F^2 .$$

Utility losses. Written with these sets, the averages in Eq. equation 2 are

$$\mathcal{L}_E = \frac{1}{|\mathcal{K}_E|} \sum_{i \in \mathcal{K}_E} \mathrm{softplus}\big(\delta_i^+ - \delta_i^{E-}\big), \qquad \mathcal{L}_H = \frac{1}{|\mathcal{K}_H|} \sum_{i \in \mathcal{K}_H} \mathrm{softplus}\big(\delta_i^+ - \delta_i^{H-}\big),$$

with $\mathcal{L}_E = 0$ if $\mathcal{K}_E = \varnothing$ and $\mathcal{L}_H = 0$ if $\mathcal{K}_H = \varnothing$. Gradients flow through all three passes, so a utility loss can decrease either by improving denoising with the example's own input or by increasing the error with the borrowed one; $\mathcal{L}_{\text{diff}}$ anchors denoising with the example's own inputs.

C.4 Reverse Diffusion and Probabilistic Sampling

At inference, the observed history and event set are fixed throughout reverse diffusion. For each generated trajectory s, we initialize

$$\mathbf{Y}_N^{(s)} \sim \mathcal{N}(\mathbf{0}, I).$$

At reverse step n, the denoiser predicts

$$\hat{\boldsymbol{\epsilon}}_n^{(s)} = \epsilon_\theta\Big(\mathbf{Y}_n^{(s)}, n, \mathbf{X}_{1:T}, E\Big),$$

and the DDPM reverse mean is

$$\boldsymbol{\mu}_\theta\Big(\mathbf{Y}_n^{(s)}, n\Big) = \frac{1}{\sqrt{\alpha_n}} \left[\mathbf{Y}_n^{(s)} - \frac{\beta_n}{\sqrt{1-\bar{\alpha}_n}} \hat{\boldsymbol{\epsilon}}_n^{(s)} \right].$$

For $n > 1$,

$$\mathbf{Y}_{n-1}^{(s)} = \boldsymbol{\mu}_\theta\Big(\mathbf{Y}_n^{(s)}, n\Big) + \sqrt{\widetilde{\beta}_n}\, \mathbf{z}, \qquad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, I), \qquad \widetilde{\beta}_n = \beta_n \frac{1-\bar{\alpha}_{n-1}}{1-\bar{\alpha}_n}.$$

At the final step, $n = 1$, we take the reverse mean without added noise, giving the trajectory $\hat{\mathbf{Y}}^{(s)} = \boldsymbol{\mu}_\theta(\mathbf{Y}_1^{(s)}, 1)$.

Because the event text does not change during reverse diffusion, its text representation is computed once per batch and reused across trajectories and diffusion steps. We generate 100 trajectories for each test input. Point forecasts, used for MSE and MAE, are the mean of these trajectories.

C.5 Optimization and Training Details

All components are optimized jointly. We use AdamW with weight decay 10^{-4}. The pretrained MiniLM parameters use learning rate 2×10^{-5}, while all remaining trainable parameters use learning rate 2×10^{-4}. Training uses minibatches of size 128, gradient clipping at 1.0, and mixed-precision computation.

	Train		Validation		Test		Total	
Event Type	**Episode Count**	**Window Count**	**Episode Count**	**Window Count**	**Episode Count**	**Window Count**	**Episode Count**	**Window Count**
Rain	2,604	1,576	541	317	605	334	3,750	2,227
Mist	2,189	1,376	504	312	531	351	3,224	2,039
Snow	1,197	740	207	116	256	152	1,660	1,008
Haze	747	576	175	131	220	183	1,142	890
Drizzle	674	467	218	161	170	132	1,062	760
Thunderstorm	295	240	195	140	205	150	695	530
Fog	304	243	92	73	88	76	484	392
Total	**8,010**	**—**	**1,932**	**—**	**2,075**	**—**	**12,017**	**—**

Table 4: Distribution of future weather event types across the train, validation, and test splits of Traffic Volume. Episode counts denote individual event occurrences, while window counts denote forecasting windows containing each event type.

We train for at most 1000 epochs on all datasets. Every 10 epochs, we generate 100 trajectories per validation input, using a fixed random seed so that all checkpoints are compared on the same sampling noise, and compute the MSE of the trajectory mean and the empirical CRPS. Training stops early when neither validation MSE nor validation CRPS has improved for 50 epochs. We evaluate the checkpoint with the lowest validation CRPS on the test split. **The test split is not used for optimization or checkpoint selection.** All experiments done in A6000 GPU.

D Dataset Description

D.1 Traffic Volume

We begin with the original UCI Metro Interstate Traffic Volume data, which provides hourly traffic measurements together with timestamps and weather information. We first order the observations chronologically by timestamp, remove duplicate timestamps, and handle missing values in the fields used downstream. From this ordered hourly sequence, each forecasting example is constructed as a 36-hour window: the first 24 hours form the observed history, and the following 12 hours form the prediction target. Consecutive windows are generated with a stride of 6 hours.

We split the timeline chronologically before constructing these windows. The earliest 70% of the hourly observations are assigned to training, the following 15% to validation, and the final 15% to testing. The 24-hour history and 12-hour future windows are then generated independently within each split, rather than constructing windows first and subsequently dividing them. This ensures that no forecasting window crosses a train, validation, or test boundary.

Finally, we construct the event descriptions from the weather observations associated with each window. We treat `Clear` and `Clouds` as non-events and identify each contiguous run of the same remaining `weather_main` category as a weather event. Only events falling in the 12-hour future portion are described; if an event crosses the history–future boundary, only its future portion is described. Events in the history portion are not provided to any model. For the clock-time representation, each event is expressed using its weather type together with its observed start and end times. A single-hour event is represented, for example, as "Rain at Saturday 03:00," while a longer event is represented as "Rain from Saturday 03:00 to Saturday 06:00." Thus, the text preserves the actual clock time and day of week associated with the event rather than replacing time with relative indices such as $T = 1$ or $T = 2$.Additional dataset statistics, including the number of forecasting windows and event episodes in each split, are reported in Table 4.

D.2 Energy Usage

We construct the EWELD dataset directly from the raw release, using the electricity-consumption, weather, extreme-weather, and user-location files. Each electricity series is associated with its corresponding city, and the city-level weather and extreme-weather records are aligned to the electricity series using their timestamps. The data are recorded at 15-minute intervals. We use the extreme-

	Train		Test		Validation		Total	
Event Type	**Episode Count**	**Window Count**	**Episode Count**	**Window Count**	**Episode Count**	**Window Count**	**Episode Count**	**Window Count**
Low temperature	8,730	8,595	1,829	1,820	1,733	1,719	12,292	12,134
High temperature	25,824	25,823	5,681	5,681	5,626	5,626	37,131	37,130
High humidity	27,440	27,166	5,064	4,855	3,540	3,427	36,044	35,448
High heat and humidity	1,588	1,588	677	677	384	384	2,649	2,649
Tropical Storm	4,557	4,557	604	604	489	489	5,650	5,650
Severe Tropical Storm	1,469	1,469	381	380	133	133	1,983	1,982
Typhoon	385	385	269	269	177	177	831	831
Total	**69,993**	**—**	**14,505**	**—**	**12,082**	**—**	**96,580**	**—**

Table 5: Energy Usage dataset statistics, showing the number of forecasting windows and extreme-weather event episodes in the training, validation, and test splits. Treatment categories are non-exclusive, as a window may contain multiple treatment types.

	Train		Validation		Test		Total	
Event Type	**Episode Count**	**Window Count**	**Episode Count**	**Window Count**	**Episode Count**	**Window Count**	**Episode Count**	**Window Count**
Chemotherapy	122,646	32,087	30,840	7,231	28,041	6,988	181,527	46,306
Radiotherapy	125,093	32,157	30,736	7,256	27,804	6,951	183,633	46,364
Chemotherapy + Radiotherapy	67,599	24,028	17,766	5,488	15,269	5,162	100,634	34,678
Total	**180,140**	**—**	**43,810**	**—**	**40,576**	**—**	**264,526**	**—**

Table 6: Distribution of future treatment events across the CRN train, validation, and test splits. Episode counts report the number of treatment occurrences, while window counts report the number of forecasting windows containing each treatment type. Treatment categories are non-exclusive, as a window may contain multiple treatment types.

weather event types provided by EWELD, including temperature and humidity extremes, severe thunderstorms, tropical storms and typhoons, heavy rain and thunderstorms, and sleet events. Each event is represented as a contiguous episode with a start and end time and is aligned to the timestamps of the corresponding electricity series. For each user, the time series is split chronologically into 70% training, 15% validation, and 15% test data before constructing forecasting windows. Within each split, we form non-overlapping windows of 144 time steps, corresponding to 36 hours. The first 96 steps, or 24 hours, form the history, and the following 48 steps, or 12 hours, form the prediction horizon. We use a stride of 144, so consecutive samples do not overlap and windows never cross train, validation, or test boundaries. For event conditioning, only events falling in the 48-step prediction horizon are described. Each event description contains the event type and definition together with its start and end times. Event timing is expressed using the day of the week and clock time, for example, "from Monday 14:30 to Monday 17:45." Additional dataset statistics, including the number of forecasting windows and event episodes in each split, are reported in Table 5.

D.3 Tumor Volume Simulator

We build the Tumor Volume dataset from the cancer simulator introduced with the Counterfactual Recurrent Network (CRN) (Bica et al., 2020). The simulator uses a pharmacokinetic–pharmacodynamic model of tumour growth for non-small-cell lung cancer, in which tumour volume evolves under the effects of Vinblastine chemotherapy and radiotherapy. In the original CRN formulation, chemotherapy corresponds to administering 5.0 mg/m^3 of Vinblastine and radiotherapy to administering a 2.0 Gy fraction, with treatment assignment probabilities determined by the patient's recent tumour trajectory.

We retain the original CRN patient-generation procedure, tumour-growth dynamics, chemotherapy concentration decay, and tumour-dependent treatment-assignment mechanism, but modify how treatment trajectories are generated. For each synthetic patient, we first generate a 30-timestep history during which no treatment is administered. From this common untreated history, we then generate alternative treatment futures. To introduce variation in treatment intensity, we extend the original treatment amounts to three Vinblastine doses (2, 5, and 7 mg/m^3) and three radiotherapy doses (1, 2, and 3 Gy). At the first future timestep, the patient may receive no treatment, Vinblastine chemotherapy at one of the three doses, radiotherapy at one of the three doses, or one of the nine combinations of chemotherapy and radiotherapy. After this initial treatment, the original CRN tumour-dependent treatment probabilities continue to determine whether treatment is administered, while our generation settings constrain which treatment modalities and dose levels may subsequently occur. These settings allow treatment intensity to remain fixed, decrease, or increase; preserve the initial treatment modality or treatment components; or progressively relax these restrictions. Additional statistics on the distribution of future treatment events across the train, validation, and test splits are reported in Table 6.

E Baselines: Implementation Details

E.1 TADiff

We implement TADiff (Gu et al., 2026) using the authors' released code and retain its diffusion architecture, text-attribution mechanism, training objectives, counterfactual fine-tuning procedure, and two-stage inference process. The original formulation assumes a historical time series x_h, a textual condition describing the history c_h, and a textual condition describing the future c_f. Our setting provides the numerical history and the future exogenous-event description, but no separate historical-condition or metadata text. We therefore provide the observed history time series unchanged, map our future-event description to TADiff's future-condition input, and leave the historical-condition text empty for every example. We do not construct a surrogate historical description from the time series or provide any additional metadata.

Following the original architecture, the historical time series and noisy future are separately patchified and concatenated before being processed by the diffusion noise estimator. The future-event text is tokenized using the LongCLIP tokenizer and encoded by the model's two-layer Transformer text encoder. Its representation is combined with the diffusion-step embedding and used to condition the DiT-style Transformer noise estimator. During forecasting training, the historical portion remains clean while Gaussian noise is applied to the future portion, and the forecasting loss is evaluated only over the future timesteps.

We retain TADiff's joint forecasting and attribution training rather than removing the attribution branch. The forecasting objective trains the model to recover noise from the future conditioned on the observed numerical history and future-event text, while the attribution objective operates on the historical sequence. Because our benchmark contains no historical-condition text, the attribution branch receives the same empty historical-text input for every example. All remaining components of the original training procedure are retained. Following the authors' configuration, we train the initial TADiff model for 700 epochs with a forecasting-to-attribution loss-weight ratio of 2:1, and subsequently apply the released counterfactual fine-tuning procedure. Training is performed on a single NVIDIA RTX A6000 GPU.

At inference, we retain TADiff's original two-stage procedure. The observed historical sequence is first passed through the attribution forward process using the empty historical-text condition to obtain the history-derived diffusion state. This state is then used to initialize the future portion of the reverse diffusion process. The model subsequently denoises the concatenated history–future sequence while conditioning on the supplied future-event description, replacing the historical portion with the observed history after each reverse-diffusion step. The final future portion is retained as the forecast. For each test example, we independently run the stochastic inference procedure 100 times to obtain 100 predictive trajectories, matching the common probabilistic evaluation protocol used across our experiments. All inference is performed on a single NVIDIA RTX A6000 GPU.

E.2 AURORA

We evaluate Aurora using the released pretrained checkpoint in a zero-shot setting and do not train or fine-tune the model on any of our datasets. Consistent with our problem formulation, each example provides only the observed numerical history $X_{1:T}$ and the natural-language description of the exogenous event occurring in the forecast horizon E. We map E directly to Aurora's native text-conditioning interface and do not provide any separate history text, metadata or domain description, channel description, or textual summary of the observed trajectory. We also do not provide an external image. When no image is supplied, the released Aurora implementation constructs its endogenous pseudo-image directly from the same numerical history by estimating a dominant period, reshaping the history into a two-dimensional periodic representation, resizing it, and repeating it across three channels; consequently, this branch introduces no information beyond $X_{1:T}$.

Aurora first normalizes the numerical history using its RevIN-style instance normalization and partitions it into non-overlapping temporal patches, which are linearly embedded and processed by the temporal Transformer. The future-event description is processed by Aurora's BERT-based text encoder. In the released implementation, the BERT token representations exclude the `[CLS]`, `[SEP]`, and padding tokens, are truncated or padded to at most 125 valid token representations, projected to the model hidden dimension, and distilled through learned cross-attention queries into 10 text tokens. The BERT encoder is part of the released pretrained Aurora model and remains fixed in our zero-shot evaluation. Aurora then uses the distilled text representation together with the history-derived visual representation to guide temporal self-attention and fuses the resulting modality-specific information with the encoded time-series representation.

We retain Aurora's released architecture and generative procedure without modification. The base configuration uses a hidden dimension of 256, eight attention heads, a 512-dimensional feed-forward layer, one temporal encoder layer, nine decoder layers, dropout 0.2, a reference token length of 48, 10 distilled modality tokens, 1,000 learned prototypes, a three-layer flow-matching network, and 50 sampling steps. The decoder constructs future-token conditions from the fused representation, while the prototype retriever uses the multimodal representation to form weighted combinations of learned trend and periodic prototypes that initialize prototype-guided flow matching.

At test time, we perform inference only with the pretrained model; no parameter updates, adaptation, retraining, or fine-tuning are performed. For each test example, Aurora is queried with the observed history and its corresponding future-event text, and we request 100 generative samples, yielding 100 forecast trajectories for that test example. All Aurora inference runs are performed on a single NVIDIA RTX A6000 GPU.

E.3 IATSF

We implement IATSF using the FIATS architecture, retaining the original forecasting architecture and training procedure while adapting its textual inputs to the information available in our setting. Each example contains only the observed numerical history $X_{1:T}$ and the future exogenous-event description(s) E; we do not provide historical text, channel descriptions, variable names, or any additional metadata. The numerical history is processed by FIATS's patch-based time-series encoder, while the future event descriptions are embedded with the released `paraphrase-MiniLM-L6-v2` text encoder (384-dimensional embeddings) and supplied through FIATS's intervention ("news") pathway.

All remaining components of FIATS are retained unchanged. The intervention encoder uses cross-attention to construct an event-conditioned representation, which is subsequently used by the CAPS decoder to query the patch-encoded historical time series and produce future patch representations. These are mapped to numerical predictions with the original token-wise linear decoder; overlapping predicted patches are averaged, and the model's original normalization and de-normalization steps are preserved. Since our setting contains a single target channel, the channel-aware operations reduce to the single-channel case, but their computation is otherwise unchanged. We train the complete model end-to-end using the released MSE forecasting objective, Adam optimization, and validation-based early stopping.

FIATS is deterministic and therefore produces a single trajectory for a given input. To obtain probabilistic forecasts under our common evaluation protocol, we use a lead-time-specific residual boot-

strap. For each forecast lead, we compute residuals on the validation set and, at inference time, sample residuals from the corresponding lead and add them to the deterministic test prediction. We repeat this procedure to construct 100 predictive trajectories for each test example. Inference is repeated 10 times, and we report the mean and standard deviation across repetitions. All experiments were performed on an NVIDIA RTX A6000 GPU.

E.4 EVENTTSF

We use the released implementation of EventTSF (Ge et al., 2026)and retain its autoregressive flow-matching forecasting procedure, event-conditioned diffusion schedule, model architecture, and training objective. The original EventTSF formulation conditions on historical time-series/event pairs together with future event descriptions. In our setting, however, only the numerical history $X_{1:T}$ and the future exogenous-event text are available; historical event text, channel descriptions, and other metadata text are not provided. We therefore retain the observed numerical history unchanged while replacing the historical textual conditions expected by EventTSF with null event conditions. The numerical history is still processed autoregressively to construct the model's latent historical state, so no time-series information is removed. The only non-null textual conditioning supplied to the model is the future exogenous-event description associated with the forecast horizon.

We otherwise preserve EventTSF's original segment-wise autoregressive training procedure. The time series is divided into non-overlapping segments and processed sequentially, with the latent state updated after each segment. During forecasting, future segments are generated autoregressively using the flow-matching denoiser conditioned on the evolving latent state and the supplied future-event embedding. The future event description is encoded using EventTSF's text-conditioning pipeline and is also used by its event-aware timestep mechanism to modulate the flow-matching process; we introduce no additional structured event attributes or textual metadata. We use the released 15-layer DiT architecture with a 48-dimensional latent state, 256-dimensional hidden representation, 128-dimensional event condition, and 8 attention heads. Training follows the released configuration using AdamW with an initial learning rate of 2×10^{-5}, peak learning rate of 2×10^{-4}, weight decay of 10^{-3}, and $(\beta_1, \beta_2) = (0.9, 0.999)$, together with the original scheduled teacher-forcing procedure and conditional flow-matching objective. At inference, the latent state is first constructed from the observed numerical history under the null historical-event condition, after which the forecast is generated autoregressively while conditioning only on the supplied future-event text. For each test example, we independently generate 100 predictive trajectories using fresh Gaussian noise and use these samples for the common probabilistic evaluation. All EventTSF training and inference experiments are run on an NVIDIA RTX A6000 GPU.

E.5 CLEF

CLEF is a deterministic, pointwise predictor: rather than generating the complete L-step future jointly, it predicts the outcome at one requested future timestamp t_{pred}. In our univariate setting, each CLEF query therefore produces a single scalar prediction. We provide CLEF with the numerical history $X_{1:T}$ and the future-event description E; no historical-event text, metadata text, or channel descriptions are supplied. Following CLEF, the condition is first mapped to an embedding using a frozen pretrained embedding model and then projected through CLEF's condition adapter, while the numerical history is encoded independently by the sequence encoder. CLEF additionally represents the temporal displacement between the end of the observed history and the requested prediction time and combines the history, condition, and prediction-time representations in its concept encoder before producing the requested future value.

For a forecasting horizon of length $L > 1$, we convert each forecasting window into L point-prediction training queries. Specifically, for the same observed history $X_{1:T}$ and the same future-event description E, we vary only the requested prediction time,

$$t_{\text{pred}} \in \{T+1, T+2, \ldots, T+L\},$$

and train CLEF against the corresponding observed value $X_{t_{\text{pred}}}$. Thus, the history and event conditioning remain fixed across the L queries, while t_{pred} and its supervision target change. At inference, we query the trained model in the same manner for $T+1, \ldots, T+L$ and concatenate the resulting scalar predictions to obtain the deterministic L-step forecast.

Because CLEF produces deterministic point forecasts, we construct predictive distributions using a lead-time-specific validation-residual bootstrap. After training, we run CLEF on the validation set and, separately for each forecast lead ℓ, collect the residuals

$$\mathcal{R}_\ell = \left\{ X^{(i)}_{T+\ell} - \hat{X}^{(i)}_{T+\ell} \right\}_{i \in \mathcal{V}}.$$

For a test example, CLEF first produces its deterministic forecast

$$\hat{X}_{T+1:T+L}.$$

To generate one predictive trajectory, at each lead ℓ we sample a residual from the corresponding validation residual pool $\mathcal{R}_\ell$ and add it to the point prediction,

$$\tilde{X}^{(s)}_{T+\ell} = \hat{X}_{T+\ell} + r^{(s)}_\ell, \qquad r^{(s)}_\ell \sim \mathcal{R}_\ell.$$

We repeat this procedure to obtain 100 predictive trajectories per test example. The bootstrap therefore adds empirical uncertainty estimated from validation errors at the same forecast lead, rather than treating CLEF itself as a probabilistic model. All training and inference experiments are run on an NVIDIA RTX A6000 GPU.

F Forecasting Factual Trajectories: Metrics

Let N denote the number of test examples, L the forecast horizon, and d the number of target variables. For each test example i, the model produces $S = 100$ predictive trajectories, $\{\hat{\mathbf{Y}}^{(s)}_i\}_{s=1}^S$, where $\hat{\mathbf{Y}}^{(s)}_i \in \mathbb{R}^{L \times d}$, and the corresponding ground-truth future is $\mathbf{Y}_i \in \mathbb{R}^{L \times d}$.

Point forecasts. For MSE and MAE, we first average the S predictive trajectories to obtain the mean forecast

$$\bar{\mathbf{Y}}_i = \frac{1}{S} \sum_{s=1}^{S} \hat{\mathbf{Y}}^{(s)}_i.$$

Writing $y_{i,t,k}$ and $\bar{y}_{i,t,k}$ for target variable k at forecast step t, we compute

$$\mathrm{MSE} = \frac{1}{NLd} \sum_{i=1}^{N} \sum_{t=1}^{L} \sum_{k=1}^{d} \left(\bar{y}_{i,t,k} - y_{i,t,k} \right)^2,$$

and

$$\mathrm{MAE} = \frac{1}{NLd} \sum_{i=1}^{N} \sum_{t=1}^{L} \sum_{k=1}^{d} \left| \bar{y}_{i,t,k} - y_{i,t,k} \right|.$$

Continuous Ranked Probability Score (CRPS). CRPS evaluates the full predictive distribution rather than only its mean. For the empirical distribution represented by the $S = 100$ predictive samples, the CRPS at a single forecast coordinate (i, t, k) is

$$\mathrm{CRPS}_{i,t,k} = \frac{1}{S} \sum_{s=1}^{S} \left| \hat{y}^{(s)}_{i,t,k} - y_{i,t,k} \right| - \frac{1}{2S^2} \sum_{s=1}^{S} \sum_{s'=1}^{S} \left| \hat{y}^{(s)}_{i,t,k} - \hat{y}^{(s')}_{i,t,k} \right|.$$

We average this score over all test examples, forecast steps, and target variables:

$$\mathrm{CRPS} = \frac{1}{NLd} \sum_{i=1}^{N} \sum_{t=1}^{L} \sum_{k=1}^{d} \mathrm{CRPS}_{i,t,k}.$$

Weighted Quantile Loss (WQL). We compute empirical predictive quantiles from the same $S = 100$ samples at the nine quantile levels

$$\mathcal{Q} = \{0.1, 0.2, \ldots, 0.9\}.$$

Let

$$\hat{q}^{(\tau)}_{i,t,k} = \mathrm{Quantile}_\tau \left(\left\{ \hat{y}^{(s)}_{i,t,k} \right\}_{s=1}^{S} \right)$$

denote the empirical τ-quantile of the predictive samples. The pinball loss at quantile level τ is

$$\rho_\tau(y, q) = \begin{cases} \tau(y - q), & y \geq q, \\ (1 - \tau)(q - y), & y < q. \end{cases}$$

For each quantile, the weighted quantile loss is

$$\mathrm{wQL}_\tau = \frac{2 \sum_{i=1}^{N} \sum_{t=1}^{L} \sum_{k=1}^{d} \rho_\tau \left(y_{i,t,k}, \hat{q}^{(\tau)}_{i,t,k} \right)}{\sum_{i=1}^{N} \sum_{t=1}^{L} \sum_{k=1}^{d} |y_{i,t,k}|}.$$

We report the mean weighted quantile loss across the nine quantile levels:

$$\mathrm{WQL} = \frac{1}{|\mathcal{Q}|} \sum_{\tau \in \mathcal{Q}} \mathrm{wQL}_\tau, \qquad |\mathcal{Q}| = 9.$$

Lower values indicate better performance for all four metrics.

G Forecasting Counterfactual Trajectories: Metrics

Setup. For query case i, let $H_i \in \mathbb{R}^L$ be the pre-event history, E_i the factual event, E_i' the alternate event, and $Y_i \in \mathbb{R}^T$ the observed post-event trajectory. Both arms are forecast from the same history, $\hat{Y}_i = f(H_i, E_i)$ and $\hat{Y}_i' = f(H_i, E_i')$; for probabilistic models, $\hat{Y}_i$ is the mean over S samples. The model-induced response is

$$U_i = \hat{Y}_i' - \hat{Y}_i \in \mathbb{R}^T.$$

On Tumor Volume, the simulator provides, for each history, a realized future under every event; each pair of futures with distinct events yields an evaluation pair, with one arm as (E_i, Y_i) and the other as $(E_i', Y_i^{\mathrm{ref}})$, so the reference is exact. On EWELD and Metro, only one arm is realized; we retrieve the donor $j(i) = \arg\min_{j: E_j = E_i'} d_{\mathrm{hist}}(H_i, H_j)$ and set $V_i = Y_{j(i)} - Y_i$, an imputed reference rather than ground truth. $\varepsilon > 0$ is a small stabilizer. A model that ignores the event text yields $U_i = 0$ regardless of its forecast accuracy, so all metrics below isolate whether the model uses the event condition.

ResponseMSE and ResponseMAE.

$$\mathrm{ResponseMSE} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{L} \sum_{t=1}^{L} (U_{i,t} - V_{i,t})^2, \qquad \mathrm{ResponseMAE} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{L} \sum_{t=1}^{L} |U_{i,t} - V_{i,t}|,$$

These ask whether the predicted response is wrong, and by how much: the error is measured on the *effect* of the event, not on the forecast level. Lower is better; ideal 0. The null baseline $U_i = 0$ gives $\frac{1}{NT} \sum_i \|V_i\|_2^2$ and $\frac{1}{NT} \sum_i \|V_i\|_1$; a model scoring above these is worse than one that ignores the event. ResponseMSE is the squared Precision in Estimation of Heterogeneous Effects (PEHE) of Hill (2011) and Shalit et al. (2017); prior work typically reports its square root. On the observational datasets we impute the missing arm from a nearest neighbour, following NN-PEHE (Schwab et al., 2018), which was proposed for model selection when counterfactual outcomes are unobserved. We extend PEHE from scalar to T-dimensional outcomes; per-timestep imputation errors are therefore correlated within a case, and we do not treat the $N \times T$ entries as independent when reporting variability.

Directional Alignment (DA).

$$\mathrm{DA} = \frac{1}{N} \sum_{i=1}^{N} \frac{\langle U_i, V_i \rangle}{\|U_i\|_2 \, \|V_i\|_2 + \varepsilon}.$$

DA asks whether the model moves in the right *direction*. It is the cosine between the predicted and reference responses: 1 if the response has the right shape, 0 if unrelated, -1 if opposite. It is

Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
TADiff	$\mathbf{0.167}_{\pm 0.009}$	$\mathbf{0.165}_{\pm 0.005}$	$\underline{0.158}_{\pm 0.004}$	$\underline{0.363}_{\pm 0.010}$
Aurora	$0.854_{\pm 0.034}$	$0.352_{\pm 0.008}$	$0.343_{\pm 0.007}$	$0.777_{\pm 0.016}$
IATSF	$2.872_{\pm 0.112}$	$1.684_{\pm 0.029}$	$0.789_{\pm 0.014}$	$1.773_{\pm 0.032}$
EventTSF	$1.019_{\pm 0.044}$	$0.944_{\pm 0.020}$	$0.793_{\pm 0.018}$	$1.800_{\pm 0.042}$
CLEF	$0.686_{\pm 0.016}$	$0.391_{\pm 0.005}$	$0.297_{\pm 0.003}$	$0.667_{\pm 0.007}$
MEETS	$\underline{0.252}_{\pm 0.012}$	$\underline{0.170}_{\pm 0.005}$	$\mathbf{0.122}_{\pm 0.003}$	$\mathbf{0.274}_{\pm 0.006}$

Table 7: Unseen chemotherapy intensity, low→high, on Tumor Volume. Models are trained on futures with chemotherapy at 2 and 5 mg/m^3 and tested on every treatment configuration that administers 7 mg/m^3, alone or with radiotherapy. Mean ± standard deviation over 10 inference repetitions; lower is better. TADiff has the lowest MSE and MAE, `MEETS` the lowest CRPS and WQL

scale-invariant by design: if the reference response is a drop of 400 vehicles, predicted drops of 4, 400, and 4000 all score 1. DA alone is therefore insufficient, which motivates DRS. DA adapts the directional CLIP loss of StyleGAN-NADA (Gal et al., 2021), $1 - \cos(\Delta I, \Delta T)$, which Gal et al. minimize during training so that the change in a generated image's CLIP embedding aligns with the requested change in text embedding. We use the cosine itself as an evaluation score and replace CLIP embedding differences with trajectory differences U_i and V_i.

Directional Response Score (DRS).

$$\mathrm{DRS} = \frac{1}{N} \sum_{i=1}^{N} \frac{\langle U_i, V_i \rangle}{\|V_i\|_2^2 + \varepsilon}.$$

DRS asks whether the model moves the right *amount*, and which way it errs. Per case, it is the through-origin least-squares slope of U_i on V_i, i.e., how far along the reference direction the model travelled, in units of the reference; in the example above, the three predictions score 0.01, 1, and 10. Ideal 1; $\mathrm{DRS} < 1$ means the model under-reacts, $\mathrm{DRS} > 1$ over-reacts, ≈ 0 no movement along the reference, and negative opposed movement. As $\varepsilon \to 0$ it factors as $\mathrm{DRS}_i = (\|U_i\|_2 / \|V_i\|_2) \cdot \mathrm{DA}_i$, amplitude times direction.

DRS is analogous to the calibration slope (COX, 1958; Van Calster et al., 2016) and the Mincer–Zarnowitz slope (Mincer, 1969), both of which take slope 1 as optimal. It departs from them in two ways. First, the Mincer–Zarnowitz regression includes an intercept and the calibration slope is fit on the logit scale for binary outcomes, whereas DRS is a through-origin linear slope. Second, those literatures regress truth on prediction; we regress prediction on reference. The two slopes are not reciprocals (their product is DA_i^2), so the transposition changes the estimand, and the under/over-reaction reading above holds only in this orientation. We transpose deliberately: with $\|V_i\|_2^2$ in the denominator, a model that barely responds scores near 0 instead of inflating the metric, and near-degenerate cases can be identified from the data before any model is evaluated. The cost is that noise in V_i enters the regressor, which biases DRS toward 0 on the observational datasets. The mean is sensitive to small $\|V_i\|_2$; ε in the denominator bounds each term, and we apply no low-effect exclusion.

Effect of imputation noise. On Energy Usage and Traffic Volume, write $V_i = V_i^\star + \eta_i$, where $V_i^\star$ is the unobserved true response and η_i is the donor-imputation error, assumed zero-mean with per-timestep variance σ^2 and independent of U_i. Three consequences follow. ResponseMSE acquires an additive floor of σ^2, so its ideal of 0 is unattainable, and the null baseline is inflated by the same amount. DA tends toward 0. The DRS numerator is unbiased, but $\mathbb{E}\|V_i\|_2^2 = \|V_i^\star\|_2^2 + T\sigma^2$, so DRS is attenuated by approximately $\|V_i^\star\|_2^2 / (\|V_i^\star\|_2^2 + T\sigma^2)$. An observational DRS below 1 therefore does not by itself indicate under-reaction. The simulator, where V_i is exact, provides the unattenuated reference and vice versa.

H Out-of-Distribution Generalization Results

All eleven out-of-distribution (OOD) splits are built on the Tumor Volume simulator, the only one of our three datasets that provides a realized future for every treatment configuration applied to

Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
TADiff	$\underline{0.128}_{\pm 0.006}$	$\underline{0.159}_{\pm 0.003}$	$\underline{0.112}_{\pm 0.003}$	$\underline{0.256}_{\pm 0.007}$
Aurora	$0.930_{\pm 0.034}$	$0.382_{\pm 0.006}$	$0.373_{\pm 0.007}$	$0.855_{\pm 0.015}$
IATSF	$1.655_{\pm 0.037}$	$1.098_{\pm 0.016}$	$0.977_{\pm 0.014}$	$2.232_{\pm 0.031}$
EventTSF	$1.428_{\pm 0.031}$	$0.846_{\pm 0.010}$	$0.728_{\pm 0.009}$	$1.665_{\pm 0.021}$
CLEF	$0.186_{\pm 0.008}$	$0.213_{\pm 0.005}$	$0.158_{\pm 0.004}$	$0.362_{\pm 0.010}$
MEETS	$\mathbf{0.094}_{\pm 0.002}$	$\mathbf{0.121}_{\pm 0.001}$	$\mathbf{0.088}_{\pm 0.001}$	$\mathbf{0.201}_{\pm 0.002}$

Table 8: Unseen radiotherapy intensity, low→high, on Tumor Volume. Models are trained on futures with radiotherapy at 1 and 2 Gy and tested on every treatment configuration that administers 3 Gy, alone or with chemotherapy. Mean ± standard deviation over 10 inference repetitions; lower is better. MEETS is best on all four metrics, and one ranking holds on every metric: MEETS, TADiff, CLEF, Aurora, EventTSF, IATSF.

Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
TADiff	$\mathbf{0.199}_{\pm 0.005}$	$\mathbf{0.149}_{\pm 0.002}$	$\mathbf{0.118}_{\pm 0.002}$	$\mathbf{0.215}_{\pm 0.003}$
Aurora	$0.274_{\pm 0.014}$	$0.233_{\pm 0.005}$	$0.224_{\pm 0.005}$	$0.412_{\pm 0.009}$
EventTSF	$0.484_{\pm 0.012}$	$0.621_{\pm 0.009}$	$0.376_{\pm 0.005}$	$0.683_{\pm 0.010}$
IATSF	$1.029_{\pm 0.042}$	$0.824_{\pm 0.018}$	$0.708_{\pm 0.017}$	$1.296_{\pm 0.032}$
CLEF	$0.294_{\pm 0.009}$	$0.536_{\pm 0.008}$	$0.391_{\pm 0.007}$	$0.716_{\pm 0.013}$
MEETS	$\underline{0.211}_{\pm 0.008}$	$\underline{0.154}_{\pm 0.003}$	$\underline{0.146}_{\pm 0.003}$	$\underline{0.270}_{\pm 0.006}$

Table 9: Unseen chemotherapy intensity, high→low, on Tumor Volume. Models are trained on futures with chemotherapy at 5 and 7 mg/m^3 and tested on every treatment configuration that administers 2 mg/m^3, alone or with radiotherapy. Mean ± standard deviation over 10 inference repetitions; lower is better. TADiff is best and MEETS second on all four metrics, the only split where MEETS leads on none.

the same history. Each simulated patient has a 30-step untreated history (timestamps 1–30). This history is branched at timestamp 31 into 15-step futures (timestamps 31–45) under sixteen treatment configurations: no treatment, Vinblastine chemotherapy at 2, 5, or 7 mg/m^3, radiotherapy at 1, 2, or 3 Gy, and the nine combinations that pair one chemotherapy dose with one radiotherapy dose. Every administered treatment is written as its own timestamped event string. An untreated future is described by a single null event stating that no treatment was provided between timestamps 31 and 45.

Each split partitions the treatment configurations along one event attribute (intensity, interaction, or timing), so that the attribute value seen at test time never appears in training. Training, validation, and test histories come from disjoint patients, exactly as in the main Tumor Volume split. The validation set is drawn from the training configurations, so checkpoint selection never sees the held-out attribute value. MEETS and the four trained baselines (TADiff, IATSF, EventTSF, CLEF) are retrained from scratch on each split; Aurora is evaluated zero-shot with the released checkpoint, as in Section 5.1.

Evaluation otherwise follows Section 5.1. Each model produces 100 trajectories per test input, with IATSF and CLEF obtaining theirs through the validation-residual bootstrap. MSE and MAE are computed on the mean trajectory, and CRPS and WQL on the full sample set. Every table reports the mean and standard deviation over 10 inference repetitions, and lower is better for all four metrics.

H.1 Low-to-High Intensity

The two low-to-high splits require each model to forecast under a dose larger than any it saw during training. In the chemotherapy split, the training futures administer chemotherapy only at 2 and 5 mg/m^3. Every configuration that administers 7 mg/m^3, alone or together with radiotherapy, is removed from training and makes up the test set. Three kinds of futures stay in training: radiotherapy-only futures at all three doses, combinations that pair radiotherapy with 2 or 5 mg/m^3 of chemotherapy, and untreated futures.

Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
Aurora	$\underline{0.149}_{\pm 0.005}$	$\underline{0.153}_{\pm 0.002}$	$\underline{0.120}_{\pm 0.002}$	$\underline{0.236}_{\pm 0.004}$
CLEF	$0.306_{\pm 0.011}$	$0.186_{\pm 0.003}$	$0.178_{\pm 0.004}$	$0.345_{\pm 0.007}$
IATSF	$0.457_{\pm 0.015}$	$0.322_{\pm 0.006}$	$0.321_{\pm 0.006}$	$0.616_{\pm 0.011}$
EventTSF	$0.382_{\pm 0.015}$	$0.267_{\pm 0.006}$	$0.238_{\pm 0.005}$	$0.472_{\pm 0.011}$
TADiff	$0.214_{\pm 0.004}$	$0.171_{\pm 0.002}$	$0.128_{\pm 0.001}$	$0.253_{\pm 0.003}$
MEETS	$\mathbf{0.104}_{\pm 0.003}$	$\mathbf{0.128}_{\pm 0.001}$	$\mathbf{0.094}_{\pm 0.001}$	$\mathbf{0.187}_{\pm 0.003}$

Table 10: Unseen radiotherapy intensity, high→low, on Tumor Volume. Models are trained on futures with radiotherapy at 2 and 3 Gy and tested on every treatment configuration that administers 1 Gy, alone or with chemotherapy. Mean ± standard deviation over 10 inference repetitions; lower is better. MEETS is best and Aurora second on all four metrics, Aurora's best placing on any dose split.

Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
TADiff	$\underline{0.268}_{\pm 0.011}$	$\underline{0.144}_{\pm 0.003}$	$0.117_{\pm 0.003}$	$\underline{0.262}_{\pm 0.006}$
Aurora	$0.815_{\pm 0.038}$	$0.148_{\pm 0.004}$	$\underline{0.115}_{\pm 0.003}$	$0.263_{\pm 0.007}$
EventTSF	$0.732_{\pm 0.033}$	$0.222_{\pm 0.005}$	$0.213_{\pm 0.005}$	$0.482_{\pm 0.010}$
IATSF	$0.993_{\pm 0.038}$	$0.667_{\pm 0.012}$	$0.427_{\pm 0.007}$	$0.965_{\pm 0.016}$
CLEF	$0.284_{\pm 0.008}$	$0.183_{\pm 0.003}$	$0.141_{\pm 0.002}$	$0.323_{\pm 0.005}$
MEETS	$\mathbf{0.204}_{\pm 0.005}$	$\mathbf{0.124}_{\pm 0.002}$	$\mathbf{0.097}_{\pm 0.001}$	$\mathbf{0.221}_{\pm 0.003}$

Table 11: Unseen interaction on Tumor Volume. Models are trained on untreated, chemotherapy-only, and radiotherapy-only futures at every dose, and tested only on the nine concurrent chemotherapy–radiotherapy configurations, none of which appear in training. Mean ± standard deviation over 10 inference repetitions; lower is better. MEETS is best on all four metrics; Aurora is within 1.2× of MEETS on MAE, CRPS, and WQL but 4× on MSE.

The radiotherapy split is built the same way along the other modality. Training contains radiotherapy at 1 and 2 Gy, and every configuration with 3 Gy forms the test set. Chemotherapy-only futures at all three doses, combinations with 1 or 2 Gy of radiotherapy, and untreated futures stay in training.

Chemotherapy (Table 7). No model is best on all four metrics. TADiff has the lowest MSE and MAE, and MEETS has the lowest CRPS and WQL. The two are much closer on MAE than on MSE: MEETS's MSE is 1.51 times TADiff's, while its MAE is only 1.03 times TADiff's. On the probabilistic metrics the order flips, with TADiff's CRPS and WQL at 1.30 and 1.32 times MEETS's.

Together these two models sit well ahead of the other four. CLEF is next on MSE, CRPS, and WQL, with an MSE 2.72 times and a CRPS 2.43 times that of MEETS. On MAE, Aurora takes third place and CLEF drops to fourth. IATSF has the highest MSE and MAE, its MSE 11.40 times MEETS's. EventTSF has the highest CRPS and WQL, although on CRPS it is level with IATSF at two decimals (0.79).

Radiotherapy (Table 8). One ranking holds on all four metrics: MEETS, TADiff, CLEF, Aurora, EventTSF, IATSF. TADiff's errors range from 1.27 to 1.36 times MEETS's across the four metrics, and CLEF's MSE is 1.98 times MEETS's. The largest gap in the ranking falls between CLEF and Aurora: Aurora's MSE is 5.00 times CLEF's and 9.89 times MEETS's. IATSF's MSE is 17.61 times MEETS's.

Across the two splits, TADiff leads the point metrics on chemotherapy but places second behind MEETS on every metric on radiotherapy. CLEF ranks above Aurora on MSE in both splits, and IATSF has the highest MSE in both.

H.2 High-to-Low Intensity

The high-to-low splits reverse the direction: each model must forecast under a dose smaller than any it saw during training. In the chemotherapy split, training futures administer chemotherapy only

Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
Aurora	$\underline{0.381}_{\pm 0.009}$	$\underline{0.196}_{\pm 0.002}$	$0.180_{\pm 0.002}$	$0.354_{\pm 0.005}$
TADiff	$0.541_{\pm 0.024}$	$0.229_{\pm 0.005}$	$0.205_{\pm 0.004}$	$0.398_{\pm 0.009}$
CLEF	$0.527_{\pm 0.016}$	$0.325_{\pm 0.006}$	$0.322_{\pm 0.006}$	$0.618_{\pm 0.012}$
IATSF	$0.602_{\pm 0.029}$	$0.288_{\pm 0.006}$	$0.258_{\pm 0.006}$	$0.501_{\pm 0.012}$
EventTSF	$0.421_{\pm 0.020}$	$0.221_{\pm 0.005}$	$\underline{0.164}_{\pm 0.003}$	$\underline{0.319}_{\pm 0.006}$
MEETS	$\mathbf{0.204}_{\pm 0.009}$	$\mathbf{0.168}_{\pm 0.004}$	$\mathbf{0.124}_{\pm 0.003}$	$\mathbf{0.241}_{\pm 0.005}$

Table 12: Unseen treatment onset, early→middle, on Tumor Volume. Models are trained on futures whose treatment begins in timestamps 31–35 and tested on futures whose treatment begins in timestamps 36–40. Mean ± standard deviation over 10 inference repetitions; lower is better. MEETS is best on all four metrics and every baseline is within 3× of it on MSE, the tightest timing split.

Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
TADiff	$0.808_{\pm 0.034}$	$0.328_{\pm 0.007}$	$0.306_{\pm 0.007}$	$0.486_{\pm 0.011}$
Aurora	$\underline{0.226}_{\pm 0.007}$	$\underline{0.155}_{\pm 0.002}$	$\underline{0.148}_{\pm 0.002}$	$\underline{0.235}_{\pm 0.004}$
CLEF	$1.189_{\pm 0.048}$	$0.480_{\pm 0.011}$	$0.477_{\pm 0.013}$	$0.742_{\pm 0.020}$
IATSF	$0.962_{\pm 0.028}$	$0.406_{\pm 0.008}$	$0.357_{\pm 0.007}$	$0.568_{\pm 0.011}$
EventTSF	$0.354_{\pm 0.015}$	$0.216_{\pm 0.005}$	$0.160_{\pm 0.004}$	$0.255_{\pm 0.007}$
MEETS	$\mathbf{0.142}_{\pm 0.005}$	$\mathbf{0.126}_{\pm 0.003}$	$\mathbf{0.094}_{\pm 0.002}$	$\mathbf{0.150}_{\pm 0.004}$

Table 13: Unseen treatment onset, early→late, on Tumor Volume. Models are trained on futures whose treatment begins in timestamps 31–35 and tested on futures whose treatment begins in timestamps 41–45. Mean ± standard deviation over 10 inference repetitions; lower is better. MEETS is best on all four metrics; TADiff's MSE is 5.7× MEETS's, its largest deficit on any split.

at 5 and 7 mg/m^3. Every configuration with 2 mg/m^3, alone or with radiotherapy, forms the test set. Radiotherapy-only futures, combinations pairing radiotherapy with 5 or 7 mg/m^3, and untreated futures stay in training.

In the radiotherapy split, training contains radiotherapy at 2 and 3 Gy, and every configuration with 1 Gy forms the test set. Chemotherapy-only futures, combinations with 2 or 3 Gy, and untreated futures stay in training.

Chemotherapy (Table 9). TADiff is best on all four metrics, MEETS is second on all four, and Aurora is third on all four. The gap between the first two is smaller on the point metrics than on the probabilistic ones: MEETS's MSE and MAE are 1.06 and 1.03 times TADiff's, while its CRPS and WQL are 1.24 and 1.26 times. Aurora's MSE is 1.30 times MEETS's.

CLEF and EventTSF trade fourth place. CLEF is fourth on MSE and MAE and fifth on CRPS and WQL, and EventTSF is the reverse. CLEF's MAE is 3.48 times MEETS's even though its MSE is only 1.39 times. IATSF is last on all four metrics, with an MSE 4.88 times MEETS's.

Radiotherapy (Table 10). One ranking again holds on all four metrics: MEETS, Aurora, TADiff, CLEF, EventTSF, IATSF. Aurora's MSE is 1.43 times MEETS's, and its MAE, CRPS, and WQL are 1.20, 1.28, and 1.26 times. TADiff's MSE is 2.06 times MEETS's, and IATSF's is 4.39 times.

Between the two splits, TADiff moves from first on every metric for chemotherapy to third on every metric for radiotherapy. Aurora moves from third to second, and IATSF is last on every metric in both.

H.3 Unseen Interaction

The interaction split tests whether a model can forecast under two treatments given together when it has only seen each one given alone. Training contains untreated futures, chemotherapy-only futures at 2, 5, and 7 mg/m^3, and radiotherapy-only futures at 1, 2, and 3 Gy. Every dose of each modality is therefore seen, but no training future ever combines the two. The test set consists solely of the

Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
Aurora	$0.252_{\pm 0.008}$	$\underline{0.136}_{\pm 0.002}$	$\underline{0.117}_{\pm 0.002}$	$\underline{0.191}_{\pm 0.003}$
TADiff	$\underline{0.238}_{\pm 0.010}$	$0.159_{\pm 0.003}$	$0.149_{\pm 0.003}$	$0.243_{\pm 0.006}$
CLEF	$0.503_{\pm 0.020}$	$0.313_{\pm 0.007}$	$0.171_{\pm 0.004}$	$0.279_{\pm 0.006}$
IATSF	$0.618_{\pm 0.016}$	$0.287_{\pm 0.004}$	$0.244_{\pm 0.003}$	$0.398_{\pm 0.005}$
EventTSF	$0.291_{\pm 0.008}$	$0.178_{\pm 0.003}$	$0.132_{\pm 0.002}$	$0.215_{\pm 0.003}$
MEETS	$\mathbf{0.144}_{\pm 0.005}$	$\mathbf{0.112}_{\pm 0.002}$	$\mathbf{0.083}_{\pm 0.002}$	$\mathbf{0.135}_{\pm 0.003}$

Table 14: Unseen treatment onset, middle→late, on Tumor Volume. Models are trained on futures whose treatment begins in timestamps 36–40 and tested on futures whose treatment begins in timestamps 41–45. Mean ± standard deviation over 10 inference repetitions; lower is better. MEETS is best on all four metrics; Aurora is second on MAE, CRPS, and WQL, TADiff on MSE only.

Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
TADiff	$\underline{0.443}_{\pm 0.015}$	$\underline{0.217}_{\pm 0.004}$	$\underline{0.201}_{\pm 0.003}$	$\underline{0.492}_{\pm 0.008}$
Aurora	$0.887_{\pm 0.033}$	$0.362_{\pm 0.007}$	$0.355_{\pm 0.007}$	$0.855_{\pm 0.017}$
EventTSF	$0.483_{\pm 0.026}$	$0.652_{\pm 0.014}$	$0.309_{\pm 0.008}$	$0.750_{\pm 0.019}$
IATSF	$1.462_{\pm 0.042}$	$0.612_{\pm 0.008}$	$0.514_{\pm 0.009}$	$1.253_{\pm 0.022}$
CLEF	$0.548_{\pm 0.019}$	$0.279_{\pm 0.005}$	$0.208_{\pm 0.005}$	$0.507_{\pm 0.011}$
MEETS	$\mathbf{0.238}_{\pm 0.008}$	$\mathbf{0.164}_{\pm 0.003}$	$\mathbf{0.121}_{\pm 0.002}$	$\mathbf{0.295}_{\pm 0.006}$

Table 15: Unseen treatment onset, middle→early, on Tumor Volume. Models are trained on futures whose treatment begins in timestamps 36–40 and tested on futures whose treatment begins in timestamps 31–35. Mean ± standard deviation over 10 inference repetitions; lower is better. MEETS is best and TADiff second on all four metrics.

nine combination configurations, each pairing one chemotherapy dose with one radiotherapy dose. Unlike the intensity splits, no dose is held out; only the co-occurrence of the two treatments is new.

Results (Table 11). MEETS is best and IATSF worst on all four metrics. On MSE the order is MEETS, TADiff, CLEF, EventTSF, Aurora, IATSF. TADiff and CLEF are at 1.31 and 1.39 times MEETS's MSE, and EventTSF, Aurora, and IATSF are at 3.59, 4.00, and 4.87 times.

Aurora's position changes across metrics more than any other model's. It is fifth on MSE but third on MAE, second on CRPS, and third on WQL, where it is level with TADiff at two decimals (0.26). On all three of those metrics it is within 1.19 times MEETS. CLEF is fourth and EventTSF fifth on MAE, CRPS, and WQL. IATSF's MAE is 5.38 times MEETS's.

H.4 Earlier-to-Later Timing

The timing splits hold out when treatment begins rather than what treatment is given. We divide the 15-step horizon into three five-step windows: early (timestamps 31–35), middle (36–40), and late (41–45). Each treated future is assigned to the window containing its onset, the first timestamp at which any treatment is administered. In a timing split, the training set contains only futures whose onset falls in the training window, and the test set only futures whose onset falls in the test window. Treatment type and dose are unrestricted, so chemotherapy, radiotherapy, and combination configurations appear on both sides.

The earlier-to-later splits train on an earlier onset window and test on a later one. This gives three pairings: early→middle, early→late, and middle→late.

Early→middle (Table 12). MEETS is best on all four metrics. Second place differs between the point and probabilistic metrics: Aurora is second on MSE and MAE, while EventTSF is second on CRPS and WQL, with a CRPS 1.32 times MEETS's. All five baselines are within 3 times MEETS on MSE: Aurora at 1.87, EventTSF at 2.06, CLEF at 2.58, TADiff at 2.65, and IATSF at 2.95. IATSF has the highest MSE, and CLEF has the highest MAE, CRPS, and WQL.

Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
TADiff	$\underline{0.438}_{\pm 0.010}$	$\underline{0.196}_{\pm 0.002}$	$\underline{0.184}_{\pm 0.002}$	$\underline{0.358}_{\pm 0.005}$
Aurora	$0.548_{\pm 0.027}$	$0.246_{\pm 0.006}$	$0.222_{\pm 0.006}$	$0.433_{\pm 0.011}$
EventTSF	$0.596_{\pm 0.022}$	$0.287_{\pm 0.005}$	$0.238_{\pm 0.005}$	$0.464_{\pm 0.010}$
IATSF	$0.947_{\pm 0.046}$	$0.412_{\pm 0.009}$	$0.361_{\pm 0.008}$	$0.706_{\pm 0.015}$
CLEF	$0.506_{\pm 0.011}$	$0.263_{\pm 0.003}$	$0.196_{\pm 0.002}$	$0.383_{\pm 0.005}$
MEETS	$\mathbf{0.281}_{\pm 0.007}$	$\mathbf{0.162}_{\pm 0.002}$	$\mathbf{0.120}_{\pm 0.001}$	$\mathbf{0.234}_{\pm 0.003}$

Table 16: Unseen treatment onset, late→middle, on Tumor Volume. Models are trained on futures whose treatment begins in timestamps 41–45 and tested on futures whose treatment begins in timestamps 36–40. Mean ± standard deviation over 10 inference repetitions; lower is better. MEETS is best and TADiff second on all four metrics; every baseline but IATSF is within 2.2× of MEETS on MSE.

Model	MSE ↓	MAE ↓	CRPS ↓	WQL ↓
TADiff	$0.853_{\pm 0.030}$	$0.341_{\pm 0.006}$	$0.328_{\pm 0.006}$	$0.795_{\pm 0.014}$
Aurora	$0.894_{\pm 0.039}$	$0.369_{\pm 0.008}$	$0.351_{\pm 0.009}$	$0.853_{\pm 0.021}$
EventTSF	$0.899_{\pm 0.035}$	$0.709_{\pm 0.013}$	$0.337_{\pm 0.007}$	$0.819_{\pm 0.016}$
IATSF	$1.286_{\pm 0.041}$	$0.497_{\pm 0.009}$	$0.441_{\pm 0.007}$	$1.073_{\pm 0.016}$
CLEF	$\underline{0.671}_{\pm 0.026}$	$\underline{0.298}_{\pm 0.006}$	$\underline{0.223}_{\pm 0.004}$	$\underline{0.542}_{\pm 0.010}$
MEETS	$\mathbf{0.368}_{\pm 0.015}$	$\mathbf{0.201}_{\pm 0.004}$	$\mathbf{0.149}_{\pm 0.003}$	$\mathbf{0.362}_{\pm 0.007}$

Table 17: Unseen treatment onset, late→early, on Tumor Volume. Models are trained on futures whose treatment begins in timestamps 41–45 and tested on futures whose treatment begins in timestamps 31–35. Mean ± standard deviation over 10 inference repetitions; lower is better. MEETS is best and CLEF second on all four metrics; this is MEETS's highest MSE on any timing split.

Early→late (Table 13). One ranking holds on all four metrics: MEETS, Aurora, EventTSF, TADiff, IATSF, CLEF. The models fall into two groups. Aurora and EventTSF have MSEs 1.59 and 2.49 times MEETS's, while TADiff, IATSF, and CLEF are at 5.69, 6.77, and 8.37 times. At the boundary between the groups, TADiff's MSE is 2.28 times EventTSF's.

Middle→late (Table 14). MEETS is best on all four metrics. TADiff is second on MSE only, at 1.65 times MEETS's. Aurora is second on MAE, CRPS, and WQL and third on MSE, at 1.75 times. EventTSF is third on CRPS and WQL and fourth on MSE and MAE. CLEF has the highest MAE, and IATSF has the highest MSE, CRPS, and WQL, with an MSE 4.29 times MEETS's.

Across the three splits, MEETS's MSE is lower on the two splits that test on the late window (0.14 on both) than on early→middle (0.20). Aurora is second on MAE in all three splits, and CLEF has the highest MAE in all three. TADiff's MSE relative to MEETS's is 2.65 on early→middle and 1.65 on middle→late. It rises to 5.69 on early→late, the pairing whose training and test windows are furthest apart.

H.5 Later-to-Earlier Timing

These splits use the same windows and onset assignment as Appendix H.4 but reverse the direction. Each model is trained on a later onset window and tested on an earlier one, giving middle→early, late→middle, and late→early.

Middle→early (Table 15). MEETS is best and TADiff second on all four metrics, with TADiff's MSE 1.86 times MEETS's. EventTSF varies most across metrics: it is third on MSE, at 2.03 times MEETS's, but has the highest MAE of any model, at 3.98 times MEETS's. CLEF is third on MAE, CRPS, and WQL and fourth on MSE. Aurora is fifth on MSE, CRPS, and WQL and fourth on MAE, with an MSE 3.73 times MEETS's. IATSF has the highest MSE, CRPS, and WQL, with an MSE 6.14 times MEETS's.

Dataset	Variant	MSE ↓	MAE ↓	CRPS ↓	WQL ↓	RespMSE ↓	RespMAE ↓	DA ↑	DRS → 1
Traffic Volume	Concatenated	1.95±0.01	1.05±0.00	1.04±0.01	1.30±0.01	2.55±0.01	1.18±0.01	0.30±0.02	0.25±0.03
	Separate (no SA)	0.85±0.00	0.58±0.00	0.44±0.00	0.55±0.01	1.10±0.01	0.65±0.01	0.62±0.01	0.55±0.02
	Separate + SA (MEETS)	**0.07**±0.00	**0.15**±0.00	**0.11**±0.00	**0.14**±0.00	**0.14**±0.00	**0.19**±0.00	**0.91**±0.00	**1.20**±0.00
Energy Usage	Concatenated	2.25±0.01	1.15±0.00	1.20±0.00	1.50±0.01	2.95±0.01	1.28±0.01	0.24±0.02	0.20±0.02
	Separate (no SA)	1.00±0.00	0.64±0.00	0.56±0.00	0.70±0.01	1.35±0.01	0.72±0.01	0.58±0.01	0.50±0.02
	Separate + SA (MEETS)	**0.08**±0.00	**0.04**±0.00	**0.03**±0.00	**0.05**±0.01	**0.12**±0.00	**0.06**±0.00	**0.89**±0.00	**0.76**±0.01
Tumor Volume	Concatenated	2.70±0.00	1.28±0.00	1.40±0.00	1.75±0.01	3.35±0.01	1.38±0.01	0.18±0.02	0.15±0.02
	Separate (no SA)	1.25±0.00	0.72±0.00	0.68±0.00	0.85±0.01	1.60±0.01	0.80±0.01	0.52±0.01	0.45±0.01
	Separate + SA (MEETS)	**0.06**±0.00	**0.03**±0.00	**0.02**±0.00	**0.02**±0.00	**0.20**±0.00	**0.06**±0.00	**0.96**±0.00	**0.98**±0.01

Table 18: Event-processing ablation. *Concatenated* joins all event descriptions in the horizon into one text encoded as a single event token; *Separate (no SA)* encodes each event into its own token but removes event-set self-attention; *Separate + SA* is MEETS. MSE–WQL are the factual metrics and RespMSE–DRS the response metrics. Mean ± standard deviation over 10 inference repetitions. DA: higher is better; DRS: closer to 1 is better; all other metrics: lower is better. Concatenating events into one text cuts DA to 0.19–0.33 of MEETS's, and removing self-attention alone cuts it to 0.54–0.68; self-attention alone accounts for a 12–21× MSE gap.

Dataset	Conditioning	MSE ↓	MAE ↓	CRPS ↓	WQL ↓	RespMSE ↓	RespMAE ↓	DA ↑	DRS → 1
Traffic Volume	Fused conditioning	0.95±0.01	0.70±0.00	0.60±0.01	0.75±0.01	1.25±0.01	0.78±0.01	0.42±0.02	0.40±0.02
	Separate conditioning (MEETS)	**0.07**±0.00	**0.15**±0.00	**0.11**±0.00	**0.14**±0.00	**0.14**±0.00	**0.19**±0.00	**0.91**±0.00	**1.20**±0.00
Energy Usage	Fused conditioning	1.45±0.01	0.92±0.00	0.88±0.00	1.10±0.01	1.90±0.01	1.00±0.01	0.28±0.01	0.24±0.02
	Separate conditioning (MEETS)	**0.08**±0.00	**0.04**±0.00	**0.03**±0.00	**0.05**±0.01	**0.12**±0.00	**0.06**±0.00	**0.89**±0.00	**0.76**±0.01
Tumor Volume	Fused conditioning	1.20±0.00	0.84±0.00	0.76±0.00	0.95±0.01	1.55±0.01	0.90±0.01	0.35±0.01	0.32±0.01
	Separate conditioning (MEETS)	**0.06**±0.00	**0.03**±0.00	**0.02**±0.00	**0.02**±0.00	**0.20**±0.00	**0.06**±0.00	**0.96**±0.00	**0.98**±0.01

Table 19: History–event conditioning ablation. *Fused conditioning* concatenates the history and event encodings into one memory read by a single cross-attention in each denoising block; *Separate conditioning* (MEETS) reads them through two separate cross-attentions. MSE–WQL are the factual metrics and RespMSE–DRS the response metrics. Mean ± standard deviation over 10 inference repetitions. DA: higher is better; DRS: closer to 1 is better; all other metrics: lower is better. Fusing history and events into one attention memory raises MSE 14–20× and cuts DRS to a third of MEETS's on every dataset.

Late→middle (Table 16). MEETS is best, TADiff second, EventTSF fifth, and IATSF last on all four metrics. TADiff's MSE is 1.56 times MEETS's. CLEF is third on MSE, CRPS, and WQL, and Aurora is third on MAE and fourth on the rest. Every baseline except IATSF is within 2.12 times MEETS on MSE; IATSF is at 3.37.

Late→early (Table 17). MEETS is best, CLEF second, and TADiff third on all four metrics. CLEF's MSE is 1.82 times MEETS's, and its MAE, CRPS, and WQL are 1.48, 1.50, and 1.50 times. TADiff, Aurora, and EventTSF are close on MSE, at 2.32, 2.43, and 2.44 times MEETS's. IATSF has the highest MSE, CRPS, and WQL, with an MSE 3.49 times MEETS's. EventTSF has the highest MAE, at 3.53 times MEETS's.

Across the three splits, MEETS's MSE is highest on late→early (0.37), compared with 0.24 on middle→early and 0.28 on late→middle. TADiff is second on middle→early and late→middle, and CLEF is second on late→early. IATSF has the highest MSE in all three splits, and EventTSF has the highest MAE on the two splits that test on the early window.

H.6 Ablation Studies

Each of the three ablations changes one design choice of MEETS and holds everything else fixed. The fixed components are the history encoder, the text encoder and its fine-tuning, the six-block denoiser, the diffusion schedule, the optimization settings, and the checkpoint rule of Appendix E. Each variant is retrained from scratch on all three datasets with the main training, validation, and test splits.

Every table reports two groups of metrics. The factual metrics of (MSE, MAE, CRPS, WQL) are computed as in Section 5.2. The response metrics (ResponseMSE, ResponseMAE, DA, DRS) are computed as in Section 5.3: with the DTW-nearest donor on Traffic Volume and Energy Usage, and over all pairs of realized futures with exact references on Tumor Volume. All entries are the mean

Dataset	Utility	MSE ↓	MAE ↓	CRPS ↓	WQL ↓	RespMSE ↓	RespMAE ↓	DA ↑	DRS → 1
Traffic Volume	None	1.25±0.01	0.80±0.00	0.72±0.01	0.90±0.01	1.75±0.01	0.90±0.01	0.08±0.02	0.10±0.02
	$\mathcal{L}_H$ only	0.62±0.00	0.56±0.00	0.40±0.01	0.50±0.01	0.95±0.01	0.62±0.01	0.45±0.02	0.42±0.02
	$\mathcal{L}_E$ only	0.35±0.00	0.42±0.00	0.28±0.00	0.35±0.01	0.52±0.01	0.46±0.01	0.76±0.01	0.67±0.02
	$\mathcal{L}_E + \mathcal{L}_H$ (MEETS)	**0.07**±0.00	**0.15**±0.00	**0.11**±0.00	**0.14**±0.00	**0.14**±0.00	**0.19**±0.00	**0.91**±0.00	**1.20**±0.00
Energy Usage	None	1.75±0.01	1.05±0.00	1.00±0.00	1.25±0.01	2.40±0.01	1.15±0.01	0.05±0.02	0.05±0.02
	$\mathcal{L}_H$ only	1.58±0.00	1.00±0.00	0.92±0.00	1.15±0.01	2.18±0.01	1.10±0.01	0.12±0.02	0.12±0.02
	$\mathcal{L}_E$ only	0.50±0.00	0.55±0.00	0.40±0.00	0.50±0.01	0.72±0.01	0.60±0.01	0.74±0.01	0.60±0.01
	$\mathcal{L}_E + \mathcal{L}_H$ (MEETS)	**0.08**±0.00	**0.04**±0.00	**0.03**±0.00	**0.05**±0.01	**0.12**±0.00	**0.06**±0.00	**0.89**±0.00	**0.76**±0.01
Tumor Volume	None	1.50±0.00	0.95±0.00	0.88±0.00	1.10±0.01	2.05±0.01	1.05±0.01	0.06±0.02	0.07±0.02
	$\mathcal{L}_H$ only	1.38±0.00	0.91±0.00	0.80±0.00	1.00±0.01	1.90±0.01	1.00±0.01	0.14±0.01	0.14±0.02
	$\mathcal{L}_E$ only	0.43±0.00	0.50±0.00	0.32±0.00	0.40±0.01	0.62±0.01	0.54±0.01	0.80±0.01	0.74±0.01
	$\mathcal{L}_E + \mathcal{L}_H$ (MEETS)	**0.06**±0.00	**0.03**±0.00	**0.02**±0.00	**0.02**±0.00	**0.20**±0.00	**0.06**±0.00	**0.96**±0.00	**0.98**±0.01

Table 20: Conditional-utility loss ablation. *None* trains with the diffusion loss alone; $\mathcal{L}_H$ *only* and $\mathcal{L}_E$ *only* add the history- or event-utility loss with weight 0.1; $\mathcal{L}_E + \mathcal{L}_H$ is MEETS. MSE–WQL are the factual metrics and RespMSE–DRS the response metrics. Mean ± standard deviation over 10 inference repetitions. DA: higher is better; DRS: closer to 1 is better; all other metrics: lower is better. With the diffusion loss alone DA is 6–9% of MEETS's; $\mathcal{L}_E$ alone recovers most of the direction, both terms together recover the error.

and standard deviation over 10 inference repetitions. Lower is better for the four factual metrics and for ResponseMSE and ResponseMAE, higher is better for DA, and DRS is best at 1.

Event processing (Table 18). This ablation varies how the event set $E = \{e_1, \ldots, e_j\}$ becomes the event encoding $\mathbf{M}_E$. In the *Concatenated* variant, all event descriptions in the horizon are joined into a single text that the text encoder reads once. $\mathbf{M}_E$ is then a single token regardless of how many events the horizon contains. In the *Separate (no SA)* variant, each event is encoded into its own token $\mathbf{v}_r$ as in MEETS, but the event-set Transformer f_{event} is removed. The encoding is then $\mathbf{M}_E = (\mathbf{v}_1, \ldots, \mathbf{v}_j)$, and the event tokens never attend to one another. *Separate + SA* is MEETS itself.

On every dataset and every error metric, Concatenated has the highest error, Separate (no SA) the next, and MEETS the lowest. DA rises in the same order, and MEETS's DRS is closest to 1 on all three datasets. Concatenated's MSE is 2.29, 2.25, and 2.16 times that of Separate (no SA) on Traffic Volume, Energy Usage, and Tumor Volume. Separate (no SA)'s MSE is in turn 12.14, 12.50, and 20.83 times MEETS's.

The two ablated variants move in the same direction across datasets. From Traffic Volume to Energy Usage to Tumor Volume, their MSE increases (1.95, 2.25, 2.70 for Concatenated; 0.85, 1.00, 1.25 for Separate (no SA)) and their DA decreases (0.30, 0.24, 0.18; 0.62, 0.58, 0.52). MEETS does not follow this pattern; its lowest MSE is on Tumor Volume. On ResponseMSE, Separate (no SA) is 7.86, 11.25, and 8.00 times MEETS. MEETS's DRS of 1.20 on Traffic Volume is the only DRS above 1 in the table.

History–event conditioning (Table 19). This ablation varies how the denoiser reads the two encodings. In the *Fused* variant, the history encoding $\mathbf{M}_H$ and the event encoding $\mathbf{M}_E$ are concatenated along the token axis into one memory. Each denoising block replaces the history and event cross-attentions with a single cross-attention over that memory, masking padded event slots. Future self-attention and the feed-forward layer are unchanged, and the encoders themselves are identical to MEETS's. *Separate conditioning* is MEETS.

Fused conditioning has higher error than MEETS on every error metric and every dataset. Its MSE is 13.57, 18.13, and 20.00 times MEETS's on Traffic Volume, Energy Usage, and Tumor Volume, and its MAE is 4.67, 23.00, and 28.00 times. On ResponseMSE it is 8.93, 15.83, and 7.75 times MEETS. Its DA is 0.46, 0.31, and 0.36 of MEETS's, and its DRS is 0.33, 0.32, and 0.33 of MEETS's. Among the three datasets, Fused conditioning has its highest MSE on Energy Usage.

Conditional-utility loss (Table 20). This ablation varies the training objective while keeping the architecture fixed.

- *None* trains with the diffusion loss $\mathcal{L}_{\text{diff}}$ alone ($\lambda_E = \lambda_H = 0$).

- $\mathcal{L}_H$ *only* adds the history-utility loss ($\lambda_E = 0$, $\lambda_H = 0.1$).
- $\mathcal{L}_E$ *only* adds the event-utility loss ($\lambda_E = 0.1$, $\lambda_H = 0$).
- $\mathcal{L}_E + \mathcal{L}_H$ is MEETS, with $\lambda_E = \lambda_H = 0.1$.

Borrowed histories and event sets are built exactly as in Appendix C.3.

The four variants keep the same order on every metric and every dataset: from None to $\mathcal{L}_H$ only to $\mathcal{L}_E$ only to MEETS, error falls, DA rises, and DRS moves toward 1. Adding $\mathcal{L}_H$ alone lowers MSE relative to None by a factor of 2.02 on Traffic Volume but only 1.11 on Energy Usage and 1.09 on Tumor Volume. Adding $\mathcal{L}_E$ alone lowers it by 3.49 to 3.57 on all three. Using both lowers MSE by 17.86, 21.88, and 25.00 relative to None, and by 5.00, 6.25, and 7.17 relative to $\mathcal{L}_E$ only.

Under None, DA is 0.08, 0.05, and 0.06. $\mathcal{L}_H$ only raises DA to 0.45 on Traffic Volume but only to 0.12 on Energy Usage and 0.14 on Tumor Volume. $\mathcal{L}_E$ only raises it to 0.76, 0.74, and 0.80.

H.7 Sensitivity to Event-Description Wording

MEETS conditions its forecast on a free-text description of each event. In practice the same event can be written up in many ways, depending on the source, the annotator, or the language model that produced the text. This experiment asks whether the forecast depends on the content of the description or on its exact wording.

Setup. For each of Traffic Volume, Energy Usage, and Tumor Volume, we sample 100 test windows uniformly at random. For the event description attached to each window, we generate ten paraphrases with GPT-5.4 mini using the prompt in Figure 5. The prompt fixes the event type, its intensity or severity, its start and end timestamps, its duration, and every other stated attribute. It forbids adding or removing information, introducing causes or consequences, and generalizing or specializing the event. Only the wording changes, through sentence structure, word order, temporal phrasing, and equivalent vocabulary. This yields 1,000 paraphrases per dataset and 3,000 in total.

Paraphrase set $k \in \{1, \ldots, 10\}$ replaces every original description in the 100-window test set with its k-th paraphrase, giving ten reworded copies of the same test set. The index k follows generation order, so each set is an independent draw of wording rather than a consistent style. The trained model is evaluated unchanged: paraphrases appear only at test time. For every description we draw 100 forecast trajectories and compute MSE, MAE, CRPS, and WQL from them. Each evaluation is repeated over ten runs, and the original descriptions on the same 100 windows serve as the reference. Wording sensitivity is measured by the shift of each paraphrase set relative to the original and by the spread across the ten sets, compared with the run-to-run standard deviation of the original.

Results. Paraphrasing shifts every metric in the same direction: the median paraphrase set scores 0.33–0.61% worse than the original on 11 of 12 dataset–metric pairs, and on Tumor Volume all ten sets are worse on all four metrics (Figure 4). The shift is bounded: no paraphrase set moves any metric by more than 1.84%, and the spread across paraphrase sets stays below 0.64 of the run-to-run standard deviation. MEETS is trained only on the original descriptions, so any rewording places the text input slightly off its training distribution, and the cost appears as a small systematic degradation rather than as noise. Wording therefore does not change what the model forecasts, but it is not free: training with paraphrased descriptions is the direct way to remove this offset, and evaluation on descriptions from a source other than the training corpus should expect it.

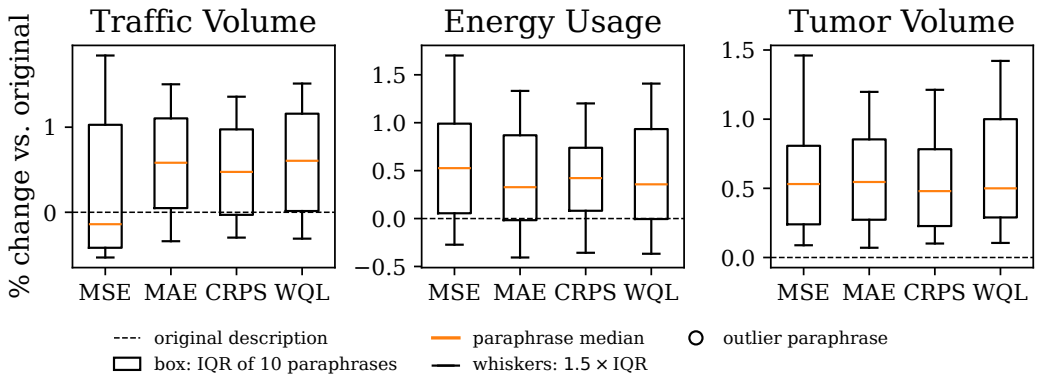

Figure 4: Change in each metric, relative to the original description, when every event description in the test set is replaced by one of ten GPT-5.4 mini paraphrases. Paraphrases score consistently but only slightly worse than the original (at most 1.84%): the model responds to event content, with a small penalty for wording it was not trained on.

```
You are given a natural-language description of an event.
**Input Event**
{EVENT_DESCRIPTION}
**Task**
Generate exactly 10 linguistically diverse paraphrases of the input event.
Each paraphrase must preserve the exact same semantic meaning as the original event.
**You MUST preserve**
* Event type
* Event intensity or severity
* Start timestamp
* End timestamp
* Duration
* Any other explicitly stated event attributes
**You MUST NOT**
* Add new information
* Remove information
* Change the event type
* Change the event intensity
* Change the timing or duration
* Introduce causes, effects, consequences, or assumptions
* Generalize or make the event more specific
* Create alternative scenarios
The 10 outputs should differ **only in their linguistic expression, not in their underlying
    meaning.**
**Diversity**
Use meaningful linguistic variation across the 10 outputs. Vary sentence structure, word order
    , temporal phrasing, and semantically equivalent vocabulary. Avoid generating near-
    duplicates that differ by only one or two words.
**Output**
Return exactly 10 paraphrases, and nothing else.
```

Figure 5: Prompt used to generate paraphrases of each event description with GPT-5.4 mini.