

NUM2EVENT: Structured Event Inference from Numerical Time-Series

Ninghui Feng^{1,3*} Songning Lai^{2*} Xinan Chen³ Zheng Chu³ Yiyi Qi^{1†}

¹International Digital Economy Academy

²Shandong University

³The University of Nottingham Ningbo China

feng65501@gmail.com lais0328eee@gmail.com

{Xinan.Chen,Zheng.Chu}@nottingham.edu.cn qiyiyan@idea.edu.cn

Abstract

Large language models (LLMs) have shown strong multimodal reasoning abilities, yet their understanding of purely numerical time-series remains limited. Existing approaches mainly focus on forecasting or trend description, providing little insight into the underlying events that have occurred or are unfolding within observed numerical dynamics. We study number-to-event reasoning and decoding, a task that infers interpretable and structured event hypotheses directly from numerical time-series when explicit event information is unavailable at inference time. The task focuses on aligned and plausible event semantics that help explain observed numerical dynamics. To address data scarcity and semantic alignment challenges, we propose a reasoning-aware framework that combines an agent-guided event extractor, a Hawkes-process-based synthetic data generator, and a multi-stage fine-tuning pipeline integrating a time-series encoder with a structured event decoder. The model generates intermediate reasoning steps and outputs constrained event representations. Experiments across multiple real-world domains show that our approach consistently outperforms strong LLM baselines in event-level precision and recall, demonstrating its effectiveness in bridging numerical reasoning and semantic event understanding. The code is available at https://github.com/feng1201/NUM2EVENT_IDEA.

1 Introduction

Numerical time-series are a primary source for monitoring complex real-world systems such as financial markets and industrial operations. Changes in these signals are often driven by underlying events, including market shocks (MacKinlay, 1997), policy interventions (Brodersen et al., 2015), or system failures (Venkatasubramanian

et al., 2003). In practice, however, explicit event information—such as structured logs or textual reports—is frequently delayed or unavailable at the moment when decisions must be made. As a result, practitioners must rely on numerical observations alone, without direct evidence of which events may be associated with the observed dynamics.

Recent advances in multimodal large language models (MLLMs) have demonstrated strong reasoning capabilities across heterogeneous inputs (Yin et al., 2024; Liu et al., 2023). When applied to time-series data, existing methods primarily focus on numerical forecasting or descriptive pattern understanding, such as anomaly or trend detection (Xie et al., 2025; Kong et al., 2025; Wang et al., 2025b). While effective for prediction, these approaches do not aim to generate plausible event hypotheses associated with numerical changes.

A related line of work is event forecasting (Guan et al., 2024; Gottschalk and Demidova, 2019), which predicts future events from historical observations or event streams. This setting fundamentally differs from the problem we study. In many real-world scenarios, event information is precisely what is missing at inference time. The practical need is to generate plausible hypotheses about what may have already occurred or may be unfolding within an observed numerical window, so that these candidates can support subsequent retrieval and human verification. We frame this setting as retrospective and contemporaneous event-hypothesis inference from time-series data.

Motivated by this gap, we introduce a new task of number-to-event reasoning and decoding. Given a window of numerical time-series, the objective is to infer a set of latent, concurrent, and structured event hypotheses that may jointly explain the observed dynamics. Event streams are available only during training for supervision and are absent at inference time. Time-series event classification typically maps numerical windows to fixed label

*Equal contribution.

†Corresponding author.

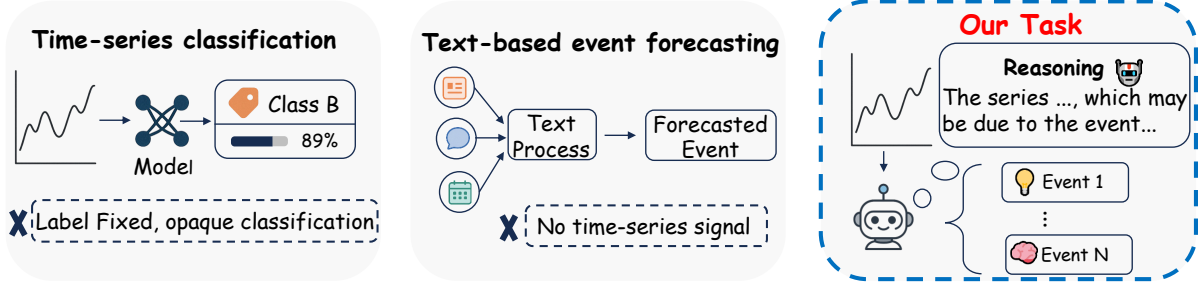


Figure 1: Comparison of our task with time-series classification and text-based event forecasting.

sets and struggles with concurrent or compositional events (Ranjan et al., 2019; Azib et al., 2024); our task requires translating numerical patterns into a set of structured semantic representations under explicit constraints. Crucially, this mapping is inherently one-to-many: similar numerical patterns may be consistent with different real-world events, so the outputs should be interpreted as auditable candidate hypotheses rather than factual event identification or causal attribution. Given a window of numerical time-series, the objective is to generate a set of concurrent and structured event hypotheses that are compatible with the observed dynamics. Because similar numerical patterns can support different event interpretations, the task represents this one-to-many relationship through a structured set of comparable and inspectable hypotheses.

Despite its importance, number-to-event reasoning poses two key challenges. First, aligned datasets pairing numerical time-series with structured event representations are extremely scarce (Liu et al., 2025), as available supervision is often embedded in unstructured historical text. Second, multiple interacting events may occur within the same time window, making fixed-label formulations inadequate. Addressing these challenges requires a framework that can leverage weak supervision while supporting structured and interpretable concurrent event decoding.

To this end, we propose a reasoning-aware number-to-event decoding framework. Our approach extracts structured AAOD (Actor, Action, Object, Direction) event representations from historical text and aligns them with numerical segments, forming an extensible event schema with explicit structural constraints. To improve coverage of event concurrency and dependency patterns under limited supervision, we employ a multivariate Hawkes process as controlled training augmentation. This generator exposes the model to diverse

event interactions. Finally, we fine-tune a large language model equipped with a time-series encoder to decode numerical windows into constrained sets of concurrent AAOD events, along with intermediate reasoning paths linking numerical dynamics to event semantics.

Our contributions are summarized as follows:

- We introduce a new task of number-to-event reasoning and decoding, which infers plausible structured event hypotheses from observed numerical time-series when explicit event information is unavailable at inference time.
- We propose a reasoning-aware framework that integrates structured event extraction, dependency-aware data synthesis, and constrained LLM decoding for interpretable event inference.
- We construct aligned numeric–event datasets across multiple real-world domains and demonstrate superior event-level precision and recall compared to strong LLM baselines.

The remainder of this paper reviews related work in Section 2, formalizes the task in Section 3, presents NUM2EVENT in Section 4, reports the experiments in Section 5, and summarizes the findings in Section 6.

2 Related Work

Recent advancements in large language models have led to rapid progress in the field of time-series data. Firstly, foundation models training by time-series data have been introduced to forecasting tasks (Rasul et al., 2024; Das et al., 2024; Ekambaram et al., 2024; Woo et al., 2024), enhancing the ability to predict time-series values through large scale dataset. Subsequently, researchers explored the alignment of time-series data with textual modality data to improve the performance of

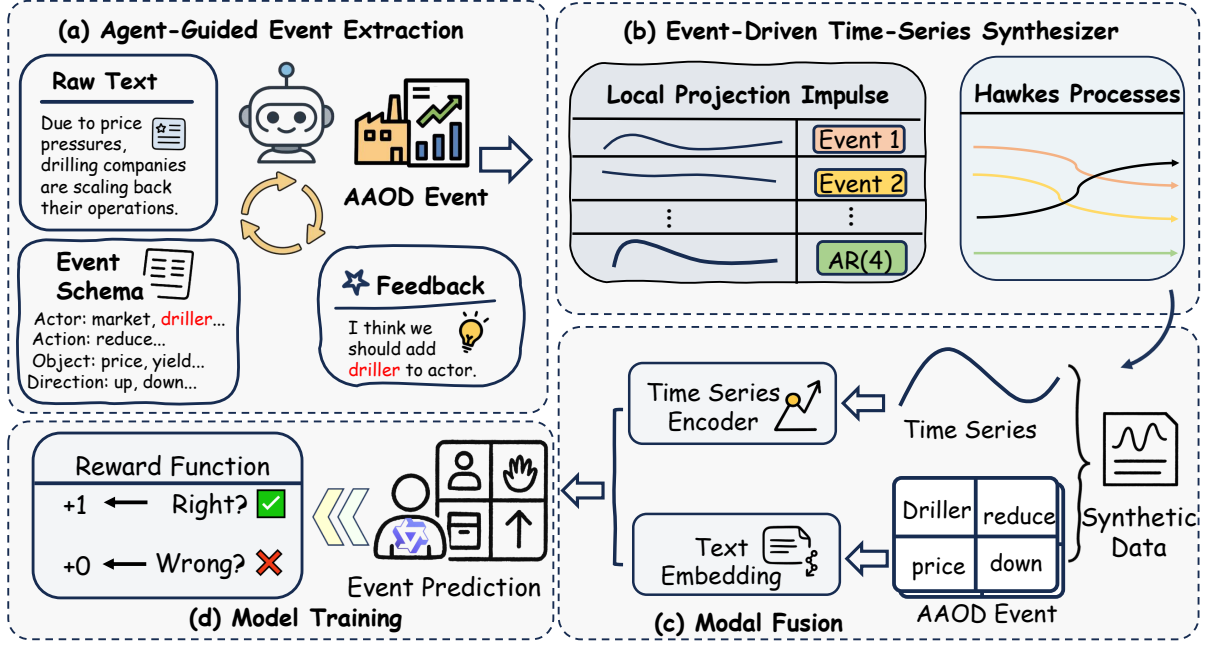


Figure 2: The proposed reasoning-aware number-to-event framework, including (a) agent-guided AAOD event extraction, (b) Hawkes-based synthetic generation, (c) modality fusion via a time-series encoder, and (d) a training pipeline for structured event decoding.

time-series tasks (Jin et al., 2024; Pan et al., 2024; Tang et al., 2025; Zhang et al., 2025b). Meanwhile, many studies have attempted to enhance LLMs’ understanding of time-series data in textual modality, particularly in the analysis of long sequence data (Xie et al., 2025; Wang et al., 2025b). Some studies have actively expanded the modeling of time-series data by incorporating additional information and modalities, such as tabular data (Ma et al., 2025) and images (Park et al., 2026; Chowdhury et al., 2025), further enhancing the processing capabilities of time-series data. Recent studies have proposed unified model frameworks designed to handle multiple time-series tasks (Zhang et al., 2025a; Wang et al., 2025a; Hao et al., 2025), such as forecasting, classification, and anomaly detection, thereby improving the generality of the model.

GAMETime (Tan et al., 2026) studies event inference from numerical time-series by asking language models to select the event sequence most consistent with a given numerical series. Its main benchmark pairs sports win-probability trajectories with ordered play-by-play events and evaluates models through a four-option selection task. NUM2EVENT considers an open-ended generative setting: given a numerical window and a domain-specific AAOD schema, the model produces an unordered set of concurrent event hypotheses without candidate event descriptions at inference time.

We evaluate this formulation across energy, health, and agriculture, where the same numerical pattern may be associated with multiple plausible event hypotheses.

3 Problem Definition

In this paper, we define number-to-event decoding as structured event-hypothesis inference from numerical time-series. Unlike conventional time-series tasks, the goal of this task is to infer the underlying event directly from numerical time-series signals. Specifically, given a time-series window $X_{(t-m+1,t)} = \{x_{t-m+1}, x_{t-m+2}, \dots, x_t\}$ of length m , the model is required to output an interpretable set of events $E_k = \{e_1, e_2, \dots, e_k\}$, where k represents the number of events. Each event is represented in a structured form that includes the key elements: Actor | Action | Object | Direction, capturing the critical components of the event. Formally, our task can be expressed as:

$$f : X_{(t-m+1,t)} \longrightarrow E_k. \quad (1)$$

Here, f maps the numerical window into a structured set of plausible event hypotheses.

4 Methodology

4.1 Overview

When addressing a new task, the primary challenge is how to obtain relevant data. In our task, we need

to extract relevant events from complex text. Event extraction in natural language processing (NLP) has traditionally followed the structured paradigm of Actor | Action | Object (Lai, 2022), with LLMs widely applied to extract event slots from text (Wei et al., 2024). However, in real-world scenarios, the textual information related to a specific time-series is often complex and comes from diverse sources (Wang et al., 2024). This makes it challenging to integrate the information into a unified event extraction framework. To address this gap, we propose an approach that integrates an agent-guided extraction mechanism with restricted schema and dynamic expansion. The restricted schema makes this new task auditable and evaluable; dynamic expansion then mitigates the limits of a fixed vocabulary by incorporating frequent new terms into the event repository.

After obtaining the relevant data, a key challenge is how to efficiently utilize the available data due to its scarcity. Data synthesis from available data has been repeatedly validated in vision and NLP tasks as an effective method for data scarcity (Bauer et al., 2024). Yet in time-series tasks, most existing generation methods primarily focus on generating numerical sequences (Fu et al., 2024). These methods generally fail to model the underlying events, as they focus solely on numerical patterns and neglect the event semantics associated with them. To address this, we incorporate a synthesis strategy based on statistical processes and impulse responses function (IRF) to construct aligned number-event training samples.

In terms of model construction and training, LLMs have demonstrated impressive capabilities in natural language understanding but exhibit limited perception of numerical time-series data (Wang et al., 2025a). Recent work in multimodal time-series forecasting has further demonstrated the effectiveness of the time-series encoder paradigm (Xie et al., 2025; Jin et al., 2024). By processing time-series data with encoders, these models are able to extract more meaningful features, thereby enhancing performance. Additionally, recent studies have shown that group relative policy optimization (GRPO) can significantly enhance the model’s reasoning capabilities (Srivastava and Aggarwal, 2025). By incorporating GRPO, models can optimize their decision-making process, enabling more accurate predictions. In our task, we propose a multi-stage training pipeline that combines these approaches to enhance the model’s ability to rea-

son over numerical time-series data and infer event semantics. Our approach specifically includes the following components:

Agent-Guided Event Extraction (AGE). An agent framework with restricted schema extracts AAOD events from historical text and dynamically proposes new terms, forming a select expand iterate loop to build an extensible event semantic repository.

Event-Driven Time-Series Synthesizer (EveDTS). To bridge the data gap, we use a generation module that separates endogenous background dynamics from exogenous event shocks. It combines Hawkes processes with local-projection impulse responses and AR dynamics to create controllable training augmentations.

Model Training. We adopt a multi-stage training pipeline: first training a time-series encoder, then fine-tuning the language model to generate natural-language reasoning and structured event hypotheses from numerical inputs.

Our framework integrates event extraction, data synthesis, and model training into a unified pipeline, designed to leverage structured supervision and synthetic samples for effective learning and reasoning from numbers to event semantics.

4.2 Agent-Guided Event Extraction

In the task of number-to-event alignment, obtaining standardized and extensible event supervision is the primary challenge. Relying solely on manual annotation is costly and often suffers from inconsistency across annotators (Horych et al., 2025). In natural language processing, event extraction can be performed with LLMs under fixed paradigms (Wei et al., 2024). However, these approaches largely remain within the textual modality, without integration with numerical signals, and they lack mechanisms for open expansion.

To address this issue, we propose the AGE module. Its goal is to provide standardized, extensible, and auditable supervision for number-to-event alignment. AGE adopts AAOD slots as the minimal structural units for extraction.

During event extraction, we use a LLM to match the input text with predefined slot that represent key components of the event. The goal of the extraction process is to identify the most likely configuration of these slots based on the input text, selecting the most relevant terms from the event schema.

Formally, for each event slot i (e.g., Actor, Action), we define a candidate schema set $\mathcal{C}_i$. The

extraction process is modeled as a constrained selection task. Specifically, for a given input text T , we instruct the LLM (parameterized by θ) to select the optimal value $\hat{v}_i$ from the candidate set $\mathcal{C}_i$ that maximizes the conditional probability:

$$\hat{v}_i = \arg \max_{v \in \mathcal{C}_i} P_\theta(v \mid T, \text{prompt}_i), \quad (2)$$

where $P_\theta(v \mid \cdot)$ represents the likelihood assigned by the LLM to the token v , and prompt_i denotes the specific instruction for slot i .

To address the issue of expanding the schema during extraction, we introduce a dynamic schema update process. When a suggested schema term v_i for a particular slot exceeds a predefined suggestion threshold s , it is added to the schema. This results in an updated event schema, denoted as $\hat{V}$, which is defined as:

$$\hat{\mathcal{C}}_i = \mathcal{C}_i \cup v \mid \text{count}(v, i) \geq \text{thr}, \quad \forall i \in \mathcal{S}. \quad (3)$$

For each specific slot type $i \in \mathcal{S}$ (where $\mathcal{S} = \{\text{Actor, Action, Object, Direction}\}$), the schema is independently expanded. The updated candidate set $\hat{\mathcal{C}}_i$ is defined as the union of the original set $\mathcal{C}_i$ and the high-frequency novel terms specific to that slot whose occurrence frequency exceeds a threshold thr .

In this way, AGE forms a closed loop of selection, expansion, and iteration: first extracting with the restricted schema, then aggregating high-frequency suggestions into the repository, and finally initiating the next round of extraction. After obtaining events, AGE further performs deduplication and filtering, outputting standardized events. Through this mechanism, AGE ensures that extraction results remain structured and consistent, while progressively expanding the event space, thereby providing stable and extensible semantic supervision for subsequent number-to-event decoding.

4.3 Event-Driven Time-Series Synthesizer

In this task, real event annotations alone are often insufficient to cover diverse dependency patterns and impact effects (Shi et al., 2025). In complex systems such as finance and industry, events may also influence the occurrence and impact of other events, producing interactions beyond independent event arrivals (Bacry et al., 2015). We therefore use synthetic data as controlled training augmentation for exposing the model to richer concurrent-event patterns. To achieve this, we must ensure

that the model possesses strong interpretability and flexibility. Interpretability ensures that the links between events and numerical fluctuations are explicit, while flexibility enables the simulation of event interactions and temporal dependencies.

To meet these requirements, we have chosen classical statistical methods, such as the Hawkes process, IRF, and autoregressive (AR) models. Hawkes processes have long been employed in domains such as financial markets and seismology to simulate clustered event arrivals (Lima, 2021), making them a natural choice for modeling event driven signals. Local projection IRF are widely used in econometrics to analyze the dynamic effects of exogenous shocks on numerical series (Coulombe and Klieber, 2025), offering interpretable pathways. Meanwhile, AR models remain a classical tool for time-series modeling and are commonly used to generate realistic background dynamics (Cai et al., 2025). Although these three approaches have been extensively studied in their respective fields, they have not previously been integrated to construct synthetic series and event paired datasets.

EveDTS module is designed to bridge this gap. EveDTS works with first differences of series, defining $\Delta y_t \equiv y_t - y_{t-1}$. We begin by modeling event arrivals with a Hawkes process:

$$\lambda_k(t) = \mu_k + \sum_{j=1}^K \sum_{t_i^j < t} \alpha_{kj} g(t - t_i^j), \quad (4)$$

where $\lambda_k(t)$ denotes the conditional intensity of type- k events at time t , μ_k is the baseline intensity, α_{kj} captures the influence of type- j events on type k , and $g(\cdot)$ is a temporal decaying kernel related to time (e.g., $g(u) = \omega e^{-\omega u}$ for $u > 0$).

Given an event at time t , its effect on the series is traced via local projections:

$$\Delta y_{t+h} = \beta_k(h) \mathbf{1}\{\text{event of type } k \text{ at } t\} + \varepsilon_{t+h}, \quad (5)$$

where $\mathbf{1}\{\text{event of type } k \text{ at } t\}$ is an indicator function that indicates whether an event of type k occurred at time t . To its left, $\beta_k(h)$ represents the impact of event k on the time-series data, specifically its effect on the value change over the next h time steps. And ε_{t+h} represents other influences on the time-series data, such as errors.

Superimposing AR background dynamics with event shocks yields the generative form in differences:

$$\Delta y_t = \sum_{j=1}^4 \phi_j \Delta y_{t-j} + \sum_{k=1}^K \sum_{t_i^k < t} \beta_k(t - t_i^k) + \varepsilon_t, \quad (6)$$

where ϕ_j represents the autoregressive coefficients, capturing the persistence of background dynamics. The second term aggregates the cumulative impact of all historical events. Specifically, the Hawkes process governs the generation of event timestamps $\{t_i^k\}$. Once an event of type k occurs at time t_i^k , it triggers a deterministic impulse response $\beta_k(\tau)$ (where $\tau = t - t_i^k$ is the elapsed time), which is superimposed onto the time series. This formulation separates the event arrival process (driven by λ) from the event impact magnitude (driven by β).

4.4 Model Training and Alignment

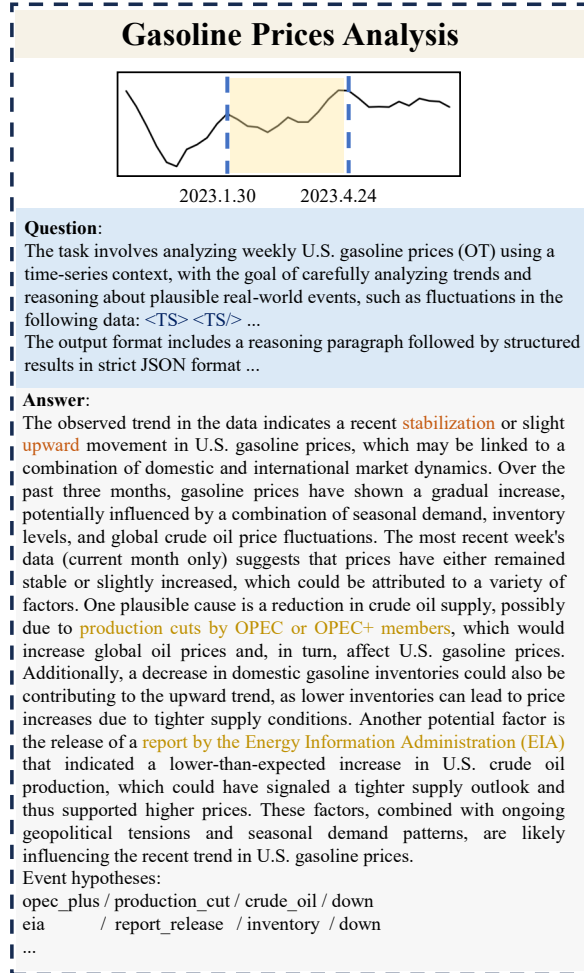


Figure 3: Case study on real-world gasoline prices data (energy dataset).

LLMs have limited performance in native numerical reasoning tasks and often fail to capture

local trends and subtle fluctuations (Shrestha et al., 2025; Boye and Moell, 2025). To address this, we introduce an alignment module that maps numerical dynamics into the LLM semantic space. Given the numerical time-series window $X_{(t-m+1,t)}$, we first segment the series into N non-overlapping patches $P \in \mathbb{R}^{N \times L_p}$. To extract numerical features, we employ an MLP-based time-series encoder f_ϕ . The processed patches are projected to the LLM’s embedding dimension D via a learnable linear projection layer $\mathbf{W}_{proj}$, yielding the numerical representation matrix $\mathbf{H}_{ts} \in \mathbb{R}^{N \times D}$:

$$\mathbf{H}_{ts} = [\mathbf{h}_1^{ts}, \dots, \mathbf{h}_N^{ts}]^\top = f_\phi(P) \mathbf{W}_{proj} + \mathbf{b}_{proj}, \quad (7)$$

where $\mathbf{h}_k^{ts} \in \mathbb{R}^D$ denotes the aligned embedding for the k -th numerical patch. Simultaneously, we construct the textual input by inserting special placeholder tokens (e.g., <TS>) into the instruction T_{text} at positions corresponding to numerical mentions. The modified sequence $T' = \{t_1, \dots, t_L\}$ is converted into dense representations using the LLM’s pre-trained embedding matrix $\mathbf{E}_{vocab}$, resulting in the textual representation matrix $\mathbf{H}_{text} \in \mathbb{R}^{L \times D}$:

$$\mathbf{H}_{text} = [\mathbf{h}_1^{text}, \dots, \mathbf{h}_L^{text}]^\top = T' \times \mathbf{E}_{vocab}, \quad (8)$$

where $\mathbf{h}_i^{text} \in \mathbb{R}^D$ represents the initial embedding of the i -th token. To enable fine-grained interaction, we employ an interleaved injection strategy. The final multimodal input embeddings $\mathbf{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_L\}$ are constructed by dynamically substituting the placeholder embeddings in $\mathbf{H}_{text}$ with the corresponding aligned patch embeddings from $\mathbf{H}_{ts}$:

$$\mathbf{z}_i = \begin{cases} \mathbf{h}_k^{ts} & \text{if } t_i \text{ is the } k\text{-th placeholder token,} \\ \mathbf{h}_i^{text} & \text{otherwise.} \end{cases} \quad (9)$$

This formulation ensures that continuous numerical signals are spatially aligned with their linguistic context, enabling the LLM to perform joint reasoning over both modalities. The decoder is trained to first produce a natural-language reasoning paragraph that summarizes the observed numerical dynamics and then output the final AAOD event set. The structured output serves as a normalized summary of the model’s reasoning trace.

In terms of training strategy, we adopt a multi-stage process to ensure stable learning of numerical representations and progressive semantic alignment. In the first stage, only the time-series encoder is trained, enabling it to acquire numerical

representations without interference from the large parameter space of the language model. In the next stage, the encoder parameters are frozen, and supervised fine-tuning (SFT) is applied to quickly adapt the model as a cold start. To foster deep reasoning, we further optimize the model using GRPO. Specifically, for each multimodal input $\mathbf{Z}$, we sample a group of G outputs $\{o_1, o_2, \dots, o_G\}$ from the current policy π_θ . We verify each output against the ground truth events to compute a reward r_i . The optimization objective is defined as:

$$\mathcal{L}_{\text{GRPO}} = \mathbb{E}_{\mathbf{Z} \sim \mathcal{D}} \left[\frac{1}{G} \sum_{i=1}^G \mathcal{J}_i \right], \quad (10)$$

$$\mathcal{J}_i = \min(\rho_i A_i, \text{clip}(\rho_i, 1 - \epsilon, 1 + \epsilon) A_i) - R_{\text{KL}}. \quad (11)$$

where $\rho_i = \frac{\pi_\theta(o_i|\mathbf{Z})}{\pi_{\theta_{\text{old}}}(o_i|\mathbf{Z})}$ is the probability ratio, $R_{\text{KL}} = \beta \mathbb{D}_{\text{KL}}$ is the KL regularization term, and ϵ is the clipping threshold. Crucially, the advantage A_i is computed using the group-relative standardization:

$$A_i = \frac{r_i - \text{mean}(\{r_1, \dots, r_G\})}{\text{std}(\{r_1, \dots, r_G\}) + \delta}. \quad (12)$$

This mechanism encourages the model to generate reasoning paths that yield higher accuracy relative to its own group samples. For further implementation details, please refer to Appendix A.2.

5 Experiments

In this section, we address the following research questions (RQs) to comprehensively evaluate the performance of our model:

RQ1 How does our model’s performance compare with other model baselines across different domains?

RQ2 What is the contribution of each component in our framework?

RQ3 Is the complex concurrent event synthetic data truly effective for the model?

RQ4 To what extent allows the framework for zero-shot generalization to unseen domains?

In the following subsections, we provide detailed experimental results and analysis to answer these questions respectively.

5.1 Datasets and Experimental Settings

We evaluate NUM2EVENT on three real-world domains: Energy (Gasoline Prices), Health

(Influenza), and Agriculture (Broiler Chicken Prices) (Liu et al., 2025). We use a chronological split: data before 2022 is used for training, and 2023–2024 is reserved for testing. Dataset descriptions are provided in Appendix A.1.

We use Qwen3-8B as the backbone LLM and fine-tune it with LoRA following the multi-stage protocol in Section 4.4. We report monthly averaged event-level Precision and Recall for all methods under the same structured decoding protocol. Matching rules, strict 4-slot evaluation, GRPO rewards, prompts, and hyperparameters are provided in Appendix A.2.

5.2 RQ1 How does our model’s performance compare with other model baselines across different domains?

Table 1 reports event-level precision and recall on the three real-world domains. We compare against closed-source LLMs, open-source LLMs, and time-series-oriented models, including ChatTS (Xie et al., 2025), TimeMQA (Kong et al., 2025), and ITformer (Wang et al., 2025b). All baselines are evaluated under the same input setting, schema constraints, output format, and matching protocol, as detailed in Appendix A.2.

The modest absolute scores across all systems show that open-ended number-to-event decoding remains challenging. Numerical patterns may support multiple plausible event hypotheses, while the aligned references record only a subset of the events associated with each window. Precision and Recall therefore measure the recovery of reference-aligned event hypotheses. The results show that task-specific adaptation improves structured event-hypothesis recovery across domains, with substantial room remaining for further improvement. Under this unified protocol, our method consistently outperforms the baselines across all domains.

5.3 RQ2 What is the contribution of each component in our framework?

We conduct three controlled ablations by removing one component at a time while keeping the remaining training and evaluation settings fixed. *w/o Synthetic data* removes the EveDTS-generated samples and trains only on real aligned data. *w/o TS encoder* removes the dedicated time-series encoder while retaining the raw numerical sequences as input to the LLM, thereby evaluating the benefit of explicit time-series representations. *w/o GRPO* omits the policy-optimization stage while retaining

Table 1: Event-level performance on real-world datasets.

Model	Size	Energy		Health		Agriculture	
		Precision(%)	Recall(%)	Precision(%)	Recall(%)	Precision(%)	Recall(%)
Closed-source models							
Claude Sonnet 4	N/A	17.50	13.76	11.76	27.06	15.00	30.94
Grok 4	N/A	12.50	10.55	2.35	3.43	12.50	28.13
Mistral Medium 3.1	N/A	15.00	11.70	11.76	19.90	10.00	26.04
Qwen3 Max	N/A	22.50	16.87	12.94	24.31	10.00	21.04
GPT-4.1 Mini	N/A	18.75	15.18	8.24	11.96	12.50	30.42
Open-source models							
Kimi k2	1T	21.25	17.16	9.41	16.67	2.50	4.69
DeepSeek V3.1	671B	11.25	7.81	4.71	6.18	12.50	25.42
GLM 4.6V	106B	18.13	12.66	7.06	9.80	6.25	11.46
Qwen3	32B	22.50	19.78	4.71	7.84	6.25	21.35
DS-R1-Distill-Qwen	32B	6.56	6.08	11.77	23.63	6.25	16.15
GPT-OSS	20B	16.56	13.72	5.88	11.37	7.50	15.63
Llama 4 Scout	17B	11.25	10.53	9.41	17.26	12.50	26.25
Qwen3	14B	10.00	8.64	7.06	15.39	2.50	2.81
DS-R1-Distill-Qwen	14B	10.99	13.75	6.96	8.43	13.75	10.99
Ministral 3	8B	8.75	7.12	8.24	12.55	7.50	16.35
Time-series models							
ChatTS	8B	16.25	11.43	4.71	5.88	3.75	5.73
TimeMQA	7B	4.38	2.14	9.31	10.49	7.19	10.42
ITformer	7B	6.25	5.36	2.35	2.65	1.25	2.08
Ours	8B	27.50	21.58	14.12	32.55	16.25	37.19

Table 2: Component ablation for event decoding.

Variant	Energy		Health		Agriculture	
	Precision(%)	Recall(%)	Precision(%)	Recall(%)	Precision(%)	Recall(%)
w/o Synthetic data	17.50	14.92	10.59	23.73	12.50	28.33
w/o TS encoder	23.75	17.72	11.76	26.47	15.00	34.06
w/o GRPO	22.50	17.63	10.59	22.84	11.25	26.77
Ours (full)	27.50	21.58	14.12	32.55	16.25	37.19

supervised fine-tuning.

As shown in Table 2, removing any of the three components reduces both Precision and Recall across all domains. The largest degradations are generally observed when synthetic augmentation or GRPO is removed, particularly for Recall on Health and Agriculture. Removing the time-series encoder produces smaller but consistent declines, showing that explicit time-series representations provide additional benefits beyond directly feeding raw numerical sequences to the LLM. These results indicate that the three components play complementary roles in the complete training pipeline.

All variants are evaluated on the same real held-out test sets. The improvements from synthetic augmentation therefore reflect the transfer of augmented training signals to real numerical windows. Additional discussion of synthetic-data diagnostics is provided in Appendix C.

5.4 RQ3 Is the complex concurrent event synthetic data truly effective for the model?

To examine whether modeling dependencies among concurrent events improves synthetic augmentation, we compare the full EveDTS generator with event interactions (WI) against a variant that uses only baseline event intensity (BI). As shown in Table 3, incorporating event interactions consistently improves both Precision and Recall across all three domains. This result indicates that modeling dependencies among event types produces synthetic training trajectories that better support downstream event decoding than modeling their background occurrence rates alone.

5.5 RQ4 To what extent allows the framework for zero-shot generalization to unseen domains?

To evaluate cross-domain generalization, we train NUM2EVENT on one source domain and evaluate

Table 3: Effectiveness of concurrent event modeling. We compare the performance of method with interactions (WI) against the variant considering only baseline intensity (BI) across three domains.

	Energy		Health		Agriculture	
	Precision(%)	Recall(%)	Precision(%)	Recall(%)	Precision(%)	Recall(%)
BI	21.25	15.00	10.59	23.82	13.75	34.06
WI	27.50	21.58	14.12	32.55	16.25	37.19

Table 4: Cross-domain experiment to validate the generalization ability of our framework. Each row represents a different training dataset, while each column corresponds to a test dataset. The row labeled "w/o Training" represents the baseline model, which has not undergone any optimization using our framework.

Training Dataset	Energy		Health		Agriculture	
	Precision(%)	Recall(%)	Precision(%)	Recall(%)	Precision(%)	Recall(%)
Energy	27.50	21.58	11.76	26.96	12.50	26.25
Health	17.50	14.10	14.12	32.55	12.50	31.25
Agriculture	16.25	13.04	10.59	24.41	16.25	37.19
w/o Training	12.50	9.60	5.88	12.45	3.75	9.38

it on each target domain without additional target-domain training. In Table 4, the diagonal cells correspond to in-domain evaluation, while the off-diagonal cells represent transfer to unseen domains. All cross-domain transfer settings improve both Precision and Recall over the unadapted model (*w/o Training*), indicating that the learned alignment between numerical dynamics and structured event hypotheses is partially transferable across domains. The in-domain model remains strongest on each target domain.

6 Conclusion

In this paper, we introduced number-to-event reasoning and decoding, which infers structured event hypotheses from numerical time-series when textual context is unavailable. We proposed NUM2EVENT, a reasoning-aware framework combining agent-guided event extraction, Hawkes-based synthetic augmentation, and multi-stage fine-tuning with a time-series encoder and LLM. Experiments on three real-world domains demonstrate that our method significantly outperforms strong LLM baselines in event-level precision, recall, and interpretability, while producing explicit reasoning traces that link quantitative changes to semantic event hypotheses.

Limitations

NUM2EVENT infers plausible event hypotheses from numerical dynamics. Numerical observations alone rarely identify unique real-world causes: multiple latent factors may produce similar time-series patterns, and the aligned reference events used for

evaluation may cover only part of the valid explanation space. Our evaluation therefore measures recovery of salient aligned event semantics under ambiguous evidence.

The framework also relies on an AAOD schema with controlled candidate sets. AGE includes dynamic schema expansion, while scaling to domains with much larger or rapidly changing event vocabularies remains an open challenge.

References

- Menouar Azib, Benjamin Renard, Philippe Garnier, Vincent Génot, and Nicolas André. 2024. [Event detection in time series: Universal deep learning approach](#). *Preprint*, arXiv:2311.15654.
- Emmanuel Bacry, Iacopo Mastromatteo, and Jean-François Muzy. 2015. [Hawkes processes in finance](#). *Preprint*, arXiv:1502.04592.
- André Bauer, Simon Trapp, Michael Stenger, Robert Leppich, Samuel Kounev, Mark Leznik, Kyle Chard, and Ian Foster. 2024. [Comprehensive exploration of synthetic data generation: A survey](#). *Preprint*, arXiv:2401.02524.
- Johan Boye and Birger Moell. 2025. [Large language models and mathematical reasoning failures](#). *Preprint*, arXiv:2502.11574.
- Kay H. Brodersen, Fabian Gallusser, Jim Koehler, Nicolas Remy, and Steven L. Scott. 2015. [Inferring causal impact using bayesian structural time-series models](#). *The Annals of Applied Statistics*, 9(1).
- Yuxi Cai, Lan Li, Yize Wang, and Guodong Li. 2025. [An efficient and interpretable autoregressive model for high-dimensional tensor-valued time series](#). *Preprint*, arXiv:2506.01658.

- Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. [Chatbot arena: An open platform for evaluating llms by human preference](#). *Preprint*, arXiv:2403.04132.
- Abdul Monaf Chowdhury, Rabeya Akter, and Safaeid Hossain Arib. 2025. [T3time: Tri-modal time series forecasting via adaptive multi-head alignment and residual fusion](#). *Preprint*, arXiv:2508.04251.
- Philippe Goulet Coulombe and Karin Klieber. 2025. [Opening the black box of local projections](#). *Preprint*, arXiv:2505.12422.
- Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. 2024. [A decoder-only foundation model for time-series forecasting](#). *Preprint*, arXiv:2310.10688.
- Vijay Ekambaram, Arindam Jati, Pankaj Dayama, Sumanta Mukherjee, Nam H. Nguyen, Wesley M. Gifford, Chandra Reddy, and Jayant Kalagnanam. 2024. [Tiny time mixers \(tms\): Fast pre-trained models for enhanced zero/few-shot forecasting of multivariate time series](#). *Preprint*, arXiv:2401.03955.
- Fanzhe Fu, Junru Chen, Jing Zhang, Carl Yang, Lvbin Ma, and Yang Yang. 2024. [Are synthetic time-series data really not as good as real data?](#) *Preprint*, arXiv:2402.00607.
- Simon Gottschalk and Elena Demidova. 2019. [Happening: Happen, predict, infer – event series completion in a knowledge graph](#). *Preprint*, arXiv:1909.06219.
- Yong Guan, Hao Peng, Xiaozhi Wang, Lei Hou, and Juanzi Li. 2024. [Openep: Open-ended future event prediction](#). *Preprint*, arXiv:2408.06578.
- Shule Hao, Junpeng Bao, and Chuncheng Lu. 2025. [A time series multitask framework integrating a large language model, pre-trained time series model, and knowledge graph](#). *Preprint*, arXiv:2503.07682.
- Tomas Horych, Christoph Mandl, Terry Ruas, Andre Greiner-Petter, Bela Gipp, Akiko Aizawa, and Timo Spinde. 2025. [The promises and pitfalls of llm annotations in dataset labeling: a case study on media bias detection](#). *Preprint*, arXiv:2411.11081.
- 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. 2024. [Time-llm: Time series forecasting by reprogramming large language models](#). *Preprint*, arXiv:2310.01728.
- Yaxuan Kong, Yiyuan Yang, Yoontae Hwang, Wenjie Du, Stefan Zohren, Zhangyang Wang, Ming Jin, and Qingsong Wen. 2025. [Time-mqa: Time series multi-task question answering with context enhancement](#). *Preprint*, arXiv:2503.01875.
- Viet Dac Lai. 2022. [Event extraction: A survey](#). *Preprint*, arXiv:2210.03419.
- Rafael Lima. 2021. [Hawkes processes modeling, inference and control: An overview](#). *Preprint*, arXiv:2011.13073.
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. [Visual instruction tuning](#). *Preprint*, arXiv:2304.08485.
- 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. 2025. [Time-mmd: Multi-domain multimodal dataset for time series analysis](#). *Preprint*, arXiv:2406.08627.
- Moxuan Ma, Muyu Wang, Lan Wei, Xiaolu Fei, and Hui Chen. 2025. Multi-modal fusion model for time-varying medical data: Addressing long-term dependencies and memory challenges in sequence fusion. *Journal of Biomedical Informatics*, 165:104823.
- A Craig MacKinlay. 1997. Event studies in economics and finance. *Journal of economic literature*, 35(1):13–39.
- Zijie Pan, Yushan Jiang, Sahil Garg, Anderson Schneider, Yuriy Nevmyvaka, and Dongjin Song. 2024. [S²ip-llm: Semantic space informed prompt learning with llm for time series forecasting](#). *Preprint*, arXiv:2403.05798.
- Sehyuk Park, Soyeon Caren Han, and Eduard Hovy. 2026. [Unicast: A unified framework for instance-conditioned multimodal time-series forecasting](#). *Preprint*, arXiv:2508.11954.
- Chitta Ranjan, Mahendranath Reddy, Markku Mustonen, Kamran Paynabar, and Karim Pourak. 2019. [Dataset: Rare event classification in multivariate time series](#). *Preprint*, arXiv:1809.10717.
- Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Hena Ghonia, Rishika Bhagwatkar, Arian Khorasani, Mohammad Javad Darvishi Bayazi, George Adamopoulos, Roland Riachi, Nadhir Hassen, Marin Biloš, Sahil Garg, Anderson Schneider, Nicolas Chapados, Alexandre Drouin, Valentina Zantedeschi, Yuriy Nevmyvaka, and Irina Rish. 2024. [Lag-llama: Towards foundation models for probabilistic time series forecasting](#). *Preprint*, arXiv:2310.08278.
- Feifei Shi, Xueyan Yin, Kang Wang, Wanyu Tu, Qifu Sun, and Huansheng Ning. 2025. [Large language models for time series analysis: Techniques, applications, and challenges](#). *Preprint*, arXiv:2506.11040.
- Safal Shrestha, Minwu Kim, and Keith Ross. 2025. [Mathematical reasoning in large language models: Assessing logical and arithmetic errors across wide numerical ranges](#). *Preprint*, arXiv:2502.08680.
- Saksham Sahai Srivastava and Vaneet Aggarwal. 2025. [A technical survey of reinforcement learning techniques for large language models](#). *Preprint*, arXiv:2507.04136.

- Mingtian Tan, Mike A. Merrill, Zack Gottesman, Tim Althoff, David Evans, and Tom Hartvigsen. 2026. [Inferring events from time series using language models](#). *Preprint*, arXiv:2503.14190.
- Hua Tang, Chong Zhang, Mingyu Jin, Qinkai Yu, Zhen-ting Wang, Xiaobo Jin, Yongfeng Zhang, and Mengnan Du. 2025. Time series forecasting with llms: Understanding and enhancing model capabilities. *ACM SIGKDD Explorations Newsletter*, 26(2):109–118.
- Venkat Venkatasubramanian, Raghunathan Rengaswamy, Kewen Yin, and Surya N Kavuri. 2003. A review of process fault detection and diagnosis: Part i: Quantitative model-based methods. *Computers & chemical engineering*, 27(3):293–311.
- Chengsen Wang, Qi Qi, Jingyu Wang, Haifeng Sun, Zirui Zhuang, Jinming Wu, Lei Zhang, and Jianxin Liao. 2025a. Chattime: A unified multimodal time series foundation model bridging numerical and textual data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 12694–12702.
- Xinlei Wang, Maike Feng, Jing Qiu, Jinjin Gu, and Junhua Zhao. 2024. [From news to forecast: Integrating event analysis in llm-based time series forecasting with reflection](#). *Preprint*, arXiv:2409.17515.
- Yilin Wang, Peixuan Lei, Jie Song, Yuzhe Hao, Tao Chen, Yuxuan Zhang, Lei Jia, Yuanxiang Li, and Zhongyu Wei. 2025b. [Ifformer: Bridging time series and natural language for multi-modal qa with large-scale multitask dataset](#). *Preprint*, arXiv:2506.20093.
- Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. 2024. [Chatie: Zero-shot information extraction via chatting with chatgpt](#). *Preprint*, arXiv:2302.10205.
- Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. 2024. [Unified training of universal time series forecasting transformers](#). *Preprint*, arXiv:2402.02592.
- Zhe Xie, Zeyan Li, Xiao He, Longlong Xu, Xidao Wen, Tieying Zhang, Jianjun Chen, Rui Shi, and Dan Pei. 2025. [Chatts: Aligning time series with llms via synthetic data for enhanced understanding and reasoning](#). *Proceedings of the VLDB Endowment*, 18(8):2385–2398.
- Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. 2024. [A survey on multimodal large language models](#). *National Science Review*, 11(12).
- Haoran Zhang, Yong Liu, Yunzhong Qiu, Haixuan Liu, Zhongyi Pei, Jianmin Wang, and Mingsheng Long. 2025a. Timesbert: A bert-style foundation model for time series understanding. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 10975–10983.
- Xiyuan Zhang, Boran Han, Haoyang Fang, Abdul Fatir Ansari, Shuai Zhang, Danielle C. Maddix, Cuixiong Hu, Andrew Gordon Wilson, Michael W. Mahoney, Hao Wang, Yan Liu, Huzefa Rangwala, George Karypis, and Bernie Wang. 2025b. [When does multimodality lead to better time series forecasting?](#) *Preprint*, arXiv:2506.21611.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

A Dataset and Experiment

A.1 Dataset

Table 5 summarizes the three real-world datasets used in our experiments, including their domain descriptions, temporal coverage, sampling frequency, and data sources.

A.2 Evaluation and Training Objectives

As mentioned in the main text, we report monthly averaged Precision and Recall. This aggregation strategy smooths out short-term volatility and provides a indicator of the model’s long-horizon temporal reasoning ability.

Event Matching and Evaluation Protocol.

Given the complexity of open-ended event decoding, requiring an exact string match for all four AAOD slots is often overly stringent. This is not only due to synonymous variations but also because many events are intrinsically neutral (e.g., "meeting held"), making the "Direction" slot latent or ambiguous. To address this, we define a predicted event as a correct hit (true positive) if it matches the ground truth in the three primary semantic slots: Actor, Action, and Object in Table 1. This criterion ensures that the evaluation prioritizes the core event semantics while allowing reasonable flexibility for auxiliary or often latent attributes, such as Direction. To provide a conservative and transparent assessment, we report the most rigorous results in Table 6, which requires an exact string match across all four AAOD slots simultaneously. Even under this strictest baseline, our model consistently outperforms existing methods. For constructing reasoning-augmented supervision, we employ GPT-4o-mini to generate intermediate reasoning chains from the paired data, which are incorporated into the training set to guide the model toward interpretable reasoning, with the specific prompt design

Table 5: Detailed Information of Datasets Used in the Study

Dataset Name	Description	Time Span	Frequency	Source
Gasoline Prices	Weekly U.S. gasoline prices, reflecting market trends influenced by factors like international oil prices, energy policies, and demand.	1996 - May 2024	Weekly	U.S. Energy Information Administration (EIA)
Influenza Patients Proportion	Proportion of influenza patients in the U.S., providing insights into seasonal trends and flu outbreaks.	1997 - May 2024	Weekly	Centers for Disease Control and Prevention (CDC)
Retail Broiler Composite	Monthly U.S. retail prices for broiler chickens, affected by production costs, demand, and supply chain changes.	1983 - May 2024	Monthly	U.S. Department of Agriculture

and generated examples illustrated in Figure 8. All baseline models are evaluated under the same task input, instruction template, schema constraints, output format, and matching script. The main comparison therefore holds the structured event-inference protocol fixed across models.

Output Schema and Post-processing. For each numerical window, the model outputs a set of AAOD events. Each event is represented as a four-slot tuple: Actor, Action, Object, and Direction. During evaluation, the generated response is first parsed into candidate event tuples. We normalize trivial surface variations such as extra whitespace and duplicated separators, and remove exact duplicate event tuples within the same window. Malformed outputs that cannot be parsed into the required slot structure are treated as invalid predictions. The predicted events are evaluated as an unordered set, since multiple events may occur within the same numerical window and their output order is not semantically meaningful. This set-based treatment is also consistent with the training objective, where the model is encouraged to recover the relevant event semantics rather than a particular ordering of event mentions. The natural-language reasoning paragraph is retained for interpretability, while the final structured AAOD set is used for automatic scoring.

Reasoning-Augmented Supervision. For supervised fine-tuning, each training instance contains the numerical input, a short reasoning chain, and the final AAOD event set. The reasoning chain summarizes salient numerical movements and links them to candidate event semantics. This format encourages the model to expose an intermediate explanation before producing the structured output, while keeping the final event representation standardized for evaluation.

GRPO Training Reward. During the subsequent GRPO training phase, we implement a fine-grained slot-level reward mechanism to provide dense feedback. Specifically, the reward $R(\hat{e}, e^*)$ for a predicted event $\hat{e}$ given the ground truth e^* is defined as:

$$R(\hat{e}, e^*) = \begin{cases} -\rho & \text{if } \hat{e} \notin \mathcal{F}_{\text{valid}} \\ \sum_{s \in \mathcal{S}} \mathbb{I}(\hat{v}_s = v_s^*) & \text{if } \hat{e} \in \mathcal{F}_{\text{valid}} \end{cases} \quad (13)$$

Here, $\mathcal{F}_{\text{valid}}$ represents the set of syntactically valid outputs. If the format is correct, the model receives **+1 point** for each slot $s \in \mathcal{S}$ (where $\mathcal{S} = \{\text{Actor, Action, Object, Direction}\}$) that matches the ground truth value v_s^* , yielding a maximum possible score of **+4**. Conversely, to enforce structural consistency, we impose a penalty ρ for format violations.

A.3 Model Parameter

We employ Qwen3-8B as our backbone large language model. To balance performance and computational efficiency, we utilize Low-Rank Adaptation for fine-tuning. The time-series encoder is implemented as a 4-layer MLP with a hidden dimension of 128. Furthermore, we investigate the impact of different encoder patch length partitions on model performance, with the detailed experimental results reported in Table 7.

A.4 EveDTS Parameter Estimation

All model parameters are empirically derived from the training data using econometric techniques, avoiding heuristic tuning.

1. Baseline Intensity (μ_k): We estimate time-varying baseline intensities $\mu_k(t)$ via Maximum Likelihood Estimation (MLE). The model explicitly incorporates week-of-year seasonal factors (γ_{woy}) to distinguish deterministic seasonal pat-

Table 6: Event-level performance on real-world datasets under the strict 4-slot match criterion.

Model	Size	Energy		Health		Agriculture	
		Precision(%)	Recall(%)	Precision(%)	Recall(%)	Precision(%)	Recall(%)
Closed-source models							
Claude Sonnet 4	N/A	15.00	10.08	5.88	6.47	6.25	8.85
Grok 4	N/A	8.75	6.17	2.35	3.43	5.00	7.29
Mistral Medium 3.1	N/A	8.75	6.23	2.35	2.35	3.75	5.73
Qwen3 Max	N/A	13.75	9.69	5.88	6.47	2.50	3.13
GPT-4.1 Mini	N/A	13.75	9.15	2.35	2.65	5.00	7.29
Open-source models							
Kimi k2	1T	13.75	9.95	1.18	1.47	0.00	0.00
DeepSeek V3.1	671B	6.25	5.17	3.53	3.82	6.25	9.58
GLM 4.6V	106B	11.56	8.78	1.17	1.17	3.75	4.68
Qwen3	32B	15.00	10.84	1.18	1.18	3.75	5.21
DS-R1-Distill-Qwen	32B	3.75	3.44	3.53	3.53	3.75	5.21
GPT-OSS	20B	11.25	8.01	1.18	2.94	2.50	3.65
Llama 4 Scout	17B	8.75	6.64	2.35	2.35	5.00	7.29
Qwen3	14B	8.75	6.34	1.18	1.47	1.25	1.56
DS-R1-Distill-Qwen	14B	8.75	7.46	1.96	1.47	1.25	1.56
Minstral 3	8B	2.50	2.14	0.00	0.00	0.00	0.00
Time-series models							
ChatTS	8B	11.25	7.16	1.18	1.18	1.25	1.56
TimeMQA	7B	4.38	2.14	5.78	5.00	0.00	0.00
ITformer	7B	6.25	4.67	0.00	0.00	0.00	0.00
Ours	8B	15.00	11.93	5.88	9.71	8.75	12.50

Table 7: Impact of patch length on event decoding performance (Energy dataset).

Patch Size	Patch = 1	Patch = 2	Patch = 3	Patch = 4
Precision(%)	17.50	27.50	25.63	21.25
Recall(%)	13.74	21.58	20.60	18.22

terns (e.g., scheduled reports) from stochastic event bursts.

2. Hawkes Interactions ($\alpha_{k,j}$): Interaction coefficients are estimated using a Multivariate Generalized Linear Model with a non-parametric discrete lag structure. To ensure stability and physical validity given data sparsity, we apply Ridge Regression combined with Non-Negative Least Squares constraints. This prevents overfitting to spurious correlations.

3. Impulse Response Functions (β_k): Causal impacts are estimated via Local Projections, which are useful to model misspecification. We utilize Newey-West HAC standard errors to identify and filter statistically insignificant responses, ensuring that synthesized samples are driven by event signals rather than noise.

B Additional Validity Evaluation

Reference-set matching is informative but incomplete for this task, because numerical observations may admit multiple plausible event interpretations and the annotated references may not cover ev-

ery valid explanation. To complement event-level precision and recall, we conduct a blind pairwise LLM-as-judge evaluation following the general evaluation paradigm of MT-Bench and Chatbot Arena (Zheng et al., 2023; Chiang et al., 2024).

For each Energy-domain test sample, the judge receives the full output from our model and GPT-4.1-mini, including both the natural-language reasoning paragraph and the final predicted event set. The two outputs are randomly ordered and anonymized. The judge scores each side on a 1–4 scale along four dimensions: hypothesis schema validity, causal logic validity, reasoning restraint, and predicted-vocabulary precision.

Table 8: Blind pairwise LLM-as-judge evaluation on the Energy domain. Scores are averaged over test samples.

Dimension	Ours	GPT-4.1-mini
Hypothesis Schema Validity	2.06	1.63
Causal Logic Validity	2.81	2.69
Reasoning Restraint	2.44	2.31
Pred. Vocabulary Precision	2.00	1.44
Total (out of 16)	9.31	8.06

Our model obtains higher average scores on all four dimensions. This result provides complementary evidence that the predicted event hypotheses are not merely artifacts of exact reference matching: an independent blind judge can distinguish the schema validity, causal consistency, and restraint of the generated outputs.

C Synthetic Data Diagnostics

We further analyze whether EveDTS introduces generator-specific event artifacts. The synthetic data are used only for training augmentation, while all reported test results are computed on real held-out windows. To inspect the event vocabulary introduced by the generator, we compare the AAOD keys appearing in synthetic training samples with those appearing in real training samples on the Energy domain.

The diagnostic shows that the synthetic event vocabulary is fully contained within the real training vocabulary: every AAOD key appearing in the synthetic samples also appears in the real aligned training data, with no synthetic-only event keys. In addition, the real data cover a broader range of actors, actions, and objects than the synthetic data. This suggests that EveDTS does not introduce novel out-of-distribution event types; instead, it provides additional training signal within the event space already observed in real data.

Combined with the w/o Synthetic data ablation in Table 2 and the cross-domain transfer results in Table 4, this diagnostic supports the interpretation that the model learns transferable numerical-to-event reasoning patterns beyond synthetic-only artifacts.

Hawkes-process and impulse-response assumptions may still miss important real-world event dynamics. Sensitivity analysis over different synthetic-data ratios and generator families remains an important direction for future work.

D Prompts and case-study

In this section, we provide detailed visualizations of the prompts employed in our framework and an additional qualitative case study to support reproducibility.

Prompt for Gasoline Prices Analysis

You are given weekly U.S. gasoline prices (OT):

- Context window (last 3 months, weekly OT):

<ts3m></ts3m>

- Current month only (weekly dOT):

<tsdot></tsdot>

Task:

1. Carefully analyze the time-series trends and reason step-by-step about plausible real-world causes.
2. Write a **reasoning paragraph** summarizing your interpretation of the data trend and potential driving factors.
 - This paragraph should sound like an analytical narrative (e.g., "The observed trend in the data indicates...").
 - It should include your reasoning path that connects the observed pattern to possible events.
3. Then, hypothesize up to 5 ****REAL-TIME events**** that occurred **WITHIN THE CURRENT MONTH ONLY**.

Output format:

- First, write your reasoning paragraph.

- After that, write the final structured results after **«FINAL»** in strict JSON format:

«FINAL»

```
{
  "hypotheses": [
    {
      "key": "NAME|ACTION|OBJECT|DIRECTION",
    }
  ]
}
```

Allowed token sets:

('NAME ∈ {market, eia, opec_plus, united_states}')

'ACTION ∈ {report_release, price_change, production_cut, forecast_lower, forecast_lower, forecast_raise, production_raise, extend_cut, sanction_impose, purchase, release, export_ban_lift}')

'OBJECT ∈ {inventory, crude_oil, gasoline, price, production, natural_gas, inventory, diesel, export, jet_fuel, rig_count, brent, forecast}')

'DIRECTION ∈ {ambiguous, up, down}')

Rules:

- Do NOT fabricate tokens outside these sets.
- Use "|" (vertical bar), ****not underscores**** to separate the fields.
- Each "key" must strictly follow this pipe-separated pattern: NAME|ACTION|OBJECT|DIRECTION .
- The reasoning paragraph should be fluent and analytic, not a list of events.
- If the same subject has multiple events, output each event separately; do not merge them.

Figure 4: Prompt for Gasoline Prices Analysis.

Prompt for Influenza Patients Proportion Analysis

You are given weekly U.S. influenza patients proportion (OT):

- Context window (last 3 months, weekly OT):

<ts3m></ts3m>

- Current month only (weekly dOT):

<tsdot></tsdot>

Task:

1. Carefully analyze the time-series trends and reason step-by-step about plausible real-world causes.
2. Write a short **reasoning paragraph** summarizing your interpretation of the data trend and potential driving factors.
 - This paragraph should sound like an analytical narrative (e.g., "The observed trend in the data indicates...").
 - It should include your reasoning path that connects the observed pattern to possible events.
3. Then, hypothesize up to 5 ****REAL-TIME events**** that occurred **WITHIN THE CURRENT MONTH ONLY**.

Output format:

- First, write your reasoning paragraph.

- After that, write the final structured results after **«FINAL»** in strict JSON format:

«FINAL»

```
{
  "hypotheses": [
    {
      "key": "NAME|ACTION|OBJECT|DIRECTION",
    }
  ]
}
```

Allowed token sets:

('NAME' ∈ {centers_for_disease_control, united_states, world_health_organization, food_and_drug_administration, advisory_committee_on_immunization_practices, health_and_human_services, national_institute_on_aging}),

'ACTION' ∈ {issue_guidance, recommend_vaccination, report_release, confirm, approve_authorize, detect, conduct_study, rate_change, outbreak}),

'OBJECT' ∈ {vaccination, influenza, h5n1, hospitalizations, h1n1, vaccine, mortality, respiratory_syncytial_virus, cases, deaths}),

'DIRECTION' ∈ {ambiguous, up, down})

Rules:

- Do NOT fabricate tokens outside these sets.
- Use "|" (vertical bar), ****not underscores**** to separate the fields.
- Each "key" must strictly follow this pipe-separated pattern: NAME|ACTION|OBJECT|DIRECTION .
- The reasoning paragraph should be fluent and analytic, not a list of events.
- If the same subject has multiple events, output each event separately; do not merge them.

Figure 5: Prompt for Influenza Patients Proportion Analysis.

Prompt for Retail Broiler Composite Analysis

You are given monthly U.S. Retail Broiler Composite (OT):

- Context window (last 3 months, monthly OT):

<ts3m><ts3m/>

- Current month only (monthly dOT):

<tsdot><tsdot/>

Task:

1. Carefully analyze the time-series trends and reason step-by-step about plausible real-world causes.
2. Write a short **reasoning paragraph** summarizing your interpretation of the data trend and potential driving factors.
 - This paragraph should sound like an analytical narrative (e.g., “The observed trend in the data indicates...”).
 - It should include your reasoning path that connects the observed pattern to possible events.
3. Then, hypothesize up to 5 **REAL-TIME events** that occurred **WITHIN THE CURRENT MONTH ONLY**.

Output format:

- First, write your reasoning paragraph.

- After that, write the final structured results after **«FINAL»** in strict JSON format:

«FINAL»

```
{
  "hypotheses": [
    {
      "key": "NAME|ACTION|OBJECT|DIRECTION",
    }
  ]
}
```

Allowed token sets:

('NAME ∈ {market, united_states, us_government_agency, poultry_industry_association, poultry_company, claxton_poultry_farms}',

'ACTION ∈ {price_action, forecast_action, production_action, trade_policy_action, health_event_action, report_action, subsidy_action, approval_action}',

'OBJECT ∈ {price, production, export, capacity, disease_event, tariff, import, poultry_product, guideline, subsidy}',

'DIRECTION ∈ {ambiguous, up, down}')

Rules:

- Do NOT fabricate tokens outside these sets.
- Use "|" (vertical bar), **not underscores** to separate the fields.
- Each "key" must strictly follow this pipe-separated pattern: NAME|ACTION|OBJECT|DIRECTION .
- The reasoning paragraph should be fluent and analytic, not a list of events.
- If the same subject has multiple events, output each event separately; do not merge them.

Figure 6: Prompt for Retail Broiler Composite Analysis.

Prompt for Reasoning Chains

You are an {{corresponding dataset}} event extraction assistant. Output strictly JSON.

Goal: Return the simplest possible events. For each distinct event in the text, extract ONLY these fields:

- {{name vocabulary}}
- {{action vocabulary}}
- {{object vocabulary}}
- {{direction vocabulary}}
- date (YYYY-MM-DD if explicitly evident; otherwise return null or omit the field; do not make up dates)
- summary (one concise sentence capturing the most important part about this event)
- Do NOT set name to a country just because the text mentions a geography (e.g., "U.S. inventories"). If it is a report or data, set name to the agency. Treat geographies as scope, not actor.
- For price-only market color, set name="market".
- Use "united_states" only when the U.S. government is the agent.
- If no clear agent, please reply with the "name" you think should be selected in the "vocab_suggestion" section.

Vocabulary suggestion rules:

- If a clearly relevant term is NOT in allowed_values, return it in "vocab_suggestion" with:
 - "term": raw phrase (lowercased),
 - "slot": one of ("names", "actions", "objects", "directions"),
 - "score": confidence 0.0–1.0.
- Return only ONE best candidate; if no candidate, return {"term": none, "slot": none, "score": 0.0}.

Hard rules:

- 1) Output JSON only, no extra text.
- 2) Use only allowed_values for name/action/object/direction.
- 3) One input may yield multiple events; avoid duplicates with same (name, action, object, direction, date, summary).

Return shape:

```
{
  "events": [
    {
      "name": "united states",
      "action": "report_release",
      "object": "price",
      "direction": "up",
      "date": "1998-01-02",
      "summary": "United States released a report indicating gradual increases in retail gasoline prices."
    }
  ],
  "vocab_suggestion": {"term": , "slot": , "score": }
}
```

Figure 7: Prompt for AGE module.

Prompt for Reasoning Chains

ROLE

You are an explainer. You justify given AAOD events from time-series patterns.

INPUT

- Last 3 months (weekly OT): {ts3m_array}
- Current month only (weekly dOT): {tsdot_array}

GOLD_ANSWER (copy EXACTLY; do not modify)

```
<answer_json>
{answer_json}
</answer_json>
```

TASK

- 1) Write ONE paragraph starting with "Reason:" (4-6 sentences, ~40–120 words).
 - Justify ONLY using OT/dOT visual patterns: trend (up/down/flat), inflection points, volatility clusters, week-over-week signs, and relative magnitude. Do NOT quote specific numbers.
 - Do NOT mention that you were given the answer. No meta-talk.
- 2) Then print EXACTLY one line: «FINAL»
- 3) AFTER «FINAL», copy the JSON inside <answer_json> UNCHANGED (verbatim). No extra text.

RULES

- Do not invent new events; use the provided JSON as-is.
- If the JSON seems inconsistent, still copy it verbatim (the evaluator will handle validity).
- End output after the JSON.

BEGIN

Reason:

Figure 8: Prompt for using GPT-4o-mini to generate reasoning chain.

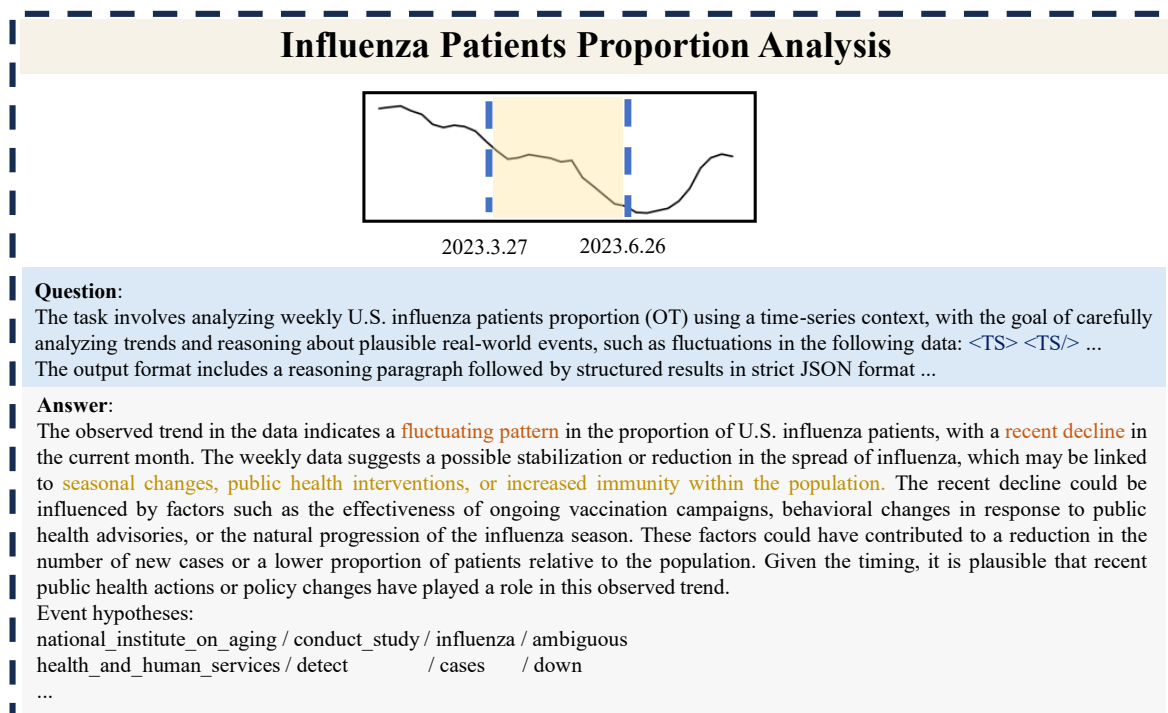


Figure 9: Case study on real-world influenza patients proportion data.