

Grammar of the Wave: Towards Explainable Multivariate Time Series Event Detection via Neuro-Symbolic VLM Agents

Sky Chenwei Wan^{1,2}, Yifei Y. Wang^{2*}, Tianjun Hou¹, Xiqing Chang¹, Aymeric Jan¹

¹AI Lab, SLB, France

²LTCl, Télécom Paris, Institut Polytechnique de Paris, France

cwan5@slb.com, thou2@slb.com

Abstract

Time Series Event Detection (TSED) aims to localize semantically meaningful events in time series data, with critical applications in high-stakes domains. Unlike statistical anomalies, events are often defined by natural-language descriptions with internal temporal-logic structures across multiple physical channels. However, in real-world settings, dense event annotations are expensive to obtain, making purely supervised learning difficult. We introduce **Language-guided TSED**, a setting where a model is given textual event descriptions and must ground them to intervals in multivariate signals with little or no labeled data. To address this problem, we propose **Event Logic Tree** (ELT), a knowledge representation framework that converts linguistic descriptions into structured temporal logic over signal primitives. Building on ELT, we present SELA, a neuro-symbolic VLM agent framework that iteratively grounds primitives from signal visualizations and composes them under ELT constraints, producing both event intervals and faithful tree-structured explanations. We further release a real-world benchmark across energy and climate domains with expert knowledge and annotations. Experiments show that SELA improves over supervised fine-tuning and existing zero/few-shot time series reasoning baselines¹.

1 Introduction

Time Series Event Detection aims to localize temporal segments in multivariate signals where specific events occur. It is important in high-stakes domains such as health (Supratak et al., 2017; Perslev et al., 2019) and energy production (Khaouja et al., 2025). Unlike time series classification, which assigns a global label to a sequence (Mohammadi Foumani et al., 2024), event detection

requires fine-grained localization at the segment level. It is also different from anomaly detection, which mainly targets statistical deviations (Zamanzadeh Darban et al., 2024). Events are semantically defined processes, often involving cross-channel temporal relations, such as “a sharp rise on channel A followed by a dip on channel B”.

Most existing approaches rely on supervised learning, including CNN or Transformer-based models (Perslev et al., 2019; Wu et al., 2023) and fine-tuning time series foundation models (Khaouja et al., 2025). However, in many industrial and scientific settings, dense event annotations are expensive to obtain, while textual event descriptions are often available from operational manuals or expert guidelines. This motivates a **language-guided setting**: given a multivariate time series and textual descriptions of target events, the model should ground the descriptions into event intervals without large amounts of labeled data (Figure 1).



Figure 1: An illustration of the Language-guided Time Series Event Detection task.

The central challenge is *grounding*. Textual descriptions are unstructured, such as the following description from the energy domain:

During the **Buildup** event, the *Volume* should remain stable across the entire phase. Simultaneously, the *Pressure* should either recover slightly with the rate of change decreasing or bounce up quickly, and then reach a steady phase.

*Work done during Yifei’s internship at SLB.

¹Our project is available at <https://github.com/cw-wan/SELA>.

This description is not a flat label. It defines atomic signal patterns, such as *Volume remains stable* and *Pressure recovers slightly*, and relates them through temporal-logic cues such as *simultaneously*, *either*, and *and then*. Therefore, an explainable detector should not only predict the global event interval, but also ground its sub-patterns to concrete signal segments as verifiable evidence.

To address this challenge, we introduce **Event Logic Tree (ELT)**, a knowledge representation framework that “translate” textual event descriptions into tree-structured temporal logic. ELT decomposes an event into atomic signal primitives and composes them through operators such as succession, synchronization, containment, and alternatives. Based on ELT, we build SELA, a neuro-symbolic agent system for zero-shot language-guided event detection. A *Logic Analyst* parses the textual description into an ELT schema, while a *Signal Inspector* grounds the schema in the observed time series by locating and refining event intervals. ELT serves as the central *intermediate representation*: it guides the grounding process and produces an instantiated proof tree that can be inspected by human experts.

To evaluate our approach, we release KITE, a real-world benchmark with expert-verified event descriptions and event annotations from energy and climate domains. Experiments show that SELA consistently improves event grounding and precise localization, demonstrating the benefit of ELT-guided reasoning.

Our contributions are summarized as follows:

- We formulate a language-guided setting for detecting semantically defined events in multivariate time series data.
- We propose **ELT**, a structured representation that helps faithfully ground textual event descriptions into time series intervals with interpretable primitive-level evidence. Based on this, we develop **SELA**, a neuro-symbolic agent system that uses ELT to perform zero-shot event semantic grounding.
- We release **KITE**, a real-world time series event detection benchmark with expert descriptions and annotations.

2 Problem Formulation

The objective is to detect all event instances in a multivariate time series that match a set of linguistic

event descriptions.

Input. The input consists of a multivariate time series $\mathbf{X} \in \mathbb{R}^{T \times C}$ and a description set $\mathcal{L} = \{\ell_e\}_{e \in \mathcal{E}}$, where T is the number of time steps, C is the number of channels, $\mathcal{E}$ is the event type set, and ℓ_e describes the signal pattern of event type e .

Ground Truth. The ground truth is a set of event instances $\hat{\mathcal{Y}} = \{\hat{y}_m\}_{m=1}^M$. Each instance is $\hat{y}_m = (\hat{I}^{(m)}, \hat{e}^{(m)})$, where $\hat{I}^{(m)} = [\hat{t}_{\text{on}}^{(m)}, \hat{t}_{\text{off}}^{(m)}]$ is the temporal interval and $\hat{e}^{(m)} \in \mathcal{E}$ is the event type.

Output. A detector f_θ predicts a set of candidate event instances $\mathcal{Y} = f_\theta(\mathbf{X}, \mathcal{L}) = \{y_k\}_{k=1}^K$. Each prediction is $y_k = (I^{(k)}, e^{(k)})$, where $I^{(k)} = [t_{\text{on}}^{(k)}, t_{\text{off}}^{(k)}]$ is the predicted interval and $e^{(k)} \in \mathcal{E}$ is the predicted event type.

Objective. Predictions are evaluated by matching them to ground-truth instances. A valid match must identify the correct event type, i.e., $e^{(i)} = \hat{e}^{(j)}$, and align with the corresponding ground-truth interval. Temporal alignment can be measured by interval IoU or boundary deviation. Thus, the task requires both event identification, determining *what* event occurs, and temporal localization, determining *when* it occurs.

3 Event Logic Tree

To achieve accurate event detection with faithful explanations, we need to explicitly model the internal structure of events. Based on the example in Section 1, we define three desiderata for such a logic structure:

D1. Hierarchical Representation. The representation should capture how atomic patterns recursively form sub-patterns and finally the global event.

D2. Semantic Quantification. The representation should quantify how well a signal morphology matches its linguistic semantics.

D3. Topological Elasticity. The representation should define events by their temporal-logic structure rather than absolute duration, allowing temporal stretching or compression while preserving logical validity.

To meet these requirements, we introduce **Event Logic Tree (ELT)**, a tree-structured representation that separates an event’s schema from its

Operator	Natural Language Description	Formalization	TS Instance
SEQ	“A <i>spike in A</i> is <i>followed by</i> a <i>drop in B</i> .” “A <i>drop in B</i> occurs <i>after</i> a <i>spike in A</i> .”	SEQ(<i>Spike_A</i> , <i>Drop_B</i>)	
SYNC	“A <i>square wave in A</i> is <i>synchronized with</i> a <i>spike in B</i> .” “A <i>square wave in A</i> and a <i>spike in B</i> occur <i>simultaneously</i> .”	SYNC(<i>SquareWave_A</i> , <i>Spike_B</i>)	
GUARD	“A <i>drop in A</i> is found <i>within</i> a <i>rise in B</i> .” “A <i>rise in B</i> <i>encompasses</i> a <i>drop in A</i> .”	GUARD(<i>Drop_A</i> , <i>Rise_B</i>)	
OR	“ <i>Either</i> a <i>spike in A</i> <i>or</i> a <i>drop in B</i> .” “ <i>At least one of</i> a <i>spike in A</i> <i>or</i> a <i>drop in B</i> occurs.”	OR(<i>Spike_A</i> , <i>Drop_B</i>)	

Table 1: Core operators of Event Logic Tree.

instantiation. The schema is parsed from the textual description only and specifies the event logic independently of any concrete time series sample. Each node is either a primitive or a composite:

$$\begin{aligned}
n_p &= \langle \tau, c \rangle, \\
n_\phi &= \langle [n_1, \dots, n_k], op \rangle, \\
op &\in \{\text{SEQ, SYNC, GUARD, OR}\}.
\end{aligned} \tag{1}$$

A primitive node n_p denotes an atomic morphology τ on a physical channel c , such as “a sharp pressure increase on the Pressure channel”. A composite node n_ϕ recursively combines child nodes through a temporal-logic operator. Inspired by Allen’s interval algebra (Allen, 1983), our operator set is adapted to the grounding need: SEQ models succession, SYNC synchronization, GUARD containment, and OR alternative descriptions of the same event. Examples are shown in Table 1, and the full schema and operator semantics are given in Appendix A.1. Figure 2 displays how the example description in Section 1 is parsed into an event logic tree schema.

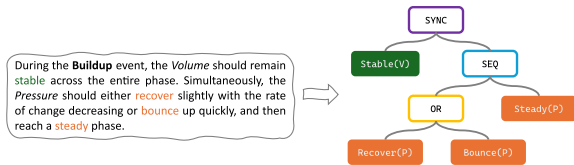


Figure 2: An example of ELT schema parsing.

Given a time series X , ELT is instantiated by grounding schema nodes to actual intervals. A primitive instance is defined as

$$\begin{aligned}
\hat{n}_p &= \langle n_p, I, \mu \rangle, \\
I &= [t_{\text{on}}, t_{\text{off}}], \\
\mu &= M_\tau(X, I, c) \in [0, 1],
\end{aligned} \tag{2}$$

where μ measures the semantic coherence between the segment $X[I, c]$ and the predicate τ . Composite

instances are computed bottom-up: their intervals span the intervals of their children, and their coherence scores are recursively aggregated by the corresponding operators. Thus, the root node represents the final event instance, with a numeric score reflecting both primitive-level grounding and global temporal consistency. Formal instantiation rules are provided in Appendix A.3.

To avoid physically invalid trees, ELT follows three validity principles: *constructive composition*, requiring each composite to add non-trivial structure; *temporal compactness*, preventing unbounded unexplained gaps inside a composite event; and *physical exclusivity*, preventing one channel from supporting incompatible primitive states at the same time. These principles constrain grounding while preserving temporal elasticity; their formal definitions are given in Appendix A.2.

Overall, ELT turns event detection into the construction of an instantiated proof tree: primitives provide signal-level evidence, operators organize the evidence into event-level logic, and confidence propagates from leaves to the root. We compare ELT with existing time series symbolic representations in Section 7.1.

4 The SELA Multi-Agent System

We build SELA, a neuro-symbolic VLM agent system that uses ELT as the intermediate representation for language-guided time series event detection. The architecture is shown in Figure 3. We explain the core concepts as follows.

ELT-guided execution. SELA directly builds on the ELT representation introduced in Section 3. The system first parses textual descriptions into logic trees. During grounding, SELA assigns intervals and semantic coherence scores to primitive

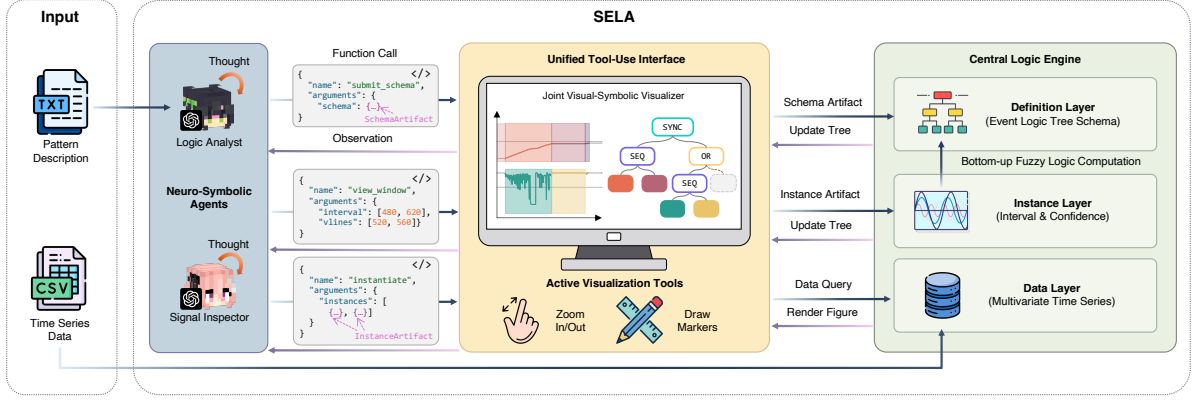


Figure 3: Overview of the SELA system.

nodes. These scores are then propagated bottom-up through the composite operator nodes, and the root node produces the final event interval and root confidence. In this way, the predictions are not just event intervals, but instantiated proof trees whose root summarizes both primitive-level semantic evidence and global temporal consistency.

Agent workflow. SELA decomposes the task into two role-specific agents. The *Logic Analyst* receives only the textual event description and submits an ELT schema. It does not access the time series, which prevents schema construction from being biased toward a particular sample. The *Signal Inspector* receives the compiled schema and grounds it in the observed time series. It uses visualization tools to inspect global and local signal views, places optional temporal markers, and submits primitive instances with intervals and confidence scores that represent the level of semantic coherence under the agent’s point of view. The backend updates the ELT status for each agent submission, recomputes all composite scores, and returns a joint visualization of the signal evidence and the current logic-tree state. Agent prompts are given in Appendix B.

Logic engine and confidence propagation. The SELA backend maintains three coupled states: the multivariate time series, the ELT schema, and the current primitive instantiations. Once the Signal Inspector submits primitive candidates, the logic engine recursively constructs the instantiated proof tree. Invalid compositions are filtered by hard gates, such as temporal-order violations or physical collisions on the same channel, while valid compositions receive confidence scores through the corresponding fuzzy operators. This makes any ground-

ing failures explicit: a low root confidence can be traced back to weak primitive evidence, poor temporal alignment, missing branches, or violated operator constraints.

When multiple mutually exclusive event candidates compete for the same record, we compare their normalized root confidence rather than the raw root confidence. This avoids unfairly penalizing a semantically richer tree only because it contains more conjunctive evidence. For an instantiated tree $\hat{T}$, we use

$$\begin{aligned} \mu_{\text{norm}}(\hat{T}) &= \mu_{\text{root}}(\hat{T})^{1/K(\hat{T})}, \\ K(\hat{T}) &= |\mathcal{L}^*(\hat{T})|, \end{aligned} \quad (3)$$

where $\mathcal{L}^*(\hat{T})$ denotes the active primitive leaves of the instantiated tree.

Error control with artifacts. Agents interact with the backend through structured function calls and typed artifacts (Qian et al., 2025). The Logic Analyst submits a schema artifact, while the Signal Inspector submits an instantiation artifact containing primitive aliases, intervals, and confidence scores. Before updating the backend state, SELA validates the schema, parses artifacts, and checks action outputs against predefined formats.

5 The KITE Dataset Collection

We introduce the KITE (Knowledge-Infused Time Series Events) benchmark, which is curated from two real-world domains: energy and climate.

5.1 Data Collection

KITE comprises two sub-datasets:

1. Pressure Test: Pressure Test is a critical well exploration operation in the oil & gas industry. The

objective is to acquire underground pressure at different depths to characterize the reservoir. During pressure test jobs, sensors are installed at certain depths in drilling wells for remote measurements. We collected a dataset of 62 multivariate time series samples from 41 real underground pressure test jobs in drilling wells in the North Sea. Pressure Test has two events: *valid test* and *lost seal*. Each time series sample has two channels: *volume* and *pressure*.

2. Climate Hazard: We created the Climate Hazard dataset by aligning two public datasets: the NOAA Integrated Surface Database (ISD)² and the NWS Storm Events database³. Through manual filtering and calibration, we acquired 86 multivariate time series samples with high-quality event labels. We selected three hazard event types with support from accessible, high-quality knowledge as labels: *dense fog*, *frost freeze* and *extreme cold*. Each climate time series sample has seven channels: *visibility*, *dew-point spread*, *air temperature*, *dew point*, *wind chill*, *wind speed*, and *sea-level pressure*.

5.2 Data Annotation

In general, the mission of data annotation is to obtain 1) the event labels (classes and boundaries), and 2) the event descriptions in natural language.

For Pressure Test, the collected data were annotated in two stages. First, two field engineers with over 20 years of experience in the oil & gas industry manually inspected every sample using Insitu Pro, a professional time series annotation software (Fries et al., 2024), to identify the classes and boundaries of all events. Their results were then cross-verified to resolve boundary disagreements. In the second stage, the field engineers documented textual guidance for event detection, assisted by an experienced AI engineer.

For Climate Hazard, we first took the event boundaries from the NWS Storm Events database. However, the original timestamps reflect zone-level conditions rather than point-station measurements, which can introduce timing offsets at the individual station level, causing minor misalignments with the NOAA database. To solve this, we aggregated and summarized the official hazard event knowl-

Set	Ratio	Length Statistics				
		Min	Q1	Median	Q3	Max
PT	<i>TS Sample</i>	467	944	1313	2228	14145
	lost seal	22.6	151	247	346	1945
	valid test	77.4	262	489	800	3386
CH	<i>TS Sample</i>	385	661	678	1093	1621
	dense fog	65.1	13	46	76	293
	extreme cold	25.6	228	259	384	453
	frost freeze	9.3	96	120	161	396

Table 2: Statistics by Dataset: **PT** = Pressure Test, **CH** = Climate Hazard. Lengths are event durations in samples; *TS Sample* is the series length. Ratio is the percentage of samples containing that class.

edge⁴⁵ as textual descriptions. We had two annotators with climate science background manually inspect the signal samples to filter out abnormal data and calibrate the remaining event boundaries to align with the event descriptions. The results were cross-verified between the two annotators to resolve disagreements. Our full pipeline of constructing climate data is detailed in Appendix C.

5.3 Dataset Statistics

We provide the statistical details of the two KITE datasets in Table 2. Overall, the time series samples are long, especially those in Pressure Test, which have a median length of 1313 timestamps. For Pressure Test and Climate Hazard, the label distributions are imbalanced, posing additional challenges for supervised learning. Moreover, both sample lengths and event durations vary substantially across the two datasets, requiring models to understand event semantics whose concrete signal realizations can be highly variable. This observation empirically supports the desideratum of *Topological Elasticity* in Section 3.

6 Experiments

We develop our analysis based on two research questions:

RQ 1. How well do LLMs/VLMs understand event structures from natural language descriptions?

RQ 2. Could Event Logic Tree improve VLM agents’ performance in time series event detection?

²<https://www.ncei.noaa.gov/data/global-hourly/access/>

³<https://www.ncei.noaa.gov/stormevents/>

⁴https://www.weather.gov/media/ajk/brochures/Wind_Chill_Temperature_Index.pdf

⁵https://www.weather.gov/media/directives/010_pdfs_archived/pd01016005c.pdf

6.1 RQ 1. Benchmarking ELT Parsing

Experimental Setup. We construct ground-truth ELTs for five events across the two KITE datasets. We evaluate four models: GPT-5⁶, GPT-4.1⁷, Qwen-3.5 27B⁸, and Gemma-4 31B⁹. Each model parses each event five times, yielding 100 parse runs in total.

Evaluation Dimensions. We first report the *pass rate* (Pass), the probability that a model produces a valid schema that can pass the syntax check within a fixed number of turns (20). Each valid parse is then compared against the human-parsed schema along four dimensions: *Structural Fidelity* (Struct), measuring whether the tree topology is logically equivalent to the ground truth; *Primitive Coverage* (Cov), measuring whether all ground-truth primitives are included; *Primitive Accuracy* (Acc), assessing whether each predicted primitive is grounded in the original description and penalising fabricated primitives; and *Semantic Alignment* (Sem), measuring how well each primitive’s description matches the corresponding part of the original event description. Scores are assigned on a 0–5 scale by two independent VLM judges, Claude Opus-4.6 and GPT-5.5, averaged and normalised to [0, 1], and validated by human. Implementation details of judges are provided in Appendix D.

Model	Pass	Struct	Acc	Cov	Sem
GPT-5	1.00	0.90	1.00	1.00	1.00
GPT-4.1	1.00	0.86	1.00	0.98	0.98
Qwen-3.5 27B	0.88	0.92	1.00	1.00	0.99
Gemma-4 31B	1.00	0.70	1.00	0.96	0.85

Table 3: ELT parse quality scores averaged over five independent runs and normalised.

Results. As shown in Table 3, all valid parses achieve perfect Primitive Accuracy (1.00). GPT-5 attains perfect Coverage and Semantic Alignment, with strong Structural Fidelity (0.90). Qwen-3.5 27B shows the weakest Pass Rate (0.88) but achieves the highest Structural Fidelity (0.92) when successful. GPT-4.1 trails in Structural Fidelity (0.86), occasionally collapsing complex operator relations into simpler ones. Gemma-4 31B scores

⁶<https://ai.azure.com/catalog/models/gpt-5>

⁷<https://ai.azure.com/catalog/models/gpt-4.1>

⁸<https://huggingface.co/Qwen/Qwen3.5-27B>

⁹<https://huggingface.co/google/gemma-4-31B-it>

Method		Pressure Test		Climate Hazard	
		$\mathcal{MF1}@0.5$	$\mathcal{MF1}@0.9$	$\mathcal{MF1}^{\text{cov}}$	$\mathcal{MF0.5}^{\text{cov}}$
Random		9.0±4.2	0.0±0.0	12.0±0.6	9.2±0.5
Sup. Models	CNN	38.6±2.3	22.4±2.5	45.1±3.2	40.7±2.2
	Trans.	27.7±2.1	17.7±1.1	46.4±6.5	40.8±6.6
	Timer	32.0±3.4	3.3±0.4	18.0±0.4	17.1±0.6
	Moment	43.6±5.4	3.6±1.2	20.9±5.3	20.4±5.6
	Chronos	43.4±5.6	2.6±0.5	37.7±6.8	39.9±8.3
	TimesFM	46.6±6.1	3.9±0.6	44.5±3.0	47.4±3.4
GPT-4.1	Num. Z_s	15.4±3.9	12.6±2.9	38.3±2.6	46.2±2.4
	Num. F_s	25.4±6.0	17.2±5.0	43.4±3.1	49.4±3.5
	VL-T. Z_s	37.5±1.2	20.3±2.5	9.6±0.1	11.0±0.3
	VL-T. F_s	40.1±2.2	24.9±3.3	12.8±1.3	15.2±2.1
	SELA Z_s	43.9±2.3	32.7±5.5	52.2±5.2	<u>53.9±5.2</u>
GPT-5 <i>thinking</i>	Num. Z_s	53.2±3.1	49.9±4.1	40.8±3.2	45.7±3.6
	Num. F_s	<u>57.8±6.0</u>	<u>54.2±3.1</u>	43.4±1.6	49.4±1.1
	VL-T. Z_s	51.9±2.1	42.8±3.5	23.8±1.9	26.0±3.3
	VL-T. F_s	53.6±3.2	46.0±2.6	30.0±3.0	33.2±4.0
	SELA Z_s	66.8±3.9	60.0±3.6	<u>50.7±1.5</u>	57.1±2.2
<i>Human</i>		82.3	79.7	69.5	75.9

Table 4: Comparison of different models and methods on the two KITE datasets. Zs=Zero-shot; Fs=Few-shot In-Context Learning.

lowest on both Structural Fidelity (0.70) and Semantic Alignment (0.85), over-compressing primitive descriptions.

6.2 RQ 2. Evaluation on KITE Datasets

Experimental Setup. We compare SELA with four groups of baselines: (i) **Random guessing**; (ii) **supervised models**, including CNN (LeCun and Bengio, 1998), Transformer (Vaswani et al., 2017), and fine-tuned time series foundation models, i.e., Timer (Liu et al., 2024), Moment (Goswami et al., 2024), Chronos (Ansari et al., 2024), and TimesFM (Das et al., 2024); (iii) **few-shot/zero-shot LLM/VLM baselines**, including Numeric (Xue and Salim, 2024; Gruver et al., 2023), which directly takes numerical time series as input, and VL-Time (Liu et al., 2025), which lets VLMs reason over time series visualizations; (iv) **human**, where two data scientists with no prior task knowledge independently identify events based only on the provided pattern descriptions.

Evaluation Metrics. For Pressure Test, the event boundaries are definite and clear, as the signals are grounded in human engineer operations. Therefore, we apply IoU-based macro F1 scores ($\mathcal{MF1}@$) as metrics to measure performance, following previous work on time series event detection in the oil & gas domain (Khaouja et al., 2025). We set two IoU thresholds, 0.5 and 0.9, to better distinguish models

Variant	Pressure Test ($\mathcal{M}$ -F1)					Climate Hazard ($\mathcal{M}$ -Coverage)				
	GPT-5		GPT-4.1		Avg.	GPT-5		GPT-4.1		Avg.
	@0.5	@0.9	@0.5	@0.9		F1	F0.5	F1	F0.5	
SELA	66.8	60.0	43.9	32.7	50.9	50.7	57.1	52.2	53.9	53.5
<i>Event-logic structure</i>										
w/o ELT (Geo. Mean)	41.8 $\downarrow^{25.0}$	39.1 $\downarrow^{20.9}$	41.5 $\downarrow^{2.4}$	32.2 $\downarrow^{0.5}$	38.7 $\downarrow^{12.2}$	29.4 $\downarrow^{21.3}$	34.5 $\downarrow^{22.6}$	48.5 $\downarrow^{3.7}$	49.7 $\downarrow^{4.2}$	40.5 $\downarrow^{13.0}$
w/o ELT (Arith. Mean)	41.8 $\downarrow^{25.0}$	39.2 $\downarrow^{20.8}$	42.9 $\downarrow^{1.0}$	<u>32.7</u> $\uparrow^{0.0}$	39.2 $\downarrow^{11.7}$	29.6 $\downarrow^{21.1}$	34.7 $\downarrow^{22.4}$	47.4 $\downarrow^{4.8}$	48.6 $\downarrow^{5.3}$	40.1 $\downarrow^{13.4}$
w/o ELT (Minimum)	34.7 $\downarrow^{32.1}$	32.0 $\downarrow^{28.0}$	38.1 $\downarrow^{5.8}$	28.9 $\downarrow^{3.8}$	33.4 $\downarrow^{17.5}$	26.4 $\downarrow^{24.3}$	31.1 $\downarrow^{26.0}$	48.6 $\downarrow^{3.6}$	50.1 $\downarrow^{3.8}$	39.1 $\downarrow^{14.4}$
<i>Confidence scoring</i>										
w/o 1/K norm	<u>66.8</u> $\uparrow^{0.0}$	<u>60.0</u> $\uparrow^{0.0}$	42.6 $\downarrow^{1.3}$	32.3 $\downarrow^{0.4}$	<u>50.4</u> $\downarrow^{0.5}$	49.4 $\downarrow^{1.3}$	55.9 $\downarrow^{1.2}$	<u>51.7</u> $\downarrow^{0.5}$	<u>53.2</u> $\downarrow^{0.7}$	<u>52.6</u> $\downarrow^{0.9}$
w/o operator penalties	67.2 $\uparrow^{0.4}$	61.8 $\uparrow^{1.8}$	42.8 $\downarrow^{1.1}$	27.8 $\downarrow^{4.9}$	49.9 $\downarrow^{1.0}$	51.0 $\uparrow^{0.3}$	58.7 $\uparrow^{1.6}$	44.3 $\downarrow^{7.9}$	44.8 $\downarrow^{9.1}$	49.7 $\downarrow^{3.8}$
<i>Inspection tool</i>										
w/o zoom	63.2 $\downarrow^{3.6}$	51.8 $\downarrow^{8.2}$	45.5 $\uparrow^{1.6}$	33.4 $\uparrow^{0.7}$	48.5 $\downarrow^{2.4}$	49.9 $\downarrow^{0.8}$	56.0 $\downarrow^{1.1}$	51.4 $\downarrow^{0.8}$	50.9 $\downarrow^{3.0}$	52.1 $\downarrow^{1.4}$

Table 5: ELT ablations on Pressure Test and Climate Hazard. Avg. averages the four backbone–metric entries. **Bold** and underline denote the best and second-best results per column.

by their ability to perform precise grounding. For Climate Hazard, event boundaries are more gradual. We therefore follow Tatbul et al. (2018) and adopt coverage-based macro F1 and F0.5 scores ($\mathcal{MF1}^{\text{Cov}}$, $\mathcal{MF0.5}^{\text{Cov}}$). More details on the metric design are provided in Appendix E.

Implementation Details. For CNN and Transformer, we add linear heads for timestamp-wise classification and fine-tune the entire models. For native time series foundation models, we freeze the model weights and fine-tune only the attached bounding-box classification heads. All supervised models are tested using 5-fold cross-validation (80% training, 20% testing), and average scores over 3-seed runs are reported. Regarding Numeric and VL-Time, we provide the prompts in Appendix F. We also report the token usage of these VLM-based settings in Appendix G.

Results. As shown in Table 4, SELA consistently outperforms zero-shot and few-shot Numeric and VL-Time baselines on both KITE datasets, despite itself operating in the zero-shot setting. The advantage is particularly clear for precise localization on Pressure Test: with GPT-5, SELA reaches 66.8 $\mathcal{MF1}@0.5$ and 60.0 $\mathcal{MF1}@0.9$, improving over few-shot VL-Time by 13.2 and 14.0 points, respectively. On Climate Hazard, SELA also achieves the strongest automated results, with GPT-4.1 obtaining the best $\mathcal{MF1}^{\text{Cov}}$ of 52.2 and GPT-5 obtaining the best $\mathcal{MF0.5}^{\text{Cov}}$ of 57.1. In contrast, supervised baselines show no consistent advantage in this low-resource setting; in particular, time series foundation models perform poorly under the strict $\mathcal{MF1}@0.9$ metric on Pressure Test, indicat-

ing limited boundary-localization ability. Overall, the consistent gains across GPT-4.1 and GPT-5 demonstrate that ELT-guided reasoning improves both event detection and precise temporal grounding, while human data scientists still retain a substantial performance margin.

Ablation Study. As shown in Table 5, the event-logic structure is the most critical component of SELA. In the w/o ELT variants, we discard the tree structure and directly aggregate primitive-level confidence scores using three basic strategies: geometric mean, arithmetic mean, and minimum. Removing ELT causes substantial performance drops under all three aggregation strategies, reducing the average score by up to 17.5 points on Pressure Test and 14.4 points on Climate Hazard. In comparison, ablating confidence normalization, operator penalties, or the zoom inspection tool leads to smaller but consistent degradations. Among them, operator penalties are particularly important for Climate Hazard, while zoom mainly benefits precise localization on Pressure Test.

6.3 Case Study

Figure 4 presents a *Frost Freeze* case from Climate Hazard, where SELA closely matches the ground-truth. The instantiated ELT grounds the event in a sequential temperature evolution from mild-above-freezing conditions through sustained cooling to a warming rebound, while synchronizing and guarding these phases with complementary wind and dew-point-spread primitives. In contrast, Numeric predicts only a short fragment, VL-Time assigns the wrong event class, and CNN and Transformer

Frost / Freeze

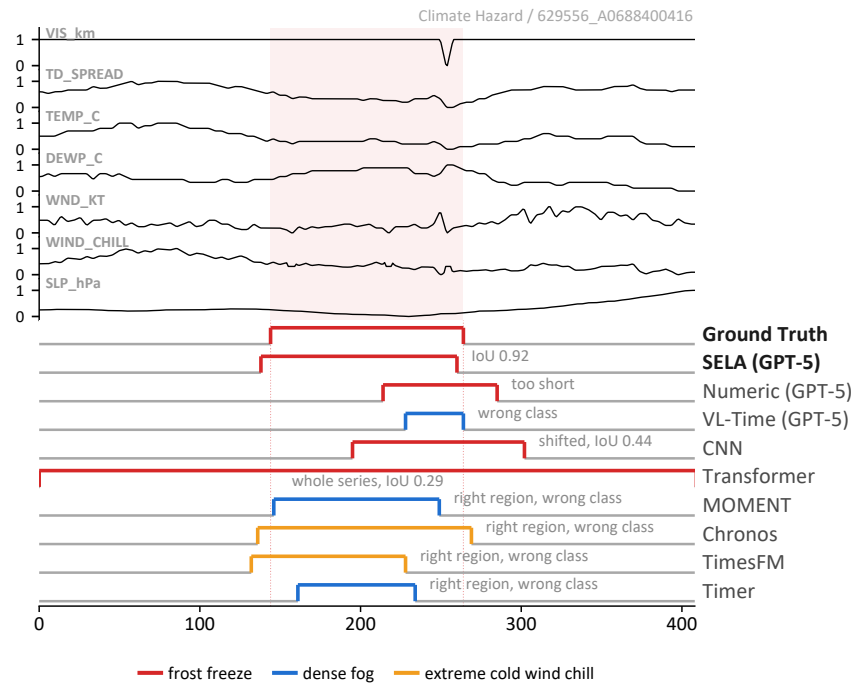
TEMP_C · DEWP_C · TD_SPREAD · WND_KT

Overnight radiational-cooling events: a Freeze Warning (temperature drops to or below freezing) or the milder Frost Advisory (near-freezing under clear skies and calm wind), typically during growing-season shoulder months (spring/fall) rather than the deep winter cold air masses of Extreme Cold / Wind Chill.

TEMP_C Descends overnight to $\leq 2^\circ\text{C}$. Shape is diurnal-radiational — steady descent through the night, minimum near sunrise, recovery afterward — rather than the abrupt arctic-airmass-arrival shape of Extreme Cold.

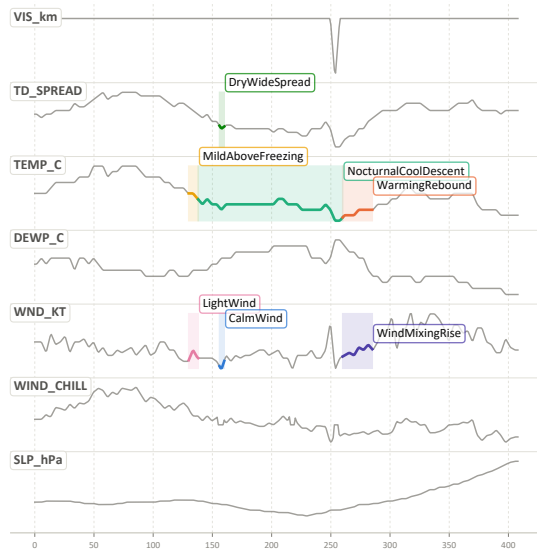
DEWP_C / TD_SPREAD Efficient radiational cooling requires dry air, so TD_SPREAD should stay wide (dry) rather than collapsing toward zero the way Dense Fog's does.

WND_KT Stays low / calm throughout — wind mixing suppresses radiational cooling, so a genuine Frost/Freeze night shows persistently light wind, distinguishing it from wind-driven cold-air-advection events.

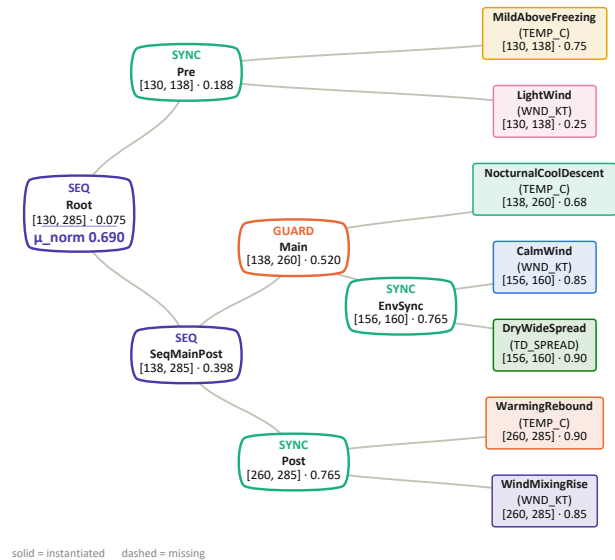


(a) Comparison of event predictions on a Climate Hazard case.

Signals & Primitive Instances



Logic Tree · frost freeze



(b) Instantiated primitive instances and Event Logic Tree for the selected case.

Figure 4: Case study from Climate Hazard, showing predictions from different methods against the ground truth, together with the instantiated primitive instances and Event Logic Tree (ELT).

	D1: Hierarchical Representation	D2: Semantic Quantification	D3: Topological Elasticity
SAX / ABBA	✗	▲	✗
Logical-Shapelets	▲	▲	✓
Z-Time	▲	▲	✓
Chronicle / STL	✓	▲	✗
Event Logic Tree	✓	✓	✓

Table 6: Comparison of frameworks against our desiderata. (✓: Satisfied; ▲: Limited / Partially satisfied; ✗: Not satisfied)

substantially mislocalize the interval; the time series foundation models identify a relevant region but also predict the wrong class. This example illustrates how ELT-guided compositional reasoning helps SELA jointly resolve event semantics and temporal boundaries in noisy multivariate signals.

7 Related Work

7.1 Time Series Symbolic Representation

We analyze existing symbolic time series representation frameworks based on our desiderata (summarized in Table 6).

(1) SAX (Lin et al., 2003; Malinowski et al., 2013) and ABBA (Elsworth and Güttel, 2020) map time series into string sequences where each character represents a specific amplitude. Moreover, the repeated sub-strings carry semantics but do not quantify coherence (partially satisfying D2), and do not allow hierarchical structure (failing D1) or complex topological relationships (failing D3).

(2) Logical-Shapelets (Mueen et al., 2011) represents entire time series with a series of discriminative sub-sequences, i.e., shapelets (satisfying D3). Z-Time (Lee et al., 2024) improves the representation by making the length variable (satisfying D3). Both the shapelets and discrete representations of Z-Time carry limited semantic abstraction (partially satisfying D2). Regarding structural modeling, Logical-Shapelets only considers boolean relations, and while Z-Time uses Allen’s algebra (Allen, 1983), only limited hierarchy structures can be represented by stacking temporal relation pairs. Therefore, both approaches partially satisfy D1.

(3) Chronicle System (Dousson and Le Maigat, 2007) applies graph structure to represent temporal events, where each sub-event is considered as a node in the graph, and STL (Maler and Nickovic, 2004) represents events as recursive logical formulas over signals intervals, both satisfying D1. However, both systems model semantic coherence as binary values (true/false) (partially satisfying

D2), and their reliance on actual interval durations does not meet D3.

In summary, existing symbolic systems can only make structured representations from time series but not language, thus not satisfying the requirements of L-TSED. In contrast, our proposed ELT framework uses tree structures based on Allen’s algebra and boolean operators to support D1. The definition of tree schema and instantiation on actual time series data are separated to satisfy D3. ELT supports quantifying semantic coherence with neural models over any combination of basic time series signal attributes, satisfying D2.

7.2 Multimodal LLMs for TS Reasoning

Although time-series foundation models show promising zero-shot or limited-supervision capabilities (Ansari et al., 2024; Liu et al., 2024; Goswami et al., 2024; Das et al., 2024), they are not natively designed for language-conditioned event localization and generally require task-specific adaptation for downstream detection tasks. Recently, LLMs’ potential to serve as few/zero-shot time series reasoners has been widely explored. ChatTS (Xie et al., 2025) designed an encoder with synthetic data to align time series with language for QA tasks. However, the domain gap between synthetic and real-world data is difficult to bridge, and embedding time series can still introduce the risk of hallucination. VL-Time (Liu et al., 2025) visualizes time series data as figures to directly adopt vision-language reasoning capabilities without extra TS-language alignment. While it has shown advantages in classification, visualization inevitably loses precision, and hallucination is still a risk. Our approach employs active visualization tools to overcome precision loss and effectively mitigates hallucination with ELT representations.

8 Conclusion

We introduce a new language-guided setting for time-series event detection and present KITE, the first real-world benchmark for this task. To address the low-resource and knowledge-intensive nature of this setting, we propose the Event Logic Tree and build SELA, a zero-shot neuro-symbolic agent system that combines ELTs with large vision-language models. Experiments, ablations, and case studies show that ELT-guided reasoning improves precise event localization and provides interpretable evidence for human-in-the-loop verification.

Limitations

Benchmark scope. KITE remains limited in scale and domain coverage because expert event descriptions and interval annotations are costly to obtain. We plan to include more real-world multi-variate time-series data from more key domains in the future.

Faithfulness of instantiation. Currently, we rely on VLMs to provide confidence scores for primitive-level text–time-series alignment during instantiation. Human evaluation of such grounding faithfulness is extremely costly since it requires domain experts to evaluate the alignment between all primitive-level textual descriptions and detected signal sections. Direct automatic evaluation is also difficult because reliable embedding models specifically designed to align domain-specific time-series primitives with textual descriptions are currently lacking. We therefore plan to reduce the impact of LLM/VLM subjectivity by developing a foundation model that can more objectively evaluate the alignment between domain-specific primitive-level textual descriptions and detected signal segments.

System complexity. SELA relies on LLM/VLM reasoning, which is costly and inherently stochastic, and may lead to variation in parsing, grounding, and confidence estimation across runs. This randomness is difficult to remove completely in an agent-based system. Future work will explore more stable and efficient models, together with improved reasoning and grounding methods, to reduce computational cost and support low-latency inference. We will also maintain the goal of transparency and explainability throughout the reasoning process, so that predictions and intermediate evidence can be inspected by domain experts and safely used under human supervision.

Potential Risks. Potential risks arise if incorrect event detections or overconfident explanations are used for operational decisions in safety-sensitive settings. SELA is intended as a research and human-in-the-loop decision-support framework rather than an autonomous safety-critical system, and its predictions should be verified by domain experts.

Acknowledgments

We thank the members of the SLB AI Lab for their support and valuable feedback throughout this

work. We are especially grateful to Josselin Kherroubi, Director of the SLB AI Lab, for providing the resources that made this research possible. We also thank the anonymous reviewers, particularly Reviewer 2qFd, for their thoughtful comments and constructive suggestions, which helped improve the paper. Finally, we thank Prof. Matthieu Labeau and Prof. Fabian Suchanek from Télécom Paris for their valuable discussions and insightful advice.

References

- James F. Allen. 1983. [Maintaining knowledge about temporal intervals](#). *Commun. ACM*, 26(11):832–843.
- Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Türkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda-Arango, Shubham Kapoor, Jasper Zschiesner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Bernie Wang. 2024. [Chronos: Learning the language of time series](#). *Trans. Mach. Learn. Res.*, 2024.
- Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. 2024. [A decoder-only foundation model for time-series forecasting](#). In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pages 10148–10167. PMLR / OpenReview.net.
- Christophe Dossou and Pierre Le Maigat. 2007. Chronicle recognition improvement using temporal focusing and hierarchization. In *IJCAI*, volume 7, pages 324–329.
- Steven Elsworth and Stefan Güttel. 2020. [ABBA: adaptive brownian bridge-based symbolic aggregation of time series](#). *Data Min. Knowl. Discov.*, 34(4):1175–1200.
- Sebastien Fries, Tianjun Hou, Amine Ennaifer, Lei Jiang, Josselin Kherroubi, and Hadrien Dumont. 2024. Methods and systems for flagging events in a time series and evaluating a downhole operation. US Patent 12,018,559.
- Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. 2024. [MO-MENT: A family of open time-series foundation models](#). In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.
- Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew Gordon Wilson. 2023. Large language models are zero-shot time series forecasters. In *Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS ’23*, Red Hook, NY, USA. Curran Associates Inc.

- Imane Khaouja, Amine EL KHAIR, Abdallah Benzine, Sebastiaan Buiting, Soumyadip Sengupta, and Youssef Tamaazousti. 2025. [Do large foundation models improve time series segmentation? an industrial case study in oil and gas drilling](#). In *1st ICML Workshop on Foundation Models for Structured Data*.
- Yann LeCun and Yoshua Bengio. 1998. *Convolutional networks for images, speech, and time series*, page 255–258. MIT Press, Cambridge, MA, USA.
- Zed Lee, Tony Lindgren, and Panagiotis Papapetrou. 2024. [Z-time: efficient and effective interpretable multivariate time series classification](#). *Data Min. Knowl. Discov.*, 38(1):206–236.
- Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. 2003. [A symbolic representation of time series, with implications for streaming algorithms](#). In *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, DMKD '03, page 2–11, New York, NY, USA. Association for Computing Machinery.
- Haoxin Liu, Chenghao Liu, and B. Aditya Prakash. 2025. [A picture is worth A thousand numbers: Enabling llms reason about time series via visualization](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 1: Long Papers, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, pages 7486–7518. Association for Computational Linguistics.
- Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. 2024. [Timer: Generative pre-trained transformers are large time series models](#). In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.
- Oded Maler and Dejan Nickovic. 2004. Monitoring temporal properties of continuous signals. In *Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems*, pages 152–166, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Simon Malinowski, Thomas Guyet, René Quiniou, and Romain Tavenard. 2013. 1d-sax: A novel symbolic representation for time series. In *International Symposium on Intelligent Data Analysis*, pages 273–284. Springer.
- Navid Mohammadi Foumani, Lynn Miller, Chang Wei Tan, Geoffrey I. Webb, Germain Forestier, and Mahsa Salehi. 2024. [Deep learning for time series classification and extrinsic regression: A current survey](#). *ACM Comput. Surv.*, 56(9).
- Abdullah Mueen, Eamonn Keogh, and Neal Young. 2011. [Logical-shapelets: an expressive primitive for time series classification](#). In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '11*, page 1154–1162, New York, NY, USA. Association for Computing Machinery.
- Mathias Perslev, Michael Hejlsbak Jensen, Sune Darkner, Poul Jørgen Jennum, and Christian Igel. 2019. [U-time: A fully convolutional network for time series segmentation applied to sleep staging](#). In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 4417–4428.
- Chen Qian, Zihao Xie, Yifei Wang, Wei Liu, Kunlun Zhu, Hanchen Xia, Yufan Dang, Zhuoyun Du, Weize Chen, Cheng Yang, Zhiyuan Liu, and Maosong Sun. 2025. [Scaling large language model-based multi-agent collaboration](#). In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.
- Akara Supratak, Hao Dong, Chao Wu, and Yike Guo. 2017. [Deepsleepnet: A model for automatic sleep stage scoring based on raw single-channel eeg](#). *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(11):1998–2008.
- Nesime Tatbul, Tae Jun Lee, Stan Zdonik, Mejbah Alam, and Justin Gottschlich. 2018. [Precision and recall for time series](#). In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.
- Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. 2023. [Timesnet: Temporal 2d-variation modeling for general time series analysis](#). In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.
- 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](#). *Proc. VLDB Endow.*, 18(8):2385–2398.
- Hao Xue and Flora D. Salim. 2024. [Promptcast: A new prompt-based learning paradigm for time series forecasting](#). *IEEE Trans. on Knowl. and Data Eng.*, 36(11):6851–6864.
- Zahra Zamanzadeh Darban, Geoffrey I. Webb, Shirui Pan, Charu Aggarwal, and Mahsa Salehi. 2024. [Deep learning for time series anomaly detection: A survey](#). *ACM Comput. Surv.*, 57(1).

TABLE OF CONTENTS

A	Event Logic Tree	12
A.1	ELT Schema	12
A.2	Constitutive Axioms	12
A.3	Instantiation of ELT	12
A.4	Operational Completeness	13
B	Implementation Details	14
C	Climate Hazard Construction Details	15
C.1	Event Type Selection Rationale . .	15
C.2	Candidate Sample Construction .	16
C.3	Event Description and Boundary Calibration	16
D	ELT Parse Quality Evaluation	17
D.1	Evaluation Dimension Details . .	17
D.2	Judge Prompt	17
E	Evaluation Metrics	17
F	Numeric and VL-Time Baseline Prompts	18
F.1	Numeric Baseline	18
F.2	VL-Time Baseline	18
F.3	Few-Shot Demonstrations	19
G	Token Budget	19

A Event Logic Tree

A.1 ELT Schema

The Event Logic Tree schema is parsed from textual descriptions only, independent of actual time series data. One event corresponds to one tree.

Definition 1 (Node Schema). *The Event Logic Tree Schema is denoted as $\mathcal{S}$. Any node in the schema $n \in \mathcal{S}$ belongs to two categories: primitives and composites.*

1. Primitive (Leaf Node) : *The atomic signal patterns over single physical channels, denoted as $n_p = \langle \tau, c \rangle$, where:*

- τ : The **semantic predicate** that describes signal morphology, which can be a compound of basic time series attributes, e.g., “a steep linear increase with high-frequency noise”.
- c : The **physical channel** index where this atomic pattern would occur.

2. Composite (Internal Node). *The temporal-logic relations that form the global event hierarchically with the primitives, denoted as $n_\phi = \langle \mathcal{N}, op \rangle$, where:*

- $\mathcal{N} = [n_1, \dots, n_k]$: *An ordered list of child nodes, either primitives or lower-level composites.*
- $op \in \{\text{SEQ}, \text{SYNC}, \text{GUARD}, \text{OR}\}$: *The **temporal-logic operator** defining the relationship between child nodes.*

A.2 Constitutive Axioms

Based on the definitions above, we establish three axioms to serve as preconditions for valid schema construction, and properly control the search space during instantiation.

Axiom 1 (Constructive Composition). *A composite node must add structural meaning. Formally, for any composite n_ϕ , $|\mathcal{N}| \geq 2$.*

Remark. Axiom 1 excludes bad structures with self-nesting, and any logical NOT as composite with a single child. We require all primitives to be defined *positively*.

Axiom 2 (Temporal Compactness). *The semantics of any node must cover its full temporal span. Any undefined gap inside the temporal spans of composites must be bounded by a hyperparameter.*

Remark. Primitives are considered to be compact by definition. For any composite, assuming the compactness tolerance is 1, a composite whose children are assigned disjoint intervals like $[1, 2]$ and $[4, 6]$ will be prohibited by Axiom 2.

Axiom 3 (Physical Exclusivity). *A physical channel c can only support one active primitive state at any time point t .*

A.3 Instantiation of ELT

Definition 2 (Node Instance). *An instance implies designating a specific time interval to a schema node n .*

1. Primitive Instance. *A primitive instance $\hat{n}_p$ is defined as: $\langle n_p, \mathcal{I}, \mu \rangle$, where:*

- $\mathcal{I} = [t_{\text{on}}, t_{\text{off}}]$: *The detected time **interval**.*
- $\mu = \mathcal{M}_\tau(X, \mathcal{I}, c) \in [0, 1]$: *The **semantic coherence score**, which quantifies the alignment between the signal and the predicate. The implementation of the semantic function*

$\mathcal{M}_\tau$ is not restricted to rule-based functions. Measurement can be evaluated based on the distance between the embeddings of linguistic description and signal pattern in an aligned latent space.

2. Composite Instance. A composite instance $\hat{n}_\phi$ is defined as $\langle n_\phi, \mathcal{I}_\phi, \mu_\phi \rangle$:

- $\mathcal{I}_\phi = [\min_{\hat{n} \in \hat{\mathcal{N}}} t_{\text{on}}^{(\hat{n})}, \max_{\hat{n} \in \hat{\mathcal{N}}} t_{\text{off}}^{(\hat{n})}]$: The **composite interval**, which is the temporal span of constituents' intervals (subject to Axiom 2). $\hat{\mathcal{N}} = [\hat{n}_1, \dots, \hat{n}_k]$ denotes the child instances.
- $\mu_\phi = \text{op}(\hat{n}_1, \dots, \hat{n}_k)$: The **recursive confidence score** by applying the operator on the constituents.

To ensure the coherence of Axiom 3, we need to introduce a collision operator Ψ . If the Axiom 3 is violated by any instance, $\Psi = 1$.

Definition 3 (Channel-Semantic Collision).

$$\Psi(\hat{n}_A, \hat{n}_B) = \max_{\substack{p \in \Pi(\hat{n}_A) \\ q \in \Pi(\hat{n}_B)}} \mathbf{1} \left[(c_p = c_q) \wedge (\ell(\mathcal{I}_p \cap \mathcal{I}_q) > \epsilon) \right] \quad (4)$$

$\Pi(\hat{n})$ denotes the set of **primitive** instances descendants of $\hat{n}$, if $\hat{n}$ is a composite instance, or the set of $\hat{n}$ itself, if $\hat{n}$ is a primitive instance. $\mathbf{1}[\cdot]$ denotes the indicator function. For discrete sampling, a change point may belong to both adjacent intervals at the same time. This violation case should be ignored. Therefore, Semantic Nullity Threshold ϵ is introduced to set up a tolerance.

Definition 4 (Temporal-Logic Operators). We define the core operator set to comprise four operators: $\mathcal{O}_{\text{core}} = \{\text{SEQ}, \text{SYNC}, \text{GUARD}, \text{OR}\}$ (Table 1). The operators aggregate confidence scores in a bottom-up manner with product T-norm, while respecting Temporal Compactness and Physical Exclusivity.

1. SEQ (Sequence). SEQ represents the temporal precedence relationship (B follows A). The confidence score is computed with product T-norm while controlled by validity gates:

$$\text{SEQ}(\hat{n}_A, \hat{n}_B) = (\mu_A \cdot \mu_B) \cdot (1 - \Psi(\hat{n}_A, \hat{n}_B)) \cdot \mathcal{G}_{\text{causal}}(\hat{n}_A, \hat{n}_B) \cdot \mathcal{G}_{\text{coh}}(\hat{n}_A, \hat{n}_B) \quad (5)$$

where the term $(1 - \Psi(\hat{n}_A, \hat{n}_B))$ enforces Physical Exclusivity. The causality gate $\mathcal{G}_{\text{causal}}(\hat{n}_A, \hat{n}_B) = \mathbf{1}[t_{\text{on}}^B > t_{\text{on}}^A] \cdot \mathbf{1}[t_{\text{off}}^B > t_{\text{off}}^A]$ enforces the temporal precedence relationship. The coherence gate $\mathcal{G}_{\text{coh}}(\hat{n}_A, \hat{n}_B) = \mathbf{1}[(t_{\text{on}}^B - t_{\text{off}}^A - \delta) < 0]$ prohibits any significant semantic gap beyond the bound δ to ensure the Temporal Compactness principle.

2. SYNC (Synchronization). SYNC represents temporal identity ($\mathcal{I}_A = \mathcal{I}_B$). A penalty term based on IoU, denoted as $\text{IoU}(\mathcal{I}_A, \mathcal{I}_B) = \frac{\ell(\mathcal{I}_A \cap \mathcal{I}_B)}{\ell(\mathcal{I}_A \cup \mathcal{I}_B)}$, is applied to penalize the joint confidence in case of temporal misalignment.

$$\text{SYNC}(\hat{n}_A, \hat{n}_B) = (\mu_A \cdot \mu_B) \cdot (1 - \Psi(\hat{n}_A, \hat{n}_B)) \cdot \exp \left(-\frac{1 - \text{IoU}(\mathcal{I}_A, \mathcal{I}_B)}{\kappa} \right) \quad (6)$$

where κ is the alignment tolerance.

3. GUARD (Containment). GUARD represents $\mathcal{I}_A \subset \mathcal{I}_B$. Boundaries of A that spill outside B are penalized:

$$\text{GUARD}(\hat{n}_A, \hat{n}_B) = (\mu_A \cdot \mu_B) \cdot (1 - \Psi(\hat{n}_A, \hat{n}_B)) \cdot \exp \left(-\frac{\Delta_{\text{on}} + \Delta_{\text{off}}}{\sigma} \right) \quad (7)$$

where $\Delta_{\text{on}} = \max(0, t_{\text{on}}^B - t_{\text{on}}^A)$ and $\Delta_{\text{off}} = \max(0, t_{\text{off}}^A - t_{\text{off}}^B)$ quantify how much A extends beyond B , controlled by temperature σ .

4. OR (Disjunction). OR models semantic alternatives for the same instance. We encourage the two alternatives to refer to the same interval by using the same IoU-based alignment penalty as SYNC:

$$\text{OR}(\hat{n}_A, \hat{n}_B) = \max(\mu_A, \mu_B) \cdot \exp \left(-\frac{1 - \text{IoU}(\mathcal{I}_A, \mathcal{I}_B)}{\kappa} \right) \quad (8)$$

A.4 Operational Completeness

From a logical perspective, SEQ, SYNC, and GUARD are *conjunctive*, representing the Boolean AND (co-existing relation (Lee et al., 2024)) projected onto temporal relationships. We can thus unify them as a generalized **Temporal Conjunction** $\text{AND}_\mathcal{K}$:

$$\text{AND}_\mathcal{K}(\hat{n}_A, \hat{n}_B) = (\mu_A \cdot \mu_B) \cdot \mathcal{K}(\hat{n}_A, \hat{n}_B) \cdot (1 - \Psi(\hat{n}_A, \hat{n}_B)) \quad (9)$$

where $\mathcal{K}$ maps to SEQ/SYNC/GUARD by working as corresponding soft/hard validity gates.

Proposition 1 (Conditional Completeness). *Under the constraints of Axioms 1-3, the operator set $\mathcal{O}_{core}$ forms a complete basis for the 13 fundamental Allen’s interval relations.*

Proof Sketch. We demonstrate that the generalized conjunction $\text{AND}_{\mathcal{K}}$ (as SEQ, SYNC, GUARD) exhaustively cover the Allen’s algebra partition:

(1) The Precedence Group (SEQ): The relations implying temporal precedence are handled by the SEQ. The causality gate ($t_{on}^B > t_{on}^A \wedge t_{off}^B > t_{off}^A$) distinguishes it from SYNC and GUARD. (i) *Meets* and *Before*: While *Meets* naturally corresponds to SEQ, the disjoint *Before* relation is bounded by horizon δ to prevent any semantic gap that would violate Axiom 2. Therefore, the *Before* relation is treated as a relaxed variant of *Meets*. (ii) *Overlaps*: a special case corresponding to SEQ if and only if Physical Exclusivity is ensured across all channels over the overlapping interval ($\Psi = 0$).

(2) The Containment Group (GUARD): The *During*, *Starts*, and *Finishes* relations can be defined by nesting $\mathcal{I}_A \subset \mathcal{I}_B$, which corresponds to the GUARD operator. Fine-grained temporal relations under GUARD like *Starts* and *Finishes* can be represented by adding constraints to the corresponding validity gates.

(3) The Identity Group (SYNC): The *Equal* relation is defined by temporal equivalence $\mathcal{I}_A = \mathcal{I}_B$, corresponding to the SYNC operator.

(4) The Inverse Group: The *After*, *Contains*, and *Overlapped-by* relations can be expressed by swapping the arguments of SEQ and GUARD (e.g., $\text{SEQ}(n_B, n_A)$ resolves *After*).

Conclusion: Since all 13 relations map to parameterizations of $\mathcal{O}_{core}$ or their constructive inverses, the set is operationally complete. $\square$

B Implementation Details

We summarize the prompts used by SELA below. Runtime-dependent inputs are represented by placeholders for readability.

ELT Schema Parser

You are a Signal Logic Architect for time-series event detection using Event Logic Trees (ELTs).

Input
event_name: <event_name>

channels: <channels>
event_specification: <description>

Task
Compile the natural-language event specification into a binary ELT.

Use the structure:
Root = SEQ(Pre, SEQ(Main, Post))

where Main is the core event phase. If Pre or Post is unspecified, use an observable background/holding state rather than inventing an event.

Primitive nodes:
- use exactly one available signal channel;
- describe an observable temporal morphology over an interval;
- have unique aliases.

Composite nodes use one of:
- SEQ(A,B): A is followed by B;
- SYNC(A,B): A and B substantially overlap;
- GUARD(A,B): A occurs within B;
- OR(A,B): either morphology may occur.

Constraints
- Use a strict binary tree with unique aliases.
- Keep the tree compact while preserving event semantics.
- Prefer SYNC for simultaneous cross-channel patterns.
- Do not use purely negative or unobservable primitives.
- Do not duplicate equivalent morphology.

Validate the schema using ‘submit_schema’, refine it if needed, and return the final schema with root_alias="Root" using ‘submit_result’.

SELA Event Inspector

You are an expert multivariate time-series analyst using an Event Logic Tree (ELT).

Input
candidate ELT(s): <tree_information>
dataset description: <description>
time series: accessed through visualization tools

Task
Ground each candidate ELT in the observed time series by assigning intervals and confidence scores to its primitive nodes.

Tools
- ‘view_full’: inspect the full multivariate signal.
- ‘view_window’: inspect a candidate region in detail.

- 'instantiate': assign primitive intervals and confidence scores.
- 'compare_trees': compare candidate ELTs when multiple classes are loaded.

Procedure

1. Inspect the full signal and identify plausible event regions.
2. Zoom into candidate regions before fixing boundaries.
3. Instantiate the relevant Pre, Main, and Post primitives.
4. Verify that each assigned interval matches its primitive morphology.
5. Refine intervals or confidence scores when ELT constraints are violated.
6. For multiple candidate classes, compare the normalized root confidence ('mu_norm') and select the best-supported ELT.

Grounding Rules

- SEQ: child phases must form a coherent temporal sequence.
- SYNC: child intervals should substantially overlap.
- GUARD: the guarded child must lie within its parent interval.
- OR: only supported branches need to be instantiated.

A primitive description must hold over its entire assigned interval.
Tighten boundaries when only a sub-region matches the morphology.

Use conservative and consistent confidence calibration:

0.9--1.0 strong match;
0.7--0.9 good match;
0.4--0.7 partial match;
0.1--0.4 weak match;
0 unsupported.

Do not inflate confidence to favor a particular event class.
Return a concise summary of the decisive evidence using 'submit_comment'.

Parallel Single-Tree Inspector

When parallel inspection is enabled, each inspector receives only one candidate event class and its ELT.

Instantiate the tree following the same grounding, operator, boundary, and confidence rules above. Other event classes are evaluated independently.

A low normalized root confidence is the correct output when the observed signals do not support the assigned event class. The final prediction is obtained by comparing the independently computed 'mu_norm' scores.

Shared ReAct Instruction

Before each tool call, briefly state what is currently observed, what will be inspected or modified next, and why the action is useful.

C Climate Hazard Construction Details

Historical weather-event records are collected from heterogeneous sources and may not always provide fully reliable event annotations. NOAA notes that some external reports in the NWS Storm Events database may remain unverified because of time and resource constraints, and therefore “does not guarantee the accuracy or validity of the information”¹⁰. Detecting historical events directly from time series data can therefore provide additional evidence for checking and improving existing event records. To construct the Climate Hazard benchmark for this setting, we first select hazard types suitable for multivariate interval detection, then align the corresponding NWS records with ISD station observations and filter low-quality samples, and finally ask data scientists to inspect the signals and calibrate the event intervals.

C.1 Event Type Selection Rationale

The Storm Events database contains dozens of hazard types, but not all of them are suitable for our task. We select *Dense Fog*, *Extreme Cold*, and *Frost Freeze* based on the following considerations.

Event frequency. Rare hazard types yield too few episodes to construct a dataset of meaningful scale, and calibrating near-singular events offers limited practical value. *Dense Fog*, *Extreme Cold* and *Frost Freeze* are among the more frequently recorded hazards in the continental US, striking a balance between data availability and operational relevance.

ISD data quality and coverage. NWS event zones and ISD station networks are independent, so a recorded event does not guarantee a co-located station with adequate coverage. Many hazard types are further hindered by their spatially concentrated impact area: when a station lies outside the event core, the recorded signal may be too weak to be representative.

¹⁰<https://www.ncei.noaa.gov/stormevents/faq.jsp>

Event duration. Short-lived hazards such as *Thunderstorm Wind* or *Lightning* last only minutes which at or below the resolution of both ISD records and NWS timestamps, reducing ground-truth boundaries to near-point annotations unsuitable for interval detection. All selected types persist over hours to days.

Multivariate necessity. Some hazards are detectable by a single threshold on one channel (e.g., *Excessive Heat* reduces to a sustained temperature exceedance), making multivariate knowledge-guided detection unnecessary.

C.2 Candidate Sample Construction

Event–station alignment. Each NWS event is associated with a geographic zone rather than a point station. For every event, we identify candidate ISD stations within or near the corresponding event zone whose active date ranges cover the event period. We retain only FM-15 (METAR) reports to ensure consistent observation timing and measurement fields.

Episode deduplication. A single meteorological episode may generate multiple NWS records across affected counties or zones. We group records belonging to the same episode and retain the station with the highest observation quality. Stations are ranked by

$$q = \frac{\bar{n}_c}{1 + \Delta}, \quad (10)$$

where $\bar{n}_c$ is the mean observation count per channel and Δ is the maximum inter-observation gap in minutes.

Quality filtering. Four filters are applied before a candidate is accepted: (1) ≥ 5 non-NaN readings per channel; (2) no gap exceeding 120 consecutive minutes; (3) ≥ 3 observations in each of the pre- and post-event sub-windows; (4) at least one observation with $\text{VIS}_{\text{km}} < 5.0$ for *Dense Fog*; $\text{TEMP}_{\text{C}} < -10^{\circ}\text{C}$ / $\text{WIND}_{\text{CHILL}} < -20^{\circ}\text{C}$ for *Extreme Cold*; $\text{TEMP}_{\text{C}} \leq 2.0^{\circ}\text{C}$ for *Frost Freeze*

Event ratio constraint. Let $r = (t_{\text{end}} - t_{\text{start}}) / L$ denote the fraction of the window occupied by the GT event, where L is the window length. Accepted ranges are $r \in [0.10, 0.80]$ for *Dense Fog* as well as *Frost Freeze*, and $r \in [0.20, 0.80]$ for *Extreme Cold*. Where ISD station records do not extend to the full target window on either side, the available data is retained as-is; no zero-padding or forward-filling beyond the last observation is applied.

C.3 Event Description and Boundary Calibration

Event descriptions. We construct textual descriptions of the selected hazard types from official NWS documentation and the American Meteorological Society Glossary¹¹¹², with additional guidance for *Extreme Cold* from the NWS Wind Chill Temperature Index¹³. The descriptions summarize the relevant signal morphology, relationships among meteorological variables, and their temporal evolution. These descriptions provide domain knowledge for both language-guided detection and subsequent expert inspection.

Manual signal inspection and calibration. The original event intervals are initialized from the NWS Storm Events database. As discussed above, these records are not always fully reliable and may therefore require further verification. Moreover, even when an event record is accurate at the geographic-zone level, its reported timestamps may not precisely match how the event appears at an individual ISD station. We therefore ask two annotators with climate-science backgrounds to independently inspect each candidate time series together with its event description. Samples with abnormal signals are removed, and the boundaries of the remaining events are manually adjusted when necessary to better match the observed meteorological evidence.

Cross-verification. The two annotators independently verify the event category and boundaries. At the event-category level, their agreement is 100%. For boundary annotations, we use an event-specific disagreement threshold T that reflects the characteristic duration and formation/dissipation behavior of each hazard. When the two annotations differ by no more than T , their mean is used as the final boundary; otherwise, the annotators review the sample jointly and resolve the disagreement through discussion. We use $T = 1$ hour for *Dense Fog*, $T = 4$ hours for *Extreme Cold*, and $T = 2$ hours for *Frost Freeze*. Across the climate hazard data, the pre-discussion boundary agreement rate is 91%.

¹¹https://www.weather.gov/media/directives/010_pdfs_archived/pd01016005c.pdf

¹²<https://glossary.ametsoc.org/>

¹³https://www.weather.gov/media/ajk/brochures/Wind_Chill_Temperature_Index.pdf

D ELT Parse Quality Evaluation

D.1 Evaluation Dimension Details

Each valid parse is scored against the ground-truth ELT and the original event description by two independent VLM judges, Claude Opus-4.6 and GPT-5.5 (thinking mode), drawn from different model families to reduce scoring bias. Each judge evaluates the same parse three times. To mitigate position bias, we randomise whether the LLM parse or the ground-truth tree is shown first in each evaluation. Scores on a 0–5 scale across four dimensions are averaged and normalised to [0, 1]. The two judges show strong agreement (Spearman $\rho = 0.86$, $p < 0.001$ across all scored parses), and VLM scores on a held-out subset of 30 parses correlate strongly with independent human annotations ($\rho = 0.83$), supporting the reliability of the evaluation.

D.2 Judge Prompt

LLM Judge Instruction

Role & Objective

You are an expert judge evaluating the quality of Event Logic Trees (ELTs) automatically parsed by LLMs from natural-language event descriptions. You will be given:

1. A ground-truth ELT (GT) parsed by a human expert
2. An LLM-parsed ELT for the same event
3. A side-by-side visualisation of the two trees
4. The original natural-language event description

Your task is to score the LLM-parsed ELT on four dimensions (0-5 scale).

Scoring Dimensions

Structural Fidelity (0-5)

How similar is the LLM tree structure to the GT in terms of operator types and tree topology?

- 5: Identical operator types and tree shape, or logically equivalent (e.g., different grouping but same logical meaning)
- 4: Minor differences in nesting or operator choice, but overall structure preserved
- 3: Similar but with extra nesting, wrong operators, or different grouping that changes logical meaning
- 1-2: Major structural differences
- 0: Completely different structure

Primitive Coverage (0-5)

Are all ground-truth primitives accounted for in the LLM parse? A primitive may be covered at a different granularity (e.g., split into sub-stages) as long as it is logically equivalent.

- 5: All GT primitives fully covered (exactly or logically equivalent)
- 3: Most GT primitives present; minor omissions
- 0: Major GT primitives missing

Primitive Accuracy (0-5)

Are all LLM-parsed primitives grounded in the original event description? This dimension penalises hallucinated or fabricated content, regardless of how many GT primitives were covered.

- 5: Every primitive has clear grounding in the event description; no hallucinated channels or fabricated signal names
- 3: Minor extras or slightly imprecise channel references
- 0: Many hallucinated or fabricated primitives

Semantic Alignment (0-5)

How well does each primitive’s description match the corresponding part of the original event description?

- 5: Precise and accurate descriptions for all primitives
- 3: Roughly correct but imprecise or incomplete
- 0: Wrong or contradicts the event definition

Important Notes

- Logical equivalence counts as a match for both Structural Fidelity and Primitive Coverage (e.g., different granularity, reordered children with same semantics)
- Primitive Accuracy and Coverage are orthogonal: a parse with one perfectly grounded primitive but all others missing scores 5 on Accuracy but 0-1 on Coverage

Output Format

Return a JSON object with the following structure:

```
{  "structural": <float 0-5>,  "coverage": <float 0-5>,  "accuracy": <float 0-5>,  "semantic": <float 0-5>,  "rationale": "<brief explanation of scores>"}
```

E Evaluation Metrics

Why F1-IoU is insufficient for Climate Hazard. NWS event times reflect zone-level condi-

tions rather than point-station measurements, and physical event boundaries are inherently gradual (e.g., fog dissipates over 30–60 min). We observe $F1@IoU-0.5$ is much larger than $F1@IoU-0.9$ across all models, confirming that hard IoU thresholds collapse meaningful performance differences: a model that detects the correct event with a 20% boundary error receives the same zero score as one that misses the event entirely.

Coverage-based metrics. We adopt a coverage-based formulation following Tatbul et al. (2018), instantiated with flat positional bias and no cardinality penalty. For a matched pair (GT interval $[s_g, e_g]$, predicted interval $[s_p, e_p]$), let

$$o = \max(0, \min(e_g, e_p) - \max(s_g, s_p)). \quad (11)$$

and define coverage recall and precision as

$$C_g = \frac{o}{e_g - s_g}, \quad C_p = \frac{o}{e_p - s_p}. \quad (12)$$

These correspond to range-based recall and precision respectively under the framework of Tatbul et al. (2018). Their harmonic mean gives the neutral aggregate:

$$F1^{Cov} = \frac{2 C_g C_p}{C_g + C_p}. \quad (13)$$

Intervals are matched one-to-one (greedy, by descending $F1^{Cov}$) within each event class; unmatched GT intervals contribute $C_g = 0$. The resulting scores are computed independently for each event class and macro-averaged across classes. We denote the reported macro-averaged metric as $\mathcal{MF1}^{Cov}$.

Precision-weighted metric $F0.5^{Cov}$. $F1^{Cov}$ treats C_g and C_p equally. We argue that precision deserves higher weight in this setting for two complementary reasons. First, NWS boundaries represent gradual physical transitions; C_g therefore penalises boundary errors that fall within the inherent uncertainty of the annotation itself, making it a noisier signal than C_p . Second, in operational hazard detection a false alarm—predicting an event outside its true temporal extent—triggers unnecessary emergency response, whereas a conservative boundary miss is less disruptive. We therefore additionally report $F0.5^{Cov}$ ($\beta = 0.5$), which weights precision twice as heavily as recall. Following Tatbul et al. (2018), this corresponds to their REWARD-LOW-FP application profile:

$$F0.5^{Cov} = \frac{1.25 C_g C_p}{0.25 C_p + C_g}. \quad (14)$$

$F1^{Cov}$ is retained as the unweighted baseline. Analogously, $F0.5^{Cov}$ is computed per event class and macro-averaged across classes, yielding the reported $\mathcal{MF0.5}^{Cov}$ metric.

F Numeric and VL-Time Baseline Prompts

F.1 Numeric Baseline

Numeric Direct TS Event Detection

```
You are an expert in time-series analysis.

# Input
event_types: <event_types>
dataset_description: <description>
time_series: <raw time-series values in CSV format>

# Task
Identify the temporal subsequence
corresponding to each target event
type and predict its start and end indices.

For every event type, output exactly one best
candidate interval.
If an event appears absent, still provide the
most plausible candidate
with a low confidence score. The event type
believed to be present
should receive the highest confidence.

# Output
Return ONLY a JSON array:

[
  {
    "className": "<event_name>",
    "start": <integer>,
    "end": <integer>,
    "confidence": <float in [0,1]>
  }
]

Output one entry per event type, with no
additional text.
```

F.2 VL-Time Baseline

VL-Time Visual TS Event Detection

```
You are an expert in signal processing and
time-series analysis.

# Input
event_types: <event_types>
dataset_description: <description>
time_series_info: <plot metadata>
time_series_visualization: <full-range
multichannel plot>

# Task
Inspect the visualization and identify the
temporal subsequence
```

corresponding to each target event type, with precise start and end indices.

For every event type, output exactly one best candidate interval.

If an event appears absent, still provide the most plausible candidate with a low confidence score. The event type believed to be present should receive the highest confidence.

Output

Return ONLY a JSON array:

```
[
  {
    "className": "<event_name>",
    "start": <integer>,
    "end": <integer>,
    "confidence": <float in [0,1]>
  }
]
```

Output one entry per event type, with no additional text.

E.3 Few-Shot Demonstrations

Numeric Few-Shot Demonstration

Annotated Example

The following time series comes from the same dataset and is accompanied by its ground-truth events. Use the example to learn the output format and the numeric signatures of the event types.

Time Series Data

<raw time-series values in CSV format>

Ground-Truth Output

```
[
  {
    "className": "<event_name>",
    "start": <integer>,
    "end": <integer>,
    "confidence": 1.0
  }
]
```

VL-Time Few-Shot Demonstration

Annotated Example

The following visualization comes from the same dataset and is accompanied by its ground-truth events. Use the example to learn the output format and the visual morphology of the event types.

Time Series Info

<plot metadata>

Set	Method	Zero-shot		Few-shot	
		Tokens	Cost	Tokens	Cost
PT	SELA (GPT-5)	233.6	0.623	-	-
	SELA (GPT-4.1)	125.9	0.189	-	-
	Num. (GPT-5)	33.6	0.081	91.9	0.143
	Num. (GPT-4.1)	29.2	0.059	92.2	0.185
	VL-T. (GPT-5)	5.1	0.033	6.7	0.031
	VL-T. (GPT-4.1)	2.3	0.005	4.7	0.010
CH	SELA (GPT-5)	295.4	0.638	-	-
	SELA (GPT-4.1)	188.5	0.296	-	-
	Num. (GPT-5)	39.6	0.091	142.3	0.203
	Num. (GPT-4.1)	35.0	0.071	139.5	0.280
	VL-T. (GPT-5)	5.5	0.033	8.2	0.033
	VL-T. (GPT-4.1)	3.0	0.008	6.5	0.014

Table 7: Average per-sample token usage (K) and inference cost (\$) for LLM-based methods. The slightly lower few-shot cost of VL-Time with GPT-5 on Pressure Test results from reduced output-token usage despite the longer input prompt.

```
# Time-Series Visualization
<full-range multichannel plot>
```

Ground-Truth Output

```
[
  {
    "className": "<event_name>",
    "start": <integer>,
    "end": <integer>,
    "confidence": 1.0
  }
]
```

G Token Budget

Table 7 reports the average per-sample token usage and inference cost of the LLM-based methods. SELA uses more tokens than Numeric and VL-Time because it performs iterative ELT reasoning, signal inspection, and confidence estimation through multiple agent interactions. Despite this additional computation, its average inference cost remains below \$0.65 per sample in all settings. Notably, SELA with GPT-4.1 uses a token budget comparable to few-shot Numeric, while achieving substantially better detection and localization performance. Numeric requires fewer tokens in the zero-shot setting, while VL-Time is the most token-efficient baseline because it reasons directly over compact time-series visualizations. Few-shot prompting increases the token usage of both baselines by including annotated demonstrations.