

LLATISA: Towards Difficulty-Stratified Time Series Reasoning from Visual Perception to Semantics

Yueyang Ding*, Haopeng Zhang, Rui Dai†, Yi Wang,
Tianyu Zong, Kaikui Liu, Xiangxiang Chu

Amap, Alibaba Group

dyyyyy494@gmail.com, zongtianyu20@mails.ucas.ac.cn

{zhanghaopeng.zhp, daima.dr, zhiyi.wy, damon, chuxiangxiang.cxx}@alibaba-inc.com

Abstract

Comprehensive understanding of time series remains a significant challenge for Large Language Models (LLMs). Current research is hindered by fragmented task definitions and benchmarks with inherent ambiguities, precluding rigorous evaluation and the development of unified Time Series Reasoning Models (TSRMs). To bridge this gap, we formalize Time Series Reasoning (TSR) via a four-level taxonomy of increasing cognitive complexity. We introduce HiTSR, a **hierarchical time series reasoning dataset** comprising 83k samples with diverse task combinations and verified Chain-of-Thought (CoT) trajectories. Leveraging HiTSR, we propose LLATISA, a strong TSRM that integrates visualized patterns with precision-calibrated numerical tables to enhance the temporal perception of Vision-Language Models (VLMs). Through a multi-stage curriculum fine-tuning strategy, LLATISA achieves superior performance and exhibits robust out-of-distribution generalization across diverse TSR tasks and real-world scenarios. Our code is available at <https://github.com/RainingNovember/LLaTiSA>.

1 Introduction

Time series data serve as a fundamental modality for capturing the temporal dynamics of complex systems, underpinning a wide range of critical applications, including medical diagnostics (Xie et al., 2025a), financial valuation (Xu et al., 2025), and industrial condition monitoring (Wang et al., 2025b). Interpreting real-world temporal dynamics necessitates Time Series Reasoning (TSR): deriving a comprehensive understanding of time series by grounding analysis in numerical evidence, perceiving salient patterns, and incorporating contextual information to support challenging tasks such as decision-making or forecasting.

Recent studies have demonstrated that Large Language Models (LLMs) can be effective for time-series understanding (Xie et al., 2025b) and provide interpretable reasoning to further enrich traditional time series analysis (Zhang et al., 2025a). Despite these advancements, building unified TSRMs faces two critical bottlenecks: **(1) The lack of a formalized taxonomy for reasoning tasks.** Existing suites like TSQA (Kong et al., 2025) or BED-Time (Sen et al., 2025) either present a mapping of traditional analytical tasks (e.g., forecasting and classification) to text formats or focus on realizing scenario-agnostic series patterns. In contrast, domain-specific datasets like Engine-QA (Wang et al., 2025b) and Heartcare-Bench (Xie et al., 2025a) tend to be numerically-insensitive, primarily relying on the semantic relationships between multiple series. **(2) Reliability deficits in existing benchmarks.** Beyond insufficient contextual information, current benchmarks are often hampered by semantic ambiguities in label definitions and low-fidelity question-answer pairs (Guan et al., 2025). To facilitate the systematic development and rigorous assessment of TSRMs, it is essential to formalize unified definitions of TSR and construct reliable benchmarks that comprehensively model capabilities across diverse scenarios.

Building on the aforementioned limitations, we posit that cultivating reliable TSR capabilities mirrors a multi-stage cognitive process: transitioning from point-level numerical grounding to series-level perception, facilitating high-level semantic interpretation, and ultimately enabling context-aware generation. Guided by this philosophy, we formalize a systematic taxonomy that decomposes TSR into four levels of increasing complexity (L1-L4), as shown in Figure 1(a). Leveraging this taxonomy, we introduce HiTSR (Figure 1(b)), a **hierarchical time series reasoning dataset** comprising 83k samples that span the foundational TSR tasks across the first three levels (L1-L3). All sam-

*Work done at internship at Amap, Alibaba.

†Corresponding author.

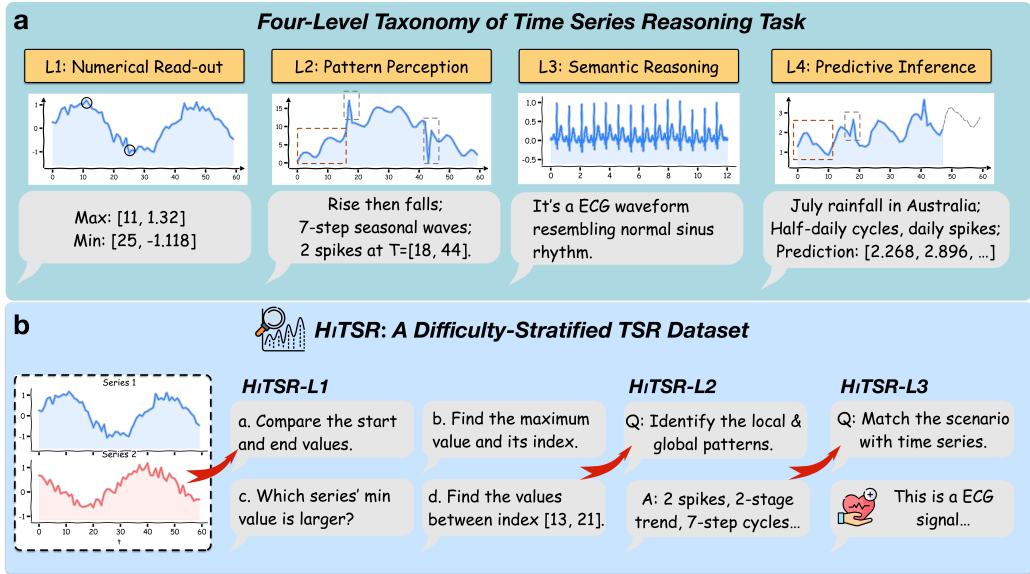


Figure 1: (a) Illustration of the four-level TSR taxonomy and HiTSR dataset. Each incorporates a simple TSR case with increasing difficulties. (b) A brief introduction to HiTSR datasets, containing three levels of TSR scenarios with diverse subtasks.

ples in HiTSR are constructed with unambiguous ground truths and verifiable reasoning chains. **This paper focuses on levels L1-L3, since we find that even advanced TSRMs struggle significantly on these foundational tasks¹**, suggesting that mastering these fundamental stages is a prerequisite for reliable time-series reasoning and remains an immediate challenge.

Recent works have shown that Vision Language Models (VLMs) can excel in basic TSR tasks by relying exclusively on time series visualizations (Zhang et al., 2025a; Sen et al., 2025). However, such vision-centric approaches frequently struggle with numerical reasoning tasks that require precise value-based evidence. Inspired by (Wei et al., 2025), which utilizes visual tokens to represent textual information efficiently, we introduce a robust VLM-based TSRM named **Large Language and Time Series Assistant (LLATISA)**. LLATISA is designed to bridge the gap between qualitative visual intuition and quantitative numerical precision. By employing a three-stage curriculum fine-tuning strategy aligned with our proposed L1-L3 hierarchy, LLATISA demonstrates that TSR benefits from incremental cognitive grounding and high-fidelity Chains-of-Thought (CoT). In summary, our contributions are summarized as follows:

- **A cognitive taxonomy and unified dataset.** We formalize a four-level cognitive taxonomy (L1-

¹We analyze these findings in Appendices A.1 and D.

L4) that captures the progressive nature of temporal understanding. Based on this, we introduce HiTSR, a large-scale, high-fidelity TSR dataset of over 83k samples spanning L1-L3, featuring diverse task settings and verified reasoning chains.

- **A numerically-enhanced TSRM.** We propose LLATISA, a VLM-based TSRM that bridges qualitative perception and quantitative precision. LLATISA is trained via a three-stage curriculum aligned with our L1-L3 hierarchy to build robust reasoning capabilities progressively.
- **Empirical validation of multi-stage reasoning.** Extensive experiments demonstrate that LLATISA consistently outperforms proprietary models and various open-source baselines. Furthermore, our model exhibits superior generalization to out-of-distribution (OOD) scenarios across existing benchmarks.

2 Related Works

2.1 Time Series Reasoning Datasets

Recently released multimodal time series datasets adopt various types of TSR tasks. For general question-answering, early efforts like TSQA (Kong et al., 2025) and TRQA (Jing et al., 2025) explore LLMs' potential in question-answering and traditional time series analysis tasks. SciTS (Wu et al., 2025) introduces a benchmark specifically designed for scientific time series analysis across

a wide range of scientific domains. RATs40k (Yang et al., 2025b) transforms classical time series anomaly detection from discriminative into a reasoning-intensive task and annotates samples with explanatory reasoning. TimeMMD (Liu et al., 2024a) is tailored for the context-aware time series forecasting task. MMTS-Bench (Anonymous, 2025) provides a multi-dimensional task classification taxonomy, yet it overlooks evaluating models’ basic numerical read-out and combining diverse semantics of real scenarios. In addition to insufficient context, current benchmarks are further hampered by semantic ambiguities in label definitions and low-fidelity question-answer pairs (Guan et al., 2025). Moreover, most of the datasets focus on either pattern description or complex reasoning scenarios such as forecasting, and overlook evaluating models’ basic ability of read-out numerical values. Our findings suggest that current models even struggle with finding point-level numerical evidence, suffering from overthinking or hallucinations with longer time horizons.

2.2 Time Series Reasoning Models

Understanding time series via language models generally mostly follows three encoding paradigms: (i) direct serialization into text tokens, (ii) visual rendering as plots for VLMs, and (iii) the integration of dedicated time-series encoders to construct Time-Series Multimodal Large Language Models (TS-MLLMs). TimeOmni-1 (Guan et al., 2025) and Time-R1 (Luo et al., 2025) employ textual inputs for context-aware forecasting. ChatTS (Xie et al., 2025b) and ITFormer (Wang et al., 2025b) incorporate an MLP-based and a Q-former (Li et al., 2023) style time series encoder to perform multivariate TSR tasks, respectively. TimeMaster (Zhang et al., 2025a) enables VLMs to perform interpretable classification based on visualized time series. While recent studies point out that TSRMs struggle with basic TSR tasks like description differentiation (Sen et al., 2025), VLMs have demonstrated superior robustness compared to their LLM and TS-MLLM counterparts. This observation motivates us to further unlock the potential of VLMs for general-purpose TSR scenarios.

3 Method

3.1 Time Series Reasoning Taxonomy

We argue that reliable TSR is inherently compositional, necessitating a hierarchy that can: (i) faith-

fully ground observations to timestamps, (ii) abstract salient temporal patterns beyond point-level evidence, and (iii) integrate with contextual knowledge to support semantic judgment and prediction. In contrast, prior TSR benchmarks are typically organized around isolated tasks and adopt heterogeneous definitions of what constitutes “reasoning”, often over-emphasizing a narrow subset of problem types (e.g., coarse pattern discrimination). Such fragmentation leads to two limitations. First, cross-benchmark comparability is compromised because mismatched capability sets and inconsistent task difficulties prevent a unified assessment of progress. Second, current TSRMs often attempt to perform high-level reasoning (e.g., forecasting) on an unsteady foundation: models still struggle with elementary grounding and verification, which undermines the credibility and interpretability of more complex reasoning.

To address these limitations, we introduce a difficulty-stratified taxonomy that organizes TSR into progressively increasing levels of complexity. This framework serves as a unified schema for categorizing diverse TSR tasks and provides a diagnostic lens to pinpoint the specific cognitive boundaries of current models. Concretely, we decompose TSR into four hierarchical levels:

- **L1: Numerical Read-out.** Establish time-aware indexing and point-level numerical retrieval.
- **L2: Pattern Perception.** Identify and differentiate multi-scale temporal patterns using quantitative evidence.
- **L3: Semantic Reasoning.** Integrate time series observations with contextual knowledge to perform domain-specific reasoning.
- **L4: Predictive Inference.** Generate high-fidelity time-series predictions.

Based on such a taxonomy, we summarize current benchmarks into different levels in Appendix A.2, posing a general standard to task definition and evaluation of TSRMs’ capabilities.

3.2 HiTSR Dataset

Building on our four-level TSR taxonomy, we introduce HiTSR, a unified dataset for both training and evaluation of TSR capabilities across levels L1-L3. Unlike prior benchmarks that may contain ambiguous queries or underspecified context, HiTSR dataset is constructed with rigorous validity checks to ensure exact problem statements and

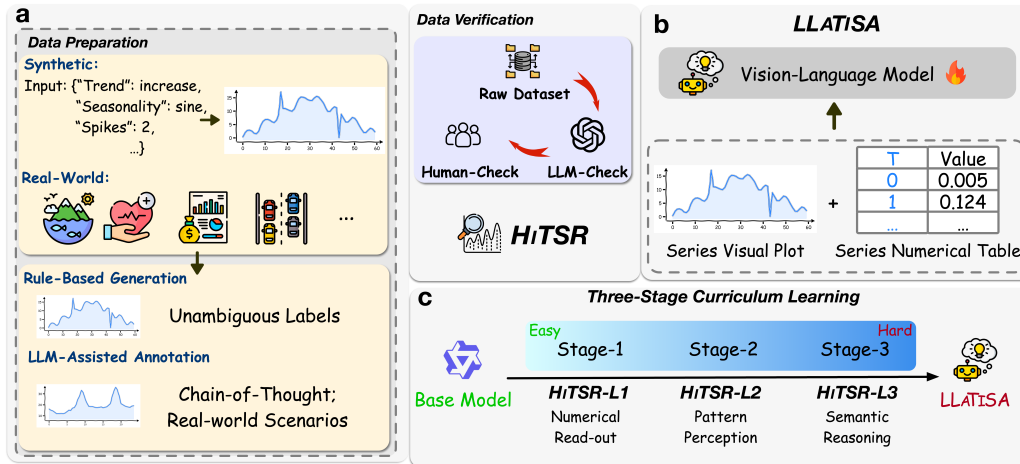


Figure 2: Overview of data pipeline and the model framework. (a) Constructions of HiTSR, including data collection, sample annotation, and multi-stage verification. (b) Framework of LLATISA, which incorporates dual-image inputs and performed (c) three-stage curriculum fine-tuning on the HiTSR.

sufficient contextual information. Overall, HiTSR comprises approximately 83k samples spanning synthetic and real-world time series, annotated and verified via a multi-stage process.

Data Sources. In total, the HiTSR dataset comprises three hierarchical levels: HiTSR-L1 (30,000 samples), HiTSR-L2 (50,703 samples), and HiTSR-L3 (3,121 samples). Each subset includes a high-fidelity test set rigorously verified by human annotators. For HiTSR-L1 and HiTSR-L2, our primary objective is to provide large-scale supervision that strengthens fundamental reasoning abilities in a controlled yet diverse manner. We therefore generate HiTSR-L1 and HiTSR-L2 instances using synthetic time series sourced from (Xie et al., 2025b). This approach allows us to systematically manipulate temporal structures and underlying patterns, while scaling the data to accommodate various training-budget requirements without being restricted by the scarcity of real-world annotations. For HiTSR-L3, we curate publicly available real-world time series from diverse domains and construct question-answer pairs written by the LLM analyzer. The complete data synthesis pipeline and the summary of real-world data information are detailed in Appendix B.

Task Formulation. Following the proposed taxonomy, we instantiate a set of subtasks within each level to cover complementary facets of the TSR as shown in Appendix C.1. L1 tasks are formulated as short-answer questions, given that

the ground truth can be generated via rule-based methods. Tasks within L2-L3 are structured as multiple-choice (MC) questions to ensure objective and reproducible evaluation. To maintain a strict instruction-following orientation and minimize biases from formatting artifacts, each supervision target is crafted as a complete natural-language statement rather than a concise label. Detailed task definitions and additional examples are provided in Appendix C. Additionally, we adopt different answer-construction protocols tailored to the degree of output determinacy. For L1 and the *Numerical Perception* subtask in L2, the correct option can be derived by explicit rules from the underlying series; we therefore generate options programmatically under fixed templates. For the two *Pattern Differentiation* subtasks in L2 and the L3 *Semantic Understanding* task, the outputs are descriptive and require higher-level interpretation. We construct these items with an LLM-assisted pipeline and subsequently verify by humans to ensure correctness and remove ambiguous cases.

Multi-Stage Verified CoT Annotation. We adopt distinct annotation and verification pipelines for L2 and L3 tasks. For L2 *Pattern Differentiation*, we first prompt GPT-5 with each time series’ visual plot and key statistics (e.g., extreme value-index pairs, spike locations with precise values, and notable local events) to produce numerically aware descriptions of local and global patterns. We then construct multiple-choice questions by sam-

pling distractor options from descriptions of other series. To ensure reliability, we employ GPT-5 to cross-validate each question, removing ambiguous descriptions and flawed items (e.g., those with non-unique correct options). Finally, we hold out 10% of the validated pool for evaluation. This subset undergoes a comprehensive human auditing, and only items that pass this rigorous review are retained, yielding a unambiguous test set. For *L3 Semantic Understanding*, we leverage LLMs to generate scenario-aware multiple-choice questions by prompting them with real-world time series and their associated metadata. Each scenario description integrates pattern recognition and numerical grounding, effectively encapsulating the TSRMs’ capabilities across L1-L3. Each item is subsequently dual-checked by GPT-5 and then reviewed by humans to comprehensively verify that the correct answer is supported by consistent numerical, pattern, and semantic evidence.

3.3 LLATISA

Time series can be presented to language models in various formats (Sen et al., 2025). While purely visual plots effectively convey global shapes and local morphological cues, they often lack the fine-grained numerical precision required for grounded reasoning. Consequently, many TSR methods typically represent time series as textualized numerical sequences (Guan et al., 2025; Kong et al., 2025) or integrate LLMs with dedicated time series encoders to develop TS-MLLMs (Xie et al., 2025b; Zhang et al., 2025b). Recent studies, however, indicate that TS-MLLMs frequently underperform relative to general-purpose LLMs or VLMs on L2 TSR tasks, while VLMs are stronger on pattern-centric tasks (e.g., local pattern differentiation) (Sen et al., 2025), underscoring the advantages of visual representations for L2 reasoning.

To empower VLMs with precise numerical grounding, we propose LLATISA (Figure 2.b), a dual-view input framework that pairs standard time series visualizations with a secondary image rendering the data as a structured index-value table. This design facilitates a progressive interpretation paradigm: the first plot enables macroscopic inspection for overall perception, the image of the structured table enables point-accurate verification of specific indices or intervals. By synthesizing these dual modalities alongside textual context, the model produces semantically grounded reasoning. Empirically, compared to processing series as tex-

tual arrays, LLATISA provides extra index information, thereby significantly mitigating numerical hallucinations and improving performance on numerical-sensitive tasks.

Leveraging HiTSR, LLATISA is trained via a three-stage curriculum designed to progressively acquire the prerequisite capabilities defined in our taxonomy. During the initial two stages, we perform sequential Supervised Fine-Tuning (SFT) on HiTSR-L1 and HiTSR-L2 to consolidate the model’s numerical read-out precision and pattern perception capabilities. In the third stage, LLATISA could be either trained on HiTSR-L3 or general-purpose domain-specific TSR benchmarks. Our experiments further show that existing TSRMs often fail even on basic L1-L2 TSR, and struggle to combine LLM priors with series-level evidence for reliable L3 semantic inference. In contrast, curriculum-trained LLATISA across L1-L2 could be easily transferred to domain-specific L3 tasks (e.g., ECG interpretation).

4 Experiment

Our experiments focus on investigating the following research questions (RQs):

- **RQ1:** How does LLATISA perform in existing TSR benchmarks across levels L1-L3?
- **RQ2:** How does LLATISA perform compared with different encoding strategies?
- **RQ3:** How does LLATISA perform in generalizing to real-world applications?
- **RQ4:** How does the curriculum tuning strategy and CoT-aware training data benefit LLATISA’s performance?

4.1 Experiment Settings

To leverage the VLM’s fundamental capability in interpreting numeric-centric images, LLATISA employs Qwen3-VL-8B-Instruct (Bai et al., 2025) as its backbone. To ensure a fair comparison, we report results exclusively on out-of-distribution (OOD) datasets across levels L1-L3 (see Table 1). To demonstrate that LLATISA serves as a strong foundation model when fine-tuned for a new domain, we further evaluate it on ECG interpretation under both in-distribution (ID) and OOD settings using the ECG-Grounding dataset (Lan et al., 2025), as shown in Table 3. Additionally, we conduct a hierarchical evaluation of baseline models

Modality	Model	L1 Numerical Read-out			L2 Pattern Perception		L3 Semantic Reasoning
		Min & Max Localization			Local	Global	Series Comparison
		Acc%	half-Acc%	SR%	Acc%	Acc%	Acc%
Text (w/o index)	GPT-4o	47.4	69.8	100.0	47.6	95.0	43.0
	Qwen3-8B	36.4	52.6	68.0	35.6	77.5	36.0
	Time-R1	0.0	16.0	100.0	32.4	46.7	43.0
	Time-MQA	0.0	0.6	17.4	19.0	28.3	10.0
Vision (plot)	GPT-4o	-	-	-	<u>72.2</u>	<u>96.7</u>	39.0
	Qwen3-VL-8B	-	-	-	38.2	85.8	41.0
Vision (plot + num)	GPT-4o	2.4	11.0	100.0	65.4	95.0	47.0
	Qwen3-VL-8B	2.0	17.4	100.0	36.4	86.7	47.0
Time Series	ChatTS	7.8	29.2	100.0	57.0	80.0	<u>59.0</u>
	OpenTSLM	-	-	-	7.8	15.0	13.0
Vision + Text (w/o index)	GPT-4o	17.4	49.8	100.0	65.8	94.2	49.0
	Qwen3-VL-8B	12.4	42.6	72.4	42.8	92.5	52.0
Vision + Text (w/ index)	GPT-4o	<u>54.2</u>	<u>88.6</u>	100.0	65.8	<u>96.7</u>	48.0
	Qwen3-VL-8B	34.2	68.0	90.6	43.0	93.3	54.0
Vision (plot + num)	LLATISA	86.8	96.0	100.0	75.6	97.5	67.0

Table 1: Performance comparison on **out-of-distribution datasets** (primarily prior benchmarks BEDTime, MMTS-Bench, and MCQ2, which differ substantially from the training data) across L1-L3. All tasks report accuracy (Acc), “half-Acc” represents the performance in locating one of the maximum or minimum values, and “SR” denotes whether the model provides valid answers or correctly maps target values with correct indices. For VLMs, “plot” and “num” denote rendering time series as a sequence plot and a numerical table, respectively. “-” denotes that the input modality cannot handle the corresponding task. The best result is **bolded** and the second best is in underlined.

on HiTSR and present findings regarding the limitations of current TSRMs (see Appendix D for details). Comprehensive training specifications, including parameter configurations and the proportions of training datasets across levels L1-L3, are also provided in Appendix E.

Baselines. We conduct extensive experiments across various time series encoding strategies, including: (i) encoding series as textual arrays; (ii) visualizing as sequence plots or numerical grids (denoted as “plot” and “num” in our tables); (iii) adopting a dual-modality input comprising both visualizations and textual sequences; and (iv) leveraging TS-MLLMs. Since LLATISA incorporates index information in the second image, we also construct two types of textual encoding methods, “w/ index” and “w/o index”, to ensure fair comparison. For closed-source models, we select GPT-4o (Achiam et al., 2023), utilizing it as both LLMs and VLMs. For open-source models, we employ Qwen3-8B (Yang et al., 2025a) and its multimodal counterpart, Qwen3-VL-8B (Bai et al., 2025), to evaluate the efficacy of our curriculum fine-tuning strategy and to conduct a comparative analysis of various training paradigms. We also incorporate textual-encoding TSRMs, such as Time-R1 (Luo et al., 2025) and Time-MQA (Kong et al., 2025). For TS-MLLMs, we evaluate ChatTS (Xie et al.,

2025b) alongside OpenTSLM (Langer et al., 2025). For ECG interpretation, we include GEM (Lan et al., 2025) and PULSE (Liu et al., 2024b) as domain-specific methods. We adopt standardized instruction formats for different time series encoding strategies to ensure a fair comparison. Detailed information on baseline models and corresponding instruction templates is provided in Appendix F.

Datasets. For the L1 extreme value localization task, we construct OOD evaluations using the real-world subset of HiTSR-L1, given that LLATISA is trained exclusively on its synthetic counterpart. Regarding L2, we employ BEDTime (Sen et al., 2025) and MMTS-Bench, which focus on local and global pattern differentiation, respectively. To balance inference overhead, we randomly sample 500 questions from BEDTime for testing, while the complete corresponding subset of MMTS-Bench (120 samples) is utilized. For L3, we use MCQ2 (Merrill et al., 2024) to assess multi-series contextual reasoning. Following the experimental setting of (Xie et al., 2025b), we randomly sample 100 questions from MCQ2. To further verify LLATISA’s potential as a robust foundation model for L3-level real-world multivariate TSR, we conduct domain-specific fine-tuning on ECG-Grounding (Lan et al., 2025), assessing the model’s transferability across both ID and OOD scenarios.

Modality	Model	L1 Numerical Read-out		L2 Pattern Perception		L3 Semantic Reasoning
		Min & Max Localization		Local	Global	Series Comparison
		Acc%	SR%	Acc%	Acc%	Acc%
Text	w/ index	87.6	100.0	39.0	94.2	47.0
	w/o index	65.2	92.4	37.6	89.2	43.0
Vision	Single image (plot)	-	-	<u>68.0</u>	97.5	<u>62.0</u>
	Single image (num)	<u>86.8</u>	100.0	32.4	75.0	32.0
Multi-Modalities	Plot + Text (w/ index)	85.2	100.0	49.8	<u>95.8</u>	60.0
	Plot + Text (w/o index)	57.2	86.4	50.4	94.2	57.0
Vision	LLATISA	<u>86.8</u>	100.0	75.6	97.5	67.0

Table 2: Encoding strategy comparison in **out-of-distribution datasets** across L1-L3. “Modality” denotes the encoding format of time series. “-” denotes that the input modality cannot handle the corresponding task. All models follow the same training settings based on the Qwen3-8B or Qwen3-VL-8B backbones. All metrics are reported as averages in all subtasks across L1-L3.

Evaluation metrics. We report accuracy (Acc) as the primary metric for evaluation across all tasks. Additionally, we report the success rate (SR) to assess the model’s ability to generate valid answers.

4.2 OOD Experiment Results

As shown in Table 1, we conduct comprehensive OOD evaluations comparing LLATISA with various baselines across the L1-L3 levels. The results demonstrate that LLATISA achieves significant improvements across diverse TSR scenarios. Specifically, in the L1 point-level value localization task, we observe a markedly lower success rate on text-based models. These models frequently struggle to output valid answers, often becoming trapped in repetitive generation loops that lead to sequences exceeding the maximum context length.

4.3 Impact of TS Representation Strategies

To empirically validate the efficacy and efficiency of LLATISA’s dual-view encoding strategy, we perform a comprehensive comparison against various time-series representation strategies (introduced in Section 4.1) under the same backbone and training configuration, and evaluate performance on OOD benchmarks at levels L1-L3.

Performance Comparison. Table 2 presents the comprehensive performance on OOD testbeds across various time series representation strategies. LLATISA consistently outperforms alternative strategies on the majority of OOD tasks across levels L1-L3, demonstrating its superior efficacy in TSR. Under the curriculum learning strategy, almost all strategies show performance

gains over their untuned counterparts. For text-based paradigms (notably the “w/o index” variant), the success rate for L1 tasks improves markedly with fine-tuning. Furthermore, incorporating explicit index information significantly improves the precision of point-level localization, leading to substantial improvements in both accuracy and SR.

4.4 Application in ECG Interpretation

To evaluate the generalization capabilities of LLATISA (previously fine-tuned on HiTSR), we further perform Supervised Fine-Tuning (SFT) on the ECG-Grounding 30k dataset. For a rigorous comparison, we also fine-tune Qwen3-VL-8B on the same dataset. Baseline results for ECG-specific models are directly sourced from (Lan et al., 2025). The full results are shown in Table 17.

As illustrated in Table 3, LLATISA exhibits a distinct advantage in lead-wise evaluation, which directly reflects its adherence to the structured, 12-lead diagnostic procedure employed by professional clinicians. Specifically, LLATISA achieves remarkable gains in lead assessment coverage and accuracy, outperforming GEM (LLaVA) by 18.14% and 14.22% in the ID evaluation, respectively. These results indicate that LLATISA more consistently executes per-lead analysis and generates more reliable lead-specific statements. Compared to Qwen3-VL-8B, LLATISA demonstrates superior performance across all metrics, with notable improvements on diagnostic accuracy (5.42%) and evidence-based reasoning (4.39%) in the OOD scenario. Notably, LLATISA is fine-tuned on only 2.5% of GEM’s instruction data (30k out of 1.186M

Type	Model	Diag. Acc.	L. Cov.	L. Acc.	Evi. Reas.
ID	PULSE	81.1	7.1	3.0	52.4
	GEM (LLaVA)	87.2	<u>71.1</u>	<u>46.4</u>	75.1
	GEM (PULSE)	<u>86.5</u>	69.8	45.3	<u>74.7</u>
	Qwen3-VL-8B	60.9	69.3	50.1	63.8
	LLATiSA	62.8	84.0	53.0	71.2
OOD	PULSE	59.2	11.2	6.3	55.5
	GEM (LLaVA)	<u>73.5</u>	80.0	<u>49.0</u>	<u>74.6</u>
	GEM (PULSE)	73.6	<u>78.9</u>	48.0	75.4
	Qwen3-VL-8B	59.0	56.4	38.1	63.8
	LLATiSA	62.2	66.5	49.2	66.6

Table 3: Experimental results on the L3 ECG-Grounding task. Diag. Acc. stands for Diagnosis Accuracy; L. Cov. and L. Acc. denote Lead Assessment Coverage and Accuracy, respectively; Evi. Reas. represents Evidence-Based Reasoning. .

samples). This high data efficiency demonstrates that LLATiSA can be efficiently specialized for grounded ECG interpretation and highlights its potential as a general TSR foundation model for real-world applications.

4.5 Ablation Study

Effectiveness of CoT. In the HiTSR dataset, we incorporate LLM-annotated CoT data, verified by humans, to enhance LLATiSA’s performance and generalization, particularly on the more complex tasks at levels L2-L3. To isolate the impact of CoT, we perform SFT on the same dataset using only the correct options as supervision (“w/o CoT”). As shown in Table 4, while ID performance does not degrade significantly, the OOD performance deteriorates markedly, with accuracy decreasing by 4.31% and 17.91% on L2 and L3, respectively. Furthermore, training solely on correct options significantly undermines the model’s original instruction-following capabilities. We observe that the model tuned without CoT data fails to generate reasoning chains, even when explicitly prompted to do so. This phenomenon highlights the critical role of HiTSR as a thinking-aware TSR training resource. By integrating high-quality CoT trajectories, our benchmark not only incentivizes robust cross-task reasoning but also synergistically preserves the foundational generative alignment of the backbone model.

Effectiveness of curriculum learning. We validate the necessity of a three-stage curriculum for

Type	Ablation	L2	L3	ECG-Grounding
ID	w/o CoT	97.0	80.0	57.4
	joint training	95.0	84.0	61.5
	LLATiSA	97.0	87.0	62.8
OOD	w/o CoT	93.3	55.0	54.3
	joint training	95.8	57.0	59.7
	LLATiSA	97.5	67.0	62.2

Table 4: Ablation results. “w/o CoT” denotes SFT on the same dataset using only the correct options as supervision during training on L2-L3, “joint training” performs a single-stage training regime.

SFT, as opposed to a single-stage regime (i.e., “joint training”) in which all tasks are pooled and shuffled. As illustrated in Table 4, single-stage joint training consistently degrades performance, yielding drops of 2.06% and 3.45% in ID tasks for L2 and L3, respectively, and reductions of 1.74% and 14.93% under OOD evaluation for L2 and L3. While the joint-training model maintains relatively stable performance on ID scenarios and simpler OOD tasks (L2), it exhibits a significant decline on OOD L3 tasks, suggesting that without curriculum learning, the model struggles to internalize the logic needed for complex semantic reasoning, thereby limiting generalization.

5 Conclusion

In this paper, we propose a difficulty-stratified view of TSR and formalize a four-level taxonomy (L1-L4) that characterizes progressive TSR capabilities. We introduce HiTSR, a large-scale hierarchical dataset with over 83k high-fidelity TSR questions spanning L1-L3, constructed with unambiguous labels and verifiable reasoning chains to support both reliable training and evaluation. Building on HiTSR, we present LLATiSA, a VLM-based TSR model that pairs time series plots with an auxiliary index-value table to bridge qualitative perception and precise numerical evidence, and is trained via a three-stage curriculum aligned with L1-L3. Extensive experiments on diverse OOD benchmarks and real-world scenarios show that LLATiSA consistently outperforms strong baselines and different modeling strategies, and transfers effectively to domain-specific semantic reasoning, suggesting a practical path toward more robust unified TSRMs.

Limitations

Our study primarily focuses on systematic dataset construction and supervised curriculum fine-tuning, leaving the exploration of Reinforcement Learning Fine-Tuning (RFT) on HiTSR as a future direction. Effectively implementing RFT in this domain remains challenging due to the complexity of reward design, which must simultaneously supervise low-level numerical precision and high-level semantic logic across our L1-L4 taxonomy. Future work includes: (i) integrating RL into the hierarchical curriculum to further refine the model’s policy; (ii) investigating robust initialization strategies to mitigate cold-start issues.

Ethics Statement

All datasets and language models used in this work are publicly available. The HiTSR dataset was constructed from well-established, publicly accessible time series benchmarks and synthetic data generation pipelines, and LLATISA is built on the open-sourced model.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Anonymous. 2025. [MMTS-Bench: A comprehensive benchmark for multimodal time series understanding and reasoning](#). Under review at ICLR 2026.
- Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, and 45 others. 2025. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*.
- Benjamin S Bloom, Max D Engelhart, Edward J Furst, Walker H Hill, David R Krathwohl, and 1 others. 1956. Handbook i: cognitive domain. *New York: David McKay*, pages 483–498.
- Yifu Cai, Arjun Choudhry, Mononito Goswami, and Artur Dubrawski. 2024. Timeseriesexam: A time series understanding exam. *arXiv preprint arXiv:2410.14752*.
- Jialin Chen, Aosong Feng, Ziyu Zhao, Juan Garza, Gaukhar Nurbek, Cheng Qin, Ali Maatouk, Leandro Tassiulas, Yifeng Gao, and Rex Ying. 2025. Mtbench: A multimodal time series benchmark for temporal reasoning and question answering. *arXiv preprint arXiv:2503.16858*.
- Rakshitha Godahewa, Christoph Bergmeir, Geoffrey I Webb, Rob J Hyndman, and Pablo Montero-Manso. 2021. Monash time series forecasting archive. *arXiv preprint arXiv:2105.06643*.
- Tong Guan, Zijie Meng, Dianqi Li, Shiyu Wang, Chao-Han Huck Yang, Qingsong Wen, Zuozhu Liu, Sabato Marco Siniscalchi, Ming Jin, and Shirui Pan. 2025. TimeOmni-1: Incentivizing complex reasoning with time series in large language models. *arXiv preprint arXiv:2509.24803*.
- Thomas Huber and Christina Niklaus. 2025. Llms meet bloom’s taxonomy: A cognitive view on large language model evaluations. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 5211–5246.
- illness. Illness Dataset. <https://gis.cdc.gov/grasp/fluview/fluportaldashboard.html>.
- Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and 1 others. 2023. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*.
- Baoyu Jing, Sanhorn Chen, Lecheng Zheng, Boyu Liu, Zihao Li, Jiaru Zou, Tianxin Wei, Zhining Liu, Zhichen Zeng, Ruizhong Qiu, and 1 others. 2025. TRQA: Time series reasoning question and answering benchmark. OpenReview. Withdrawn submission to ICLR 2026. Available at <https://openreview.net/forum?id=ULQt51DRug>.
- 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. In *Association for Computational Linguistics (ACL)*.
- Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. 2018. Modeling long-and short-term temporal patterns with deep neural networks. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 95–104.
- Xiang Lan, Feng Wu, Kai He, Qinghao Zhao, Shenda Hong, and Mengling Feng. 2025. GEM: Empowering mllm for grounded ecg understanding with time series and images. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Patrick Langer, Thomas Kaar, Max Rosenblattl, Maxwell A Xu, Winnie Chow, Martin Maritsch, Aradhana Verma, Brian Han, Daniel Seung Kim, Henry Chubb, and 1 others. 2025. OpenTSLM: Time-series language models for reasoning over multivariate medical text-and time-series data. *arXiv preprint arXiv:2510.02410*.

- Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.
- Haoxin Liu, Shangqing Xu, Zhiyuan Zhao, Lingkai Kong, Harshvardhan Prabhakar Kamarthi, Aditya Sasanur, Megha Sharma, Jiaming Cui, Qingsong Wen, Chao Zhang, and 1 others. 2024a. Time-mmd: Multi-domain multimodal dataset for time series analysis. *Advances in Neural Information Processing Systems*, 37:77888–77933.
- Ruoqi Liu, Yuelin Bai, Xiang Yue, and Ping Zhang. 2024b. Teach multimodal llms to comprehend electrocardiographic images. *arXiv preprint arXiv:2410.19008*.
- Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. 2024c. Timer: Generative pre-trained transformers are large time series models. *arXiv preprint arXiv:2402.02368*.
- Yucong Luo, Yitong Zhou, Mingyue Cheng, Jiahao Wang, Daoyu Wang, Tingyue Pan, and Jintao Zhang. 2025. Time series forecasting as reasoning: A slow-thinking approach with reinforced llms. *arXiv preprint arXiv:2506.10630*.
- Mike A Merrill, Mingtian Tan, Vinayak Gupta, Thomas Hartvigsen, and Tim Althoff. 2024. Language models still struggle to zero-shot reason about time series. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 3512–3533.
- Mark Monmonier. 1985. *Semiology of graphics: Diagrams, networks, maps*.
- Paul Quinlan, Qingguo Li, and Xiaodan Zhu. 2025. Chat-ts: Enhancing multi-modal reasoning over time-series and natural language data. *arXiv preprint arXiv:2503.10883*.
- Medhasweta Sen, Zachary Gottesman, Jiaying Qiu, C Bayan Bruss, Nam Nguyen, and Tom Hartvigsen. 2025. BEDTime: A unified benchmark for automatically describing time series. *arXiv preprint arXiv:2509.05215*.
- Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, and 1 others. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.
- 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.
- Yilin Wang, Peixuan Lei, Jie Song, Yuzhe Hao, Tao Chen, Yuxuan Zhang, Lei Jia, Yuanxiang Li, and Zhongyu Wei. 2025b. ITFormer: Bridging time series and natural language for multi-modal qa with large-scale multitask dataset. In *International Conference on Machine Learning (ICML)*.
- Haoran Wei, Yaofeng Sun, and Yukun Li. 2025. Deepseek-ocr: Contexts optical compression. *arXiv preprint arXiv:2510.18234*.
- Wetterstation. 2020. Weather dataset. <https://www.bgc-jena.mpg.de/wetter/>.
- Andrew Robert Williams, Arjun Ashok, Étienne Marcotte, Valentina Zantedeschi, Jithendaraa Subramanian, Roland Riachi, James Requeima, Alexandre Lacoste, Irina Rish, Nicolas Chapados, and 1 others. 2024. Context is key: A benchmark for forecasting with essential textual information. *arXiv preprint arXiv:2410.18959*.
- Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. 2021. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430.
- Wen Wu, Ziyang Zhang, Liwei Liu, Xuenan Xu, Junlin Liu, Ke Fan, Qitan Lv, Jimin Zhuang, Chen Zhang, Zheqi Yuan, and 1 others. 2025. SciTS: Scientific time series understanding and generation with llms. *arXiv preprint arXiv:2510.03255*.
- Yihan Xie, Sijing Li, Tianwei Lin, Zhuonan Wang, Chenglin Yang, Yu Zhong, Wenqiao Zhang, Haoyuan Li, Hao Jiang, Fengda Zhang, and 1 others. 2025a. Heartcare suite: Multi-dimensional understanding of ecg with raw multi-lead signal modeling. *arXiv preprint arXiv:2506.05831*.
- Zhe Xie, Zeyan Li, Xiao He, Longlong Xu, Xidao Wen, Tiejing Zhang, Jianjun Chen, Rui Shi, and Dan Pei. 2025b. ChatTS: Aligning time series with llms via synthetic data for enhanced understanding and reasoning. *Proceedings of the VLDB Endowment*, 2025.
- Wenyan Xu, Dawei Xiang, Yue Liu, Xiyu Wang, Yanxiang Ma, Liang Zhang, Shu Hu, Chang Xu, and Jiaheng Zhang. 2025. FinMultiTime: A four-modal bilingual dataset for financial time-series analysis. *arXiv preprint arXiv:2506.05019*.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Yiyuan Yang, Zichuan Liu, Lei Song, Kai Ying, Zhiguang Wang, Tom Bamford, Svitlana Vyetrenko, Jiang Bian, and Qingsong Wen. 2025b. Time-ra: Towards time series reasoning for anomaly with llm feedback. *arXiv preprint arXiv:2507.15066*.

Wen Ye, Jinbo Liu, Defu Cao, Wei Yang, and Yan Liu. 2025. When llm meets time series: Can llms perform multi-step time series reasoning and inference. *arXiv preprint arXiv:2509.01822*.

Junru Zhang, Lang Feng, Xu Guo, Yuhang Wu, Yabo Dong, and Duanqing Xu. 2025a. Timemaster: Training time-series multimodal llms to reason via reinforcement learning. In *NeurIPS 2025 Workshop on Behavioral Foundation Models, Representational Learning, and Time Series (BERT2S)*.

Zhirui Zhang, Changhua Pei, Tianyi Gao, Zhe Xie, Yibo Hao, Zhaoyang Yu, Longlong Xu, Tong Xiao, Jing Han, and Dan Pei. 2025b. Timesense: Making large language models proficient in time-series analysis. *arXiv preprint arXiv:2511.06344*.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 11106–11115.

Luca Zhou, Pratham Yashwante, Marshall Fisher, Alessio Sampieri, Zihao Zhou, Fabio Galasso, and Rose Yu. 2025. CaTS-Bench: Can language models describe numeric time series? *arXiv preprint arXiv:2509.20823*.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, and 1 others. 2025. InternV3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*.

Appendix

A Findings

A.1 Model Analysis

Our research is primarily motivated by a critical appraisal of existing TSR benchmarks. Preliminary evaluations of GPT-4o on local pattern differentiation tasks reveal significant deficiencies in handling foundational TSR operations, echoing the observations in (Sen et al., 2025). As illustrated in Table 18, GPT-4o fails to perform accurate numerical grounding when interpreting a visual plot; it struggles to discern the precise relationship between start and end points or characterize the specific rate of change (e.g., confusing abrupt drops with "smooth" decreases). This diagnostic case underscores a pivotal prerequisite: establishing robust numerical read-out capabilities for TSRMs is essential before tackling the higher-order reasoning required in L3 and L4 tasks. Furthermore, our empirical results across both ID (Table 15) and OOD

(Table 1) scenarios demonstrate that current LLMs, VLMs, and even specialized TSRMs fall short in foundational L1-L2 tasks, which suggests a fundamental perception-reasoning gap.

A.2 Dataset Analysis

We systematically categorize existing TSR datasets according to our proposed four-level taxonomy in Table 16, through which we identify several critical deficiencies in current benchmarks. For instance, certain questions exhibit label ambiguity, containing multiple valid options (Tables 20 and 21), while others lack sufficient contextual information for logical reasoning (Table 22). Furthermore, most TSR datasets are designed exclusively for evaluation rather than providing comprehensive training-test splits. Among the limited training data available, the absence of Chain-of-Thought (CoT) annotations is a major drawback, while solely relying on correct labels for training tends to degrade the generative reasoning capabilities of TSRMs.

B Data Source

B.1 Synthetic Time Series

We use the complete time series synthesis pipeline introduced by (Xie et al., 2025b). To avoid ambiguity in options and control question quality, we merge some of the preset attributions to ensure that the remaining attributes do not introduce multiple correct options when used to create local pattern recognition problems. The new attribution pool is summarized in Table 5.

B.2 Real-world Time Series

We utilize real-world time series data for: (i) creating OOD test sets for the L1 task; (ii) constructing semantic understanding subsets. For the L1 evaluations reported in Table 1, we select datasets as follows:

ETT (4 subsets) (Zhou et al., 2021) contains a group of four subsets of oil temperature and power load collected by electricity transformers from July 2016 to July 2018, with minutes or hourly recorded frequency.

Weather (Wetterstation, 2020) records meteorological time series with 21 weather indicators collected every 10 minutes from the Weather Station of the Max Planck Biogeochemistry Institute in 2020.

Exchange Rate (Lai et al., 2018) includes the daily exchange rates of eight different countries, ranging from 1990 to 2016.

Traffic (Wu et al., 2021) encompasses the hourly measures of road occupancy rates obtained from 862 sensors situated in the San Francisco Bay area freeways between January 2015 and December 2016.

Electricity (Wu et al., 2021) mainly consists of hourly electricity consumption data from 321 customers.

To facilitate the semantic understanding task, we augment our dataset with large-scale time series repositories, including UTSD (Liu et al., 2024c), Monash (Godaheva et al., 2021), and Time-MMD (Liu et al., 2024a). These datasets provide rich meta-annotations that serve as the basis for generating ground-truth options. Each time series is subjected to random cropping to generate sub-series, thereby expanding the data volume and enhancing descriptive diversity. Comprehensive statistics of these datasets are provided in Table 13.

For each sample, we prompt GPT-5 with the raw time series (represented as textual arrays) and its associated metadata to synthesize scenario-aligned MCQs. The ground truth is derived from the actual data source, while distractors are engineered to be erroneous in terms of numerical precision, temporal pattern perception, or contextual consistency, which correspond to levels L1–L3 in our TSR taxonomy. We further define two semantic evaluation settings: (i) single-scenario, where all options describe the series within the same metadata context, and (ii) multi-scenario, where the LLM is instructed to generate cross-contextual wrong options that deviate from the ground truth.

C Details of HiTSR

C.1 Task definition

The full task definition is summarized in Table 14.

C.2 Data Verification

L1: Numerical Read-out. The question-answer (QA) pairs utilized in this stage are completely constructed by strict rule-based annotations based on each time series’ attribution pool.

L2: Pattern Perception. For numerical perception, it is similar to L1, which generated the complete QA pairs through an explicit rule-based

Type	attribution	Caption
Global	Trend	increase, decrease, steady, multi-stage;
	Seasonality	sin, square, triangle, none;
	Frequency	high, low;
	Noise	noisy, clean;
Local	Spike	upward, downward;
	Turning point	rises then falls, falls then rises;
Numerical	Min/Max	[index, value] pairs;
	Start/End	values;
	Key event positions	spike, turning point;

Table 5: Overview of synthetic time series attributions.

pipeline. The distractors are created by slightly perturbing the numerical evidence.

For local pattern differentiation, distractors are synthesized using rule-based perturbations, such as numerical manipulation or random attribute sampling from heterogeneous series. To ensure label uniqueness, we perform cross-verification by benchmarking each candidate option against the ground-truth attributes of the series. Subsequently, we use GPT-5 with both the metadata and visual plots of the given time series to annotate the correct answers with CoT reasoning. For quality assurance, the training set only includes samples correctly resolved by GPT-5, while the test set undergoes rigorous human verification to ensure high fidelity.

Regarding global pattern differentiation, we first prompt GPT-5 to generate comprehensive descriptions of the series. Distractors are then constructed by incorporating annotations from disparate sequences. Similarly, the original dataset is processed by GPT-5 to produce reasoning-enhanced ground truth, ensuring the depth and accuracy of the labels.

L3: Semantic Understanding. The raw QA pairs are initially synthesized by GPT-5 in a single pass. To ensure data integrity, we implement a cross-validation stage where GPT-5 evaluates the generated pairs against a set of rigorous criteria. This evaluation focuses on two critical dimensions: (i) verifying the ground truth, ensuring that numerical evidence, pattern characterization, and semantic logic are technically sound; and (ii) auditing distractors, where each incorrect option is re-examined to eliminate potential overlaps or counterexamples that might inadvertently align with the

time series. Finally, a manual dual-check is performed on the validated candidates to finalize a high-fidelity, real-world TSR dataset.

C.3 Prompt Templates

The prompt templates used during the construction of HiTSR are summarized as follows:

- **L2: Annotation:** Figure 3.
- **L2: CoT Generation:** Figure 4.
- **L3: QA Generation:** Figure 5.
- **L3: Question Examination:** Figure 6.

C.4 Illustration of HiTSR

We provide a QA sample for each subtask within HiTSR, which is summarized as follows:

- *Numerical Read-out:* Tables 23 to 26.
- *Pattern Perception:* Tables 27 to 29.
- *Semantic Reasoning:* Table 30.

C.5 Comparison to Existed TSR Datasets

The differences among HiTSR and existing TSR datasets are summarized as follows:

- **The Enrichment of Verified Chain-of-Thought.** Different from evaluation-only benchmarks (e.g., BEDTime (Sen et al., 2025), TimeMCQ2 (Merrill et al., 2024), and MMTS-Bench (Anonymous, 2025)), HiTSR provides 83k samples enriched with verified Chain-of-Thought (CoT) trajectories. By transforming raw data into high-quality instruction-tuning pairs, we enable TSRMs to learn the "process of reasoning" rather than solely mapping correct answers. Our results (Section 4.5) show that this approach significantly enhances the LLATISA’s performance in complex reasoning tasks while preserving the model’s general instruction-following capabilities.
- **Unambiguous and Logic-Grounded CoT.** In contrast to the datasets with noisy (see Table 13-14 in the manuscript) or purely semantic annotations, our CoT follows a strict “Perception-to-Reasoning” logic (Numerical \rightarrow Pattern \rightarrow Semantic). Each step is cross-verified against ground-truth data, ensuring that the reasoning path is grounded in numerical reality rather than textual hallucination.
- **Emphasis on Foundational TSR Capabilities.** Most existing studies (e.g., Time-MMD (Liu et al., 2024a), TSR-Suite (Guan et al., 2025))

focus on L2-L4 high-level TSR tasks but overlook the verification of a model’s numerical read-out ability, especially in long-horizon settings. HiTSR uniquely introduces L1 tasks (precise grounding and localization) as a diagnostic anchor. This ensures that complex TSR abilities are evaluated only after verifying the model’s foundational perceptual accuracy.

- **A clear difficulty-Stratified taxonomy for TSR.** Guided by our 4-level taxonomy (grounded in Bloom’s Taxonomy (Bloom et al., 1956) and Bertin’s Levels of Reading (Monmonier, 1985)), HiTSR acts as a systematic diagnostic framework. Unlike fragmented benchmarks, HiTSR’s stratified design allows for curriculum learning (L1 \rightarrow L3) and enables researchers to pinpoint exactly where a TSRM’s reasoning chain breaks down.

D Evaluation on the HiTSR

We evaluate representative baseline models on the HiTSR test set, comprising 100 randomly sampled questions, with results summarized in Table 15. Key observations are as follows: First, incorporating temporal indices into textualized numerical arrays significantly enhances the reasoning performance of LLMs; for instance, GPT-4o’s L1 accuracy improves from 6.0% to 23.0% with indices, particularly aiding numerical grounding tasks. Second, ChatTS demonstrates a strong capability to estimate approximate values while being poor at mapping the correct index for the target value, achieving a competitive 70.0% in semantic understanding (L3), which is merely a 1.0% performance gap compared to the best-performing multimodal GPT-4o.

A consistent performance disparity exists between local and global pattern differentiation, revealing the limitations of single-modality reasoning. While pure vision or text-only models excel in global perception (e.g., GPT-4o reaching 93.0% and 94.0% in L2-Global, respectively), they both struggle with local granularity, as evidenced by their stagnant performance in L2-Local (hovering around 55.0%). This bottleneck is exemplified in Table 18, where pure vision fails to discern precise temporal transitions, such as confusing a sharp step-drop with a “smooth” decrease.

Crucially, our results demonstrate a significant performance leap when Vision and Text modalities are integrated, with GPT-4o’s L2-Local accuracy rising to 69.0%. This synergistic gain

highlights the effectiveness of a dual-stream input paradigm—utilizing visual plots for overall pattern recognition and textual sequences for numerical precision. These findings suggest that future TSRMs should prioritize the fusion of visual global context and numerical fine-grained features to overcome the inherent perceptual gaps of isolated modalities.

For the “Numerical Perception task” in L2, we introduce subtle numerical perturbations into local pattern differentiation to synthesize challenging distractors. Experimental results in Table 15 show that Qwen3-VL-8B achieves the state-of-the-art (SOTA) performance of 78.0% when equipped with both vision and indexed textual series inputs. Besides, we observe a significant performance degradation for Qwen3-8B in the “Text (w/o index)” setting (36.0%); this is primarily due to its inefficient regressive value retrieval patterns, which often exceed the maximum output length, thus fail to answer the question. Furthermore, the Vision-only modality consistently yields the lower scores across models (e.g., 58.0% for GPT-4o and 47.0% for Qwen3-VL). Such a failure of visual heuristics underscores the success of our dataset construction: the “Numerical Perception” task necessitates a reliance on precise numerical evidence rather than coarse visual patterns, thereby establishing a more rigorous benchmark for fine-grained TSR.

E Training Details

We provide the training parameter settings and training data proportions of the three-stage SFT in the Table 12.

F Baseline Methods

F.1 Time Series Reasoning Models

Time-MQA (Kong et al., 2025). Time-MQA is an LLM-based TSRM that employs LoRA to fine-tune different backbone models on a large-scale TSQA dataset that contains traditional time series analysis and QA tasks.

Time-R1 (Luo et al., 2025). Time-R1 is an LLM-based TSRM, which utilizes RFT to transfer the LLM’s reasoning abilities to the forecasting task.

ChatTS (Xie et al., 2025b). ChatTS is a TS-MLLM built on Qwen2.5-14B-Instruct, which encodes time series via a 5-MLP encoder, training on large-scale synthetic datasets and generalizing to various TSR scenarios.

OpenTSLM (Langer et al., 2025). OpenTSLM is a domain-specific TS-MLLM built on that adopts two types of architectures. It employs a curriculum training paradigm, which first trains on TS-QA datasets and then transfers to fine-tune on medical-specific TSR tasks.

F.2 Instruction Templates

Our instruction templates for different types of TSRMs are listed as follows:

- text models: Figure 7.
- LLATISA and VLMs with numerical table image: Figure 8.
- vision models with textual series input: Figure 9.
- For existing TSRMs that are already trained or fine-tuned, we adapt their official prompt templates for the evaluation.

G Case Study

We provide valuable cases of how LLATISA responds to different TSR tasks in the OOD setting. Cases are summarized as follows:

- **L1. Min/Max Grounding.** Table 31.
- **L2. Local Pattern Differentiation.** Table 32.
- **L2. Global Pattern Differentiation.** Table 33.
- **L3. Semantic Understanding.**

H AI Assistants Usage

In this paper, the AI Assistants are only used for text polishing.

I Theoretical Underpinning of the Difficulty-Stratified Taxonomy

Our four-level decomposition of TSR is not arbitrary but is grounded in two well-established theoretical frameworks: Bloom’s Taxonomy (Bloom et al., 1956) (cognitive psychology) and Bertin’s Levels of Reading (Monmonier, 1985) (visual analytic).

- **Cognitive Reasoning (Bloom’s Taxonomy):** Our hierarchy mirrors the progression from Low-Order Thinking Skills (LOTS) to High-Order Thinking Skills (HOTS). Specifically: (i) L1 (Numerical Read-out) maps to Remembering/Retrieving (discrete fact extraction); (ii) L2 (Pattern Perception) maps to Understanding (identifying

structural relationships and trends); (iii) L3 (Semantic Reasoning) maps to Applying/Analyzing (contextualizing patterns within domain-specific knowledge); and (iv) L4 (Predictive Inference) maps to Creating (generating future states or logical relationships based on historical temporal data). Recent studies (Huber and Niklaus, 2025) have formally advocated for using Bloom’s Taxonomy to evaluate the reasoning capabilities of LLMs.

- **Visual Perception (Bertin’s Levels of Reading):** From a data visualization perspective, Jacques Bertin’s classic framework justifies our structural necessity: (i) Elementary Level (L1) focuses on a single data element; (ii) Intermediate/Global Level (L2) focuses on relationships and macroscopic trends across multiple elements; and (iii) Extrapolative Level (L3/L4) involves deducing hidden semantics or future trends beyond the given data.

By aligning with these frameworks, HiTSR provides a principled road-map to evaluate whether a TSRM’s failure stems from a perceptual bottleneck (L1-L2) or a reasoning deficit (L3-L4). This structured decomposition is necessary for diagnosing the precise cognitive boundaries of TSRMs.

J Position of Our Work

We emphasize that our paper intentionally focuses on TSR-Understanding (L1-L3) rather than TSR-Generation (L4). We argue that these are two fundamentally different paradigms.

- **Paradigm Distinction.** In the broader multi-modal field, understanding models (e.g., Qwen3-VL) and generative model (e.g., the diffusion model) are typically treated as distinct research directions. Integrating L4 (especially numerical forecasting) would necessitate a transition from a Reasoning Model to a Unified Understanding-and-Generation Model, which presents distinct technical challenges (e.g., specialized generative losses) and constitutes a separate research frontier.
- **Diagnostic Purity.** Our primary goal is to provide a diagnostic lens to pinpoint the cognitive boundaries of current TSRMs. By focusing on L1-L3, we ensure a rigorous evaluation of the model’s analytical capabilities without the confounding factors of generative precision.

Modality	Model	Acc%
Text (w/o index)	GPT-4.1	79.1
	GPT-4o	75.6
	Claude-3.5-Sonnet	82.2
	Qwen3-8B	67.1
	LLaMA3.1-8B	41.0
Vision (plot + num)	GPT-4o	78.3
	Qwen3-VL-8B	42.1
	LLATISA (L1→L2→L3)	54.2
	LLATISA (L1→L2→L3→L4)	83.3

Table 6: Performance comparison on **out-of-distribution datasets** in L4 (Generative Inference).

K Evaluation on L4 Task

We have conducted evaluations on the L4 (Predictive Inference) task and incorporated L4 into HiTSR. We then trained LLATISA on this subset and evaluated its out-of-distribution (OOD) performance on real-world time series data.

K.1 Task Introduction

Given a time series segment x , the model is tasked with a Multiple Choice (MC) question to identify the unique chronological successor from a pool of four candidate patches y_1, y_2, y_3, y_4 . While the correct successor follows x immediately in the original series, the negative candidates are sampled from distinct time series with diverse characteristics. This setup evaluates the model’s ability to extrapolate underlying temporal dynamics and distinguish the true evolution of a series from statistically distinct alternatives.

K.2 Dataset Construction

We extended our synthetic generation pipeline to ensure that L4 tasks are both challenging and unambiguous. To prevent “shared ground truth” (where multiple options could be plausible), we implemented a multi-stage filtering protocol for negative candidates:

- **Statistical Uniqueness.** We filter out candidates with high Pearson Correlation (≥ 0.8) or small Euclidean Distance to the ground truth to ensure distinctiveness.
- **Strategic Truncation.** To avoid introducing excessive ambiguity, we ensure the split point between x and y does not occur during high-variance events, such as mid-spike or at sharp turning points. Splitting at such critical junctures may cause multiple plausible answers; by selecting structurally stable partition points, we ensure

the ground truth is the only logically consistent successor.

- **Trend Smoothness.** We use a weighted continuation score, which incorporates gradient consistency and tail-to-head difference to verify that the positive sample is the most semantically and numerically plausible successor.
- **Mutual Exclusion.** Negative candidates are cross-checked to ensure low mutual correlation, preventing redundant options within a single question.

K.3 Transition to Vision Modalities

To align with the dual-view strategy of LLATISA, we transform the L4 predictive task into a dual-image input format. Specifically, the visualized patterns are rendered as a 2×2 grid of subplots, each representing one of the four options. Within each subplot, the lookback context and the candidate prediction patch are visualized in blue and red, respectively, allowing the model to perform visual comparative analysis of temporal continuity. Complementing this, a precision-calibrated numerical table provides exact values for the lookback window. This dual-view setup ensures the model can leverage global trend intuition from the plots while maintaining the fine-grained numerical grounding necessary for rigorous successor identification.

K.4 Experiment Results

Following this rigorous protocol, we expanded our curriculum to include a fourth stage (L1→L2→L3→L4). We trained LLATISA on 4,993 L4-specific questions for 2 epochs (learning rate 1×10^{-5}) and evaluated its OOD performance on the real-world TRQA (Jing et al., 2025) subset. As results shown in table 6, we summarize key observations as follows:

- **Surpassing Text-only TSRMs.** LLATISA achieves superior performance compared to both close-source and open-source TSRMs with textual time series input settings. These results further underscore the effectiveness of LLATISA’s dual-view multimodal paradigm, high-quality TSR dataset (HiTSR), and multi-stage curriculum fine-tuning.
- **Comparison with General VLMs.** LLATISA significantly outperforms its backbone model Qwen3-VL-8B. This discrepancy highlights the

Ablation	Acc%
only L1	16.9
only L2	23.0
only L3	43.0
L1+L2	21.4
L1+L2+L3	36.8
L1+L2+L3+L4	75.4
L1→L2	32.8
L1→L2→L3	54.2
LLATISA (L1→L2→L3→L4)	83.3

Table 7: Ablation study on the L4 task. All experiments are conducted on OOD datasets.

necessity of our dual-view representation and multi-stage training on the difficulty-stratified TSR datasets.

- **Efficacy of the L4 Curriculum Stage.** The transition from the L3-trained model (54.2%) to the L4-trained model (83.3%) reveals a substantial 29.1% improvement. This confirms that while foundational understanding (L1-L3) provides a necessary base, specific predictive inference training (L4) is essential for generalizing the TSR capability from understanding to the generative tasks.
- **Robustness in OOD Scenarios.** Since the evaluation was conducted on real-world data from TRQA (which differs significantly from our synthetic training distributions), the high accuracy (83.3%) serves as strong evidence of LLATISA’s cross-domain generalization. The model has learned the underlying principles of temporal continuity rather than memorizing specific patterns.

Overall, these results provide a definitive answer to the reviewers’ concerns: LLATISA is not only a diagnostic tool for understanding but also has the potential to generalize to L4 predictive tasks.

K.5 Ablation Study

To further investigate the synergy between our difficulty-stratified taxonomy and the curriculum training regime, we conducted additional ablation studies on the L4 task. We compared LLATISA against single-stage training and joint multi-task

learning to verify if foundational levels (L1–L3) are prerequisites for predictive inference (L4). Here, “+” refers to joint training on shuffled datasets from multiple levels (e.g., “L1+L2”), while “→” denotes a sequential curriculum regime (e.g., “L1→L2”). Evaluation results (table 7) underscore the critical importance of our sequential curriculum: single-stage training fails on L4 tasks without the foundational grounding provided by L1–L3. Notably, **LLaTiSA** outperforms joint training (“L1+L2+L3+L4”) by 7.9%, confirming that a step-by-step mastery of temporal primitives and semantics effectively improves the performance of complex predictive inference. This validates that our difficulty-stratified taxonomy acts as a necessary cognitive scaffolding, ensuring robust generalization in complex OOD scenarios.

L Expansion of Experiments

L.1 Ablation Study

To justify the necessity of the multi-stage SFT, we provide intermediate evaluations of **LLaTiSA** across different training stages on both ID and OOD benchmarks (Table below). Here, “L1 + L2” refers to joint training on shuffled datasets from L1-L2, while “L1 → L2” denotes a two-stage curriculum fine-tuning regime. The evaluation results (table 10) reveal three critical insights that justify the necessity of our three-stage curriculum:

- As shown in the “Only L2” row, although the model masters pattern perception, its L1 performance significantly collapses. This proves that high-level TSR does not naturally “backwards inherit” foundational perception. The L1 stage is essential to “anchor” the TSRM’s grounding capability.
- While training on L2 alone provides some reasoning capability, the performance on L3 improves when L1 and L2 are combined (from 66.0% to 70.0% ID). The full curriculum (**LLaTiSA**) achieves the highest L3 score (87.0% ID, 67.0% OOD).
- **LLaTiSA** consistently outperforms “L1 → L2” and “L1 + L2” strategies, indicating the effectiveness of the training on the L3 part and multi-stage curriculum regime, respectively.

In conclusion, the curriculum regime is necessary for **LLaTiSA** because it ensures to maintain a balanced training across different cognitive levels,

which a single-stage training on high-level data fails to provide.

L.2 Baseline Comparison

We further evaluate ChatTime (Wang et al., 2025a) on the datasets we used across table 15 and explain the differences between TSRMs and Time-LLM (Jin et al., 2023). We clarify our evaluation results (table 8) and the architectural distinctions below:

- **Evaluation on ChatTime.** We evaluated ChatTime on our OOD benchmarks. As shown in table 8, ChatTime performs poorly across all levels. Notably, it fails to adhere to instruction-defined output formats (e.g., the “index-value” requirement for L1 tasks). Furthermore, as ChatTime only accepts a single series, we had to manually concatenate multiple series and specify segment lengths to enable it to finish the “Series Comparison” (L3) task.
- **Distinction with Time-LLM.** **LLaTiSA** significantly outperforms its backbone model Qwen3-VL-8B. This discrepancy highlights the necessity of our dual-view representation and multi-stage training on the difficulty-stratified TSR datasets.

L.3 Expansion of VLM Baselines

To provide a more comprehensive evaluation, we have extended our benchmarks to include InternVL3-8B (Zhu et al., 2025) and Gemma3-12B (Team et al., 2025) across various visual input modalities. Following the same experimental settings introduced in Section 4.1, additional results are shown in table 9.

Empirical results demonstrate that **LLaTiSA** consistently and significantly outperforms various VLM-based baselines of a comparable parameter scale, regardless of the input modalities or representation strategies employed to numerical series. Notably, these newly incorporated VLM-based models exhibit a similar performance scale to the Qwen3-VL baselines reported in table 15, which all fall significantly short of **LLaTiSA** in both L1 numerical grounding and L3 semantic reasoning tasks. This further validates that our performance gains stem from both the specialized dual-view architecture and three-stage curriculum training paradigm.

Model	Modality	L1 Numerical Read-out		L2 Pattern Perception		L3 Semantic Reasoning
		Min & Max Localization		Local	Global	Series Comparison
		Acc%	half-Acc%	Acc%	Acc%	Acc%
ChatTime	Time Series	0.0	0.0	9.8	14.2	6.0
LLATISA	Vision(plot+num)	86.8	96.0	75.6	97.5	67.0

Table 8: Performance comparison on **out-of-distribution datasets** across L1-L3 with ChatTime.

Model	Modality	L1 Numerical Read-out		L2 Pattern Perception		L3 Semantic Reasoning
		Min & Max Localization		Local	Global	Series Comparison
		Acc%	half-Acc%	Acc%	Acc%	Acc%
InternVL3-8B	Vision(plot)	-	-	49.0	82.0	46.0
	Vision + Text(w/o index)	16.8	59.2	41.8	85.8	56.0
	Vision + Text(w/ index)	36.0	72.4	44.0	87.5	55.0
	Vision (plot+num)	6.2	36.8	27.6	88.3	58.0
Gemma3-12B	Vision(plot)	-	-	64.4	91.7	51.0
	Vision + Text(w/o index)	24.0	73.2	59.8	90.0	49.0
	Vision + Text(w/ index)	51.4	81.0	61.2	92.5	53.0
	Vision (plot+num)	7.8	43.4	55.3	95.8	41.0
LLATISA	Vision(plot+num)	86.8	96.0	75.6	97.5	67.0

Table 9: Performance comparison on **out-of-distribution datasets** across L1-L3 with superior VLMs. The “-” denotes that the single plot input cannot handle L1 tasks.

Type	Ablation	L1	L2	L3
ID	only L1	98.2	87.5	47.0
	only L2	14.2	96.5	66.0
	L1 + L2	98.0	96.0	70.0
	L1 → L2	97.8	96.5	72.0
	LLATISA (L1 → L2 → L3)	98.0	97.0	87.0
OOD	only L1	97.2	87.5	47.0
	only L2	10.0	96.5	52.0
	L1 + L2	96.0	96.0	53.0
	L1 → L2	96.0	97.5	56.0
	LLATISA (L1 → L2 → L3)	96.0	97.5	67.0

Table 10: Extended ablation study results on the curriculum learning strategy.

M Data Quality Assurance

M.1 Human Verification

To ensure the data quality, we employed a hybrid verification pipeline combining deterministic scripts and human expert audits:

- **L1.Numerical Read-out:** 3,000/30,000 (10%). Given the objective nature of read-out tasks, we primarily utilized deterministic scripts for data

Task	GPT-5	Qwen3	Cohen’s Kappa
L2 (Local)	16,121 / 20,000	16,149 / 20,000	0.865
L2 (Global)	24,654 / 25,000	24,572 / 25,000	0.847
L3 (Semantic)	3,321 / 4,000	3,378 / 4,000	0.831

Table 11: Agreement results and Cohen’s Kappa for different tasks.

verification.

- **L2.Pattern Perception:** 5,100/50,703 (10%). This subset includes 1,600 Local, 2,500 Global verified by human experts, and 1,000 Numerical Perception samples verified by deterministic scripts.
- **L3.Semantic Reasoning:** 3,121/3,121 (100%). Due to the high cognitive complexity of L3, every single sample underwent manual human verification to ensure reasoning integrity.

Our experts followed a three-fold protocol: (1) Numerical Faithfulness to the source series, (2) Logical Consistency of the CoT trajectories, and (3) Ground-truth Correctness. To evaluate the reliability of our protocol, two independent annota-

Training Stage	Hyperparameter	Assignment
Shared	Base model	Qwen3-VL-8B-Instruct
	Computing environment	8*H20 GPUs
	Optimizer	AdamW
	Learning rate scheduler	Cosine
	Weight decay	0.1
Stage-1	Epochs	1
	Warm-up steps	50
	Batch size per device	2
	Gradient accumulation steps	2
	Learning rate	1e-5
	#Samples	2,7000
Stage-2	Epochs	2
	Warm-up steps	50
	Batch size per device	2
	Gradient accumulation steps	2
	Learning rate	1e-5
	#Samples	4,5000
Stage-3	Epochs	2
	Warm-up steps	50
	Batch size per device	1
	Gradient accumulation steps	1
	Learning rate	1e-5
	#Samples	2,700
Stage-ECG	Epochs	2
	Warm-up steps	50
	Batch size per device	1
	Gradient accumulation steps	1
	Learning rate	1e-5
	#Samples	3,0000

Table 12: Hyperparameter settings for each training stage. “#Samples” denotes the number of training items within the corresponding subsets of the HiTSR dataset.

tors re-verified a random subset of 500 samples from L2 and L3. **We achieved a Cohen’s Kappa of 0.853, signifying “almost perfect agreement.”** This score demonstrates that our verification criteria are objective and consistent across different human experts.

M.2 Cross-Model Validation

To ensure the annotations are objective and free from model-specific biases, we conducted a cross-model validation using Qwen3-235B-A22B-Instruct (Qwen3) on the subset of HiTSR which contains verifiable CoT trajectories. The following table reports the successfully validated instance counts (number of passed samples relative to the total evaluated) and the inter-annotator agreement scores (Cohen’s Kappa calculated between GPT-5 and Qwen3). **All subsets across L2-L3 achieve “almost perfect agreement.” with an average score of 0.848.** Combined with the full human verification on L3, such cross-model evidence demonstrates that advanced language models converge on a consistent standard of validity, effectively serving as a robust, model-agnostic filter to maximize the efficiency and reliability of subsequent human verification.

Data	Meta Information	Crop Length	Source
ETTh (1/2)	Hourly measured time series of the transformer oil temperature in the Electricity Transformer Temperature dataset.	[96, 192, 336, 720]	(Zhou et al., 2021)
ETTm (1/2)	15-minute measured time series of the transformer oil temperature in the Electricity Transformer Temperature dataset.	[96, 192, 336, 720]	(Zhou et al., 2021)
Weather	10-minute measured time series of a weather-related target variable, commonly treated as air temperature, from a weather station dataset.	[96, 192, 336, 720]	(Wetterstation, 2020)
Exchange Rate	1-daily measured time series of the Singapore exchange rate.	[60, 120, 240, 360]	(Lai et al., 2018)
Traffic.A	Hourly measured time series of the road occupancy rate recorded by San Francisco freeway sensors from 2015 to 2016.	[96, 192, 336, 720]	(Wu et al., 2021)
Electricity	Hourly measured time series of the electricity consumption of one customer selected from 321 customers collected from 2012 to 2014.	[96, 192, 336, 720]	(Wu et al., 2021)
Illness	Weekly measured time series covering from 2002 to 2021 of patient counts reported in influenza-related surveillance statistics.	[24, 36, 52, 104]	(illness)
Economy	Monthly measured time series that represents the international trade balance.	[96, 120, 180, 447]	(Liu et al., 2024a)
Energy	1-weekly measured time series that represents gasoline prices.	[full, 120, 240, 480]	
Environment	1-daily measured time series that represents the air quality index.	[full, 240, 480, 576]	
Health US	1-weekly measured time series that represents influenza patients.	[full, 120, 240, 480]	
Security	Quantifying pattern attributes (e.g., amplitude, slope).	[full, 96, 120, 240]	
SocialGood	A monthly measured time series that represents the unemployment rate.	[96, 120, 180, 360, 512]	
Traffic.B	A monthly measured time series that represents travel volume.	[96, 120, 180, 360, 480]	
Electricity Demand	This is a time series representing electricity demand in Victoria, Australia, for 2014, measured at 30-minute intervals.	[96, 192, 336, 720]	(Liu et al., 2024c) (Godahewa et al., 2021)
Australia Rainfall	This is an hourly measured time series of the temperature change in some part of Australia.	[96, 192, 336, 720]	
London Smart Meters	This is a half-hourly measured time series that represents of the energy consumption readings of London households in kWh.	[96, 192, 336, 720]	
Wind Farms	This is a time series representing the wind power production of an Australian wind farm, measured every 4 seconds.	[128, 256, 512, 1024]	
Benzene Concentration	This is an hourly measured time series of the benzene concentration.	[96, 192, 336, 720]	
Atrial Fibrillation	This is a time series representing atrial fibrillation, measured every 0.008 seconds.	[256, 512, 720]	
IEEPPG	This is a time series of heart rate estimation measured every 0.008 seconds.	[128, 256, 512, 720]	
SDWPF	This is a time series representing the active power output (Patv, kW) of a wind turbine in a single wind farm in China, measured every 10 minutes.	[60, 120, 240, 360]	
Wind Speed	This is a time series representing the nacelle-measured wind speed (Wspd, m/s) for a wind turbine in a single wind farm in China, measured every 10 minutes.	[60, 120, 240, 360]	
Saugeen Day	This is a time series representing the mean daily flow of the Saugeen River at Walkerton in cumecs (m ³ /sec), measured daily.	[60, 180, 365, 730]	
Starlight	This is a synthetic time series that simulates the brightness of a celestial object in the night sky, measured daily.	[60, 180, 365, 730]	
KDD Cup 2018	This is a time series representing the PM2.5 concentration in Beijing from 2017 to 2018, measured hourly.	[72, 120, 240, 360]	
Pedestrian Counts	This is a time series representing the pedestrian count in Melbourne city, measured hourly, starting in May 2009.	[72, 120, 240, 360]	
Kaggle Web Traffic	This is a time series representing the daily views of a random Wikipedia article, measured daily.	[60, 120, 180, 365]	

Table 13: Summary of real-world time series data utilized in HiTSR. “Crop Length” denotes the random cropping length of the subseries utilized to create an MCQ. “full” denotes using the full time series. The “Meta Information” is utilized for creating L3 QA pairs.

Subtask	Capability Target	Data	#Samples	Length Range
<i>L1: Numerical Read-out</i>				
Min/Max Grounding	Index-aware value grounding and verification	Synthetic	10,000	[64, 512]
Multi-series Comparison	Cross-series grounding under aligned time/value queries	Synthetic	10,000	[64, 256]
Start/End Comparison	Boundary grounding (initial vs. terminal states)	Synthetic	5,000	[64, 512]
Subseries Localization	Interval-based retrieval with temporal alignment	Synthetic	5,000	[64, 512]
<i>L2: Pattern Perception</i>				
Local Pattern Differentiation	Multi-scale morphology and trend alignment	Synthetic	16,098	[64, 256]
Global Pattern Differentiation	Multi-scale morphology and trend alignment	Synthetic	24,605	[64, 1024]
Numerical Perception	Quantifying pattern attributes (e.g., amplitude, slope)	Synthetic	10,000	[64, 256]
<i>L3: Semantic Reasoning</i>				
Semantic Understanding	Contextual interpretation and decision-making over series evidence	Real-world	3,121	[24, 2048]

Table 14: Task overview of HiTSR (L1-L3). Each level groups multiple subtasks.

Modality	Model	L1. Min/Max Grounding		L2. Local Pattern Differentiation	L2. Global Pattern Differentiation	L2. Numerical Perception	L3. Semantic Understanding
		Acc%	half-Acc%	Acc%	Acc%	Acc%	Acc%
Text (w/o index)	GPT-4o	6.0	23.0	53.0	94.0	<u>73.0</u>	67.0
	Qwen3-8B	33.0	67.0	59.0	66.0	36.0	54.0
Text (w/ index)	GPT-4o	23.0	66.0	55.0	<u>98.0</u>	62.0	62.0
	Qwen3-8B	60.0	80.0	55.0	89.0	64.0	66.0
Vision	GPT-4o	-	-	55.0	93.0	58.0	61.0
	Qwen3-VL-8B	-	-	65.0	88.0	47.0	35.0
Time Series	ChatTS	<u>37.0</u>	54.0	67.0	88.0	63.0	<u>70.0</u>
Vision+Text (w/o index)	GPT-4o	8.0	33.0	69.0	97.0	68.0	71.0
	Qwen3-VL-8B	5.0	29.0	<u>68.0</u>	87.0	51.0	48.0
Vision+Text (w/ index)	GPT-4o	29.0	<u>77.0</u>	55.0	99.0	61.0	54.0
	Qwen3-VL-8B	19.0	70.0	69.0	89.0	78.0	46.0

Table 15: Evaluation results on key tasks in the HiTSR.

Type	Dataset	Data	Modalities	CoT	Train-Test	Level	Open source
General	TSQA (Kong et al., 2025)	Real	Text	✗	✓	L2-L4	✓
	TimeMCQ2 (Merrill et al., 2024)	Syn.	Text	✗	✗	L3	✓
	TimeSeriesExam (Cai et al., 2024)	Syn.	Text	✗	✗	L2	✓
	ChatTime (Wang et al., 2025a)	Syn.&Real	Text	✗	✓	L2, L4	✓
	ChatTS (Xie et al., 2025b)	Syn.	TS	✗	✓	L2-L3	✓
	BEDTime (Sen et al., 2025)	Syn.&Real	Text&Vision	✗	✗	L2	✓
	MMTS-Bench (Anonymous, 2025)	Syn.&Real	Text	✗	✗	L2-L4	✓
	CaTS-Bench (Zhou et al., 2025)	Real	Text&Vision	✗	✓	L2	✓
	TRQA (Jing et al., 2025)	Real	Text	✗	✓	L3	✓
	TimeSense (Zhang et al., 2025b)	Syn.	TS	✗	✓	L2-L3	✗
	TSR-Suite (Guan et al., 2025)	Real	Text	✓	✓	L3-L4	✗
	TimeMMD (Liu et al., 2024a)	Syn.&Real	TS	✗	✓	L4	✓
	Chat-TS (Quinlan et al., 2025)	Real	TS	✗	✓	L2-L3	✗
	TSAIA (Ye et al., 2025)	Syn.&Real	Text	✗	✓	L3-L4	✓
CiK (Williams et al., 2024)	Syn.&Real	Text	✗	✗	L4	✓	
MTBench (Chen et al., 2025)	Real	Text	✗	✗	L3-L4	✓	
Domain & Task-Specific	RAT40k (Yang et al., 2025b)	Real	Text&Vision	✓	✓	L2-L3	✓
	ITFormer (Wang et al., 2025b)	Syn.	TS	✗	✓	L3	✓
	ECG-Grounding (Lan et al., 2025)	Real	TS&Vision	✓	✓	L3	✓
	SciTS (Wu et al., 2025)	Real	TS	✗	✓	L3-L4	✗
General	HiTSR	Syn.&Real	Text&Vision	✓	✓	L1-L3	✓

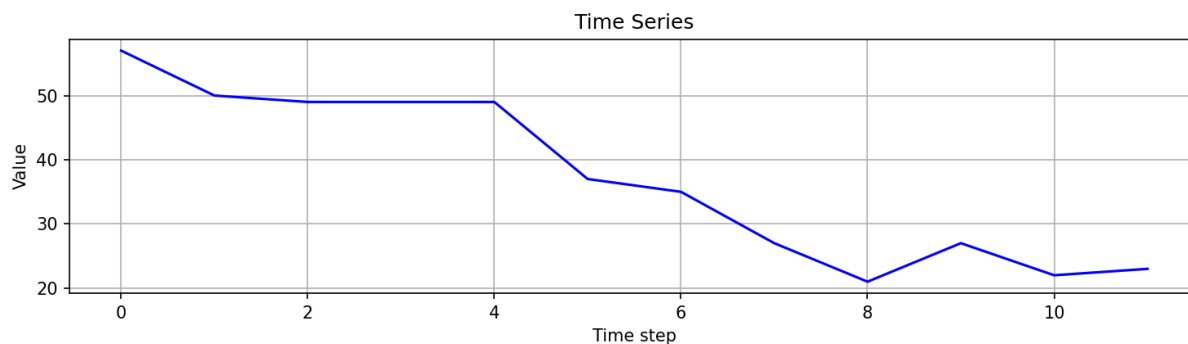
Table 16: Dataset properties. “Syn.” denotes the synthetic time series data, and “Train-Test” represents whether the corresponding dataset provides train-test splits instead of as an evaluation benchmark.

Type	Model	Diagnosis Accuracy	Analysis Completeness	Analysis Relevance	Lead Assessment Coverage	Lead Assessment Accuracy	ECG Feature Grounding	Evidence Based Reasoning	Clinical Diagnostic Fidelity
ID	PULSE	81.14	2.37	2.39	7.11	2.95	50.18	52.40	51.63
	GEM (LLaVA)	87.24	4.41	<u>5.01</u>	<u>71.07</u>	<u>46.44</u>	75.48	75.09	75.28
	GEM (PULSE)	<u>86.49</u>	<u>4.43</u>	4.91	69.80	45.33	<u>74.95</u>	<u>74.70</u>	<u>74.87</u>
	Qwen3-VL-8B	60.85	4.01	3.97	69.28	50.10	69.89	63.82	69.10
	LLATiSA	62.77	5.64	7.14	84.03	53.03	71.41	71.20	70.10
OOD	PULSE	59.24	2.20	2.06	11.20	6.27	52.52	55.48	53.85
	GEM (LLaVA)	<u>73.53</u>	4.19	<u>2.96</u>	79.54	<u>49.01</u>	<u>74.48</u>	<u>74.61</u>	<u>73.84</u>
	GEM (PULSE)	73.59	4.19	3.00	<u>78.86</u>	47.96	74.97	75.41	74.24
	Qwen3-VL-8B	59.01	3.22	1.82	56.41	38.12	46.43	63.81	49.89
	LLATiSA	62.22	<u>3.62</u>	2.48	66.50	49.20	66.19	66.61	64.91

Table 17: Full results on the L3 ECG-Grounding task. The best result is **bolded** and the second best is in underlined.

Model-Case1

Question: Carefully analyze the provided time series plot in the image and choose the single best option that most accurately describes the pattern depicted. Follow these rules strictly: (1) Look at the image before deciding. (2) Only output the chosen option, highlighted as A, B, C, or D; (3) Avoid adding extra text or explanations.



Options:

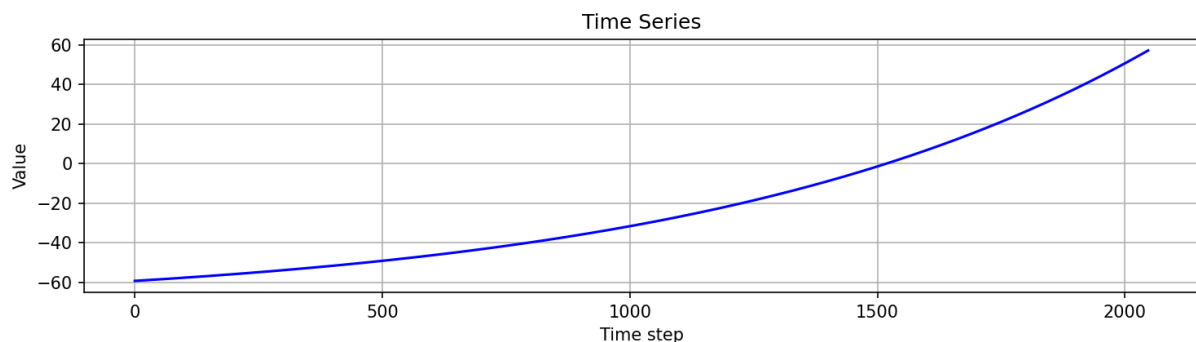
- A. Peaks towards the end.
- B. Ends at same value as beginning.**
- C. Highest point at the end.
- D. Decreases smoothly in the beginning.**

Answer: B

Table 18: Bad cases generated by GPT-4o. Options in red denote the correct answer, and the blue one is the wrong option chosen by the model.

Model-Case2

Question: Carefully analyze the provided time series plot in the image and choose the single best option that most accurately describes the pattern depicted. Follow these rules strictly: (1) Look at the image before deciding. (2) Only output the chosen option, highlighted as A, B, C, or D; (3) Avoid adding extra text or explanations.



Options:

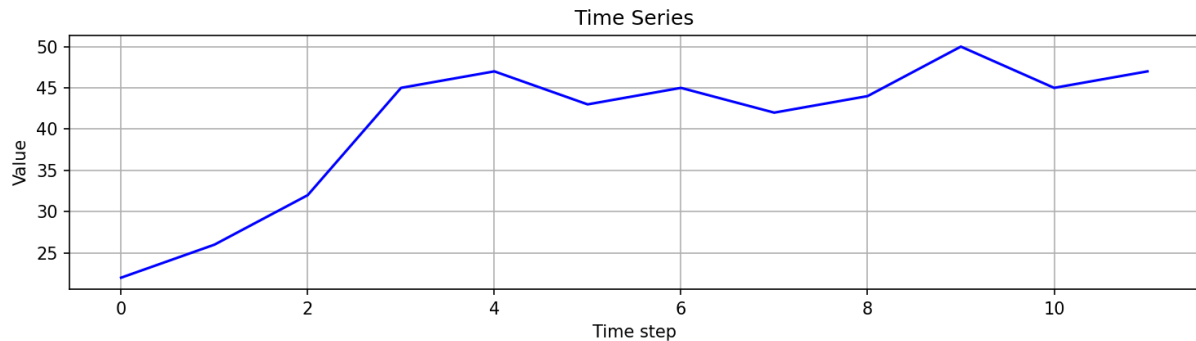
- A. A periodic pattern that is consistently repeated at regular intervals, similar to a square wave, is presented by the signal. Significant step changes punctuate the signal.
- B. The course is marked by exponential amplification, resulting in a sharp spike at the end.**
- C. Like an inverted sigmoid curve that forms a mirrored S-shape, the trend shows a monotonically decreasing pattern, with its rate of decrease gradually slowing down, eventually converging to a certain value. For the most part, the signal is smooth with slight disturbances.
- D. The trend is straight in its growth. Usually, the signal is smooth with minor spikes.

Answer: B

Table 19: Bad cases generated by GPT-4o. Options in red denote the correct answer, and the blue one is the wrong option chosen by the model.

Data-Case1: Multiple Correct Options

Question: Carefully analyze the provided time series plot in the image and choose the single best option that most accurately describes the pattern depicted. Follow these rules strictly: (1) Look at the image before deciding. (2) Only output the chosen option, highlighted as A, B, C, or D; (3) Avoid adding extra text or explanations.



Options:

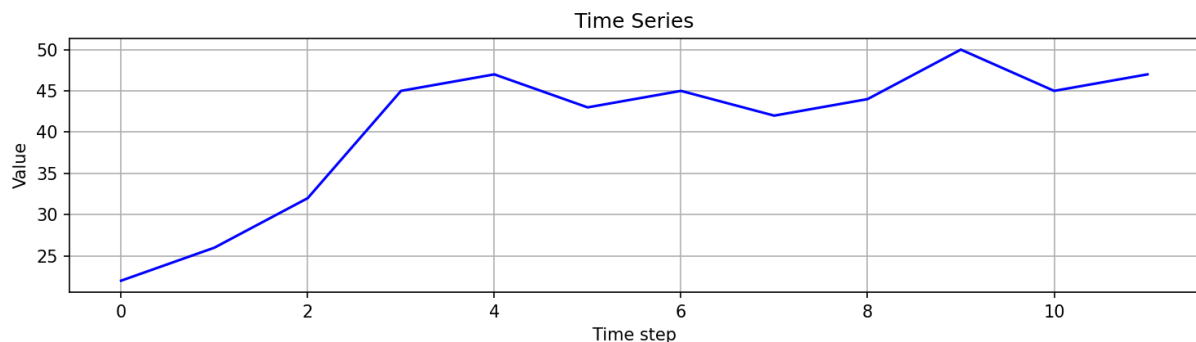
- A. endpoint is higher than the beginning.
- B. mostly flat in the middle.
- C. maximum value at the end.
- D. big climb towards the end.

Answer: B

Table 20: Ambiguous questions observed from existing datasets. Options in red denote the correct answer.

Data-Case2: The unclear definition of Location Terms

Question: Carefully analyze the provided time series plot in the image and choose the single best option that most accurately describes the pattern depicted. Follow these rules strictly: (1) Look at the image before deciding. (2) Only output the chosen option, highlighted as A, B, C, or D; (3) Avoid adding extra text or explanations.



Options:

- A. maximum value at the end.
- B. dips at the end.
- C. sharp decline near the end.
- D. ends lower than it started.

Answer: C

Table 21: A question sample that lacks a clear definition of location terms. Options in red denote are possibly correct, depending on the definition of “end”. The option marked in blue is the wrong option chosen by GPT-4o, which is the “guessing the boundary of volatility” problem indicated by (Guan et al., 2025).

Data-Case3: Lack of Information

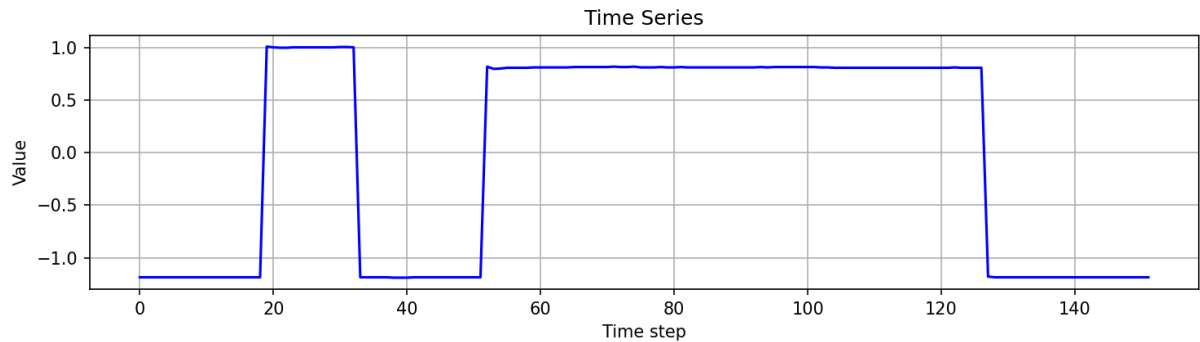
Question: Classify the given time series into one of the categories below.

Respond ONLY with the letter of the correct choice (A, B).

Choices:

A: normal process

B: abnormal process



Answer: B

Table 22: A question sample that lacks sufficient information, i.e., does not introduce the definition of “normal process”.

L1- Numerical Read-out: Min/Max Grounding

Question: Given the time series: <ts>, find the maximum and minimum values in the time series and report their first occurrence indices (if max/min repeats, use the earliest index). Provide the exact values and state which appears first. The first index is 0.

You MUST exactly follow the output format as:

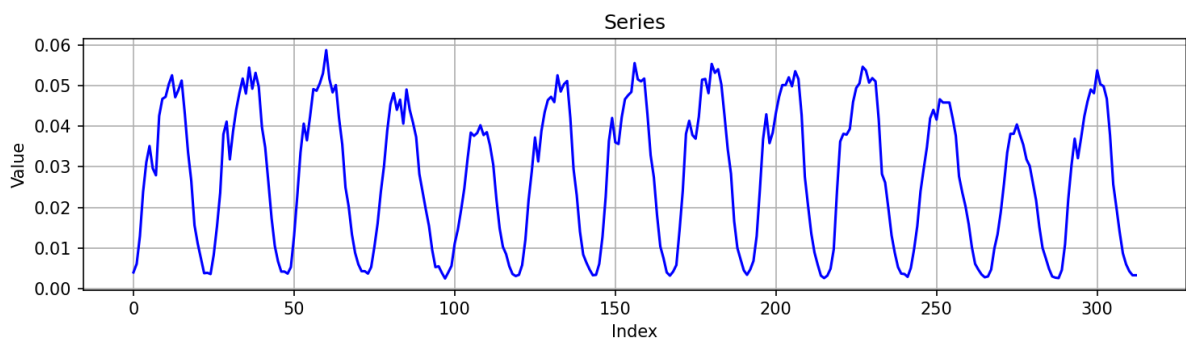
<max>[index, value]</max>

<min>[index, value]</min>

The max value appears first.

OR

The min value appears first.



Answer:

<max>[60, 0.0587]</max>

<min>[97, 0.0025]</min>

The max value appears first.

Table 23: Illustration of “Min/Max Grounding” task. “<ts>” denotes various input types of time series, e.g., numerical tables, textual arrays, or specific time series tokens.

L1- Numerical Read-out: Start/End Comparison

Question: Given the time series: <ts>, compare the value at the start of the time series (index 0) and the value at the end (last index). Report both [index, value] pairs and then state the comparison result.

Output format (MUST follow exactly):

<start>[index, value]</start>

<end>[index, value]</end>

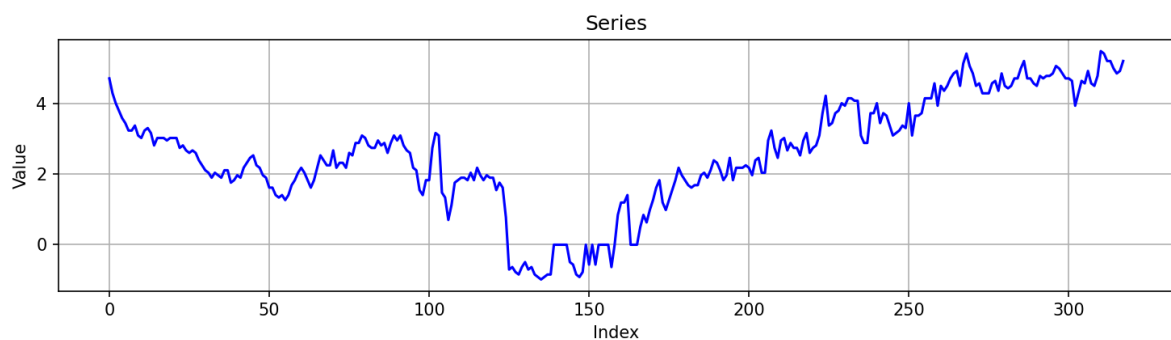
The start value is larger than the value at the end.

OR

The start value is smaller than the value at the end.

OR

The start value is equal to the value at the end.



Answer:

<start>[0, 4.713]</start>

<end>[317, 5.206]</end>

The start value is smaller than the value at the end.

Table 24: Illustration of “Start/End Comparison” task. “<ts>” denotes various input types of time series, e.g., numerical tables, textual arrays, or specific time series tokens.

L1- Numerical Read-out: Multi-series Comparison

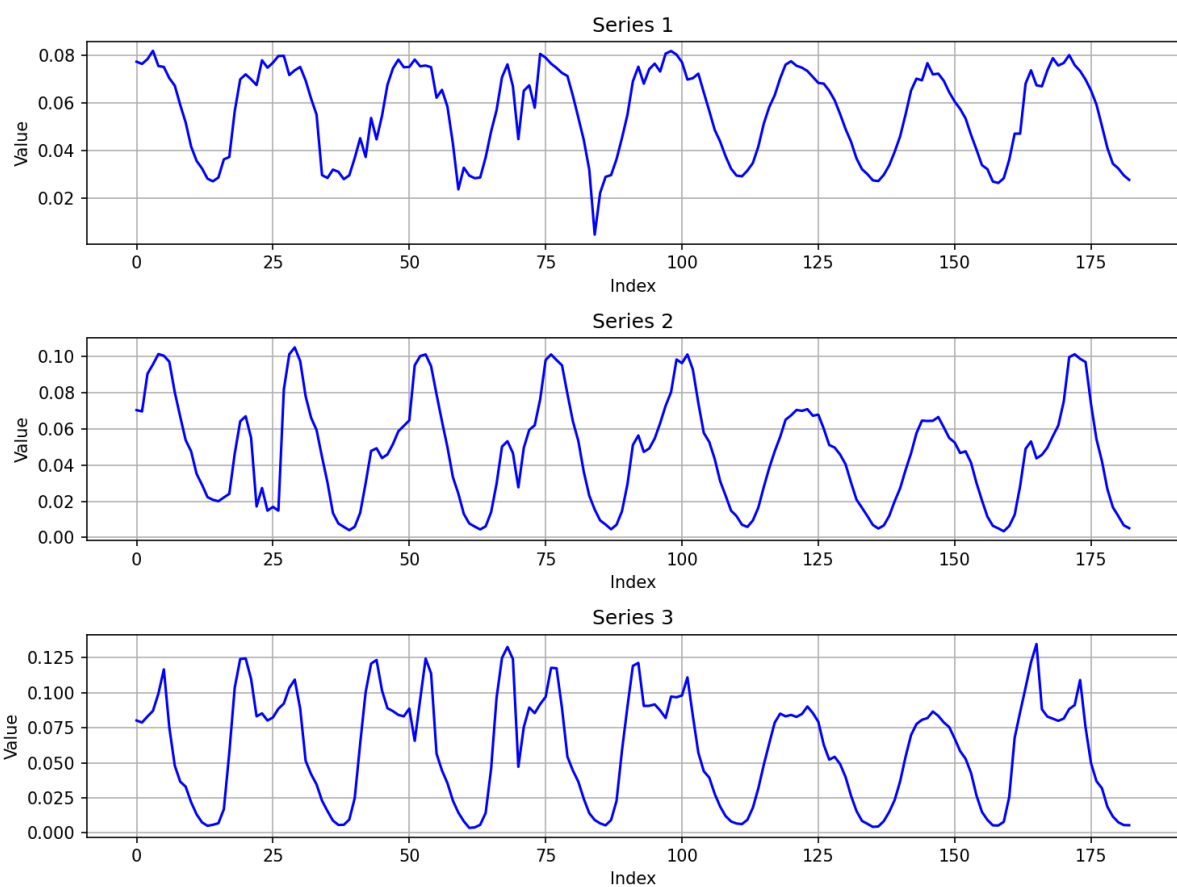
Question: Given the time series: <ts>, answer the question based on the provided multiple time series. You MUST exactly follow the output format as:

<answer>[index, value]</answer>

<series>X</series>,

where X refers to the index of the time series.

Question: Compare the minimum values of the multiple given time series. Which series has the lowest minimum value, and what is that specific value?



Answer:

<answer>[159, 0.0034]</answer>

<series>2</series>

Time Series 2 has the lowest minimum value among all the series. Its exact minimum value is 0.0034.

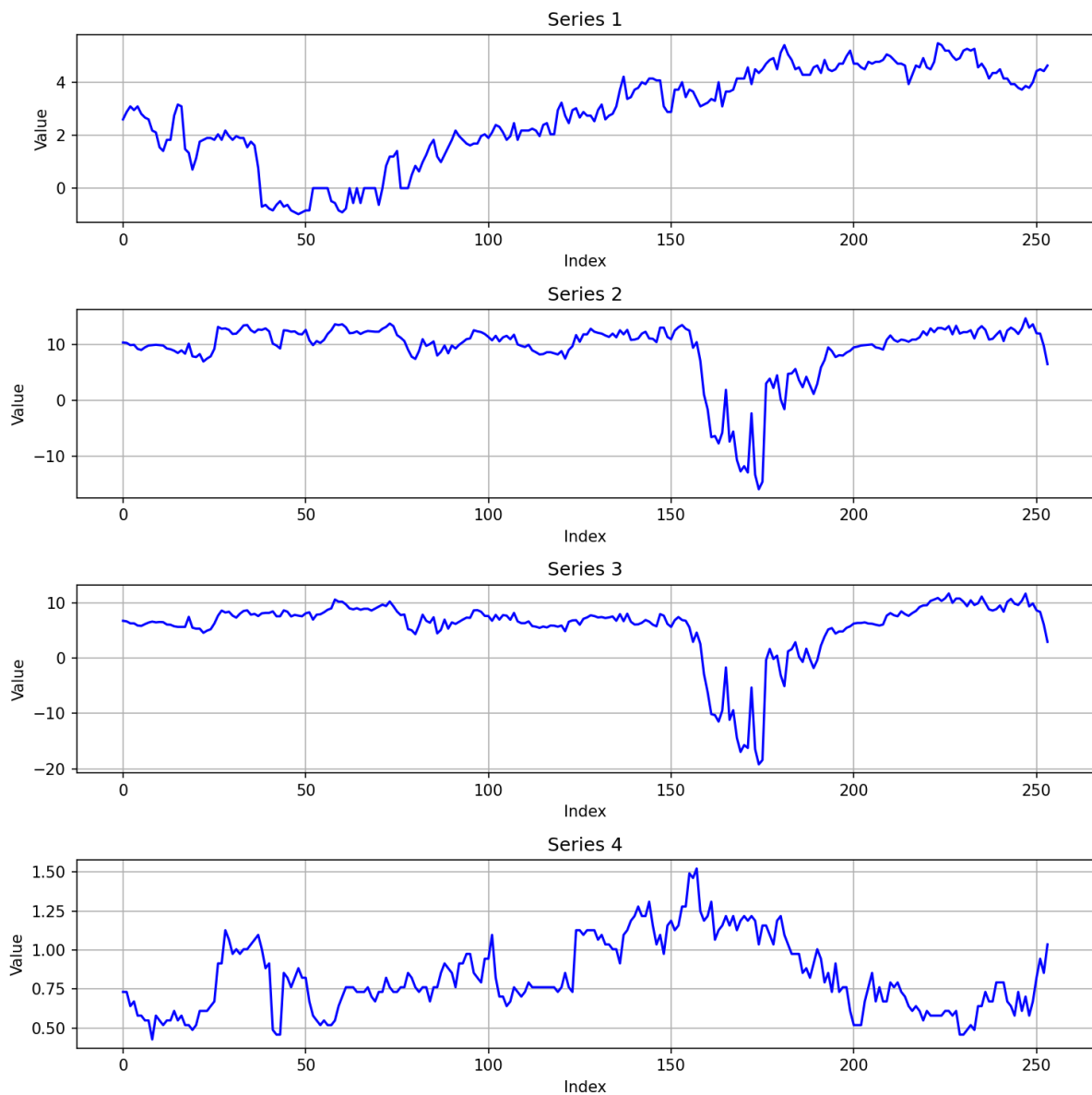
Table 25: Illustration of “Multi-series Comparison” task. “<ts>” denotes various input types of time series that represent numerical information, e.g., numerical tables, textual arrays, or specific time series tokens.

L1- Numerical Read-out: Subseries Localization

Question: Given the time series: <ts>, answer the question based on the provided multiple time series. You MUST exactly follow the output format as:

<answer>[v1, v2, ..., vk]</answer>

Recover the values of series 4 from index 201 to index 217 (inclusive). Return the values as a list in the exact order.



Answer:

<answer>[0.518, 0.518, 0.67, 0.761, 0.853, 0.67, 0.761, 0.67, 0.67, 0.792, 0.761, 0.792, 0.731, 0.701, 0.64, 0.609, 0.64]</answer>

Table 26: Illustration of “Subseries Localization” task. “<ts>” denotes various input types of time series that represent numerical information, e.g., numerical tables, textual arrays, or specific time series tokens.

L2-Pattern Perception: Local Pattern Differentiation

Question: Given the time series: <ts>, carefully analyze each option, then choose the single best option that most accurately describes the pattern.

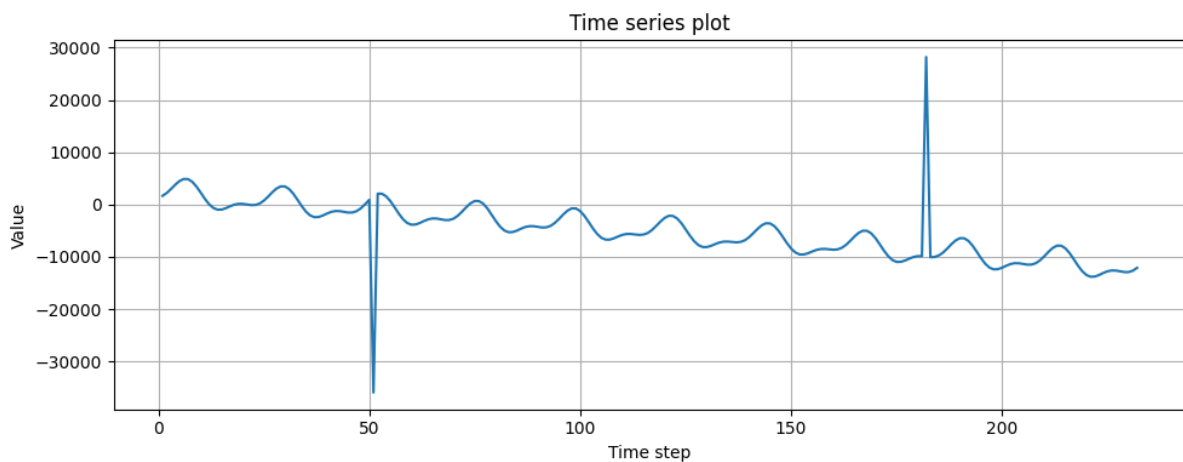
Options:

A: The approximate minimum value of the series is about -24059.

B: The time series exhibits a square-wave periodic fluctuation, with an approximate minimum period length of 23.1.

C: The value at the end of the series is higher than at the start.

D: The time series contains 0 overall trend turning points. Turning points refer to changes in the overall trend direction.



Answer: Step-by-step analysis of each option:

A: The series has a deep negative spike around time ~ 50 that reaches close to -34,000 (near the -35,000 axis tick). -24,059 is far above this minimum, so this is inaccurate.

B: The plot is not a square wave; it looks more like a smooth oscillation with noise and two sharp outliers. Claiming a square-wave periodic fluctuation and a specific period (23.1) does not match the visual pattern, so this is inaccurate.

C: The series starts near a positive value (a few thousand) and ends well below zero (around -10,000 to -12,000). The end value is lower than the start, so this is false.

D: The overall trend drifts downward from start to end without a sustained reversal in direction; the large spikes are brief anomalies, not changes in long-term trend. Thus, the number of overall trend turning points is effectively 0.

The correct answer is D.

Table 27: Illustration of “Local Pattern Differentiation” task in L2. “<ts>” denotes various input types of time series that represent numerical information, e.g., numerical tables, textual arrays, or specific time series tokens.

L2-Pattern Perception: Global Pattern Differentiation

Question: Given the time series: <ts>, carefully analyze each option, then choose the single best option that most accurately describes the pattern.

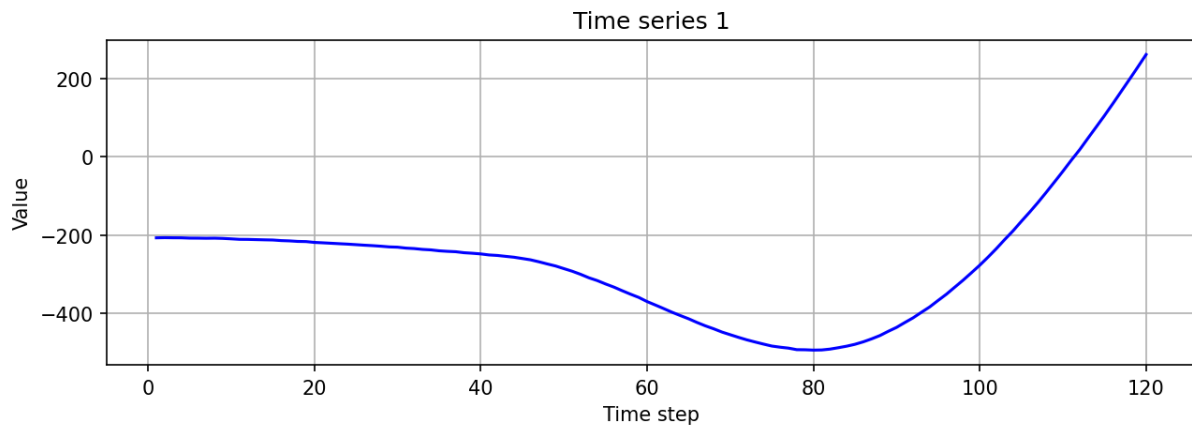
Options:

A. This time series steadily decreases from about 91 at index 0 (its maximum) to about 13 at index 162 (its minimum), with a smooth low-volatility path and no apparent turning points or seasonality.

B. This time series gently declines from index 0 around -200 to its minimum near -500 at index 80, then steadily climbs with moderate volatility to finish at its maximum around 260 at index 120, showing no apparent seasonality.

C. Starting near 200, the series spikes to around 2000 at index 11, then plunges below -1200 by around index 17 and continues as a steady, low-volatility sine-like oscillation with a roughly 26-point period around -1100 , reaching a valley near -1380 at index 194 and ending close to -1000 without significant turning points.

D. This time series gently declines from index 390 around -42 to its minimum near -979 at index 615, then steadily climbs with moderate volatility to finish at its maximum around 554 at index 634, showing no apparent seasonality or major turning points.



Answer: Step-by-step analysis:

A. Describes a steady decrease from about 91 at index 0 to about 13 at index 162, with no turning points. The provided plot starts around -200 , decreases to roughly -500 by ~ 80 , then rises to about 260 by 120. It is not monotonic and ends positively, so A does not match.

B. Describes a gentle decline from around -200 at index 0 to a minimum near -500 at index 80, then a steady climb with moderate volatility to finish at a maximum around 260 at index 120, with no seasonality. This matches the plot very well: start near -200 , bottom around -500 at ~ 80 , and end around 260 at 120, with smooth behavior and no seasonality.

C. Describes a series starting near 200, spiking to ~ 2000 at index 11, plunging below -1200 by ~ 17 , then continuing as a sine-like oscillation around -1100 , with a valley near index 194 and ending near -1000 . The plot shows none of these extreme values or oscillatory behavior, nor does it extend to index 194. Not a match.

D. Describes a series starting at index 390 around -42 , falling to -979 at index 615, then climbing to 554 at index 634. The indices and values are completely different from the plot (which spans 0 to 120). Not a match.

The correct answer is B.

Table 28: Illustration of “Global Pattern Differentiation” task in L2. “<ts>” denotes various input types of time series that represent numerical information, e.g., numerical tables, textual arrays, or specific time series tokens.

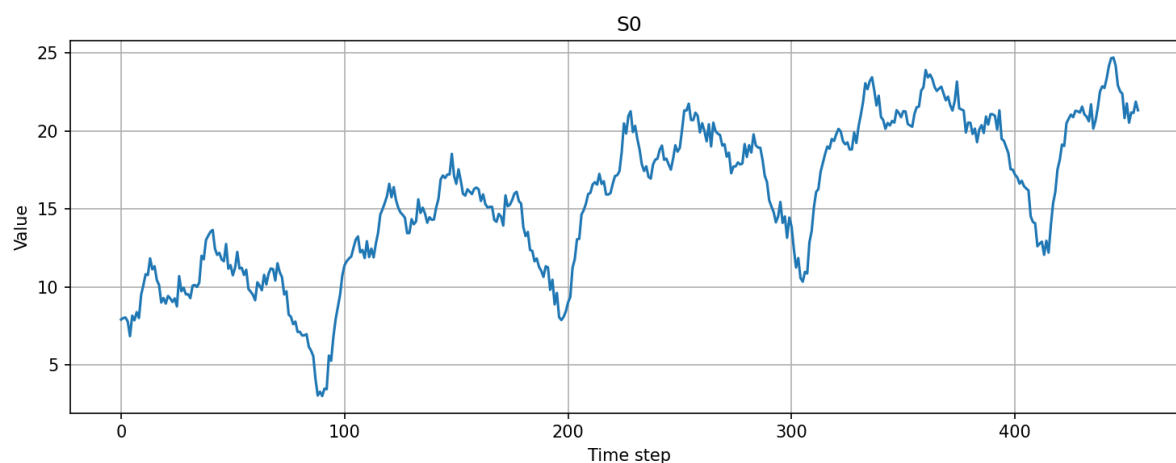
L2-Pattern Perception: Numerical Perception

Question: Given the time series: <ts>, carefully analyze each option, choose the option that correctly describes **BOTH** the numerical features and the overall shape.

Constraint: You must select the option with the most precise numerical value.

Options:

- A. The starting value is 7.950, and the overall trend is increasing.
- B. The minimum value is 3.295 at index 90, and the overall trend is increasing.
- C. The minimum value is 3.016 at index 90, and the overall trend is increasing.
- D. The minimum value is 3.016 at index 90, and the overall trend is decreasing.



Answer:

Option A: incorrect. Although the shape description matches, the numerical value 7.950 is imprecise. According to the numerical information, the correct value for min is 3.016.

Option B: incorrect. Although the shape description matches, the numerical value 3.295 is imprecise. According to the numerical information, the correct value for min is 3.016.

Option D: incorrect. While the numerical value is correct, the shape description ‘the overall trend is decreasing’ contradicts the visual plot. The actual trend is that the overall trend is increasing.

The correct answer is C.

Table 29: Illustration of “Numerical Perception” task in L2. “<ts>” denotes various input types of time series that represent numerical information, e.g., numerical tables, textual arrays, or specific time series tokens.

L3-Semantic Reasoning

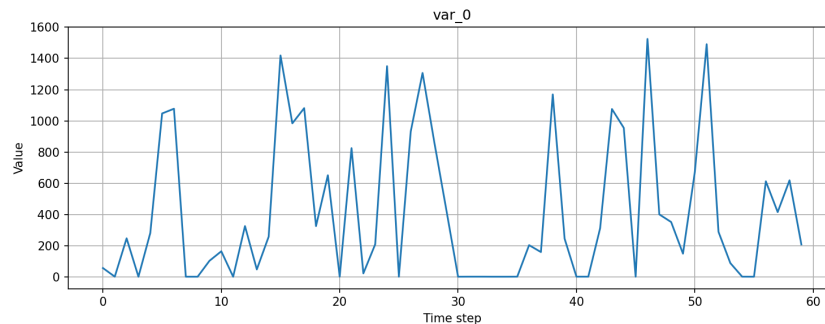
Question: Given the time series: <ts>, considering the overall pattern, precise numerical values, and semantic logic, which of the following scenarios best fits the given time series? You need first to check the overall pattern, then verify the precise values, and finally think whether the series exactly matches the given background knowledge.

A: The time series shows a periodic daily cycle with peak power outputs consistently occurring every 12th observation, specifically at index 12 (324.45 kW) and index 24 (1350.66 kW), reflecting regular wind patterns in a Chinese wind farm.

B: There are multiple zero or near-zero power readings (e.g., index 1: 0.0 kW, index 33: -0.3 kW) that align with turbine shutdowns during low wind conditions, and the highest recorded power output is 1525.51 kW at index 46, which is physically plausible for a modern wind turbine.

C: The power output increases monotonically after index 30, rising from -0.3 kW to 618.11 kW by index 58, indicating a steady wind acceleration over 48 minutes, consistent with sudden gusts in arid regions of China.

D: The value at index 5 is 1047.14 kW, and it decreases to 1077.57 kW at index 6, which represents a typical fluctuation due to wind turbulence, and this behavior occurs every 5 observations, forming a repeating short-term pattern.



Answer: Step-by-step analysis:

Option A: The claim of a periodic daily cycle every 12 observations is invalid because 12 observations correspond to 2 hours ($12 * 10$ minutes), not a full day; additionally, peak values do not consistently occur every 12 steps (e.g., index 15: 1419.46 kW, index 46: 1525.51 kW), and the pattern is irregular. This is a Pattern Error.

Option B: The statement correctly identifies multiple zero and near-zero values (e.g., index 1: 0.0, index 33: -0.3), which are plausible during turbine idling or low wind; the maximum value 1525.51 kW at index 46 is consistent with typical wind turbine capacity (1.5 MW class), and the context aligns with real-world operation in China. No errors detected.

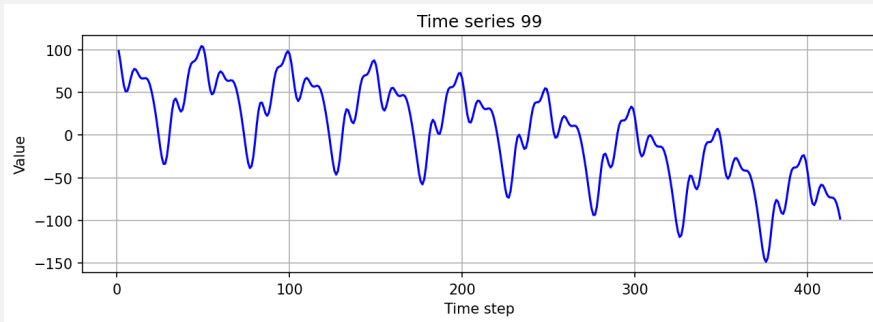
Option C: The claim of monotonic increase after index 30 is false: from index 30 (-0.3) to 31 (0.0) to 32 (0.0) to 33 (-0.3), the series does not rise steadily; further, from index 34 onward, values fluctuate (e.g., index 35: 0.0, index 36: 202.17) but not monotonically. The assertion of a steady rise is contradicted by data. This is a Value Error.

Option D: The statement claims index 5 (1047.14) decreases to index 6 (1077.57), but 1077.57 is greater than 1047.14, so it is an increase, not a decrease. Additionally, no repeating 5-observation pattern is evident in the data. This contains a Value Error due to a correct comparison.

Therefore, after eliminating options with incorrect or unsupported interpretations, only **Option B** remains fully consistent with the data in terms of pattern, values, and semantics.

<answer>B</answer>

Table 30: Illustration of “Semantic Understanding” task in L3. “<ts>” denotes various input types of time series that represent numerical information, e.g., numerical tables, textual arrays, or specific time series tokens.



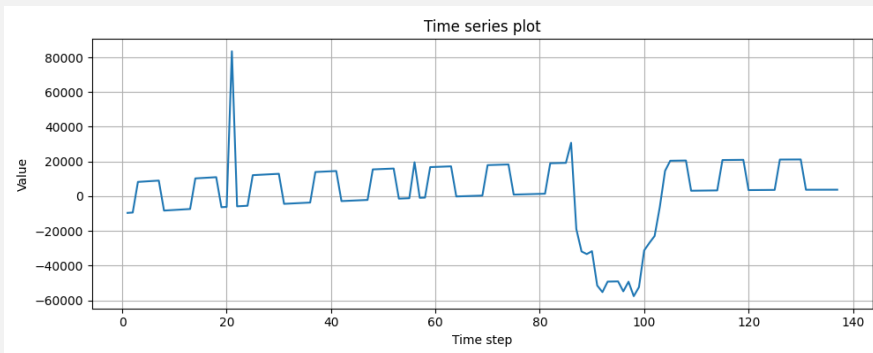
You are a time series analysis expert. Generate a clear and brief natural language description of the given time series plot. You can describe its shape, main phases, and key events based on the metadata below. As key numerical characteristics must be included, the precise values represented in the metadata are recommended to be referenced from the coordinate axis division in the plot, and higher precision should be omitted. Only describing approximate values in natural language and do not use '~' notation. Do not summarize every characteristic in the metadata; generate a concise and specific description in one sentence, like the following example:

This time series shows gradual growth from index 0 to 50, followed by a steady decline until index 100, then exhibits exponential growth peaking at index 280 with a maximum value of 1.259 before sharply dropping.

Metadata:

- Length: 419 points
- Overall trend: decreasing
- Seasonality: seasonality pattern of type 'sin' with period of 49.9 points
- maximum value 104.2374 at position 49
- minimum value -148.294 at position 376
- starts at 98.42943844179285 and ends at -97.69269541940011 (overall change 'down')
- Turning points: no significant turning points
- Local events: no major local events
- Trend segments: decrease from index 0 to 418 (values 98.42943844179285→-97.69269541940011)
- volatility level is 'high'.
- Cycle stats: peaks between -90.98002956701706 and 104.23737349499228;
- valleys between -37.91800010327716 and 102.68065545342499;
- mean amplitude:-36.82626304378404; 8.4 cycles in total.

Figure 3: Prompt template for global pattern annotation.



You are a time series analysis expert. Carefully analyze the provided time series plot and choose the single best option that most accurately describes the pattern depicted.

Please analyze each option step-by-step, and explain every option whether it matches the given time series plots, and point out any inaccuracies. After you have analyzed all options, output the best answer in the format: "The correct answer is X".

Options:

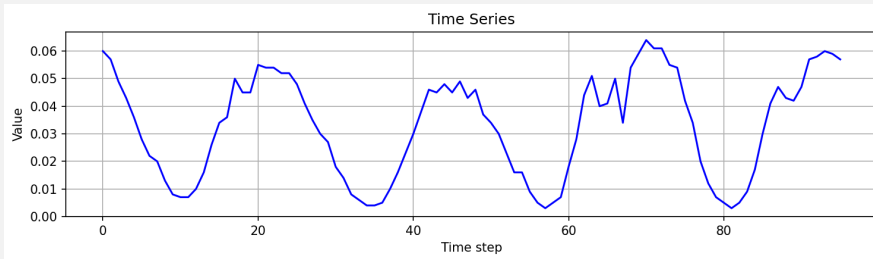
A: The time series exhibits a sinusoidal periodic fluctuation, with an approximate minimum period length of 11.3.

B: The overall trend of the time series is steady, which means no significant increasing or decreasing overall trend.

C: The value at the end of the series is lower than at the start.

D: The time series contains 3 overall trend turning points. Turning points refer to changes in the overall trend direction.

Figure 4: Prompt template for generating CoT for L2 tasks.



You are an expert in Time Series Analysis. Analyze the given numerical time series and generate one high-quality multiple-choice question (MCQ) with a detailed Chain-of-Thought evaluation for each option.

Output **MUST** be a single, valid JSON object with exactly the following structure: question: string, options:

A: string,
B: string,
C: string,
D: string

chain of thought:

A: evaluation: string, error type: none or Pattern/Value/Semantic,

B: evaluation: string, error type: none or Pattern/Value/Semantic,

C: evaluation: string, error type: none or Pattern/Value/Semantic,

D: evaluation: string, error type: none or Pattern/Value/Semantic

final verification:

checked values: [list of [index, value] verified],

neighbor consistency: boolean ,

Final answer: A/B/C/D

RULES: - DO NOT include any text before or after the JSON.

- ALL strings must be properly escaped.

- Use zero-based indexing to verify all [index, value] claims.

- One and only one correct answer; assign it randomly, but ensure others have one clear error:

- Pattern Error: wrong overall behavior (e.g., says periodic when irregular)

- Value Error: incorrect index-value pair or broken neighbor logic

- Semantic Error: impossible real-world context (e.g., room temp at 8ms)

- In options: combine Pattern (P), Values (V), Semantics (S) into one fluent sentence per choice.

- Academic English only. No markdown, no line breaks.

<input-data>

Time series (length=96, sampling interval=hourly,element format: [index,value]):

[[0, 0.06], [1, 0.057], [2, 0.049], [3, 0.043], [4, 0.036], ...]

Source Info:

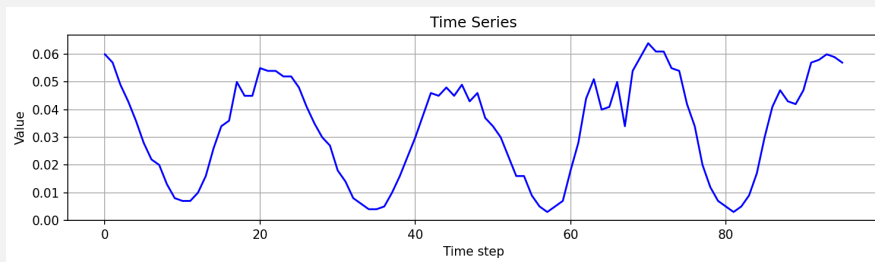
frequency: hourly,

true scenario: This is an hourly measured time series that represents road occupancy rate recorded by San Francisco freeway sensors from 2015 to 2016.

</input-data>

Now output ONLY the JSON.

Figure 5: Prompt template for generating L3 QA pairs.



You are a meticulous Question Examiner. Your task is to evaluate the correctness of the analysis in the “chain of thought” section for a multiple-choice time series question. You must verify whether each option’s evaluation is strictly accurate – based **only on the raw time series data**, not on assumptions about real-world timing or formatting conventions.

You can refer to the time series plot to support your evaluation.

The input format is fixed:

timeseries: [[index,value],...],

question: {...},

options: {...},

chain of thought: {...},

final verification: {...},

final answer: X,

sampling interval: {...}

Follow these rules RIGOROUSLY. Do NOT use external reasoning that violates these rules.

1. NO ABSOLUTE TIME INFERENCE ALLOWED – BUT RELATIVE TIME FRAME MUST BE RESPECTED WHEN INTERVAL IS GIVEN

- NEVER assume that index=0 corresponds to 00:00, 06:00, or any clock time.

- HOWEVER, IF the question specifies a measurement interval (e.g., every 30 minutes), then:

- Elapsed time from start = index × interval duration in hours.

- Terms like “morning”, “evening”, “peak”, “night” must be evaluated for relative plausibility within realistic human activity patterns.

- Example: “morning rise” should occur between approximately 6–11 hours from a plausible start; “evening peak” between 17–22 hours.

- If two events are labeled as same-day morning and evening, their elapsed times must differ by roughly 6–12 hours, not 30+ hours.

- Most critically: claims of “daily pattern” or “bimodal daily cycle” require that peaks recur approximately every 48 indices (for 30-min data), or strong evidence of repetition within the series.

- Do NOT accept narrative consistency if temporal spacing makes it physically implausible.

- You must calculate elapsed times whenever an interval is given, and reject interpretations where:

- A claimed “same-day” morning/evening pair spans more than 24 hours,

- Or a “daily” pattern lacks repeating structure at 48-step intervals.

1.1: DAILY PERIODICITY REQUIRES EVIDENCE OF REPETITION

- For claims involving “daily”, “bimodal daily”, “repeats each day”:

- There must be observable recurrence at intervals close to 48 steps (for 30-min data) or an appropriate multiple.

- At least one full cycle should be visible.

- Values at corresponding phases (e.g., index i and $i+48$) should be similar within a reasonable tolerance ($\sim \pm 5\%$).

- Absence of such repetition invalidates the claim, regardless of individual peak shapes.

2. NUMERICAL VERIFICATION WITH CONTEXTUAL TOLERANCE

- For any claimed value (e.g., “value at index 2 is 5014.8”):

- Accept it if the actual value rounds to the stated value within one decimal place.

- Specifically: if the option states a number with fewer decimals (e.g., 5014.8), and the true value is 5014.835 → this is acceptable.

- Do NOT require exact string match or full precision.

- Only demand exactness if the option uses words like “exactly”, “precisely”, or “equals down to three decimals”.

- This rule applies to all numerical claims, including maxima, minima, and comparisons.

3. FULL VALUE & EXTREMUM CHECK

- For claims involving “highest”, “lowest”, “maximum”, “minimum”:

- YOU MUST scan all points to confirm.

- But: still apply Rule #2 – allow rounded reporting.

- Example: option says “maximum is about 7983.9” → actual max 7983.885 → Valid

4. PATTERN & TREND: ACCEPT PLAUSIBLE NARRATIVES

- Do NOT reject an option because another interpretation exists.

- Accept a pattern description if:

- It matches the data trend in the relevant window,

- The terminology is not factually wrong,

- And the shape is realistic under the described scenario.

- Reject only if:

- The trend is opposite (e.g., claims increase but value drops),

- Or generalizes a short fluctuation to the whole series,

- Or uses clearly false periodicity (e.g., “every 6 hours” with no recurrence).

5. SEMANTIC PLAUSIBILITY (REAL-WORLD SENSE)

- Evaluate whether explanations make sense:

- e.g., “due to sunrise” → should align with the rising trend at that point

- “industrial growth” → should show sustained increase, not fluctuations

- But: never override numerical or pattern correctness due to personal belief.

6. RESIST CONFIRMATION BIAS

- Do NOT trust the “chain of thought” just because it looks detailed.

- Re-check every claim independently.

- But: if it correctly identifies errors and supports the right answer, mark as valid.

7. FINAL OUTPUT FORMAT (STRICTLY ENFORCED)

Output exactly one of:

“valid”: “true”, “reason”: All analyses are correct.

valid: false, reason : [Error type]: [Option X] has an issue because [specific reason with index/value or pattern mismatch].

- Do NOT say “index 50 is 25 hours from start”.

- Do NOT reject for 5014.835 vs 5014.8 unless the claim is exact.

</input-data>

</input-data>

Figure 6: Prompt template for L3 QA pairs verification.

Given the time series data:

[64.234214, 104.221399, 104.478061, 104.747216, 104.972426, 105.241486, 105.503337, ...],

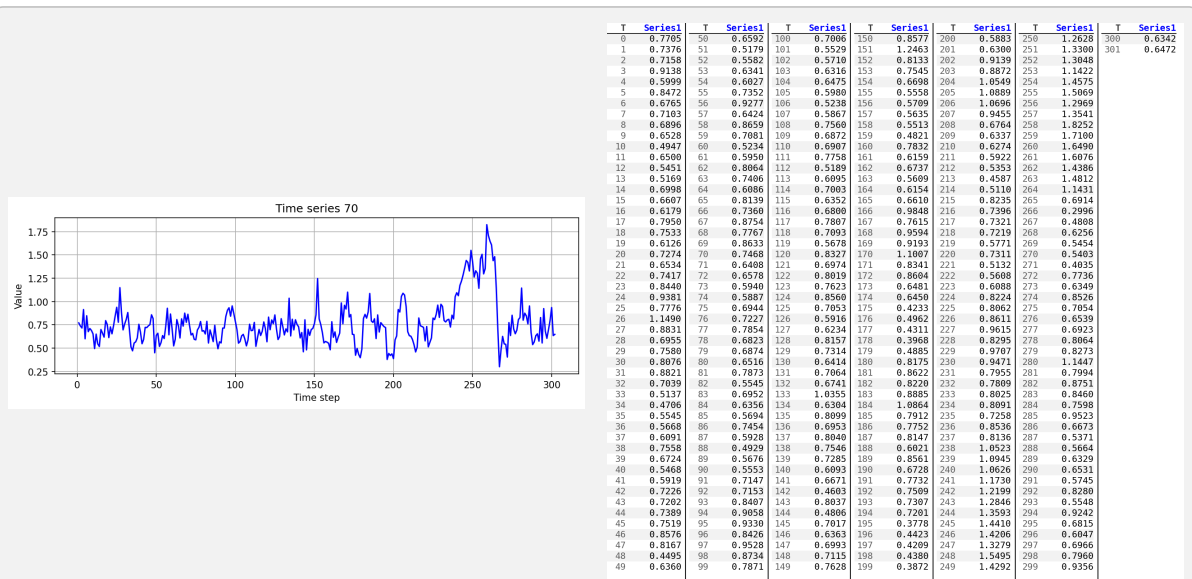
<question>

Given the time series data (**element format[index,value]**), start from T = 0:

[[0, 64.234214], [1, 104.221399], [2, 104.478061], [3, 104.747216], [4, 104.972426], [5, 105.241486], [6, 105.503337], ...],

<question>

Figure 7: Two types of templates for text models, i.e., “w/o index” and “w/ index”.



<image><image>You must refer to both of the two time series images to answer the question.

The first image plots the visual trends. The second image is a **High-Density Numeric Grid** that provides precise values.

How to read the second image (Numerical Grid):

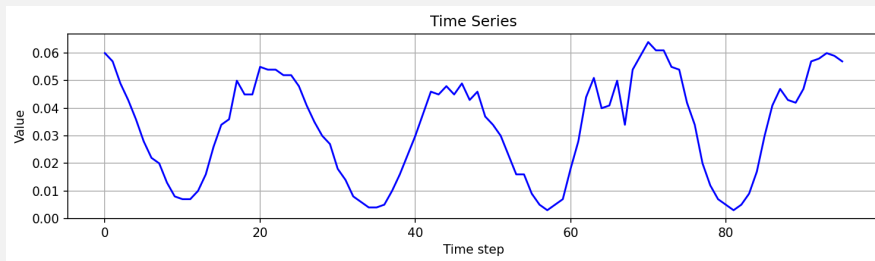
1. **Layout:** The data is organized in a multi-column newspaper layout. Read the columns from **left to right**.

2. **Vertical Flow:** Within each major column, read the data vertically from **top to bottom**. When a column reaches the bottom, the sequence continues at the top of the next column to the right.

3. **Structure:** Inside each column block, the **leftmost sub-column labeled 'T'** represents the time index. The subsequent sub-columns (e.g., Series-1) represent the values of different time series.

<question>

Figure 8: Prompt template for numerical table input, e.g., LLaTISA.



<image>You must refer to the time series' plot and its numerical series to answer the question. Given the time series data:

[64.234214, 104.221399, 104.478061, 104.747216, 104.972426, 105.241486, 105.503337, ...],

<question>

-

<image>You must refer to the time series plot and its numerical series to answer the question. Given the time series data (**element format[index,value]**), **start from T = 0**:

[[0, 64.234214], [1, 104.221399], [2, 104.478061], [3, 104.747216], [4, 104.972426], [5, 105.241486], [6, 105.503337], ...],

<question>

Figure 9: Two types of templates for dual-input of image and textual arrays.

Case: L1. Min/Max Grounding

Prompt: <image><image> You must refer to the second provided image to answer the question. The first image plots the visual trends. The second image is a **High-Density Numeric Grid** that provides precise values.

How to read the second image (Numerical Grid):

- Layout:** The data is organized in a multi-column newspaper layout. Read the columns from **left to right**.
- Vertical Flow:** Within each major column, read the data vertically from **top to bottom**. When a column reaches the bottom, the sequence continues at the top of the next column to the right.
- Structure:** Inside each column block, the **leftmost sub-column labeled 'T'** represents the time index. The subsequent sub-columns (e.g., Series) represent the values of different time series.

Question: Find the maximum and minimum values in the time series and report their first occurrence indices (if max/min repeats, use the earliest index). Provide the exact values and state which appears first. The first index is 0.

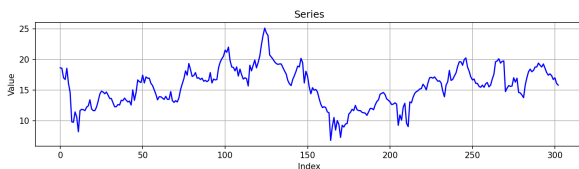
You MUST exactly follow the output format as:

<max>[index, value]</max> <min>[index, value]</min>

The max value appears first.

OR

The min value appears first.



T	Series	T	Series	T	Series	T	Series	T	Series	T	Series	T	Series	T	Series	T	Series
1	18.37200	51	16.18900	101	21.74800	151	15.54700	201	12.60200	251	16.81300	301	16.18900	351	16.81300	401	16.18900
2	17.92400	52	17.10500	102	22.03900	152	14.42100	202	12.73300	252	16.10900	302	16.10900	352	16.10900	402	16.10900
3	16.74300	53	16.95400	103	19.83800	153	15.46600	203	12.94400	253	16.10900	303	16.10900	353	16.10900	403	16.10900
4	18.37200	54	16.95400	104	18.78700	154	14.96400	204	12.73300	254	15.61700	304	15.61700	354	15.61700	404	15.61700
5	16.10900	55	16.18900	105	18.71200	155	15.12500	205	9.28600	255	15.47600	305	15.47600	355	15.47600	405	15.47600
6	14.82000	56	15.75000	106	18.25000	156	14.70300	206	10.70800	256	15.82000	306	15.82000	356	15.82000	406	15.82000
7	9.84900	57	15.85400	107	18.78300	157	13.64700	207	10.13800	257	15.47600	307	15.47600	357	15.47600	407	15.47600
8	9.77800	58	14.25100	108	17.26500	158	12.73200	208	12.24900	258	16.25000	308	16.25000	358	16.25000	408	16.25000
9	11.46700	59	11.43600	109	18.43100	159	12.17900	209	12.87400	259	16.25000	309	16.25000	359	16.25000	409	16.25000
10	10.42100	60	11.92900	110	17.56800	160	12.31100	210	8.63800	260	15.54700	310	15.54700	360	15.54700	410	15.54700
11	8.23100	61	13.92900	111	16.81300	161	12.17900	211	9.87500	261	15.82000	311	15.82000	361	15.82000	411	15.82000
12	11.07800	62	11.71800	112	17.02400	162	11.46700	212	13.06800	262	16.81300	312	16.81300	362	16.81300	412	16.81300
13	11.80900	63	13.50700	113	16.08300	163	11.25000	213	11.94400	263	17.58700	313	17.58700	363	17.58700	413	17.58700
14	11.01800	64	11.99900	114	15.68700	164	6.82400	214	13.85800	264	19.62700	314	19.62700	364	19.62700	414	19.62700
15	11.07800	65	11.50700	115	19.06400	165	9.87500	215	14.49100	265	19.76600	315	19.76600	365	19.76600	415	19.76600
16	12.24000	66	11.58700	116	18.15000	166	18.52200	216	14.77300	266	20.11900	316	20.11900	366	20.11900	416	20.11900
17	12.45100	67	14.77200	117	19.06400	167	6.51200	217	14.49100	267	19.41600	317	19.41600	367	19.41600	417	19.41600
18	13.43600	68	11.29000	118	19.90800	168	18.06000	218	15.29500	268	19.69700	318	19.69700	368	19.69700	418	19.69700
19	11.95900	69	13.81400	119	19.64200	169	9.42000	219	15.26500	269	19.76600	319	19.76600	369	19.76600	419	19.76600
20	11.07800	70	11.29000	120	19.48600	170	7.31000	220	15.96900	270	14.77300	320	14.77300	370	14.77300	420	14.77300
21	11.07800	71	13.89500	121	20.54100	171	9.29000	221	15.67900	271	15.39600	321	15.39600	371	15.39600	421	15.39600
22	12.24000	72	11.78800	122	22.37000	172	9.88400	222	15.95400	272	15.75800	322	15.75800	372	15.75800	422	15.75800
23	13.21500	73	13.12500	123	23.01000	173	9.49700	223	16.16800	273	15.61700	323	15.61700	373	15.61700	423	15.61700
24	14.35100	74	16.83900	124	25.11400	174	9.56700	224	17.02400	274	15.68700	324	15.68700	374	15.68700	424	15.68700
25	14.46300	75	16.88300	125	24.34000	175	11.13000	225	17.04800	275	17.04800	325	17.04800	375	17.04800	425	17.04800
26	14.79300	76	18.15800	126	23.91800	176	11.39600	226	16.95400	276	16.31300	326	16.31300	376	16.31300	426	16.31300
27	14.42100	77	17.44800	127	26.71300	177	11.88900	227	17.16100	277	18.95400	327	18.95400	377	18.95400	427	18.95400
28	14.79300	78	19.34500	128	28.40100	178	11.07800	228	16.81300	278	14.63200	328	14.63200	378	14.63200	428	14.63200
29	14.21000	79	18.38100	129	20.64900	179	12.52200	229	16.46100	279	14.56200	329	14.56200	379	14.56200	429	14.56200
30	13.64700	80	17.25200	130	19.67900	180	11.81600	230	16.52100	280	14.21000	330	14.21000	380	14.21000	430	14.21000
31	13.64700	81	17.37600	131	19.34500	181	11.07800	231	16.18900	281	13.78800	331	13.78800	381	13.78800	431	13.78800
32	12.94400	82	17.86800	132	19.20500	182	11.07800	232	14.83900	282	15.82000	332	15.82000	382	15.82000	432	15.82000
33	12.31100	83	16.95400	133	19.27500	183	11.39600	233	13.92800	283	17.04800	333	17.04800	383	17.04800	433	17.04800
34	12.31100	84	17.82400	134	19.27500	184	11.32600	234	14.82800	284	18.00900	334	18.00900	384	18.00900	434	18.00900
35	12.96200	85	16.74300	135	18.71200	185	11.25000	235	16.39100	285	18.43100	335	18.43100	385	18.43100	435	18.43100
36	12.95900	86	16.95400	136	18.25000	186	10.94000	236	18.20800	286	18.00900	336	18.00900	386	18.00900	436	18.00900
37	13.96600	87	16.46100	137	17.05700	187	11.46700	237	16.60200	287	18.22000	337	18.22000	387	18.22000	437	18.22000
38	13.96600	88	16.60200	138	16.60200	188	12.02000	238	16.74300	288	18.78300	338	18.78300	388	18.78300	438	18.78300
39	13.71800	89	16.39100	139	16.03900	189	12.82500	239	17.38900	289	18.78300	339	18.78300	389	18.78300	439	18.78300
40	13.86600	90	16.88300	140	15.75800	190	11.81600	240	17.26600	290	18.41600	340	18.41600	390	18.41600	440	18.41600
41	13.86500	91	17.86800	141	16.74300	191	12.59200	241	18.85300	291	19.66400	341	19.66400	391	19.66400	441	19.66400
42	13.22500	92	18.18900	142	17.76600	192	13.15000	242	18.67900	292	18.79300	342	18.79300	392	18.79300	442	18.79300
43	12.59200	93	16.81300	143	18.07900	193	13.43600	243	19.62700	293	19.27500	343	19.27500	393	19.27500	443	19.27500
44	15.05600	94	16.67200	144	18.92300	194	14.35100	244	19.96400	294	18.37200	344	18.37200	394	18.37200	444	18.37200
45	13.36600	95	16.74300	145	18.74300	195	14.49100	245	19.77800	295	17.58700	345	17.58700	395	17.58700	445	17.58700
46	16.63200	96	18.64200	146	20.18000	196	14.63200	246	20.26800	296	17.44600	346	17.44600	396	17.44600	446	17.44600
47	16.07200	97	15.55000	147	19.88600	197	14.29000	247	18.71800	297	17.05700	347	17.05700	397	17.05700	447	17.05700
48	16.39100	98	20.11900	148	18.18000	198	13.57700	248	20.87900	298	17.37600	348	17.37600	398	17.37600	448	17.37600
49	16.25000	99	20.54100	149	18.07900	199	13.36600	249	17.16100	299	16.74300	349	16.74300	399	16.74300	449	16.74300

Answer:

<max>[124, 25.11400]</max>

<min>[164, 6.82400]</min>

The max value appears first.

Table 31: “Min/Max Grounding” OOD example with LLATISA’s response.

Case: L2. Local Pattern Differentiation

Prompt: <image><image> You must refer to the second provided image to answer the question. The first image plots the visual trends. The second image is a **High-Density Numeric Grid** that provides precise values.

How to read the second image (Numerical Grid):

- Layout:** The data is organized in a multi-column newspaper layout. Read the columns from **left to right**.
- Vertical Flow:** Within each major column, read the data vertically from **top to bottom**. When a column reaches the bottom, the sequence continues at the top of the next column to the right.
- Structure:** Inside each column block, the **leftmost sub-column labeled ‘T’** represents the time index. The subsequent sub-columns (e.g., Series) represent the values of different time series.

Question:

Carefully analyze the given time series description and choose the single best option that most accurately describes the pattern for the time series.

Options:

- A: decreased in the middle
- B: slow and steady decrease
- C: ends with a drastic increase
- D: begins with a shallow increase



T	Series	T	Series	T	Series	T	Series	T	Series	T	Series	T	Series	T	Series
1	18.57200	51	16.18900	101	21.17400	151	15.54700	201	12.66200	251	16.81300	301	16.19900	351	25.82800
2	17.80400	52	17.18500	102	22.20900	152	14.42300	202	12.23200	252	16.39900	302	16.92800	352	25.82800
3	16.74300	53	16.95400	103	19.83800	153	15.40600	203	12.94400	253	16.10900	303	16.10900	353	25.82800
4	18.27100	54	16.25400	104	18.38800	154	14.98800	204	12.21800	254	15.61700	304	16.10900	354	25.82800
5	16.10900	55	16.10900	105	18.71200	155	15.12500	205	9.28600	255	15.47600	305	16.10900	355	25.82800
6	14.63200	56	15.75800	106	18.23600	156	14.70300	206	10.97400	256	15.82800	306	16.10900	356	25.82800
7	9.84900	57	15.85400	107	18.78300	157	14.44700	207	10.13800	257	15.47600	307	16.10900	357	25.82800
8	9.77800	58	14.35100	108	17.30500	158	12.73300	208	12.24800	258	16.03900	308	16.03900	358	25.82800
9	11.46700	59	13.43600	109	18.31300	159	12.17000	209	12.87800	259	16.25600	309	16.25600	359	25.82800
10	18.02200	60	13.92900	110	17.31600	160	12.31100	210	9.63800	260	15.54700	310	16.25600	360	25.82800
11	9.23100	61	13.92900	111	16.81300	161	12.17000	211	9.67900	261	15.82800	311	16.25600	361	25.82800
12	11.67800	62	13.71900	112	17.62400	162	11.46700	212	13.86500	262	16.81300	312	16.81300	362	25.82800
13	11.90900	63	13.50100	113	16.80300	163	11.37000	213	12.94400	263	15.54700	313	16.81300	363	25.82800
14	11.81800	64	13.99900	114	15.68700	164	6.82400	214	13.85800	264	19.62700	314	19.62700	364	25.82800
15	11.87800	65	13.50100	115	19.66400	165	9.97500	215	14.84100	265	19.76800	315	19.76800	365	25.82800
16	12.24900	66	13.58700	116	18.15800	166	16.55200	216	14.77300	266	20.11900	316	20.11900	366	25.82800
17	12.45100	67	14.77300	117	19.66400	167	6.51200	217	14.84100	267	19.41600	317	19.41600	367	25.82800
18	13.43600	68	13.29600	118	19.90800	168	16.06600	218	15.18500	268	19.60700	318	19.60700	368	25.82800
19	11.90900	69	13.81800	119	18.44100	169	9.42600	219	15.26500	269	19.76800	319	19.76800	369	25.82800
20	11.67800	70	13.29600	120	19.48600	170	7.31600	220	15.96900	270	14.77300	320	14.77300	370	25.82800
21	11.67800	71	13.88500	121	20.34300	171	9.28600	221	15.61700	271	15.33600	321	15.33600	371	25.82800
22	12.94400	72	13.78900	122	22.78900	172	9.89400	222	15.96900	272	15.75800	322	15.75800	372	25.82800
23	13.25500	73	15.12500	123	23.91800	173	9.49700	223	16.18800	273	15.61700	323	15.61700	373	25.82800
24	14.35100	74	16.92900	124	21.81000	174	9.56100	224	17.85800	274	15.68700	324	15.68700	374	25.82800
25	14.84300	75	16.88300	125	24.34000	175	11.11500	225	17.89400	275	17.02400	325	17.02400	375	25.82800
26	14.70300	76	16.12500	126	23.81000	176	11.39600	226	16.96400	276	16.32100	326	16.32100	376	25.82800
27	14.42100	77	17.44600	127	20.75200	177	11.89900	227	17.16500	277	16.95400	327	16.95400	377	25.82800
28	14.70300	78	16.34500	128	20.60100	178	11.67800	228	16.81300	278	14.63200	328	14.63200	378	25.82800
29	14.21000	79	16.36100	129	20.64900	179	12.52200	229	16.46100	279	14.56200	329	14.56200	379	25.82800
30	13.64700	80	17.22900	130	19.62700	180	11.81600	230	16.52100	280	14.29600	330	14.29600	380	25.82800
31	13.64700	81	17.37600	131	19.34500	181	11.67800	231	16.18800	281	13.78900	331	13.78900	381	25.82800
32	12.94400	82	17.86800	132	19.25600	182	11.67800	232	14.84100	282	14.84100	332	14.84100	382	25.82800
33	12.31100	83	16.95400	133	19.27500	183	11.99600	233	13.92900	283	17.02400	333	17.02400	383	25.82800
34	12.31100	84	17.62400	134	19.27500	184	11.28600	234	15.82800	284	18.90900	334	18.90900	384	25.82800
35	12.66200	85	16.74300	135	18.71200	185	11.25600	235	16.39100	285	18.41300	335	18.41300	385	25.82800
36	12.94400	86	16.95400	136	18.23600	186	10.96400	236	16.28800	286	18.09900	336	18.09900	386	25.82800
37	13.36600	87	16.46100	137	17.62400	187	10.46700	237	16.60300	287	18.20000	337	18.20000	387	25.82800
38	13.78900	88	16.60200	138	16.60200	188	12.80200	238	16.74300	288	18.78300	338	18.78300	388	25.82800
39	13.71900	89	16.39100	139	16.93900	189	12.80200	239	17.29200	289	18.78300	339	18.78300	389	25.82800
40	13.36600	90	16.60200	140	15.75800	190	11.81800	240	17.79800	290	19.41600	340	19.41600	390	25.82800
41	13.88500	91	15.86800	141	16.74300	191	12.55200	241	18.82300	291	19.66400	341	19.66400	391	25.82800
42	13.25500	92	16.18900	142	17.37600	192	13.15500	242	19.62700	292	18.78300	342	18.78300	392	25.82800
43	12.50200	93	15.81300	143	18.07900	193	13.45600	243	15.61700	293	19.27400	343	19.27400	393	25.82800
44	15.65400	94	16.97200	144	18.92300	194	14.35100	244	16.96400	294	18.57200	344	18.57200	394	25.82800
45	13.36600	95	16.74300	145	18.78300	195	14.49100	245	19.57600	295	17.86600	345	17.86600	395	25.82800
46	14.63200	96	18.84200	146	20.19900	196	14.83200	246	20.26800	296	17.44600	346	17.44600	396	25.82800
47	16.67200	97	15.55600	147	19.86600	197	14.28600	247	18.78300	297	17.02400	347	17.02400	397	25.82800
48	16.99100	98	20.11900	148	18.18800	198	13.57700	248	18.97800	298	17.37600	348	17.37600	398	25.82800
49	16.25600	99	20.54100	149	18.07900	199	13.36600	249	17.16500	299	16.74300	349	16.74300	399	25.82800

Answer:

Option A: “decreased in the middle” — The series does dip in the middle, but the description is vague and doesn’t capture the later strong rebound and final rise. It’s partially right, but not the best summary.

Option B: “slow and steady decrease” — Incorrect. The series is not steadily decreasing; it has pronounced rises and falls, a trough around the middle, and a strong rise at the end.

Option C: “ends with a drastic increase” — Correct. After a low near the middle and a modest rise, the final segment jumps sharply from mid-50s to around 80, clearly a drastic increase.

Option D: “begins with a shallow increase” — The start does show a small rise (around 1–2 steps), but this is a minor detail and doesn’t describe the overall pattern as well as the strong final increase.

The correct answer is C.

Table 32: “Local Pattern Differentiation” OOD example with LLATISA’s response. The question is from BEDTime (Sen et al., 2025).

Case: L2. Global Pattern Differentiation

Prompt: <image><image> You must refer to the second provided image to answer the question. The first image plots the visual trends. The second image is a **High-Density Numeric Grid** that provides precise values.

How to read the second image (Numerical Grid):

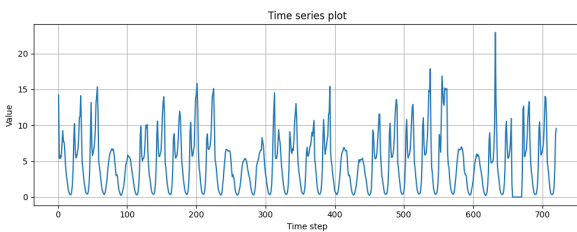
- Layout:** The data is organized in a multi-column newspaper layout. Read the columns from **left to right**.
- Vertical Flow:** Within each major column, read the data vertically from **top to bottom**. When a column reaches the bottom, the sequence continues at the top of the next column to the right.
- Structure:** Inside each column block, the **leftmost sub-column labeled ‘T’** represents the time index. The subsequent sub-columns (e.g., Series) represent the values of different time series.

Question:

Given the provided time series plot with its value table, select the description from the four options that best corresponds to the time series without overthinking.

Options:

- The data shows regular oscillatory behavior with consistent amplitude variations, featuring recurring peaks around 20-30 and baseline values near 1-2 throughout the sequence.
- This time series exhibits cyclical patterns with periodic spikes reaching peaks around 15-20, followed by gradual declines to near-zero values before repeating the cycle.
- This series demonstrates relatively stable fluctuations around 2-3 with occasional upward excursions, showing gradual trending behavior and moderate variability across the observation period.
- The data maintains high baseline values around 87-94 with occasional step-like transitions and brief periods of lower values, exhibiting overall stability with discrete level changes.



T	Series 1	Series 2	Series 3	Series 4	Series 5	Series 6	Series 7	Series 8	Series 9	Series 10	Series 11	Series 12	Series 13	Series 14	Series 15	Series 16	Series 17	Series 18	Series 19	Series 20	Series 21	Series 22	Series 23	Series 24	Series 25	Series 26	Series 27	Series 28	Series 29	Series 30	Series 31	Series 32	Series 33	Series 34	Series 35	Series 36	Series 37	Series 38	Series 39	Series 40	Series 41	Series 42	Series 43	Series 44	Series 45	Series 46	Series 47	Series 48	Series 49	Series 50	Series 51	Series 52	Series 53	Series 54	Series 55	Series 56	Series 57	Series 58	Series 59	Series 60	Series 61	Series 62	Series 63	Series 64	Series 65	Series 66	Series 67	Series 68	Series 69	Series 70	Series 71	Series 72	Series 73	Series 74	Series 75	Series 76	Series 77	Series 78	Series 79	Series 80	Series 81	Series 82	Series 83	Series 84	Series 85	Series 86	Series 87	Series 88	Series 89	Series 90	Series 91	Series 92	Series 93	Series 94	Series 95	Series 96	Series 97	Series 98	Series 99	Series 100
0	3.2	4.1	5.0	6.0	7.0	8.0	9.0	10.0	11.0	12.0	13.0	14.0	15.0	16.0	17.0	18.0	19.0	20.0	21.0	22.0	23.0	24.0	25.0	26.0	27.0	28.0	29.0	30.0	31.0	32.0	33.0	34.0	35.0	36.0	37.0	38.0	39.0	40.0	41.0	42.0	43.0	44.0	45.0	46.0	47.0	48.0	49.0	50.0	51.0	52.0	53.0	54.0	55.0	56.0	57.0	58.0	59.0	60.0	61.0	62.0	63.0	64.0	65.0	66.0	67.0	68.0	69.0	70.0	71.0	72.0	73.0	74.0	75.0	76.0	77.0	78.0	79.0	80.0	81.0	82.0	83.0	84.0	85.0	86.0	87.0	88.0	89.0	90.0	91.0	92.0	93.0	94.0	95.0	96.0	97.0	98.0	99.0	100.0		

Answer:

Option A: Not a good match. While the series does oscillate, the peaks are not consistently in the 20–30 range (many are 10–15, and there’s a maximum around ~23). The baseline is not persistently near 1–2; it sits closer to 3–6 with frequent oscillations. The “consistent amplitude variations” phrasing is also inconsistent with the plot’s changing peak heights.

Option B: Best match. The plot shows recurring cyclical behavior with spikes that often reach roughly 15 units (some cycles peak higher, around 18–22). After spikes, values frequently drop toward or briefly touch zero before the cycle repeats. This description aligns well with the observed oscillatory pattern and magnitude.

Option C: Not a match. The series is not centered around 2–3, nor are the fluctuations “gradual” with a stable mean; instead, there are pronounced oscillations and spikes up to well above 10, with changing baseline levels.

Option D: Clearly incorrect. The baseline is nowhere near 87–94; values range roughly from 0 to about 23, with no step-like transitions between high plateaus.

The correct answer is B.

Table 33: “Global Pattern Differentiation” OOD example with LLaTISA’s response. The question is from MMTS-Bench (Anonymous, 2025).