The Landscape of Agentic Time Series Systems: Architectures, Reliability, and Frontiers

Anonymous authors
Paper under double-blind review

Abstract

Time series analysis is moving beyond the classical goal of accurate forecasting toward interpretation, verification, and decision-making in dynamic environments. Time series foundation models improve cross-domain generalization through reusable temporal representations, while Large Language Models for Time Series (LLM4TS) introduce language interfaces, multimodal alignment, and explicit reasoning. However, most systems remain model-centric, mapping observations to outputs without a closed loop over evidence acquisition, tool or action selection, feedback, and state updates. This survey presents a systematic landscape of *agentic time series systems*, which treat time series analysis as closed-loop interaction with temporal environments. We trace the progression from foundation models to LLM-based translators, temporal reasoners, and closed-loop agents, and define a time series agent as an operational system that observes temporal evidence, selects tools or actions over evolving states, receives feedback, and adapts subsequent behavior. We organize the literature through five capability layers: perception, reasoning, planning and action, memory and knowledge, and temporal world models. This framework serves as a capability map rather than a mandatory checklist. We further treat benchmarks as evolving evaluation infrastructure, considering reliability and trustworthiness as cross-cutting requirements that span predictive quality, reasoning faithfulness, tool use, grounding, robustness, safety, auditability, and reproducibility. Building on an extensive literature corpus, we synthesize representative methods, benchmarks, applications, and research frontiers, and argue for reliable, auditable, and adaptive agents that can analyze, decide, learn, and act under dynamic uncertainty.

1 Introduction

Time series data are one of the main interfaces through which computational systems observe dynamic environments (Wang et al., 2026a; Kottapalli et al., 2025; Wen et al., 2022; Zhang et al., 2024; Chang et al., 2025), including finance (Zhu et al., 2024; Hu et al., 2025b; Cheng et al., 2022), healthcare (Soyiri & Reidpath, 2013; Vargas-Santiago et al., 2025; Patel & Rahimi, 2026), transportation (Zhang et al., 2026c; Liu et al., 2023), industry (Fatima & Rahimi, 2024; Hu & Man, 2023), and environmental science (Niu et al., 2025; Bi et al., 2022). In these settings, the goal is rarely only to predict the next value. Practical systems must inspect data quality, contextualize temporal patterns, quantify uncertainty, select tools, and sometimes turn forecasts into decisions whose outcomes unfold later. Time series analysis is therefore moving from isolated prediction toward a sequential process of evidence acquisition, reasoning, action, and adaptation.

The methodological evolution of the field reflects this widening scope, as sketched in Fig. 1. Classical statistical and deep models learn temporal dependencies for forecasting, classification, imputation, and anomaly detection (Zhou et al., 2021; Wu et al., 2021; Zhou et al., 2022; Zeng et al., 2023; Hu et al., 2025d; 2026; 2025c; Liu et al., 2024c). Time series foundation models extend this model-centric paradigm through large-scale pretraining, reusable representations, and zero- or few-shot transfer (Liu et al., 2026c; 2025a; Das et al., 2023; Ansari et al., 2025; Khwaja et al., 2026). LLM4TS methods then broaden the interface: translator-style systems align numerical series with prompts, symbols, text, visualizations, or multimodal representations (Zhou et al., 2023; Jin et al., 2024; Cao et al., 2023; Liu et al., 2024d; 2025b; Xie et al., 2024; Jiang et al., 2026), while reasoner-style systems study temporal inference through chain-of-thought, slow

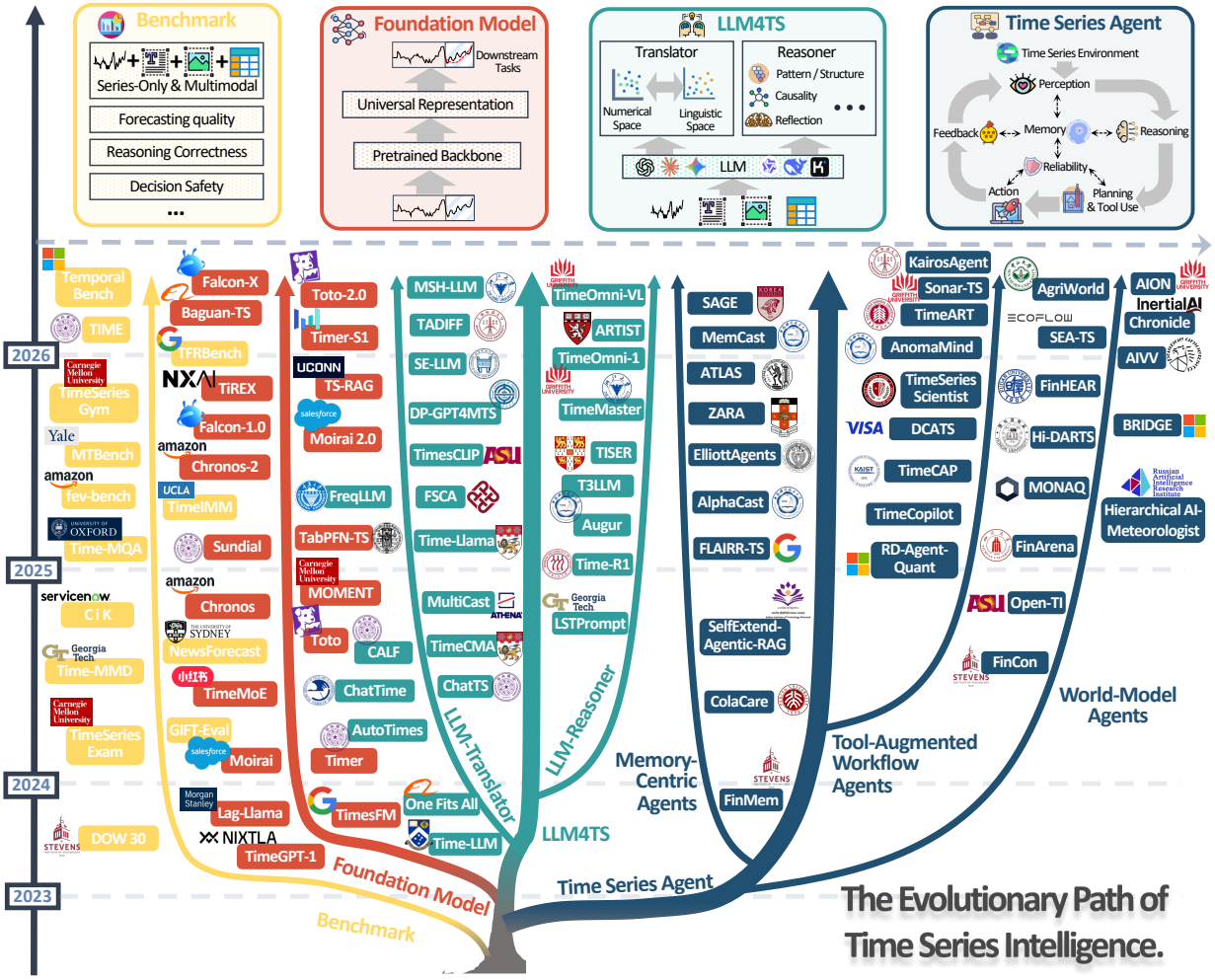

Figure 1: The evolutionary path of time series intelligence across benchmarks, foundation models, LLM4TS, and time series agents. The field progresses from model-centric representation learning toward language-mediated reasoning and closed-loop agents that perceive temporal environments, use tools, update state, and act under feedback. Representative works illustrate this transition.

thinking, reinforcement learning, process verification, and reviewer-style correction (Cai et al., 2024; Yu et al., 2025a; Cheng et al., 2026b; Luo et al., 2025b; Guan et al., 2025; Messica et al., 2026; Lu et al., 2026). These developments make time series more interpretable and language-accessible, but they do not by themselves create agency.

The missing ingredient is a closed loop. A model may align a series with text or explain a pattern without deciding what evidence to acquire, which tool to invoke, whether an intermediate result is trustworthy, what should be remembered, or how future behavior should change after feedback (Cheng et al., 2026a; Kong et al., 2025b). This distinction matters under non-stationarity (Liu et al., 2025c; Ryan et al., 2025), delayed feedback (Long et al., 2024), costly or irreversible actions (Hu et al., 2025a; Li et al., 2026d), accumulating uncertainty (Shi et al., 2024), and event-driven regime shifts (Ge et al., 2025; Wang et al., 2025a). We use *agentic time series systems* to refer to systems that operate in such environments by observing temporal evidence, selecting tools or actions over evolving states, receiving feedback, and updating subsequent behavior. Richer systems may further incorporate explicit reasoning, memory, world models, and verification (Luo et al., 2025a; Wang et al., 2024a). Recent work already exposes these capabilities across perception (Ruan & Liang, 2026; Trirat et al., 2026; Lee et al., 2025; Sukhorukov et al., 2026), reasoning (Wu et al., 2026b; Liu et al., 2025d; Tao et al., 2026c;b), planning and action (Zhao et al., 2025; Yeh et al., 2025; Li et al., 2026d; Xiao et al., 2024), memory (Yu et al., 2025b; Jalori et al., 2025; Zhang et al., 2026a; Tao et al., 2026a; Pan et al.,

2026), and world modeling (Zhang et al., 2026b; Quinlan et al., 2026; Zhang et al., 2025b). The detailed boundary among backbone models, translators, reasoners, and agents is formalized in Sec. 2.

Evaluation evolves in parallel. Forecasting benchmarks measure predictive quality and generalization (Aksu et al., 2024; Shchur et al., 2025; Qiao et al., 2026); multimodal, question-answering, and reasoning benchmarks test contextual understanding and faithful inference (Liu et al., 2024b; Kong et al., 2025a; Cai et al., 2024; Chen et al., 2025a; Weng et al., 2026; Ahamed et al., 2026); and system-level benchmarks increasingly evaluate workflow construction, tool use, delayed feedback, and decision quality (Cai et al., 2025b; Nechepurenko & Shuvalov, 2026; Mostafa et al., 2026; Karger et al., 2025). Benchmarks should therefore be treated as diagnostic infrastructure rather than as a substage of time series modeling (Wang et al., 2026a). As systems become more agentic, evaluation must move from output accuracy toward reasoning correctness, tool-use reliability, decision safety, auditability, and deployment trustworthiness.

Despite this progress, the literature still lacks a clear organizing framework. Existing surveys mainly organize LLM4TS around prompting, tokenization, representation learning, alignment, forecasting, or multimodal interfaces (Zhang et al., 2025c; 2024; Shi et al., 2025a), while recent work has begun to discuss reasoning and agentic forecasting (Kong et al., 2025b; Ning et al., 2025a; Chang et al., 2025; Cheng et al., 2026a). Three boundaries remain under-specified: ① when an LLM-enhanced time series model becomes a closed-loop agent; ② how architectural capabilities differ from mechanisms such as RAG, tool use, and multi-agent coordination; and ③ how failures propagate across perception, reasoning, tools, memory, and actions. A survey of agentic time series systems must therefore explain what constitutes time series agency, how its capabilities interact, and how reliability should be evaluated.

Motivated by this gap, this survey reconciles definitions across foundation models, LLM4TS, temporal reasoners, and closed-loop agents. It also connects multimodal alignment, tool-augmented reasoning, memory-centric adaptation, world models, benchmarks, applications, and reliability principles for trustworthy temporal intelligence. Specifically, we aim to address the following key questions:

Key Questions

- ❶ **Paradigm Boundary.** What distinguishes agentic time series systems from time series foundation models, LLM-based translators, and time series reasoners?
- ❷ **Agentic Architecture.** What core capabilities are necessary for a time series system to operate as a closed-loop agent in temporal environments?
- ❸ **Reliability and Trustworthiness.** How should reliability be defined and evaluated for systems that perceive temporal evidence, reason over dynamics, use tools, update memory, and take actions?
- ❹ **Evaluation and Applications.** What do existing benchmarks and real-world applications reveal about the current maturity, limitations, and deployment readiness of agentic time series systems?
- ❺ **Future Frontiers.** What research directions are most critical for moving from agentic prototypes toward trustworthy autonomous temporal intelligence?

To answer these questions, we first distinguish *LLM as Backbone*, *LLM as Translator*, *LLM as Reasoner*, and *Time Series Agent*. We then organize the literature through five capability layers: perception, reasoning, planning and action, memory and knowledge, and temporal world models. Reliability and trustworthiness are treated as cross-cutting requirements spanning prediction, reasoning, grounding, tool use, action safety, auditability, and reproducibility. Benchmarks and applications are synthesized to identify measurable progress and deployment gaps, and the final section discusses frontiers in temporal reasoning, memory, decision-centric evaluation, world models, standardized tooling, and time series data-agent engineering.

Contributions This survey makes four main contributions. (1) We distinguish *LLM as Backbone*, *LLM as Translator*, *LLM as Reasoner*, and *Time Series Agent*, while treating benchmarks as parallel evaluation infrastructure. (2) We organize the literature through five capability layers, including perception, reasoning, planning and action, memory and knowledge, and time series world models. (3) We treat reliability and trustworthiness as cross-cutting requirements and analyze failure propagation across evidence, reasoning, tools, actions, and state updates. (4) Based on more than 200 papers, we synthesize architectures, benchmarks, applications, and frontiers, identifying gaps in evaluation, deployment, time series agents, and world modeling.

Table 1: Four paradigms of LLM-based time series systems. The key distinction lies in whether the system operates in a closed loop involving temporal evidence, tool/action selection, feedback, and state updates. Explicit reasoning, memory, world modeling, and verification are treated as progressively richer capabilities.

Paradigm	**Closed loop**	**Core role**	**Representative systems**	**Core limitation**
Pretrained Backbone	No	Learns temporal representations or predictive distributions directly from numerical series.	Chronos (Ansari et al., 2024), MOMENT (Goswami et al., 2024), TimesFM (Das et al., 2023)	Strong general-purpose forecasters, but no explicit tool use, memory, or action loop.
LLM as Translator	No	Maps time series into text, tokens, prompts, images, or semantic representations for an LLM.	Time-LLM (Jin et al., 2024), TEST (Sun et al., 2024), ChatTS (Xie et al., 2024)	Natural-language interfaces improve usability, but alignment errors can propagate downstream.
LLM as Reasoner	Partial	Performs explicit inference over temporal evidence, often through chain-of-thought, review, or reinforced reasoning.	TimeSeriesExam (Cai et al., 2024), PATRA (Lu et al., 2026), ARTIST (Messica et al., 2026)	Reasoning is visible, but the system often lacks persistent memory, tool-grounded action, and feedback-driven adaptation.
Time Series Agent	Yes	Maintains a closed-loop policy over temporal evidence, tool/action selection, feedback, and state update; may incorporate explicit reasoning, memory, world models, and verification.	TimeART (Wu et al., 2026b), Cast-R1 (Tao et al., 2026b), MemCast (Tao et al., 2026a), KairosAgent (Feng et al., 2026), TimeSeriesScientist (Zhao et al., 2025)	More autonomous and auditable, but more costly and harder to evaluate and reproduce.

2 Preliminaries: Formalizing Agentic Time Series Systems

Recent surveys often group many methods under "large language models for time series" (Chang et al., 2025; Zhang et al., 2024). For this survey, however, the central distinction is not whether an LLM appears in the pipeline, but whether the system maintains a closed loop over temporal evidence, tool or action selection, feedback, and state update. A pretrained forecaster, a numerical-to-language translator, a temporal reasoner, and an autonomous analytical workflow therefore instantiate different paradigms. This section first separates these paradigms (Sec. 2.1), then formalizes the time series agent as a closed-loop system in a temporal environment (Sec. 2.2), contrasts it with standard NLP-agent templates (Sec. 2.3), and finally connects the definition to the five-layer architecture used throughout the survey (Sec. 2.4).

2.1 Four Paradigms of LLM-Based Time Series Systems

The four paradigms in Tab. 1 and Fig. 1 are separated by one question: does the system only map inputs to outputs, or does it adapt through evidence acquisition, action, feedback, and state update?

The LLM-style multi-layer Transformer Backbone models such as Chronos (Ansari et al., 2024), MOMENT (Goswami et al., 2024), and TimesFM (Das et al., 2023) provide reusable temporal representations or predictive distributions, but normally terminate after a prediction. Translator-style systems such as Time-LLM (Jin et al., 2024), TEST (Sun et al., 2024), and ChatTS (Xie et al., 2024) align numerical series with text, prompts, tokens, or multimodal representations, thereby expanding the interface without necessarily adding stateful interaction. Reasoner-style systems ask models to infer patterns, anomalies, causal hypotheses, or future dynamics, as in TimeSeriesExam (Cai et al., 2024), PATRA (Lu et al., 2026), ARTIST (Messica et al., 2026), and chain-of-thought or review-based approaches (Su et al., 2025; Messica et al., 2026; Parker et al., 2025). Reasoning makes temporal inference more explicit, but a reasoner is not yet an agent if it does not decide what evidence to acquire, which tool or action to execute, what to remember, and how to adapt

after feedback. The time series agent paradigm (Zhao et al., 2025; Wu et al., 2026b; Tao et al., 2026b;a; Ruan & Liang, 2026) is therefore architectural rather than terminological: agency arises from a closed loop that observes a temporal environment, structures evidence, selects tools or actions, receives feedback, verifies outcomes, and updates future behavior.

2.2 Formalizing the Time Series Agent

A time series agent is a closed-loop decision system situated in a temporal environment. It is not defined merely by the presence of an LLM, a reasoning trace, or a tool call. We write the functional object as

$$\mathcal{A}_{TS} = \langle \mathcal{P},\, \mathcal{C}; \mathcal{R},\, \mathcal{M},\, \mathcal{W} \rangle_{\mathcal{E}_{TS}}, \tag{1}$$

where $\mathcal{P}$ and $\mathcal{C}$ are the minimal operational core: perception and planning/action. The optional but increasingly important layers $\mathcal{R}$, $\mathcal{M}$, and $\mathcal{W}$ denote explicit time series reasoning, memory and knowledge, and temporal world modeling. The subscript $\mathcal{E}_{TS}$ indicates that the agent is defined relative to an environment with evolving observations, feedback, delays, and constraints. These five functional roles organize Secs. 3 to 7. Verification is not placed inside the tuple; we denote it by $\mathcal{V}$ when needed and treat it as a cross-cutting reliability mechanism in Sec. 8.

At step t, the agent maintains a temporal information state

$$s_t = \langle x_{1:t},\, c_{1:t},\, z_t,\, \eta_t,\, b_t,\, \Omega_t \rangle, \tag{2}$$

where $x_{1:t}$ is the observed series, $c_{1:t}$ aligned context or events, z_t an inferred regime or latent environment state, η_t the internal agent state, b_t a belief or uncertainty state, and Ω_t the available tool and action space. A generic closed-loop step is

$$e_t = \mathcal{P}(x_{1:t}, c_{1:t}; s_t), \tag{3}$$

$$r_t = \mathcal{R}_{\text{opt}}(e_t, \eta_t, \mathcal{W}_{\text{opt}}), \tag{4}$$

$$(u_t, a_t) \sim \mathcal{C}(\,\cdot \mid s_t, e_t, r_t, \Omega_t\,), \qquad y_{t+\Delta} = \text{Env}(a_t, u_t), \tag{5}$$

$$\eta_{t+1} = \mathcal{U}\big(\eta_t, \langle e_t, r_t, u_t, a_t, y_{t+\Delta} \rangle\big), \qquad s_{t+1} = \mathcal{T}(s_t, a_t, u_t, y_{t+\Delta}). \tag{6}$$

Here e_t is structured temporal evidence, r_t an optional reasoning state, u_t a tool invocation, a_t an analytic or external action, and $y_{t+\Delta}$ delayed feedback. If explicit memory is implemented, the update may include

$$m_{t+1} = \mathcal{M}\big(m_t, \langle e_t, r_t, u_t, a_t, y_{t+\Delta} \rangle\big). \tag{7}$$

Externalized memory is useful but not mandatory; the minimal requirement is that subsequent decisions can depend on the evolving state rather than only on the original input.

Reliability enters as an evaluator or constraint over the trajectory:

$$v_t = \mathcal{V}(s_t, e_t, r_t, u_t, a_t, y_{t+\Delta}). \tag{8}$$

Such verification may be an internal self-check, external evaluator, statistical diagnostic, human review process, or safety constraint. It assesses whether evidence is grounded, reasoning is faithful, tools are used correctly, actions are safe, memory updates are auditable, and simulations are valid. This boundary allows backbone models, translators, and temporal reasoners to serve as components inside an agent without being agents themselves. Current systems instantiate different subsets of the abstraction, including tool-grounded reasoning in TimeART (Wu et al., 2026b) and TS-Agent (Liu et al., 2025d), workflow planning in TimeSeriesScientist (Zhao et al., 2025), sequential decision policies in Cast-R1 (Tao et al., 2026b), memory-augmented forecasting in MemCast (Tao et al., 2026a), and executable world-tool environments in AgriWorld (Zhang et al., 2026b).

2.3 Time Series Agents $\neq$ NLP Agents

Time series agents and NLP agents share a closed-loop template: both may maintain histories, invoke tools, observe intermediate results, and revise later actions. A general tool-using agent can be written as

$$(u_t, a_t) \sim \pi(u, a \mid \eta_t, o_t, \Omega_t)\,, \qquad \eta_{t+1} = \mathcal{T}_{\mathcal{E}}(\eta_t, o_t, u_t, a_t, y_{t+\Delta})\,, \tag{9}$$

Table 2: How temporal environments change the requirements of closed-loop agents. The differences are best understood as pressures on the observation, transition, feedback, policy, memory, and reliability components of a shared agentic abstraction.

Temporal property	**Where it affects the agent loop**	**Requirement for time series agents**	**Representative systems / benchmarks**
Non-stationarity	Observation and transition dynamics change over time.	Detect regime shifts, update retrieval and memory, and route tools or models under drift.	**MemCast** (Tao et al., 2026a); **FinMem** (Yu et al., 2025b); **TemporalBench** (Weng et al., 2026)
Delayed feedback	Reward, error, or outcome signals arrive after the environment evolves.	Maintain trajectories, assign delayed credit, and verify decisions after outcomes are observed.	**Cast-R1** (Tao et al., 2026b); **TradingAgents** (Xiao et al., 2024); **CryptoTrade** (Li et al., 2024)
Costly or irreversible action	The action space is constrained by risk, cost, and safety.	Constrain policies by confidence, risk budgets, human oversight, and pre-execution checks.	**FinArena** (Xu et al., 2025c); **Hubble** (Shi et al., 2026a); **AIVV** (Kwon et al., 2026)
Uncertainty accumulation	Beliefs and errors propagate across horizons, tools, and decisions.	Track calibration, propagate uncertainty, and trigger replanning, abstention, or fallback policies.	**Cast-R1** (Tao et al., 2026b); **TimeCopilot** (Garza & Rosillo, 2025); **Foresight Arena** (Nechepurenko & Shuvalov, 2026)
Multi-scale and multimodal evidence	The observation model must align heterogeneous signals across time scales.	Align series, events, logs, charts, tables, and domain signals across windows, frequencies, and regimes.	**AI-Meteorologist** (Sukhorukov et al., 2026); **AGCD** (Wu et al., 2026a); **HEARTS** (Li et al., 2026c)
Numerical and statistical grounding	Reliability depends on executable checks rather than textual plausibility alone.	Ground claims about trends, lags, thresholds, anomalies, and causality in diagnostics and raw series.	**TimeART** (Wu et al., 2026b); **TS-Agent** (Liu et al., 2025d); **TS-Reasoner** (Ye et al., 2026)

where η_t is the history or internal state, o_t the current observation, Ω_t the tool and action space, and $y_{t+\Delta}$ feedback from the environment. The difference lies in the structure of o_t, the transition $\mathcal{T}_{\mathcal{E}}$, the feedback delay Δ, and the constraints on actions. For time series agents, the observation is not merely a symbolic artifact or a larger prompt; it includes evolving numerical trajectories, aligned events, regimes, uncertainty, temporal context, and tool outputs, summarized as

$$s_t = \left\langle x_{1:t},\, c_{1:t},\, z_t,\, \eta_t,\, b_t,\, \Omega_t \right\rangle. \tag{10}$$

This environmental distinction changes where the closed-loop pressure appears. Translator and reasoner systems such as Time-LLM (Jin et al., 2024), TEST (Sun et al., 2024), ChatTS (Xie et al., 2024), PATRA (Lu et al., 2026), and TimeSeriesExam (Cai et al., 2024) improve alignment or inference, but do not necessarily model feedback, action constraints, or state evolution. Systems such as TimeSeriesScientist (Zhao et al., 2025), TimeART (Wu et al., 2026b), Cast-R1 (Tao et al., 2026b), TS-Agent (Liu et al., 2025d), TS-Reasoner (Ye et al., 2026), TimeCopilot (Garza & Rosillo, 2025), MemCast (Tao et al., 2026a), MAS4TS (Ruan & Liang, 2026), FinMem (Yu et al., 2025b), TradingAgents (Xiao et al., 2024), and AgriWorld (Zhang et al., 2026b) instantiate different parts of the temporal agent loop. Tab. 2 summarizes the resulting requirements.

Two consequences follow. First, time series reasoning must often be coupled to temporal operators such as decomposition, autocorrelation, change-point detection, frequency analysis, cross-correlation, causal-lag discovery, backtesting, and rolling validation. TimeART (Wu et al., 2026b), TS-Agent (Liu et al., 2025d), and TS-Reasoner (Ye et al., 2026) therefore bind reasoning to executed observations, while Cast-R1 (Tao et al., 2026b) treats forecasting as a sequential reasoning and decision process. The same grounding requirement extends to multimodal temporal context, including TimeCAP (Lee et al., 2025) and ColaCare (Wang et al.,

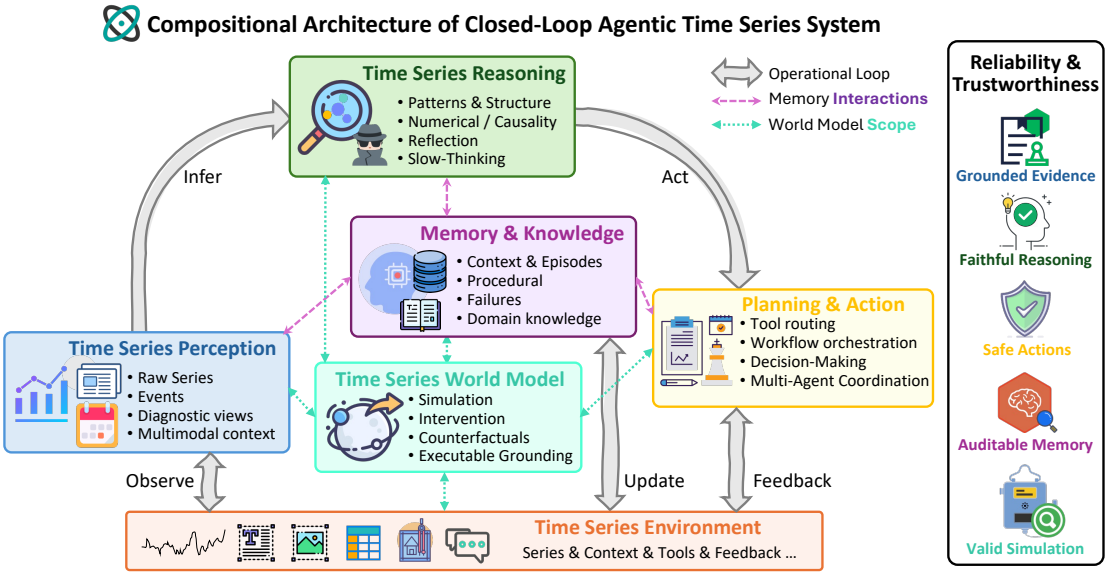

Figure 2: The components of the closed-loop agentic time series system. The system coordinates time-series perception, reasoning, planning and action, memory and knowledge, and temporal world modeling around interaction with the temporal environment. Reliability and trustworthiness provide cross-cutting constraints on evidence grounding, reasoning faithfulness, action safety, memory auditability, and simulation validity.

2025b). Second, deployment domains expose delayed and constrained consequences: financial agents combine market series, news, filings, and portfolio feedback (Yu et al., 2025b; 2024; Xiao et al., 2024; Papadakis et al., 2025; Li et al., 2024; Shi et al., 2026a; Xu et al., 2025c); traffic and energy systems convert forecasts into policies (Da et al., 2024); and AgriWorld embeds agents in executable agricultural world tools (Zhang et al., 2026b). They all demonstrate that temporal agency is shaped by the environment in which feedback and constraints arise. Thus, the relevant distinction is not whether a system calls itself an agent, but whether it closes the loop among temporal evidence, tools, actions, feedback, and state update.

2.4 The Five-Layer Architecture

The formal object $\mathcal{A}_{TS}$ organizes the survey into five capability layers and one cross-cutting reliability dimension. Perception and planning/action form the minimal core: the agent must make temporal evidence observable and decide what to do with it. Reasoning, memory, and world modeling enrich this core by supporting inference over dynamics, reuse of temporal experience, and simulation or counterfactual analysis of possible futures. These are functional capabilities rather than a checklist of required software modules; mechanisms such as RAG, tool use, multi-agent coordination, reinforcement learning, code execution, and procedural memory may realize different layers.

The architecture in Fig. 2 visualizes this organization. The subsequent sections elaborate the layers in order: perception constructs structured evidence (Sec. 3); reasoning turns evidence into grounded temporal claims (Sec. 4); planning and action select tools, workflows, and interventions (Sec. 5); memory and knowledge retain cases, regimes, procedures, and failures (Sec. 6); and world models support simulation, intervention analysis, and counterfactual reasoning (Sec. 7). Sec. 8 then treats reliability and trustworthiness as system-level constraints on evidence grounding, reasoning faithfulness, action safety, memory auditability, simulation validity, and reproducibility.

3 Time Series Perception: Observing and Structuring Temporal Environments

Time series perception makes temporal environments observable to an agent by constructing task-conditioned evidence rather than merely encoding observations for predefined prediction tasks (Goswami et al., 2024;

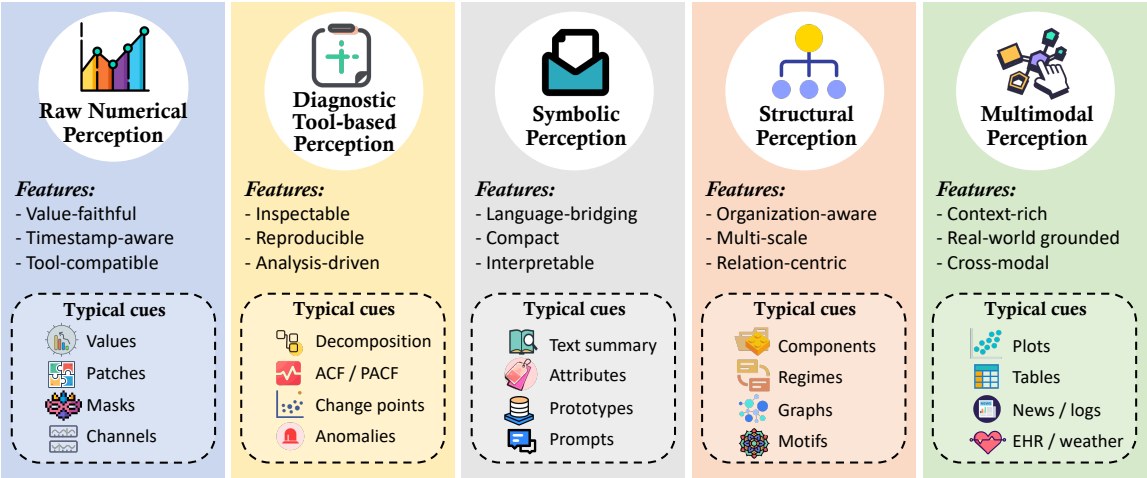

Figure 3: Taxonomy of time series perception in agentic systems. Perception can be categorized as raw numerical, diagnostic tool-based, symbolic, structural, and multimodal forms, which expose values, decompositions, summaries, regimes, graphs, motifs, plots, tables, and contextual records as structured evidence.

Ansari et al., 2024). Given heterogeneous inputs such as numerical trajectories, irregular events, contextual records, and domain knowledge (Xie et al., 2024; Chen et al., 2026), perception determines which signals should be observed, at what granularity, under which context, and with what reliability.

Formally, at task step t, the available observations are

$$\mathcal{O}_t = \{X_t,\, U_t,\, D_t,\, S_t,\, B_t,\, \Gamma_t\}\,. \tag{11}$$

where X_t denotes numerical observations, U_t external context, D_t diagnostic observations, S_t structural observations, B_t multimodal artifacts, and Γ_t task constraints or domain priors. The perception layer maps them into a structured evidence state:

$$E_t = \Phi_\theta(\mathcal{O}_t \mid g,\, m_t,\, \eta_{<t})\,, \tag{12}$$

where g is the task goal, m_t the agent memory, and $\eta_{<t}$ the interaction history. Unlike a single embedding, E_t is a perceptual record that may contain numerical representations, diagnostic summaries, symbolic abstractions, structural descriptions, multimodal alignments, and uncertainty estimates.

The remainder of this section reviews five complementary perceptual interfaces, as shown in Fig. 3. Together, they define how temporal environments become observable through numerical, diagnostic, symbolic, structural, and multimodal evidence (Jin et al., 2024; Xie et al., 2024), as summarized in Tab. 3.

3.1 Raw Numerical Perception

Raw numerical perception preserves the time series as a numerical object. It exposes values, timestamps, channels, and local temporal structures with minimal semantic abstraction (Ansari et al., 2024; Goswami et al., 2024). Its central question is how much numerical structure can be retained while making the sequence usable by a model or agent (Das et al., 2023; Liu et al., 2024e).

Recent time series foundation models instantiate this interface through tokenization, patching, quantization, masked encoding, and large-scale temporal pretraining. Chronos converts real-valued time series into discrete tokens through scaling and quantization, casting forecasting as sequence modeling over numerical tokens (Ansari et al., 2024). MOMENT learns general-purpose representations via masked time series modeling (Goswami et al., 2024). TimesFM, Moirai, Lag-Llama, and Timer similarly represent sequences through patches, windows, decoder-style modeling, or universal forecasting architectures (Das et al., 2023; Woo et al., 2024; Rasul et al., 2023; Liu et al., 2024e). Although these models are not agents, they provide reusable numerical substrates that can be queried, forecasted, reconstructed, or compared.

Table 3: Perception targets in the time series perception layer.

Perception target	Observable evidence	Perceptual function	Representative systems
Raw numerical evidence	Values, windows, patches, tokens.	Preserves numerical fidelity.	Chronos (Ansari et al., 2024), MOMENT (Goswami et al., 2024), Timer (Liu et al., 2024e).
Diagnostic evidence	Trend, anomalies, statistics, quality indicators.	Exposes diagnostic evidence.	TimeSeriesScientist (Zhao et al., 2025), TimeART (Wu et al., 2026b), TS-Agent (Liu et al., 2025d).
Symbolic evidence	Text summaries, labels, prototypes, semantic tokens.	Produces language-compatible abstractions.	Time-LLM (Jin et al., 2024), TEST (Sun et al., 2024), LLMTime (Gruver et al., 2023), ChatTS (Xie et al., 2024).
Structural evidence	Components, regimes, motifs, graphs, causal relations.	Reveals temporal organization.	PATRA (Lu et al., 2026), Augur (Cui et al., 2025), SGCMA (Sun et al., 2025), LeRet (Huang et al., 2024).
Multimodal evidence	Text, tables, images, external context.	Integrates cross-modal evidence.	TimeXL (Jiang et al., 2026), TSAD-Agents (Xu et al., 2026g), Time-VLM (Zhong et al., 2025), MILM (Chung et al., 2026).

For agents, raw numerical perception is useful because it preserves quantities against which derived interpretations can be verified (Liu et al., 2025d). This is critical when magnitude, timing, and calibration matter, and supports the use with forecasting models, anomaly detectors, statistical diagnostics, and backtesting tools.

Its limitation is that numerical fidelity does not guarantee semantic observability. A sequence may preserve values while obscuring whether a pattern is seasonal, meaningful, or externally driven. Tokenization, patching, and normalization may also introduce perceptual distortions (Ansari et al., 2024; Nie et al., 2022). Thus, raw numerical perception is necessary but insufficient: it provides a faithful view of the signal, but not necessarily an interpretable view of the temporal environment.

3.2 Diagnostic Tool-based Perception

Diagnostic tool-based perception exposes temporal properties that are not directly visible from raw observations. The agent invokes analytical procedures that transform a sequence into diagnostic evidence, such as trend and seasonal components, correlations, change points, anomalies, and data-quality indicators (Wu et al., 2026b; Liu et al., 2025d; Cleveland et al., 1990). The perceptual object is therefore not only the sequence itself, but also tool-produced views of its latent temporal properties.

Classical diagnostics form the basis of this interface. STL separates trend, seasonal, and residual components (Cleveland et al., 1990); ACF and PACF reveal lag-dependent correlations (Box et al., 2015); change-point detection identifies regime shifts or distributional breaks (Killick et al., 2012); and anomaly detection locates unusual observations or intervals (Chandola et al., 2009). When embedded in agentic workflows, these tools become perception operators that make hidden temporal evidence observable.

Agentic systems increasingly rely on such operators. TimeSeriesScientist decomposes analysis into stages such as data curation, planning, forecasting, and reporting, using diagnostics to guide processing decisions (Zhao et al., 2025). TimeART and TS-Agent ground temporal analysis in executable tool calls and intermediate observations, obtaining statistical and diagnostic evidence before producing conclusions (Wu et al., 2026b; Liu et al., 2025d). In these systems, diagnostic tools act as perceptual instruments at task time.

The strength of diagnostic perception lies in interpretability and auditability. Its outputs are easier to inspect, verify, and reproduce than latent representations, making it useful for data-quality assessment, anomaly investigation, model selection, and exploratory analysis. However, diagnostic evidence inherits the tool-used assumptions, including modeling choices, parameters, preprocessing, resampling, smoothing, normalization, and interpolation (Cleveland et al., 1990; Box et al., 2015; Aminikhanghahi & Cook, 2017). Thus, it should be treated as one contingent view of the temporal environment rather than a complete description.

3.3 Symbolic Perception

Symbolic perception converts temporal patterns into language-compatible or discrete semantic forms. Instead of exposing only numerical values or diagnostic outputs, it represents time series evidence as textual summaries, labels, prototypes, captions, or natural-language descriptions (Xue & Salim, 2023; Gruver et al., 2023). This interface is central to LLM-based time series systems because it allows temporal information to enter language-centric reasoning pipelines (Jin et al., 2024; Sun et al., 2024; Xie et al., 2024).

Several mechanisms instantiate symbolic perception, including textualization, prototype-based abstraction, attribute synthesis, and language-oriented representation. Time-LLM, TEST, PromptCast, LLMTime, ChatTS, and DualSG make time series accessible to language models through reprogramming, prompts, textual descriptions, captions, or time-series-to-language interfaces (Jin et al., 2024; Sun et al., 2024; Xue & Salim, 2023; Gruver et al., 2023; Xie et al., 2024; Ding et al., 2025). These approaches turn temporal observations into symbolic evidence for language-based reasoning and interaction.

The value of symbolic evidence lies in compactness, interpretability, and compatibility with LLM infrastructures. Symbolic descriptions can express trends, anomalies, and salient temporal properties in forms usable by both humans and language models (Gruver et al., 2023). They also support user interaction, retrieval, memory organization, and cross-domain reasoning (Xue & Salim, 2023).

However, symbolic perception is inherently lossy. It often preserves qualitative patterns while omitting quantitative details, uncertainty, and fine-grained timing. Generated descriptions may also introduce ambiguous, overconfident, or hallucinated interpretations when insufficiently grounded. For example, describing a series as "sharply increasing" is underspecified without magnitude, baseline, and temporal window. Symbolic perception should therefore be treated as an abstraction layer, not a replacement for numerical or diagnostic evidence. Its reliability depends on links to the observations and tools from which symbols are derived.

3.4 Structural Perception

Structural perception makes explicit the organization of a temporal environment. Rather than exposing only values or descriptions, it represents components, regimes, motifs, relational structures, and causal dependencies (Sun et al., 2025; Cui et al., 2025). The perceptual target is the structure underlying observations and the relations that organize temporal behavior (Lu et al., 2026; Huang et al., 2024).

This interface is important because many temporal environments are not collections of independent sequences. Industrial systems and traffic networks, for instance, contain interacting entities coupled across time and space. Structural perception exposes such dependencies through graph construction, regime segmentation, motif discovery, decomposition, and causal modeling (Sun et al., 2025; Cui et al., 2025).

Recent LLM-related time series systems increasingly instantiate this interface. SGCMA exploits state-transition graph structures shared by temporal patches and linguistic representations, making structural consistency part of perception (Sun et al., 2025). LeRet uses a causal dependence learner to capture sequential causal evolution across patches (Huang et al., 2024). PATRA and Augur similarly treat temporal components, pattern structures, and causal associations as explicit objects that can be aligned, refined, and reused (Lu et al., 2026; Cui et al., 2025). Compared with purely numerical representations, these approaches expose higher-order temporal organization as evidence for reasoning and forecasting.

Structural perception helps agents reason over dependencies, regime transitions, recurring patterns, and state evolution instead of observations (Sun et al., 2025; Cui et al., 2025; Lu et al., 2026; Huang et al., 2024). Its limitation is assumption dependence: inferred structures reflect modeling choices, estimation procedures, and data quality, not necessarily stable mechanisms. In nonstationary environments, structures valid in one period may fail later. Thus, structural evidence should be treated as provisional: valuable because it makes organization inspectable, but reliable only when the inferred structure is robust and stable (Cui et al., 2025).

3.5 Multimodal Perception

Multimodal perception extends the observable environment beyond numerical sequences. In many real-world settings, the meaning of a signal depends on external context: price movements may depend on news, while

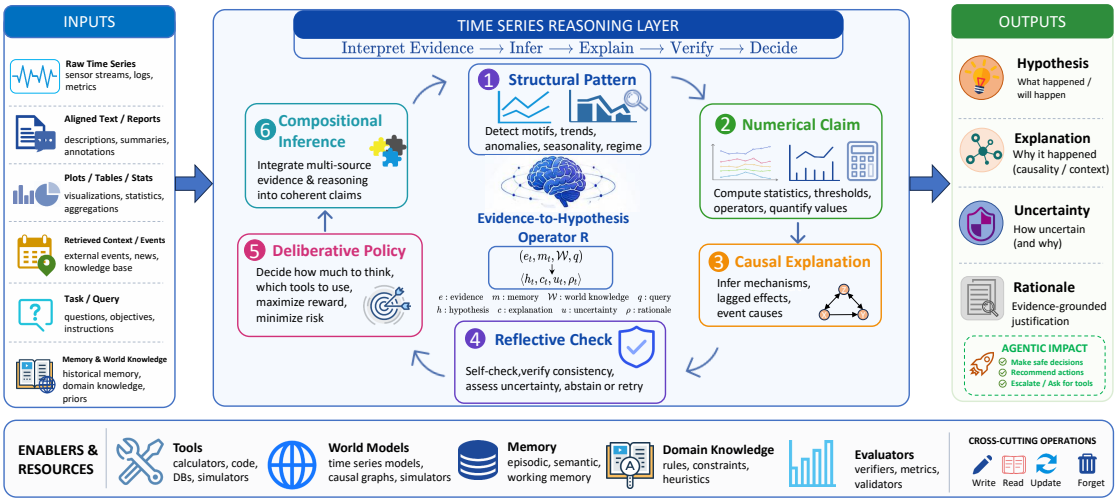

Figure 4: Overview of the time series reasoning layer, which transforms heterogeneous temporal evidence into hypotheses, explanations, uncertainty estimates, and evidence-grounded rationales for agentic decision-making.

sensor anomalies may depend on maintenance logs (Xie et al., 2024; Chen et al., 2026). This interface aligns time series with text, tables, images, spatial records, and other domain-specific artifacts (Jiang et al., 2026).

Systems such as TimeXL, MAS4TS, AGCD, TSAD-Agents, Time-VLM, and MILM illustrate the growing importance of multimodal temporal evidence, where the observable state is not contained in the sequence alone (Jiang et al., 2026; Ruan & Liang, 2026; Wu et al., 2026a; Xu et al., 2026g; Zhong et al., 2025; Chung et al., 2026). These approaches allow external artifacts to become part of the evidence available to an agent.

The main benefit of multimodal perception is contextualization. External evidence can explain why a signal changes, whether an apparent anomaly is expected, and which factors should be considered part of the temporal environment (Chen et al., 2026). For example, an abnormal load curve may become understandable under extreme weather, and a financial decline may align with a macroeconomic event rather than an unexpected anomaly.

The main challenge is alignment. Modalities differ in temporal resolution, uncertainty, noise, and semantic granularity. External context may be unsynchronized, irrelevant, or conflicting, and vision-based perception may be sensitive to chart scaling, resolution, or visual design (Chen et al., 2026). Thus, multimodal perception is not simply the addition of more context. It requires selecting external signals that are temporally aligned, semantically relevant, and reliable enough to support the current evidence state.

4 Time Series Reasoning: Inferring Temporal Dynamics and Causality

Time series reasoning connects temporal perception with downstream planning. It turns evidence from raw series, plots, tables, text, and retrieved context into claims about what happened, why it may have happened, how reliable the inference is, and what action it can support. Fig. 4 summarizes this role as an evidence-to-hypothesis interface for agentic time series systems. We formalize this interface as

$$\mathcal{R} : (e_t, m_t, \mathcal{W}, q) \rightarrow \langle \eta_t, c_t, \nu_t, \rho_t \rangle. \tag{13}$$

Here, e_t is the available evidence, m_t is task or episodic memory, $\mathcal{W}$ is domain knowledge, and q is the query. The outputs denote a temporal hypothesis η_t, a causal or contextual explanation c_t, an uncertainty statement ν_t, and an evidence-grounded rationale ρ_t. The goal is not longer reasoning traces, but traceable claims grounded in windows, statistics, events, tool outputs, or domain constraints.

Table 4: Reasoning targets in the time series reasoning layer, organized by the evidence each target should consume and the agentic risk if the inference is left unverified.

Reasoning target	**Evidence that should ground it**	**Agentic implication**	**Representative systems**
Structural pattern	Windows, plots, decomposition, motif or anomaly labels.	Determines what the agent can safely summarize or escalate.	TimeSeriesExamAgent (Gwiazda et al., 2026), PATRA (Lu et al., 2026), MTBench (Chen et al., 2025a), MAS4TS (Ruan & Liang, 2026), Segment Selection (Messica et al., 2026).
Numerical claim	Statistics, thresholds, operators, executable tool observations.	Turns fluent explanations into auditable evidence traces.	TS-Agent (Liu et al., 2025d), TimeART (Wu et al., 2026b), TS-Reasoner (Ye et al., 2026), Argos (Gu et al., 2025), VeriTime (Zhou et al., 2026).
Causal explanation	Temporal order, lag, event context, graph or test-based constraints.	Controls whether the agent may recommend intervention or only report association.	Augur (Cui et al., 2025), MTBench (Chen et al., 2025a), TimeOmni-1 (Guan et al., 2025), TS-Reasoner (Ye et al., 2026).
Reflective check	Intermediate traces, raw-series evidence, verifier comments, output contracts.	Decides whether to answer, retry, ask for tools, or abstain.	T3LLM (Su et al., 2025), TS-Agent (Liu et al., 2025d), CastFlow (Pan et al., 2026), AlphaCast (Zhang et al., 2026a), TS-Debate (Trirat et al., 2026), VeriTime (Zhou et al., 2026).
Deliberative policy	Process rewards, task taxonomies, forecast error, trend and seasonal scores.	Allocates slow thinking to hard temporal questions rather than every query.	Time-R1 (Luo et al., 2025b), Cast-R1 (Tao et al., 2026b), TimeOmni-1 (Guan et al., 2025), VeriTime (Zhou et al., 2026), PATRA (Lu et al., 2026), CastFlow (Pan et al., 2026).

Tab. 4 organizes the following discussion. Pattern reasoning shapes η_t, numerical reasoning strengthens ρ_t, causal reasoning refines c_t, reflection checks the consistency of $\langle \eta_t, c_t, \nu_t, \rho_t \rangle$, and slow-thinking methods decide when additional deliberation is worth the cost.

4.1 Pattern and Structural Reasoning

Pattern reasoning is the entry point of time series reasoning. Before an agent can explain a forecast, diagnose a fault, or compare causes, it must identify structural evidence such as trend, seasonality, local shape, similarity, volatility regime, anomaly, changepoint, or recurring motif. Benchmarks and segment-selection methods make this explicit: the agent must not only name a pattern, but also decide which temporal window supports the claim (Gwiazda et al., 2026; Messica et al., 2026).

Recent systems treat such structure as the interface between time series encoders and language reasoning. PATRA aligns decomposed temporal components with text, rationale-grounded in-context learning retrieves observation-to-implication precedents, and ZARA prunes sensor-level evidence before forming a decision insight (Lu et al., 2026; Liu et al., 2026b; Li et al., 2026e). Multimodal benchmarks and agents further extend this setting to plotted curves, tables, and textual context (Chen et al., 2025a; Ruan & Liang, 2026). The shared lesson is that pattern evidence is multi-scale: a global trend may hide a local reversal, seasonality may obscure an anomaly, and similar shapes may differ in lag or amplitude. Therefore, qualitative pattern descriptions should hand off numerical or ambiguous claims to statistic-grounded verification.

4.2 Numerical and Statistic-Grounded Reasoning

Many time series claims are directly checkable. Statements about thresholds, rates of change, volatility, correlation, anomaly scores, lags, or forecast horizons should be grounded in computed quantities rather

than fluent impressions. Tool-augmented agents such as TS-Agent, TimeART, and TS-Reasoner make this principle operational by separating candidate answers from the observations, operators, and execution traces that support them (Liu et al., 2025d; Wu et al., 2026b; Ye et al., 2026).

This evidence contract also appears in anomaly and domain-specific agents. Argos turns anomaly explanations into reproducible detection rules; AnomaMind and specialist-analyzer systems decompose anomaly evidence before aggregation; TSAG checks whether financial claims are backed by the intended quantitative operator (Gu et al., 2025; Tao et al., 2026c; Kang et al., 2026; Kolonin et al., 2026). Process-verifiable training systems such as VeriTime further reward intermediate validity rather than only final-answer correctness (Zhou et al., 2026). Still, numerical grounding is not automatic reliability: an agent can choose the wrong operator, pass the wrong window, or misread a tool result. Thus, ρ_t should specify which computation supports which claim, while ν_t should reflect tool coverage and failure modes.

4.3 Causal and Compositional Reasoning

Causal reasoning asks not only what pattern exists, but which variables, events, or mechanisms may have produced it. This requires temporal order, lag structure, confounding control, exogenous context, and domain constraints. A time series agent should therefore distinguish causal hypotheses from correlational summaries and make the evidence type behind each explanation explicit.

Existing systems approach this by making causal or event structure inspectable. Augur represents candidate causal associations as graph-like objects that can be pruned, judged, refined, and reused for forecasting (Cui et al., 2025). Grammar of the Wave compiles natural-language event definitions into event-logic trees before reporting explanations (Wan et al., 2026). Broader benchmarks and models such as MTBench, TimeOmni-1, and TS-Reasoner test whether temporal signals, retrieved context, and numerical operators can be composed under domain constraints (Chen et al., 2025a; Guan et al., 2025; Ye et al., 2026). The main risk is causal overreach: synchronized shocks, shared seasonality, and correlation do not establish causation. Without lag-sensitive tests, graph constraints, interventions, or verified events, causal claims should remain bounded.

4.4 Reflective and Metacognitive Reasoning

Reflective reasoning lets an agent inspect intermediate conclusions before producing an answer. This is especially useful for time series tasks because errors often occur early: the model may choose the wrong window, confuse seasonality with anomaly, ignore scale, or infer causality from coincidence. Reflection therefore checks whether the trace has enough evidence, whether the final claim follows from observations, and whether the output satisfies the task format.

Systems implement this separation differently. T3LLM reviews reasoning traces against the original series; TS-Agent uses critic and quality-gate checks; CastFlow and AlphaCast separate generation from reflective revision; TS-Debate turns multi-agent disagreement into a verification signal (Su et al., 2025; Liu et al., 2025d; Pan et al., 2026; Zhang et al., 2026a; Trirat et al., 2026). Process-verifiable pipelines such as VeriTime and execution harnesses such as AION make this review more structured, while CALM adds a streaming version by delaying anomaly judgment until sufficient context is available (Zhou et al., 2026; Zhan et al., 2026; Devireddy & Huang, 2025). Reflection is useful only when it can reject a claim for a concrete reason tied to raw windows, tools, schemas, or labels; otherwise, it becomes another layer of polished but unsupported text.

4.5 Slow-Thinking and Reinforcement-Enhanced Reasoning

Prompting can expose intermediate steps, but recent systems increasingly train temporal deliberation policies. Time-R1, Cast-R1, VeriTime, and TimeOmni-1 use supervised warm-up, tool-policy learning, process rewards, or task-grounded reinforcement to improve multi-step temporal reasoning (Luo et al., 2025b; Tao et al., 2026b; Zhou et al., 2026; Guan et al., 2025). PATRA and CastFlow show reward design must balance heterogeneous tasks, evidence alignment, and forecasting quality (Lu et al., 2026; Pan et al., 2026). These methods suggest that temporal reasoning can be shaped by format constraints, task correctness, process validity, and tool feedback. Yet slow thinking should remain adaptive: simple pattern or arithmetic questions should be answered directly, while causal, long-horizon, or event-conditioned tasks should expose intermediate evidence.

Table 5: Mechanism map for using reasoning inside an agentic time series system. Unlike Tab. 4, which classifies inference targets, this table summarizes when each mechanism should be invoked and what evidence contract it should leave for planning.

Mechanism	**When to invoke**	**Evidence contract**	**Planning handoff**
Pattern abstraction	The agent must name structure or choose a relevant segment.	Segment or plot id, pattern label, local counter-evidence, and ambiguity note.	Summarize if qualitative; call tools for thresholds, lags, or amplitudes.
Executable verification	The claim concerns a value, statistic, lag, volatility, anomaly score, or forecast number.	Operator name, parameters, input window, observation, and supported sentence.	Accept only within tool coverage; escalate if the statistic is used causally.
Causal composition	The agent explains direction, event attribution, confounding, or mechanism-level relationships.	Temporal order, lag evidence, graph or test constraints, event context, and association-versus-causation wording.	Recommend intervention only when the causal standard is met; otherwise report association.
Reflective critique	A generated trace may contain unsupported steps, format errors, evidence gaps, or inconsistent forecasts.	Reviewer decision tied to raw series, tool observations, output schema, or process labels, plus retry reason.	Retry, ask for more tools, abstain, or pass only the verified prefix to planning.
Deliberative training	The task requires multi-step temporal deliberation or repeated tool-policy decisions.	Rewards for task correctness, process validity, format, and calibrated forecast error, with over-thinking checks.	Route easy tasks directly; reserve slow policies for causal, long-horizon, or event-conditioned decisions.

Tab. 5 summarizes reasoning mechanisms as design choices rather than separate model families. A mature time series agent may identify a motif, compute a statistic, compare a causal hypothesis with retrieved context, check the trace reflection, and invoke slow thinking only when the task warrants additional deliberation.

Three design implications follow. First, reasoning should be evidence-grounded, with rationales tied to raw windows, statistics, retrieved events, tool observations, or domain constraints. Second, it should preserve uncertainty, since pattern recognition, forecasting, and causal attribution require different evidentiary standards. Third, it should be process-evaluable, because final accuracy alone cannot show whether the agent used the right segment, calculation, or tool trace (Zhou et al., 2026; Wu et al., 2026b; Su et al., 2025). Reasoning thus serves as the controllable handoff from observed dynamics to planning: supported hypotheses can guide action, while uncertain ones should trigger evidence collection, tool use, reflection, or abstention.

5 Planning and Action: From Forecasting to Autonomous Decision-Making

Planning and action form the operational layer of an agentic time series system: **they determine how temporal evidence is acquired, which computations are executed, how intermediate results are revised, and when a prediction becomes a consequential decision.** This layer is where time series intelligence stops being a single mapping from history to future and becomes a controlled process over data, tools, models, memories, and environments (Cheng et al., 2026a; Zhan et al., 2026; Poikarayil et al., 2026). A reasoning model may describe a trend or justify a forecast in one pass, but a time series agent must choose diagnostics, retrieve or construct context, invoke forecasting or anomaly-detection modules, inspect validation feedback, revise its plan, and, in decision-centric settings, issue actions whose consequences may only appear later (Wu et al., 2026b; Zhao et al., 2025; Garza & Rosillo, 2025; Tao et al., 2026b). Thus, the central object of this layer is not a chain-of-thought transcript, but an executable and auditable control process.

We formalize the planning and action layer as

$$\mathcal{C} = \langle \mathcal{G}, \mathcal{S}, \mathcal{A}, \pi, \mathcal{O}, \mathcal{V} \rangle, \tag{14}$$

where $\mathcal{G}$ is the task goal, $\mathcal{S}$ is the evolving evidence state, $\mathcal{A}$ is the typed action space, π is the planning or action policy, $\mathcal{O}$ is the set of observations returned by tools or environments, and $\mathcal{V}$ denotes verification signals. At step t, the agent selects an action by

$$a_t \sim \pi_\theta(a_t \mid g, s_t, \eta_{<t}), \qquad o_t = \mathrm{Env}(a_t), \qquad s_{t+1} = \mathrm{Update}(s_t, a_t, o_t), \tag{15}$$

where $\eta_{<t}$ denotes the memory and the prior interaction trace. The action a_t may be a statistical diagnostic, a retrieval operation, a model call, a data-selection decision, a code execution step, an inter-agent delegation,

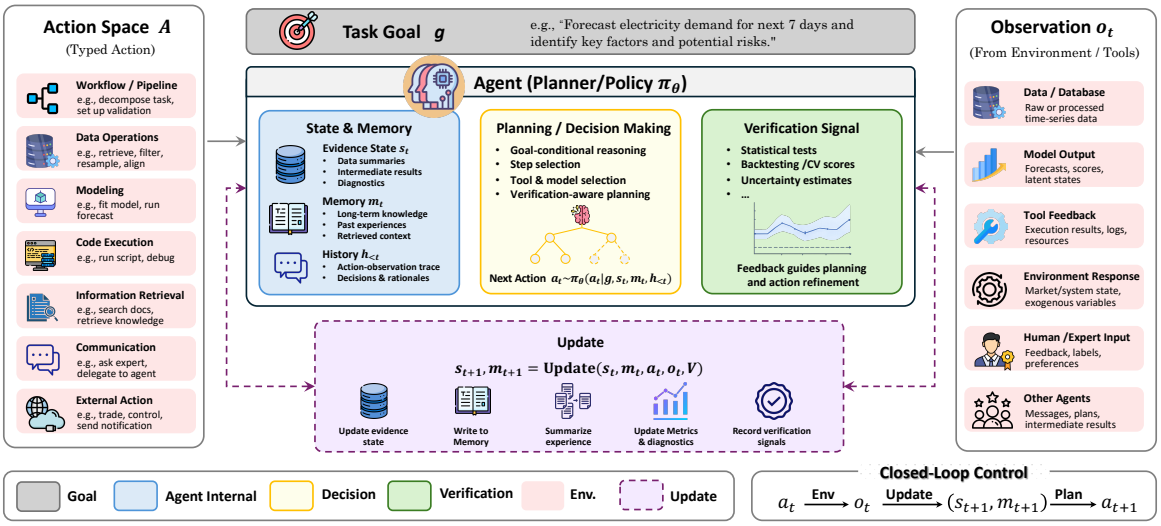

Figure 5: Closed-loop planning and action in agentic time series systems. The agent conditions on the task goal g, evidence state s_t, memory m_t, and history $h_{<t}$ to select typed actions from $\mathcal{A}$ under policy π_θ. Actions invoke workflows, data operations, models, code, retrieval, communication, or external interventions. Observations o_t and verification signals $\mathcal{V}$ update the agent state and memory, enabling iterative, auditable decision-making over time.

a report-generation step, or an external intervention such as a trade or traffic-control action. This view makes planning and action a closed-loop interface: every action changes either the agent's evidence state, the computational pipeline, or the external temporal environment.

5.1 Workflow Planning: Making Forecasting Pipelines Explicit

The most immediate form of planning is **workflow decomposition**. Real time series analysis rarely reduces to a single forecast call: even conventional tasks may require frequency inference, missing-value treatment, seasonal or regime diagnosis (Zhao et al., 2025; Li et al., 2026b; Yang et al., 2025a), feature construction (Li et al., 2026d; Yeh et al., 2025; Lee et al., 2025; Das et al., 2026), model-space pruning (Yang et al., 2025e; Tao et al., 2026c), cross-validation, ensembling, error attribution (Cai et al., 2025b; Kang et al., 2026; Gu et al., 2025), and reporting (Zhan et al., 2026; Zhao et al., 2025). Workflow planning makes these intermediate decisions explicit by translating an underspecified temporal goal into executable stages.

Several systems turn forecasting into a white-box pipeline. TimeSeriesScientist separates curation, planning, forecasting, and reporting; TimeCopilot lets an LLM coordinate feature analysis, model selection, cross-validation, and forecast generation through a unified API; and AutoLFM adapts this pattern to building-load forecasting through retrieval, reasoning, code generation, and executable checking (Zhao et al., 2025; Garza & Rosillo, 2025; Li et al., 2026b). A second line treats planning as data construction: DCATS selects auxiliary traffic series, TimeCAP converts series into event-centered textual descriptions, and Nexus decomposes evidence into macro fluctuations, micro fluctuations, and textual events before prediction (Yeh et al., 2025; Lee et al., 2025; Das et al., 2026). These choices can improve forecasting by constructing useful evidence before model fitting, but they can also introduce leakage or spurious correlations without temporal constraints.

A third line studies iterative workflow refinement. CastFlow learns role-specialized workflows where planning, forecasting, reflection, and memory update repeat; FLAIRR-TS and AlphaCast revise forecasts using retrieved examples, interaction, external evidence, or prior experience; and SEA-TS shifts revision into code through a generate–evaluate–review–improve loop (Pan et al., 2026; Jalori et al., 2025; Zhang et al., 2026a; Xu et al., 2026f). Workflow planning extends beyond forecasting: AD-Agent converts anomaly-detection requests into executable pipelines, CityGPT decomposes urban IoT analysis into temporal, spatial, fusion, and visualization stages, and AION abstracts these patterns into task files, workspaces, validation interfaces, agents, skills, rules, memory, evaluation, and protocols (Yang et al., 2025d; Feng et al., 2025; Zhan et al., 2026).

5.2 Tool Routing and Evidence Acquisition

The second form of action is **tool routing**. Tool use is important for time series because many relevant facts are numerical, statistical, and local in time, while pure language reasoning can hallucinate averages, trends, anomalies, correlations, or future values. Tool-augmented agents reduce this risk by converting intermediate reasoning into executable observations, so that claims about trend, seasonality, anomaly, correlation, or forecast quality can be checked against diagnostics rather than only asserted in language (Wu et al., 2026b; Liu et al., 2025d; Ye et al., 2026; Tao et al., 2026c; Tan et al., 2026). The key question is how a tool call changes the agent's evidence state and what verification signal it returns.

The clearest pattern is *ReAct-style evidence gathering*. TimeART defines a Query–Thought–Action–Observation–Final Answer loop over decomposition, autocorrelation, change-point, anomaly, and lightweight forecasting tools; TS-Agent keeps raw series outside the LLM while operators extract statistics, trends, anomalies, and relations under an evidence log and quality gate; and TS-Reasoner translates natural-language questions into operator programs over model, numerical, retrieval, and custom-generation tools (Wu et al., 2026b; Liu et al., 2025d; Ye et al., 2026). Tool routing also supports retrieval and knowledge grounding: Agentic RAG for Time Series Analysis routes forecasting, anomaly detection, classification, and imputation to specialized sub-agents with task-specific prompt memories; TSAG aligns quantitative tools with financial documents; and Hubble restricts financial alpha discovery to operator trees checked by deterministic evaluators (Ravuru et al., 2024; Kolonin et al., 2026; Shi et al., 2026a).

Anomaly and event-detection agents use tools to *localize and verify abnormal patterns*. AnomaMind follows a coarse-to-fine workflow in which broad evidence acquisition precedes diagnostic tool selection; SAGE assigns point, structural, seasonal, and pattern anomalies to specialized analyzers; and Grammar of the Wave and TS-Debate cross-check symbolic, visual, numerical, and textual evidence (Tao et al., 2026c; Kang et al., 2026; Wan et al., 2026; Trirat et al., 2026). Thus, anomaly-oriented tool routing is evidence arbitration: the agent must choose the diagnostic family, reconcile conflicting signals, and decide when the evidence is sufficient. Domain agents extend tool routing into executable environments, as in Open-TI for traffic simulation and control, CityGPT for urban IoT analysis, AgriWorld for geospatial and crop-simulation tools, and RFID systems for retail supply execution (Da et al., 2024; Feng et al., 2025; Zhang et al., 2026b; Yang et al., 2025g). Once tool outputs become simulated queue lengths, crop-model checks, or constraint violations, planning shifts from evidence acquisition toward closed-loop control.

5.3 Model, Data, and Code Orchestration

The third action family is **orchestration over computational resources**. As statistical baselines, neural models, time series foundation models, external data sources, and domain-specific toolkits proliferate, the bottleneck shifts from having a predictor to deciding which resource to use, when to trust it, and how to audit its artifact. This is broader than ordinary model selection: an orchestrating agent must define the optimization target, choose the search space, expose feedback, and preserve traces for rollback and comparison.

Model orchestration starts from *choosing among forecasting engines*. TimeCopilot treats the LLM as a controller over statistical models, machine-learning models, neural models, time series foundation models, and ensembles, using feature analysis and cross-validation to decide when to escalate model complexity (Garza & Rosillo, 2025). TSOrchestra similarly casts the LLM as a metacognitive controller that evaluates optimization history and decides whether to continue, reweight, or accept the current forecast ensemble (Cao et al., 2025). The agentic action is the allocation of computation across candidate engines under validation feedback.

Data orchestration *changes the intervention target*. DCATS selects auxiliary traffic series from metadata and validation feedback; R&D-Agent-Quant jointly explores data-centric factor construction and modeling choices; and Human-in-the-Loop Adaptive Optimization adds user feedback to post-training forecast correction (Yeh et al., 2025; Li et al., 2026d; Tiomoko et al., 2025). These methods act on the data set, factor space, or correction rule before model parameters, but auxiliary series can leak future information, reflect transient regimes, or encode spurious calendar artifacts.

Code orchestration *turns the agent into a time series ML engineer*. SEA-TS generates forecasting algorithms, evaluates them, performs automated code review, and uses best/worst memories to refine later generations;

AD-Agent selects data-processing steps, libraries, models, and debugging actions for anomaly detection; and TimeSeriesGym evaluates whether agents can edit code, tune pipelines, manage artifacts, and satisfy task-specific validation in realistic codebases (Xu et al., 2026f; Yang et al., 2025d; Cai et al., 2025b). The target expands from prediction error to engineering competence: executability, log interpretation, failure recovery, and resistance to weak validation protocols.

Constrained orchestration provides a safer alternative to arbitrary code generation. Hubble restricts alpha discovery to operator trees evaluated by deterministic pipelines; MONAQ queries neural architectures for resource-constrained time series analysis; AegisTS uses hierarchical reinforcement learning for multivariate cleaning and imputation; and CALM adapts anomaly-detection pipelines over streams (Shi et al., 2026a; Trirat & Lee, 2025; Shi et al., 2026b; Devireddy & Huang, 2025). Narrower action spaces improve reproducibility and safety, but make the exposed operators, reward signals, and validators the bottleneck; open-ended code gives more flexibility, but requires stronger sandboxing, testing, and audit trails.

5.4 Multi-Agent Coordination: Separating Incompatible Responsibilities

Multi-agent coordination is useful when a time series task contains responsibilities that should not be collapsed into a single prompt. Generation, execution, critique, domain translation, and reporting often require different tools, incentives, and stopping criteria. The value of multiple agents is not the number of LLM calls, but the separation of incompatible roles under an explicit communication protocol.

General-purpose systems often *divide perception, reasoning, and execution*. MAS4TS uses Analyzer, Reasoner, and Executor agents, linking VLM-based plot perception to latent trajectory reasoning and task-specific tool routing; TS-Debate uses modality-specific agent debate; MERIT applies coordination to transferable unsupervised representations; and MAFS divides forecasting by horizon-, component-, or pattern-specific responsibilities (Ruan & Liang, 2026; Trirat et al., 2026; Zhou et al., 2025b; Huang et al., 2025).

Anomaly systems specialize by *failure mode and validation role*. Argos separates Detection, Repair, and Review; SAGE decomposes diagnosis into point, structural, seasonal, and pattern analyzers; AnomaMind combines tool-backed evidence construction with RL-optimized anomaly decisions; and ChatAD improves anomaly reasoning through multi-turn instruction evolution (Gu et al., 2025; Kang et al., 2026; Tao et al., 2026c; Sun et al., 2026). Across these systems, one role proposes, another executes or repairs, and a third validates or adjudicates, reducing reliance on a single visual or statistical cue.

Semantic, multimodal, and domain-facing systems specialize by *interpretation role*. TESSA separates cross-domain from domain-specific annotation and review; TimeCAP separates event contextualization from prediction; ZARA and FETA use evidence-grounded or in-context agent pipelines for classification; and weather systems use role separation across temporal scales, captions, and cross-modal decoding (Lin et al., 2026; Lee et al., 2025; Li et al., 2026e; Sui et al., 2025; Sukhorukov et al., 2026; Liu, 2026; Wu et al., 2026a).

Decision systems add analysts, reviewers, and risk managers because their outputs may change the environment. TradingAgents simulates a trading firm with market, news, sentiment, technical, bull/bear, trader, and risk-management roles (Xiao et al., 2024). R&D-Agent-Quant, ATLAS, FinCon, FinMem, FinArena, and FinHEAR preserve this financial multi-agent pattern while varying the adaptation mechanism, including prompt optimization, verbal reinforcement, layered memory, human-agent collaboration, and risk-aware expertise (Li et al., 2026d; Papadakis et al., 2025; Yu et al., 2024; 2025b; Xu et al., 2025c; Chen et al., 2025b). CityGPT and ColaCare apply the same principle beyond finance by aligning agents with urban-IoT or EHR data modalities and decision roles (Feng et al., 2025; Wang et al., 2025b). The main risk is communication overhead: inconsistent intermediate states can accumulate, so multi-agent systems require stronger state management than single-agent pipelines.

5.5 Sequential Decision-Making and External Action

The most agentic form of planning is **sequential decision-making**. Here the output is not merely a forecast or explanation, but an action in a process with state, delayed feedback, and potential external consequences.

Cast-R1 bridges forecasting and sequential decision-making: the agent maintains a decision state, chooses actions such as analysis, feature extraction, model invocation, forecast update, and revision, and learns

long-horizon policies through supervised and reinforcement training (Tao et al., 2026b). ARTIST applies a related policy view to time series QA, where a controller sequentially selects informative time segments before the reasoner answers (Messica et al., 2026). VeriTime, COUNTS, TimeOmni-1, and Time-R1 do not always execute external tools, but they show that RL and process-verifiable rewards can train longer temporal reasoning trajectories (Zhou et al., 2026; Parker et al., 2025; Guan et al., 2025; Luo et al., 2025b). This line connects slow-thinking reasoning to action policies: the model learns not only what answer to give, but what evidence-selection path to follow.

Financial systems make the transition from prediction to external action concrete. ATLAS uses an order-aware action space and adapts prompts from delayed market feedback; FinMem emphasizes layered memory for trading decisions; FinCon abstracts prior episodes into reusable investment concepts with risk control; and CryptoTrade, Hi-DARTS, Trading-R1, and ElliottAgents extend this action view through reflection, hierarchical reinforcement trading, RL-trained investment reasoning, and natural-language stock-market agents (Papadakis et al., 2025; Yu et al., 2025b; 2024; Li et al., 2024; Sagong et al., 2025; Xiao et al., 2025b; Chudziak & Wawer, 2025). Agentic Trading surveys this design space and stresses that execution, risk, adaptation, and reproducibility must be evaluated together (Xia et al., 2026).

Sequential action is broader than trading. Open-TI connects natural-language traffic requirements to map retrieval, simulation, and signal-control actions (Da et al., 2024). StockAgent and RFID model stock trading, retail demand control, or ensemble weighting as MDP/Dec-POMDP-style decision processes (Zhang et al., 2025b; Yang et al., 2025g). SocioDojo extends the interaction setting to lifelong socio-economic analysis agents that combine text, time series, tools, and memory across tasks (Cheng & Chin, 2024). Across these domains, delayed feedback and irreversible decisions make action quality, calibration, and safety more important than forecasting error alone.

5.6 Evidence Map and Design Implications

Table 6 summarizes representative planning and action mechanisms. It is organized by action space rather than application domain, because the same design question recurs across forecasting, anomaly detection, trading, traffic, agriculture, and urban IoT: **what can the agent do, what feedback does it observe, and how is the action verified?**

Several design implications follow from this evidence map. First, the action space of a time series agent must be **typed**. Actions such as "compute autocorrelation", "fit a baseline", "retrieve weather covariates", "generate code", and "place an order" have different risk profiles and should not be represented as undifferentiated text. Second, planning should be **diagnosis-aware**. A planner that ignores sampling frequency, missingness, seasonality, multivariate dependence, horizon length, and operational constraints is unlikely to produce reliable workflows. Third, every action should leave an **evidence trace**. Time series agents are often deployed in domains where users need to know not only the final forecast, but also which diagnostics, data, tools, and validation checks supported it. Fourth, **verification** must be part of the action loop rather than a post-hoc evaluation step. Lots of the systems all point toward the same principle: generation must be paired with *executable checking*.

Planning and action also define the *boundary* between time series reasoning models and time series agents. A system that only emits a forecast or explanation remains a reasoning or prediction model. A system that decomposes temporal goals, executes tools, updates state from observations, revises intermediate decisions, coordinates specialized agents, and acts under validation and risk constraints is better viewed as an agentic time series system. This distinction motivates the rest of the taxonomy: memory determines what experience can be reused, world models determine what environments can be simulated, and verification determines whether actions are trustworthy enough for deployment.

6 Memory and Knowledge: Accumulating Temporal Experience

Memory is the layer that determines whether a time series agent accumulates experience or merely reacts to the present. Unlike a conventional forecasting pipeline whose parameters encode past experience once, an agent must decide what to store, retrieve, revise, forget, and reuse as distributions shift. Recent work

Table 6: Representative planning and action mechanisms in agentic time series systems.

Category	Action space	Representative systems	Feedback and limitation
Workflow planning	Preprocessing, feature engineering, model selection, validation, reporting	TimeSeriesScientist (Zhao et al., 2025), AutoLFM (Li et al., 2026b), AION (Zhan et al., 2026), CastFlow (Pan et al., 2026)	Validation metrics and report traces improve auditability, but planning quality depends strongly on the exposed tool and model space.
Tool routing	Statistical diagnostics, decomposition, anomaly tools, forecasting tools, external-data operators	TimeART (Wu et al., 2026b), TS-Agent (Liu et al., 2025d), TS-Reasoner (Ye et al., 2026), AnomaMind (Tao et al., 2026c), AgriWorld (Zhang et al., 2026b)	Tool observations ground reasoning and reduce hallucination, but tool selection errors and tool-output misuse remain central failure modes.
Model and data orchestration	TSFM routing, statistical baselines, ensembles, auxiliary-series selection, context integration	TimeCopilot (Garza & Rosillo, 2025), DCATS (Yeh et al., 2025), MoiraiAgent (Liu et al., 2025a), Nexus (Das et al., 2026)	Cross-validation and benchmark scores guide selection, but cost-performance trade-offs and regime shift are still difficult.
Code and formula generation	Pipeline code, forecasting architecture code, anomaly-detection scripts, alpha-factor formulas	SEA-TS (Xu et al., 2026f), AD-Agent (Yang et al., 2025d), Hubble (Shi et al., 2026a), TS-Agent (financial) (Ang et al., 2025)	Sandboxes and deterministic evaluators improve reliability, but generated artifacts can overfit validation environments or exploit weak metrics.
Multi-agent coordination	Delegation among analyzers, reasoners, executors, reviewers, risk managers, and reporters	MAS4TS (Ruan & Liang, 2026), Argos (Gu et al., 2025), TimeCAP (Lee et al., 2025), R&D-Agent (Yang et al., 2025e), CityGPT (Feng et al., 2025)	Role separation improves interpretability and control, but communication overhead and inconsistent intermediate states can accumulate.
External environment action	Trading orders, traffic control, energy trading, crop-simulation intervention, inventory or demand-control actions	Cast-R1 (Tao et al., 2026b), TradingAgents (Xiao et al., 2024), ATLAS (Papadakis et al., 2025), Open-TI (Da et al., 2024), AgriWorld (Zhang et al., 2026b)	Delayed feedback and irreversible actions make verification and risk constraints essential; offline forecasting metrics are insufficient.

therefore treats structured memory as a core module beside perception, reasoning, and planning (Cheng et al., 2026a; Xu et al., 2026b; Zhang et al., 2025c). The central claim of this section is that *stuffing a longer context window is not memory*: without selection, indexing, updating, forgetting, and reuse, past observations remain only a log.

6.1 Beyond Conversation History: Why Time Series Agents Need Structured Memory

NLP agent memory is usually textual, relatively stable, and judged by factual recall. Time series memory differs in five ways that make direct transfer insufficient.

Tab. 7 summarizes the key distinctions: memory entries may expire under regime shift, must be indexed across temporal scales, receive delayed feedback, preserve statistical structure, and remain numerically verifiable rather than merely textually plausible.

To formalize these requirements, we define the memory layer of a time series agent as a six-tuple

$$\mathcal{M} = \langle \mathcal{M}_{\text{ctx}}, \mathcal{M}_{\text{epi}}, \mathcal{M}_{\text{temp}}, \mathcal{M}_{\text{know}}, \mathcal{M}_{\text{proc}}, \mathcal{M}_{\text{conf}} \rangle, \tag{16}$$

where each component supports write$(\cdot)$, read$(\cdot)$, update$(\cdot)$, and forget$(\cdot)$. Context memory stores short-term task evidence; episodic memory stores complete cases for analogy; temporal memory stores regime-, season-, and rare-event patterns; knowledge memory stores rules, manuals, and causal structure; procedural memory stores strategies and workflows; and confidence memory stores failures, uncertainty, and calibration. This taxonomy extends prior memory discussions (Cheng et al., 2026a; Zhang et al., 2025c; Xia et al., 2026) by making temporal and confidence memory first-class subtypes.

Table 7: Why time series agents require specialized memory designs: five fundamental dimensions that distinguish TS-agent memory from NLP-agent memory.

Dimension	NLP Agents	Time Series Agents
Non-stationarity	Relatively stable knowledge base; facts update infrequently	Continuous distribution drift; patterns valid under one regime may be harmful under another
Multi-scale dependency	Single granularity; conversation turns are uniform	Second-level, hourly, seasonal patterns coexist at different resolutions
Delayed feedback	Immediate evaluation; tool output confirms/refutes action	Consequences appear hours/days later; credit assignment spans time steps
Statistical structure	Text fragments; retrieval by semantic similarity	Carries trend, seasonality, changepoints; retrieval must respect numerical structure
Numerical grounding	Facts accepted/rejected by language reasoning	Patterns must be verifiable by statistical tools, not textual plausibility

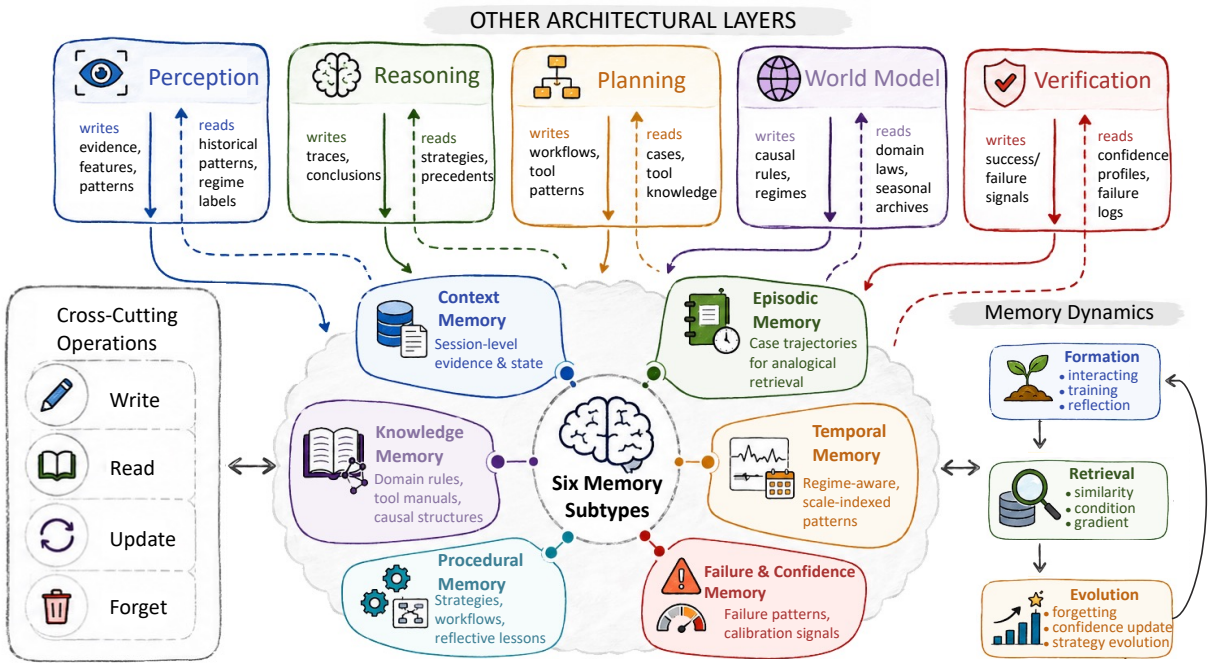

Figure 6: Architecture of the memory layer in agentic time series systems. The six memory subtypes, context, episodic, temporal, knowledge, procedural, and failure/confidence, each support four cross-cutting operations (write, read, update, forget). The upper row shows the five surrounding architectural layers (perception, reasoning, planning, world model, verification) and their bidirectional data flows with the memory subtypes: perception writes structured evidence into context and episodic memory; reasoning writes traces and conclusions into procedural and confidence memory; planning writes workflow patterns into procedural memory; the world model writes causal rules and regime information into knowledge and temporal memory; verification writes success and failure signals into confidence memory. The lower row shows the three phases of memory dynamics, formation, retrieval, and evolution, which operate across all subtypes in a closed loop.

Memory is coupled to the whole agent stack. Perception writes statistical features and anomaly labels; reasoning writes traces and reflections; planning writes workflows; the world model writes causal and regime information; and verification writes success or failure signals. In return, retrieved cases, strategies, domain knowledge, temporal patterns, and confidence profiles guide perception, reasoning, planning, modeling, and verification. Fig. 6 illustrates these flows, the six subtypes, and the formation–retrieval–evolution cycle.

The following subsections review each subtype, then synthesize memory dynamics and open challenges.

6.2 Context and Working Memory

Context memory is short-lived task evidence: tool outputs, reasoning steps, decision states, and messages kept within a session. It appears as evidence logs in TS-Agent and TS-Reasoner (Liu et al., 2025d; Ye et al., 2026), state buffers in Cast-R1 and AIVV (Tao et al., 2026b; Kwon et al., 2026), shared working memory in AD-AGENT and VisualTimeAnomaly (Yang et al., 2025d; Xu et al., 2026g), and session persistence in PatchTST-based analysis, AI-Meteorologist, and AnomaMind (Buratto et al., 2026; Sukhorukov et al., 2026; Tao et al., 2026c). Its weakness is unchanged: unless entries are selected, compressed, or promoted, context memory cannot handle regime shifts, rare events, or cross-session reuse.

6.3 Episodic and Case-Based Memory

Episodic memory stores complete cases, including evidence, reasoning, and outcomes, so that new tasks can retrieve comparable precedents. MemCast, financial TS-Agent, and Argos retrieve past patterns, methodologies, or contrastive samples by statistical relevance rather than text alone (Tao et al., 2026a; Ang et al., 2025; Gu et al., 2025). Other systems build augmented episodes: SAGE synthesizes anomaly references, MERIT stores reviewed positive views, and MarsTSC maintains tiered multimodal knowledge (Kang et al., 2026; Zhou et al., 2025b; Li et al., 2026a). RationaleTS and FinSrag retrieve rationales or historical segments (Liu et al., 2026b; Xiao et al., 2025a), while FinHEAR and cross-source decoding connect cases with expert precedents and transferable annotations (Chen et al., 2025b; Lin et al., 2026). The key design problem is similarity: text embeddings may miss numerical structure, while purely numerical matches may miss semantic intent.

6.4 Long-Horizon Temporal and Regime Memory

Temporal memory stores regime-specific, season-aware, and rare-event patterns by time scale. It must support regime detection, temporal decay, and rare-event recall. FinMem uses scale-dependent forgetting, MemCast separates concrete historical patterns from reasoning wisdom and general laws, SocioDojo combines sliding-window and lifelong stores, CALM curates verified drift data, and AgriWorld records full interaction trajectories for recurrent agricultural conditions (Yu et al., 2025b; Tao et al., 2026a; Cheng & Chin, 2024; Devireddy & Huang, 2025; Zhang et al., 2026b). TimeOmni-1 encodes temporal priors through training, while CryptoTrade and TradingAgents preserve reflections or debate histories for later market decisions (Guan et al., 2025; Li et al., 2024; Xiao et al., 2024). The defining requirement is regime awareness: a pattern may be correct in one regime and harmful in another, yet only a few systems explicitly tie forgetting or adaptation to distribution change.

6.5 Knowledge Memory

Knowledge memory stores domain rules, manuals, causal structures, and other task-general information. Structured stores appear in AutoLFM and R&D-Agent-Quant (Li et al., 2026b;d); graph-based forms appear in ChatKG, ElliottAgents, and its Neo4j-backed extension (Christino & Paulovich, 2024; Chudziak & Wawer, 2025; Wawer & Chudziak, 2025); diagnostic stores appear in AnomaMind and ZARA (Tao et al., 2026c; Li et al., 2026e). RAG systems retrieve external pattern knowledge at inference time, as in agentic RAG, TS-RAG, and TimeRAG (Ravuru et al., 2024; Ning et al., 2025b; Park, 2026). Parameterized memory instead internalizes knowledge through pattern banks or latent prototypes, as in TS-Memory and MEMTS (Lyu et al., 2026; Yu et al., 2026b). The trade-off is update flexibility versus retrieval cost and precision.

6.6 Procedural and Reflective Memory

Procedural and reflective memory store what to do next: strategies, workflows, code changes, and lessons about success or failure. MemCast distills reasoning wisdom and general laws (Tao et al., 2026a); CastFlow, SEA-TS, and TS-Agent store workflows, code-review corrections, or refinement heuristics (Pan et al., 2026; Xu et al., 2026f; Ang et al., 2025). FinCon, MarsTSC, and ATLAS generalize experience across episodes through verbal reinforcement, self-evolving knowledge, or prompt optimization histories (Yu et al., 2024; Li et al., 2026a; Papadakis et al., 2025). CryptoTrade, TradingAgents, ChatAD, and StockAgent preserve

reflection, debate, evolved instructions, or expert-weight feedback (Li et al., 2024; Xiao et al., 2024; Sun et al., 2026; Zhang et al., 2025b). T3LLM internalizes corrected reasoning traces, SocioDojo stores reusable analysis strategies, and TSOrchestra or autonomous LLM-agent systems keep audit trails and session histories (Su et al., 2025; Cheng & Chin, 2024; Cao et al., 2025; Buratto et al., 2026). Recent position and trading-agent surveys frame such memory as essential for selective recall and cross-session strategy improvement (Cheng et al., 2026a; Xia et al., 2026).

6.7 Failure and Confidence Memory

Failure memory records bad strategies, overconfident predictions, and misleading patterns; confidence memory records how much each stored pattern should be trusted. Hubble is the clearest explicit design, with separate success and failure factor libraries (Shi et al., 2026a). MemCast calibrates entry confidence, SEA-TS stores code-review failures, and Nexus learns calibration rules through gated backtesting (Tao et al., 2026a; Xu et al., 2026f; Das et al., 2026). FinHEAR, CryptoTrade, and CALM add partial failure awareness through outcome-based precedent refinement, strategy reflection, and LLM-judged drift curation (Chen et al., 2025b; Li et al., 2024; Devireddy & Huang, 2025). The gap remains large: most systems remember successes more explicitly than failures, limiting reliability in high-stakes domains.

6.8 Memory Dynamics: Formation, Retrieval, and Evolution

Across subtypes, memory quality depends on formation, retrieval, and evolution (see Fig. 6).

Memory Formation. Memories are formed through interaction, training, and reflection: examples include TS-Agent logs, Cast-R1 states, AgriWorld trajectories, TS-Memory pattern banks, TS-RAG retrieval bases, MEMTS latent prototypes, FinCon conceptual reinforcement, and CryptoTrade reflection (Liu et al., 2025d; Tao et al., 2026b; Zhang et al., 2026b; Lyu et al., 2026; Ning et al., 2025b; Yu et al., 2026b; 2024; Li et al., 2024).

Memory Retrieval. Retrieval varies by metric and architecture: embedding search in CastFlow and FAISS-based agents (Pan et al., 2026; Buratto et al., 2026), DTW in SAGE (Kang et al., 2026), time-series encoder distance in TS-RAG (Ning et al., 2025b), hybrid statistical-semantic retrieval in RationaleTS (Liu et al., 2026b), conditional retrieval in MemCast and FinMem (Tao et al., 2026a; Yu et al., 2025b), and differentiable prompt-pool retrieval in agentic RAG (Ravuru et al., 2024).

Memory Evolution. Evolution updates memory under drift and obsolescence. FinMem applies layer-specific forgetting, AIVV and SocioDojo use sliding windows, MemCast downweights low-utility entries, ATLAS and SEA-TS refine strategies, and SocioDojo or TS-Agent support cross-session transfer (Yu et al., 2025b; Kwon et al., 2026; Cheng & Chin, 2024; Tao et al., 2026a; Papadakis et al., 2025; Xu et al., 2026f; Ang et al., 2025). Overall, formation and retrieval are more common than evolution; MemCast is unusual in spanning multiple subtypes, while recent surveys still identify layered memory as underdeveloped (Cheng et al., 2026a; Zhang et al., 2025c).

6.9 Evidence Map and Open Challenges

Tab. 8 summarizes the six memory subtypes and representative implementations. Several patterns emerge. Context memory is common but shallow; episodic memory is useful when case-based reasoning matches the task; temporal memory is distinctive but under-implemented; knowledge memory is dominated by RAG; procedural memory lacks evaluation of cross-session improvement; and failure/confidence memory is the largest gap, with Hubble as the rare explicit success/failure design. These patterns imply eight open challenges: **Memory Evaluation** should measure formation, retrieval, and evolution, not only downstream accuracy. **Regime-Aware Memory** needs regime labels, shift monitoring, and retrieval weights that change when old patterns become unsafe. **Cross-Session Memory Reuse** remains open beyond exceptions such as SocioDojo and TS-Agent (Ang et al., 2025). **Memory-Reasoning Alignment** requires retrieved entries to match reasoning intent, not just statistical similarity; RationaleTS and MemCast offer partial solutions (Liu et al.,

Table 8: Summary of memory subtypes in agentic time series systems: characteristics and representative systems.

Memory subtype	Characteristics	Representative systems
Context	Short-term task evidence, tool outputs, reasoning traces within a session; typically implemented as session logs or evidence buffers without explicit selection, compression, or forgetting	**TS-Agent** (Liu et al., 2025d); **Cast-R1** (Tao et al., 2026b); **AD-AGENT** (Yang et al., 2025d); **AIVV** (Kwon et al., 2026)
Episodic	Complete case trajectories for analogical retrieval; case-based reasoning and pattern matching; similarity metrics must respect statistical properties rather than relying solely on text embeddings	**MemCast** (Tao et al., 2026a); **SAGE** (Kang et al., 2026); **Argos** (Gu et al., 2025); **FinSrag** (Xiao et al., 2025a); **RationaleTS** (Liu et al., 2026b)
Temporal	Regime-aware, scale-indexed, and season-aware patterns indexed by time scale; requires regime detection, temporal decay, and rare-event recall, the subtype most distinctive to time series agents	**FinMem** (Yu et al., 2025b); **MemCast** (Tao et al., 2026a); **CALM** (Devireddy & Huang, 2025); **SocioDojo** (Cheng & Chin, 2024); **AgriWorld** (Zhang et al., 2026b)
Knowledge	Domain rules, tool manuals, and causal structures delivered via RAG, knowledge graphs, or structured knowledge bases; also includes parameterized memory that internalizes domain knowledge into model weights	**AutoLFM** (Li et al., 2026b); **ChatKG** (Christino & Paulovich, 2024); **ElliottAgents** (Chudziak & Wawer, 2025); **TS-Memory** (Lyu et al., 2026); **MEMTS** (Yu et al., 2026b)
Procedural	Strategies, workflows, code modifications, and reflective conclusions about what works; stores "how to do" rather than "what happened"; supports cross-session improvement through strategy evolution and prompt refinement	**MemCast** (Tao et al., 2026a); **CastFlow** (Pan et al., 2026); **SEA-TS** (Xu et al., 2026f); **FinCon** (Yu et al., 2024); **ATLAS** (Papadakis et al., 2025); **T3LLM** (Su et al., 2025)
Failure	Failure patterns, uncertainty estimates, and confidence calibration signals; the most scarce but most consequential subtype; without it, agents repeat mistakes and cannot distinguish high- from low-confidence predictions	**Hubble** (Shi et al., 2026a); **MemCast** (Tao et al., 2026a); **Nexus** (Das et al., 2026); **SEA-TS** (Xu et al., 2026f)

2026b; Tao et al., 2026a). **Memory Cost-Efficiency** requires systematic study of latency, storage, update, and accuracy trade-offs between external and parameterized memory. **Multi-Agent Shared Memory** remains unresolved because AD-AGENT and VisualTimeAnomaly show shared stores (Yang et al., 2025d; Xu et al., 2026g) but not concurrent consistency. **Memory Conflict Resolution** requires explicit rules for contradictory memories instead of delegating numerical conflicts to parametric reasoning. **Cross-Domain Memory Transfer** remains untested for strategies, patterns, and failures across domains.

Together, these challenges reinforce the section's main claim: a log is not memory unless it supports structured selection, indexing, updating, forgetting, and reuse. Most current systems stop at context memory or simple episodic retrieval, leaving temporal, procedural, and failure memory underdeveloped. Closing the loop from verification feedback to confidence and failure memory is therefore necessary for reliable long-lived time series agents.

7 Time Series World Models: Simulating Dynamic Temporal Environments

Time series world modeling concerns how temporal systems move from representing observed sequences toward modeling the dynamic environments that generate them. Conventional time series models are usually optimized for task-specific outputs, such as predictions, classifications, risk scores, or decisions. World modeling asks a different question: what is the temporal environment being modeled, how may it evolve, and how would its evolution change under different conditions? This distinction matters for agentic time series systems because an agent often needs to reason about consequences, interventions, and alternative futures rather than merely produce an output for the current task.

Table 9: Current routes toward time series world models. Each route exposes a different interface for representing, simulating, comparing, or acting on temporal environments.

Route	World-model interface	Representative systems	Core capability and limitation
Environment understanding	Structured state over temporal covariates, dependencies, and domain context.	Augur (Cui et al., 2025).	Organizes the environment for relational reasoning, but inferred dependencies do not establish controllable dynamics or intervention-level causality.
Temporal simulation	Probabilistic rollout from history or generation conditioned on textual descriptions and semantic prototypes.	Chronicle (Quinlan et al., 2026); BRIDGE (Li et al., 2025).	Produces plausible or controllable futures, but the mechanism connecting conditions to temporal transitions may remain implicit.
Counterfactual simulation	Baseline and alternative trajectories under factual or hypothetical future conditions.	Counterfactual Time Series Forecasting (Gu et al., 2026).	Supports what-if comparison, but alternative futures are unobserved and textual conditions do not identify the causal pathway.
Executable grounding	Tool- or simulator-defined environment with callable actions, returned artifacts, and verification signals.	AgriWorld (Zhang et al., 2026b).	Enables intervention and execution-based verification, but requires costly domain engineering and remains difficult to scale.

Let $x_{1:t}$ and $c_{1:t}$ denote, respectively, the observed temporal evidence and the relevant context available up to time t (as in Sec. 2.3). A world model maps this information into a state representation

$$z_t = \mathrm{Enc}_\psi(x_{1:t}, c_{1:t}), \tag{17}$$

and characterizes how this state evolves under condition-changing factors:

$$z_{t+1} \sim p_\theta(\cdot \mid z_t, \xi_t). \tag{18}$$

Here z_t denotes the system's representation of the temporal environment at time t, which may be latent, structured, symbolic, observation-based, or simulator-defined. The variable ξ_t denotes a factor that affects the environment's evolution, such as an agent action a_t, an external event, or a contextual condition, depending on the setting. The purpose of z_t is not necessarily to reconstruct raw observations, but to provide a state representation from which the system can reason about how the temporal environment may evolve under different conditions.

Recent visual and embodied world models provide a useful reference point for this formulation. Genie-style generative environments and newer world foundation models show how learned representations can support action-controllable or physically grounded simulation in visual domains (Bruce et al., 2024; NVIDIA et al., 2025; Team et al., 2025). Predictive-representation models such as V-JEPA 2 further suggest that world modeling need not be tied to reconstructing observations directly; useful dynamics can also be learned in abstract representation spaces (Assran et al., 2025). Time series world modeling shares this general logic of state representation, transition, and imagined rollout, but the modeled environment is different. Rather than visual scenes, robotic settings, or interactive 3D worlds, the target is often a partially observed temporal system shaped by multivariate dependencies, exogenous conditions, domain constraints, and possible interventions.

This difference motivates a distinct set of time series world-modeling directions. Current studies approach the objective through environment understanding (Cui et al., 2025), temporal simulation via multimodal forecasting or controlled generation (Quinlan et al., 2026; Li et al., 2025), counterfactual reasoning under changed conditions (Gu et al., 2026), and executable grounding in constructed domain environments (Zhang et al., 2026b). The following discussion synthesizes these directions and clarifies how existing work contributes to time series world modeling.

Table 9 summarizes these complementary routes. Environment understanding structures the state, temporal simulation rolls it forward, counterfactual simulation compares changed conditions, and executable grounding exposes environment transitions to agent actions and verification.

7.1 Environment Understanding: Structuring Temporal Dependencies

Environment understanding concerns what must be represented for a temporal environment to be modeled. A time series is not only an ordered sequence of values; it may also reflect interactions among variables, delayed effects, external events, and domain constraints. For a temporal world model, such a structure is important because future evolution is shaped not only by temporal continuation but also by how components of the environment influence one another.

Augur (Cui et al., 2025) represents this direction. It uses large language models to help model directed associations among temporal covariates. Rather than treating covariates as flat auxiliary inputs, Augur organizes them into relational structures that make the environment state more explicit. Its value for world modeling lies in giving this state an internal organization: possible futures are shaped not only by the continuation of past values, but also by how components of the environment are related. At the same time, these inferred relations should be treated as modeled dependencies or causal hypotheses rather than fully validated mechanisms. Thus, Augur illustrates how relational environment understanding can enrich temporal world states and provide constraints for later reasoning about temporal evolution.

7.2 Temporal Simulation: From Forecasting to Controlled Generation

Temporal simulation concerns how a modeled temporal environment can be rolled forward from available evidence. Here, simulation does not require a complete physical simulator; it refers more broadly to producing future trajectories that are consistent with observed history, contextual information, or specified semantic conditions. Forecasting becomes relevant to world modeling when it is interpreted as such a rollout rather than only as a task-specific output. Existing methods differ in how explicitly they expose this process: forecasting-oriented models extend observed histories under uncertainty, whereas controlled generation synthesizes trajectories according to semantic constraints.

Chronicle (Quinlan et al., 2026) represents a forecasting-oriented route. It learns a shared modeling space for language and time series and uses this shared backbone for autoregressive time series forecasting. Given historical temporal evidence, Chronicle can predict future time series patches and produce probabilistic forecasts through quantile outputs. This combination of multimodal context and probabilistic rollout allows the model to use numerical history and available textual context to estimate how a temporal process may continue under uncertainty.

However, this rollout remains different from explicit intervention-based world modeling. Chronicle can incorporate textual information into temporal modeling, but such text mainly functions as contextual evidence rather than as an explicitly modeled event, action, or intervention. The model therefore approximates future trajectories from multimodal evidence, rather than learning how a specified condition-changing factor ξ_t transforms a world state from z_t to z_{t+1}.

BRIDGE (Li et al., 2025) represents a more explicitly generative route. It uses textual descriptions together with semantic prototypes to control time series generation, connecting natural-language conditions and pattern-level temporal representations with generated trajectories. Its contribution is not only that text can guide generation, but also that semantic prototypes provide reusable pattern-level structures for producing time series under different descriptions.

Together, Chronicle and BRIDGE show how temporal simulation can be approximated without a complete physical simulator. Chronicle rolls forward from observed histories under uncertainty, while BRIDGE generates trajectories under semantic conditions. The common difficulty is that plausibility and controllability do not automatically identify the mechanism of evolution. A generated trajectory may be consistent with the observed history or satisfy a textual condition, but the link between that condition and the underlying temporal dynamics can remain implicit. This leaves an important gap for world modeling: the model may

approximate how a process could continue, while still lacking a grounded account of why the environment would evolve in that particular way.

7.3 Counterfactual Simulation: Alternative Futures Under Changed Conditions

This direction extends temporal simulation from producing possible futures to comparing how futures diverge when conditions change. Forecasting-oriented rollout estimates how a process may continue from observed history, and controlled generation can synthesize trajectories under specified descriptions. The additional step here is contrastive: the model must relate a baseline evolution to an alternative one and estimate what difference the changed condition makes. This contrastive structure makes counterfactual simulation closer to causal reasoning, because the question is no longer only whether a future is plausible, but how a change in condition is associated with a change in future evolution.

Counterfactual time series forecasting (Gu et al., 2026) provides a direct formulation of this problem by using natural language to specify factual or hypothetical future conditions. The model takes historical observations together with historical and future textual conditions, and generates a future trajectory that both preserves the intrinsic characteristics of the past sequence and reflects the external influence described by the future text. Compared with binary or categorical interventions, textual conditions can express more complex real-world events and scenarios; compared with ordinary controlled generation, the goal is not only to satisfy a condition, but to estimate how the future would differ when that condition changes. This makes the task a useful approximation to causal what-if reasoning in temporal environments.

The main limitation is that this remains an approximation rather than a fully identified causal model. The text may describe a what-if event, but it does not necessarily specify the mechanism or pathway through which the event affects the temporal process. Because the alternative future is usually unobserved, evaluation must rely on indirect signals such as semantic consistency, attribution quality, or domain constraints. Thus, textual counterfactual forecasting moves time series modeling toward causal world-modeling questions, while leaving grounding and validation as open challenges.

7.4 Executable Grounding: Tool-Exposed World-Model Interfaces

Executable grounding refers to executable domain environments that can be used by agents as temporal world models. The preceding subsections focus on learned approximations of temporal dependencies, future rollout, and counterfactual evolution. Here the emphasis is different: the temporal environment is organized outside the language model as a computational substrate, and the agent interacts with it through executable interfaces. In this setting, the world model is valuable not only because it encodes domain structure, but also because an agent can use it: domain data, executable procedures, returned artifacts, and verification results become part of the reasoning loop.

AgriWorld (Zhang et al., 2026b) represents this route. It provides a Python-based agricultural world environment that integrates heterogeneous agricultural data, remote-sensing time series analysis, crop simulation, task-specific prediction, and verification within a unified world–tools–protocol framework. In this formulation, AgriWorld is the constructed agricultural environment, while its tools serve as the executable access layer through which the agent can operate on that environment. The Agro-Reflective agent uses this layer to issue code-based operations, observe returned artifacts, and refine its analysis through execution feedback. This separation is central to executable grounding: the agent can reason with a transparent and monitorable world substrate, while intermediate claims remain traceable to domain procedures and verification signals.

At the same time, this route is difficult to reproduce and scale. Constructing such a world requires substantial domain engineering, because the relevant data sources, domain procedures, simulators, predictors, and verification rules must be explicitly defined and integrated. When an important process cannot be captured by well-defined tools or a reliable execution environment, the constructed world model has limited ability to extend to that process. Thus, AgriWorld demonstrates an alternative route to time series world modeling: instead of learning the whole temporal environment as a unified transition model, it constructs an executable domain world that an agent can use for grounded reasoning. Its strength comes from the transparency and

operability of this constructed world, while its main bottleneck lies in the cost of building, maintaining, and scaling such an environment.

7.5 Synthesis and Outlook

Taken together, these studies suggest that time series world modeling is not a fixed model family, but an emerging capability layer that extends temporal modeling toward environment-level reasoning. Current approaches develop this capability from different directions, including structured environment representation, temporal simulation, counterfactual comparison, and executable grounding. However, each direction still faces its own constraints: relational methods can support forecasting but do not by themselves provide fully controllable rollout, simulation methods may leave dynamics implicit, counterfactual methods are difficult to validate, and executable environments are costly to scale. A natural next step is to combine these capabilities, rather than treating them as separate routes, so that temporal world models can better represent, simulate, compare, and verify possible evolutions of dynamic systems.

8 Reliability and Trustworthiness in Agentic Time Series Systems

Reliability is not a downstream add-on for agentic time series systems; it is a design constraint on how an agent observes temporal evidence, reasons under uncertainty, invokes tools, updates memory, and acts under delayed feedback (Cheng et al., 2026a; Ning et al., 2025a; Dai et al., 2026a). Compared with conventional forecasters, agentic systems have a wider failure surface: they may select the wrong tool, fabricate an intermediate statistic, rely on stale memory, leak private context, overreact to a transient shock, or turn an uncertain prediction into an unsafe action (Liu et al., 2025d; Debenedetti et al., 2024; Shi et al., 2026a; Xia et al., 2026). We call this risk *failure propagation*: an interpretation error can trigger an invalid tool call; the tool output can become unsupported evidence; the evidence can then justify an action whose consequences are observed only later (Wu et al., 2026b; Tao et al., 2026b; Zhan et al., 2026).

We organize reliability as a stack of contracts rather than a flat metric list. Forecasting quality is the base condition. Reasoning correctness, tool-use reliability, and grounding determine whether intermediate claims are supported. Robustness and security determine whether the system tolerates shift, adversarial inputs, and poisoned context. Decision safety determines whether predictions can be converted into constrained actions. Human alignment and auditability determine whether the episode can be inspected, corrected, and reproduced. Figure 7 illustrates the loop, and Tab. 10 gives a compact checklist.

8.1 Forecasting, Reasoning, and Tools

The first contract is predictive: autonomy should not hide weak forecasting, poor calibration, leakage, or brittle horizon behavior. GIFT-Eval, FLAIRR-TS, TimeSeriesGym, and sample-scenario evaluation broaden assessment from point error toward contamination control, retrieval-refinement workflows, executable ML artifacts, and deployment-relevant temporal scenarios (Aksu et al., 2024; Jalori et al., 2025; Cai et al., 2025b; Dai et al., 2026b). Still, accurate forecasts do not certify temporal understanding. TemporalBench and TFRBench show that event-informed reasoning, rationale quality, and contextual consistency must be checked separately from numerical accuracy (Weng et al., 2026; Ahamed et al., 2026).

The second contract covers reasoning and tools. Time series agents produce rationales, causal narratives, diagnostic summaries, and action recommendations, so each claim should be tied to a raw window, statistic, retrieved event, or executable observation. TS-Agent, TimeART, TS-Debate, and Cast-R1 move from free-form explanation toward process-level verification through evidence logs, tool calls, repair, or reviewer feedback (Liu et al., 2025d; Wu et al., 2026b; Trirat et al., 2026; Tao et al., 2026b). TSAG, TimeCopilot, AD-AGENT, AutoLFM, and AION extend the same contract to tool routing, validation interfaces, shared memory, and layered review (Kolonin et al., 2026; Garza & Rosillo, 2025; Yang et al., 2025d; Li et al., 2026b; Zhan et al., 2026). Adjacent evidence on multi-agent skill distillation further suggests that reliability and efficiency gains are metric- and trace-dependent rather than automatic (Xu et al., 2026c). The reliable artifact is therefore a trace, not only a prediction.

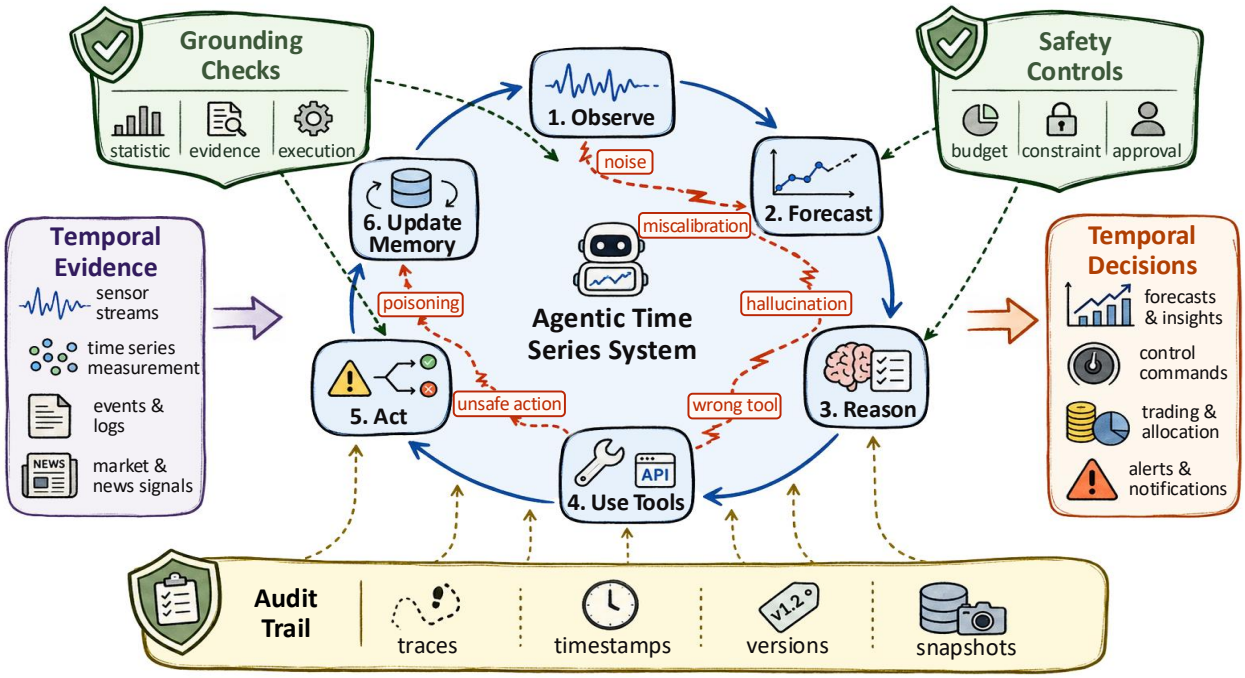

Figure 7: Failure propagation and reliability controls in agentic time series systems. Errors can enter through evidence, forecasting, reasoning, tool use, actions, or memory updates; grounding checks, safety controls, and audit trails interrupt the loop.

Table 10: Reliability stack for agentic time series systems.

Layer	What to check	Characteristic failure	Representative anchors
Forecasting quality	Accuracy, calibration, horizon stability, leakage control	Good average error but weak tails, long horizons, or contaminated splits	**GIFT-Eval** (Aksu et al., 2024), **TFRBench** (Ahamed et al., 2026)
Robustness	Shift, extreme events, corrupted sensors, recovery	Regime break or sensor drift causes confident failure	**FOIL** (Liu et al., 2024a), **SocGate** (Dai et al., 2025b)
Reasoning correctness	Arithmetic, temporal ordering, causal discipline	Correct-looking answer with invalid steps or correlation-as-causation	**TS-Agent** (Liu et al., 2025d), **VeriTime** (Zhou et al., 2026)
Tool use	Tool routing, argument validity, execution recovery	Wrong tool, malformed call, failed execution, or fabricated result	**TimeART** (Wu et al., 2026b), **TSAG** (Kolonin et al., 2026)
Grounding	Support from raw series, events, statistics, retrieval, tools	Invented trend, event, seasonality, causal story, or anomaly cause	**CaTS-Bench** (Zhou et al., 2025a), **AXIS** (Lan et al., 2025)
Security	Provenance, least privilege, instruction-data isolation, memory hygiene	Untrusted text, tool metadata, or poisoned memory steers action	**ToolHijacker** (Shi et al., 2025c), **AgentDojo** (Debenedetti et al., 2024)
Decision safety	Risk budgets, constraints, abstention, escalation	Useful forecast becomes unsafe intervention, trade, or allocation	**FinHEAR** (Chen et al., 2025b), **Trading-R1** (Xiao et al., 2025b)
Auditability	Replayable traces, timestamps, prompts, seeds, tools, artifacts	Fluent summary without replayable evidence or comparable protocol	**TimeSeriesGym** (Cai et al., 2025b), **TimeSeek** (Mostafa et al., 2026)

8.2 Robustness, Grounding, and Security

Reliability becomes harder under non-stationarity. Distribution shift may appear as drift, abrupt regime change, seasonal phase shift, missingness, sensor replacement, policy intervention, or exogenous shock (Liu et al., 2024a; Fan et al., 2024; Weng et al., 2026). For agents, robustness is also a policy property: the system must decide whether to trust the forecast, retrieve related episodes, switch tools, widen uncertainty, abstain, or escalate (Tao et al., 2026a; Qiao et al., 2025). Domain constraints can help, as physics-guided battery models improve stability under temperature variation, sensor noise, incorrect initial values, aging, and long multicycle operation (Dai et al., 2025a;b). Evaluation should report regime-conditional performance, tail risk, calibration under shift, degradation curves, and recovery time (Mostafa et al., 2026; Nechepurenko & Shuvalov, 2026).

Grounding prevents fluent but unsupported explanations. A time series agent may invent a trend, seasonality, event attribution, causal relation, or tool result. TS-Agent preserves numerical evidence; TSAG evaluates financial tool selection and hallucination; CaTS-Bench and AXIS test numeric and contextual fidelity; Argos and AnomaMind make anomaly explanation an iterative diagnostic process with rules, review, and self-reflection (Liu et al., 2025d; Kolonin et al., 2026; Zhou et al., 2025a; Lan et al., 2025; Gu et al., 2025; Tao et al., 2026c). A faithful explanation should satisfy evidence grounding, temporal consistency, and decision relevance.

Security is coupled with grounding because many systems read untrusted news, reports, logs, clinical notes, web pages, and user incident descriptions. Prompt injection, malicious tool metadata, poisoned memories, adversarial spikes, transferable adversarial attacks, and crafted market narratives can all steer forecasts or actions (Debenedetti et al., 2024; Zhang et al., 2025a; Ferrag et al., 2025; Shi et al., 2025c; Xu et al., 2026e; 2025a). Practical defenses include least-privilege tool access, typed schemas, sandboxed execution, provenance tracking, instruction-data separation, memory hygiene, approval gates, clean-data separation, and external semantic auditing (Xu et al., 2025b; 2026d). Hubble illustrates this pattern by constraining financial factors with a domain-specific language, AST checks, and success/failure memory (Shi et al., 2026a).

8.3 Decision Safety and Auditability

The final contract appears when predictions become actions. In trading, energy, traffic, clinical, and industrial settings, the target is a policy that respects risk budgets, constraints, and delayed feedback, not merely an accurate forecast. FinHEAR, TradingAgents, Trading-R1, and AIVV move toward role-specialized analysis, risk-aware debate, confidence-aware sizing, reinforcement-trained decisions, and auditable validation artifacts (Chen et al., 2025b; Xiao et al., 2024; 2025b; Kwon et al., 2026). Trace-economic underwriting offers an adjacent risk-transfer view: autonomous deployment becomes acceptable only when episode-level traces define bounded authority, expected benefit, control cost, premium, and residual loss (Xu et al., 2026a). A reliable interface should expose uncertainty, evidence, candidate actions, constraint checks, and escalation status. A risk manager can reject hard-constraint violations, shrink actions under poor calibration, require human approval, and log decisions for replay (Shi et al., 2026a; Xia et al., 2026).

Auditability connects deployment reliability to scientific evaluation. A trustworthy episode should preserve prompts, retrieved context, tool versions, code artifacts, random seeds, input timestamps, memory state, intermediate observations, actions, feedback, and final decisions (Cai et al., 2025b; Zhan et al., 2026; Ang et al., 2025). This is essential for memory-augmented agents: storing only successes amplifies survivorship bias, while storing unverified rationales can institutionalize hallucinations (Tao et al., 2026a; Xu et al., 2026b; Zhou et al., 2025a). Mature benchmarks should report where a failure entered, whether it propagated, whether the agent detected it, and whether the system recovered or escalated. Existing resources cover pieces of this agenda, including GIFT-Eval, TemporalBench, TimeSeriesGym, Foresight Arena, TimeSeek, ForecastBench, and agentic trading audits (Aksu et al., 2024; Weng et al., 2026; Cai et al., 2025b; Nechepurenko & Shuvalov, 2026; Mostafa et al., 2026; Karger et al., 2025; Xia et al., 2026). The remaining gap is integration: deployed agents fail through interactions among forecasting, reasoning, tools, memory, security, and action, so trustworthy evaluation must be episode-level rather than answer-level.

Table 11: Summary of Time Series Benchmark Taxonomy. As the paradigm evolves, the evaluation object shifts from numerical predictors to text-aligned reasoners, and ultimately to autonomous agents, fundamentally changing the input and output spaces as well as the core evaluation metrics.

Category	Core Evaluation Target	Input Space	Output Space	Key Metrics	Representative Works
Series-Only Foundation (Sec. 9.1)	Evaluate generalization capabilities of time series foundation models.	Univariate and multivariate time series	Numerical sequences or discrete labels	MSE, MAE, F1-score	GIFT-Eval (Aksu et al., 2024), fev-bench (Shchur et al., 2025), It's TIME (Qiao et al., 2026)
Heterogeneous Augmented (Sec. 9.2)	Evaluate whether heterogeneous signals enhance standard time series tasks.	Time series paired with textual modalities (e.g., news, events) or/and images	Numerical sequences or discrete labels	MSE, MAE, F1-score	Time-MMD (Liu et al., 2024b), CiK (Williams et al., 2024), FNSPID (Dong et al., 2024), TTC (Kim et al., 2024)
Understanding & Reasoning (Sec. 9.3)	Assess the cognitive capacity of models to comprehend, explain, and perform reasoning over temporal dynamics via static question answering.	Time series aligned with textual prompts and queries	Natural language (e.g., choice, open-ended answers, chains of thought)	Answer Score, Text Generation Metrics (e.g., BLEU, ROUGE)	Time-MQA (Kong et al., 2025a), TSAQA (Jing et al., 2026), TSRBench (Yu et al., 2026a), MMTS-Bench (Yin et al., 2026), TimeSage-MT (Kong et al., 2026)
System-Level Agentic Tasks (Sec. 9.4)	Evaluate the autonomous end-to-end behavioral trajectories of agents in machine learning engineering, evidence discovery, open-world forecasting.	High-level user queries coupled with open environments and tool interfaces	Behavioral trajectories, executable code, and synthesized empirical evidence	Result Score, Completion Rate, Process Score, Forecasting Score	TimeSeriesGym (Cai et al., 2025b), Dr-CiK (Tang et al., 2026), AutoCast (Zou et al., 2022), FutureX (Zeng et al., 2025)

9 Benchmarks and Evaluation Protocols

Benchmarks determine which capabilities count as progress and which failures remain invisible. As time series systems move from foundation models to LLM4TS and then to agents, the evaluation object shifts from numerical prediction, to language-aligned understanding, to behavioral trajectories involving tools, code, evidence acquisition, and feedback. We therefore treat benchmarks as an orthogonal evaluative pillar in Fig. 1, rather than as a subordinate component of any single model family. Tab. 11 summarizes the resulting taxonomy, and Sec. 9.5 maps it back to the agentic layers.

9.1 Series-Only Foundation Benchmarks

Series-only benchmarks define the numerical baseline that later multimodal or agentic systems must not obscure. Classical TSLib-style evaluation covers forecasting, imputation, anomaly detection, and classification on fixed splits (Wang et al., 2026a); TSFM-era protocols instead emphasize contamination control, scale, freshness, domain coverage, and statistically meaningful comparison. GIFT-Eval, fev-bench, and It's TIME illustrate this shift through contamination-free zero-shot evaluation, larger task collections with uncertainty estimates, and dynamically updated datasets (Aksu et al., 2024; Shchur et al., 2025; Qiao et al., 2026). Capability probes such as compositional reasoning tests further ask whether a forecaster has learned reusable temporal structure rather than only benchmark-specific accuracy (Potosnak et al., 2025). These benchmarks remain essential, but they mainly measure numerical fidelity; they cannot by themselves evaluate contextual understanding, tool use, or decision quality.

9.2 Heterogeneous-Augmented End-to-End Benchmarks

Heterogeneous-augmented benchmarks add news, events, reports, metadata, images, or logs to numerical sequences while keeping the output as a forecast, label, or anomaly decision. They test whether external context improves standard tasks, not whether an agent can reason or act. General-domain datasets increasingly stress alignment quality, contextual necessity, data fidelity, and cold-start settings, as in Time-MMD, CiK, NewsForecast, TimeIMM, Fidel-TS, and MoTime (Liu et al., 2024b; Williams et al., 2024; Wang et al., 2024b; Chang et al., 2026; Xu et al., 2025e; Zhou et al., 2025c). Domain-specific benchmarks concentrate on finance, retail, and observability: FinMultiTime and FNSPID provide market-text alignment, StockNet and Hybrid Deep Sequential establish cross-modal financial baselines, DOW 30 and FinTexTS use LLMs or semantic retrieval for financial context, VISUELLE connects product images with retail demand, and LEMMA-RCA pairs metrics with system logs for root-cause analysis (Xu et al., 2025d; Dong et al., 2024; Xu & Cohen, 2018; Wu et al., 2018; Chen et al., 2023; Lee et al., 2026; Skenderi et al., 2024; Zheng et al., 2024). Boundary cases such as TIME-RA and TTC begin to attach explanations or text outputs to time series tasks (Yang et al., 2025f; Kim et al., 2024), but their main evaluation logic is still inherited from supervised prediction.

9.3 From Label to Text: Time Series Understanding and Reasoning

Understanding and reasoning benchmarks move the output space to natural language, including binary answers, multiple choice, open-ended explanations, captions, and reasoning traces. Outcome-oriented TSQA datasets such as Time-MQA, MTBench, TSAQA, TimeSage-MT, and EngineMT-QA unify forecasting, classification, anomaly, or domain-specific sensor tasks into question-answering interfaces (Kong et al., 2025a; Chen et al., 2025a; Jing et al., 2026; Kong et al., 2026; Lei et al., 2025). Capability-deconstruction benchmarks then probe perception, reasoning, alignment, and domain-specific cognition more explicitly: TSRBench, MMTS-Bench, TemporalBench, ARFBench, HEARTS, TimeSeriesExam, ODTQA-FoRe, and TFRBench evaluate structural transitions, contextual layers, clinical or observability reasoning, template-driven sensitivities, open-domain evidence, and verifier-mediated reasoning (Yu et al., 2026a; Yin et al., 2026; Weng et al., 2026; Xie et al., 2026; Li et al., 2026c; Cai et al., 2024; Wang et al., 2026b; Ahamed et al., 2026). Other studies show that LMs still struggle with zero-shot numerical reasoning and that visual modalities or VLMs introduce their own alignment failures (Merrill et al., 2024; Zhou & Yu, 2025; Zhou et al., 2025a). These benchmarks are stronger probes of temporal cognition, but most remain single-turn and static: the model answers an exam question rather than acquiring evidence or revising a plan.

9.4 Beyond Static Exam: Working on System-Level Agentic Tasks

System-level benchmarks evaluate trajectories rather than isolated outputs. Machine-learning engineering benchmarks ask agents to build executable time series pipelines, manage artifacts, and submit working solutions; TimeSeriesGym and its diagnostic companion show both the promise and fragility of such agents under long-horizon code and modeling workflows (Cai et al., 2025b;a). Evidence-discovery settings such as Dr-CiK, TimeSeriesExamAgent, and TSAIA test whether agents can retrieve predictive context, generate evaluation tasks, or solve CodeAct-style temporal tasks under constraints (Tang et al., 2026; Gwiazda et al., 2026; Ye et al., 2025). Open-world event forecasting further shifts the target from the time series itself to future events, with AutoCast, AutoCast++, ForecastQA, PROPHET, ForecastBench, FutureX, FutureX-Pro, and Foresight Arena evaluating temporally valid information use, unresolved queries, settlement, and agent-generated forecasting tasks (Zou et al., 2022; Yan et al., 2024; Jin et al., 2021; Yang et al., 2025c; Karger et al., 2025; Zeng et al., 2025; Liu et al., 2026a; Nechepurenko & Shuvalov, 2026). This tier is closest to agentic evaluation, but delayed feedback, memory, world modeling, and action safety are still only partially captured.

9.5 Evaluation Boundaries: Mapping Benchmarks to the Layers

Coverage on Perception and Reasoning Layers. Existing benchmarks mainly cover perception and reasoning. Series-only and heterogeneous-augmented datasets evaluate numerical fidelity, feature alignment, and context integration, while QA and capability-deconstruction benchmarks evaluate whether systems

identify patterns, verify claims, and produce explanations. However, most of these protocols remain passive and single-turn; they do not measure how evidence acquisition, tool selection, memory retrieval, or feedback changes later behavior.

The Frontier of Planning and Action Evaluation. Planning and action evaluation begins with engineering and evidence-discovery benchmarks such as TimeSeriesGym, Dr-CiK, and TSAIA (Cai et al., 2025b; Tang et al., 2026; Ye et al., 2025). These tasks measure decomposition, code generation, tool routing, and artifact production, but often return immediate feedback from a simplified environment. Realistic temporal agents require evaluation under delayed outcomes, rolling replanning, operational constraints, and credit assignment across multiple decisions.

Gaps in Memory and World Model Assessment. Memory and world-model layers remain the least standardized. Current benchmarks rarely evaluate cross-session memory, forgetting, regime-conditioned retrieval, causal simulation, counterfactual validity, or intervention outcomes. A mature evaluation stack should therefore combine predictive quality, reasoning faithfulness, tool-use reliability, memory dynamics, simulation validity, and decision-centric metrics in one reproducible protocol.

10 Applications of Agentic Time Series Systems Across Domains

Building on the architecture and reliability framework, this section examines how agentic time series systems appear across application domains. The purpose is not to provide an industry catalogue, but to show which parts of the agent stack each domain stresses: general-purpose analysis emphasizes workflow construction and auditability; finance emphasizes delayed, costly decisions; transportation emphasizes spatial reasoning, simulation, and control; physical, clinical, observability, agricultural, retail, and supply-chain settings emphasize different forms of grounding, safety, and executable feedback. Across domains, the value of agency lies in turning forecasts into trustworthy workflows that can analyze, decide, adapt, and act under domain-specific constraints.

10.1 General-Purpose Time Series Analysis

General-purpose time series analysis is the lowest-prior setting: the agent receives a raw series and an underspecified request without knowing the task type, frequency, data quality, relevant horizon, or temporal scale (Yang et al., 2026a;b; Ding et al., 2023). The system must diagnose the environment before modeling it, because errors in frequency inference, missingness handling, seasonality detection, or regime identification propagate into model selection, validation, and reporting. TimeSeriesScientist and TimeCopilot instantiate this workflow-oriented view by combining curation, planning, model narrowing, cross-validation, forecasting, and reporting (Zhao et al., 2025; Garza & Rosillo, 2025).

Open-ended analysis systems extend this setting from forecasting to evidence-grounded question answering and executable workflows. TimeART, TS-Agent, and TS-Reasoner use tool calls, statistical operators, retrieval, and generated programs to answer temporal queries with intermediate observations rather than unsupported text (Wu et al., 2026b; Liu et al., 2025d; Ye et al., 2026). TimeSeriesGym and AION further treat the agent as a time series engineering system that edits code, manages artifacts, tunes pipelines, and validates submissions (Cai et al., 2025b; Zhan et al., 2026). The main reliability requirement is therefore auditability: broad autonomy should not hide weak forecasting quality, as shown by GIFT-Eval (Aksu et al., 2024), nor should it return reports whose data processing, diagnostics, model choices, and validation artifacts cannot be rerun.

10.2 Finance and Trading

Finance is a mature stress test for agentic time series systems because decisions are delayed, costly, non-stationary, and often irreversible. Market feedback arrives after positions are taken, while transaction costs, leverage, liquidity, regime shifts, and macro events can invalidate historical patterns. Research-oriented agents therefore focus on reproducible signal discovery: R&D-Agent-Quant uses a research-development-feedback loop for factor mining, modeling, code generation, and backtesting, while Hubble constrains factor generation

through a domain-specific operator language, sandboxed execution, diagnostics, and memory (Li et al., 2026d; Shi et al., 2026a).

Decision-oriented financial agents add role structure, memory, personalization, risk control, and reinforcement learning. TradingAgents coordinates analyst, trader, and risk roles; FinCon uses hierarchical conceptual reinforcement; FinMem separates short-, mid-, and long-term market memory; FinArena and FinHEAR add user preferences, expert retrieval, confidence-aware sizing, and human-agent decision support; Trading-R1 trains reasoning policies under trading feedback; and TSAG evaluates tool delegation and hallucination avoidance in financial time series QA (Xiao et al., 2024; Yu et al., 2024; 2025b; Xu et al., 2025c; Chen et al., 2025b; Xiao et al., 2025b; Kolonin et al., 2026). These systems make decision safety, provenance, and reproducibility central: forecasts can be unsafe under leverage or ignored uncertainty, while code-executing agents are sensitive to splits, execution timing, cost assumptions, and factor protocols. Reliable financial agents therefore need risk constraints, comparable backtesting, secure provenance, and standardized execution semantics (Bao et al., 2026; Xia et al., 2026).

10.3 Traffic, Urban Systems, and Transportation

Transportation settings connect forecasting with spatial reasoning, simulation, and control. Traffic actions can reshape future network states, and urban sensing requires temporal signals to be interpreted together with maps, mobility patterns, demand shifts, and operational constraints. Open-TI addresses this by letting an augmented language model interpret user requests, invoke traffic-analysis tools, and configure workflows for simulation, diagnosis, or signal-control experiments (Da et al., 2024). CityGPT extends this logic to broader urban intelligence through spatial knowledge and urban context (Feng et al., 2025).

DCATS highlights a complementary data-centric action: selecting and refining auxiliary traffic series from metadata and validation feedback rather than only changing the forecasting model (Yeh et al., 2025). Together, these systems show that data selection, spatial context construction, tool invocation, and simulator interaction are first-class actions. The missing evaluation piece is a decision-centric closed loop that connects forecasting, spatial reasoning, simulation, and control under realistic feedback delays, using metrics such as travel time, throughput, queue length, delay, robustness, and safety rather than pointwise forecast error alone.

10.4 Weather, Energy, and Buildings

Weather, energy, and building systems expose agents to physical processes that change across multiple time scales. The output is often not only a forecast, but also a report, warning, model-building workflow, optimization plan, or control-relevant recommendation that should remain consistent with physical constraints. Existing systems therefore move from grounded reporting to operational forecasting. Hierarchical AI-Meteorologist reasons over hourly, six-hour, and daily forecasts to generate keywords and explainable reports; AGCD injects state-conditioned physics priors through agent-guided cross-modal decoding; and WeatherTGD improves forecast captions through statistical, physical, and meteorological agent feedback (Sukhorukov et al., 2026; Wu et al., 2026a; Liu, 2026).

Building and energy agents make the workflow more executable. AutoLFM automates building load forecasting through retrieval, reasoning, code generation, and validation, while SEA-TS, CastFlow, and AlphaCast show how self-evolving code, memory-supported workflows, and human-LLM co-reasoning can make forecasting more adaptive in operational settings (Li et al., 2026b; Xu et al., 2026f; Pan et al., 2026; Zhang et al., 2026a). StockAgent-style optimization for prosumers further illustrates the transition from forecasting and explanation to optimization, action, and physical feasibility (Zhang et al., 2025b). Reliability in this domain combines forecasting quality, grounding, tool-use correctness, and decision safety: a weather report must be faithful to forecast tables, and an energy recommendation must respect operational constraints rather than only reduce pointwise error.

10.5 Healthcare

Healthcare time series agents face conservative decision-support requirements. Physiological signals, EHR streams, clinical notes, and guideline knowledge must be aligned without turning uncertainty into unsafe

recommendations. ColaCare is the clearest agentic example: DoctorAgents reason from different clinical perspectives, expert models process structured EHR data, a MetaAgent aggregates outputs, and RAG retrieves guideline knowledge for mortality and readmission prediction (Wang et al., 2025b). GEM complements this at the ECG level by unifying ECG time series, 12-lead images, and text; OpenTSLM treats medical time series as a native modality for text-time-series reasoning; and HEARTS provides a broad benchmark across health signals, modalities, and reasoning tasks (Lan et al., 2026; Langer et al., 2025; Li et al., 2026c).

The reliability concern is direct: a fluent clinical explanation is unsafe if it cites the wrong segment, ignores uncertainty, or conflicts with retrieved medical knowledge. AUROC and AUPRC remain necessary for prediction, but they do not evaluate whether evidence was used faithfully or whether the system should defer to a clinician. The missing loop is therefore abstention and escalation: healthcare agents need evaluation settings that connect prediction, guideline grounding, uncertainty-aware deferral, and clinician review.

10.6 Industrial IoT, Observability, and Root Cause Analysis

Industrial IoT and cloud observability turn anomaly detection into streaming incident diagnosis (Yang et al., 2025b). Metrics, logs, traces, alerts, and sensor signals shift as workloads and operating conditions change, and small early indicators may only become meaningful after symptoms propagate. Agentic systems respond by making anomaly detection more explainable, reproducible, and evidence-grounded. Argos generates inspectable anomaly-detection rules; AnomaMind uses coarse-to-fine localization, adaptive feature construction, tool interaction, and self-reflection; SAGE and specialist analyzer systems compare candidate explanations; and MAS-LSTM shows why distributed perception must scale for heterogeneous IIoT traffic (Gu et al., 2025; Tao et al., 2026c; Shi et al., 2025b; Kang et al., 2026; Qin et al., 2025).

Evaluation reinforces the same need for grounded incident reasoning. ARFBench tests time-series QA over production incidents, while LEMMA-RCA aligns metrics and logs for root-cause analysis across IT and OT systems (Xie et al., 2026; Zheng et al., 2024). Reliability here means reasoning correctness, grounding, tool validity, and auditability: an incident agent can waste operator time if it invents a causal chain, misaligns logs and metrics, or reverses leading and lagging signals. A useful output should therefore be a replayable diagnostic trace, not only an anomaly label or a polished summary.

10.7 Agriculture, Autonomous Systems, Retail, and Supply Chain

Agriculture, autonomous systems, retail, and supply chains test whether forecasts can support decisions whose consequences unfold later. These domains share multi-scale dependency, delayed feedback, and costly action. AgriWorld is the clearest world-model example: it provides geospatial queries, remote-sensing time-series analytics, crop simulation, and task predictors, while an Agro-Reflective agent writes code, observes execution results, and refines its analysis over lookup, forecasting, anomaly detection, and counterfactual tasks (Zhang et al., 2026b). AIVV brings a similar verification logic to autonomous underwater vehicles, where role-specialized LLM agents validate flagged anomalies against natural-language requirements and produce verification artifacts (Kwon et al., 2026).

Retail and supply-chain systems are less LLM-agentic, but they clarify why forecast accuracy must be evaluated with downstream decisions. Multi-agent deep reinforcement learning jointly optimizes demand forecasting and inventory control, while VISUELLE-style multimodal forecasting uses product images, metadata, sales signals, and Google Trends for cold-start retail prediction (Yang et al., 2025g; Skenderi et al., 2024). The reliability signature is decision safety, simulation validity, counterfactual consistency, and auditability. For AgriWorld and AIVV, the answer should be grounded in tool execution, simulator feedback, or verification traces; for retail and supply chains, a lower forecast error is insufficient if the resulting inventory policy increases stockouts or costs. The common gap is executable evaluation of the full chain from prediction to simulation, intervention, and action outcome.

11 Positions and Frontiers

This survey argues that agentic time series systems shift the field from learning a mapping from history to forecast toward running a closed loop that perceives, reasons, acts, and adapts in temporal environments. This

Table 12: The six frontiers of agentic time series systems. Each rides a shift the field is already undergoing, bookends one chapter of this survey, and leaves a concrete open question.

Frontier	Ch.	The shift it rides	The open question it leaves
Verifiable temporal reasoning	Sec. 4	From final-answer accuracy to step-level grounding.	Can every claim about trend, lag, or cause be bound to an executable check?
End-to-end agentic training	Sec. 5	From prompt-engineered pipelines to policies learned over trajectories.	Can perception, planning, memory, and action be optimized jointly?
Memory and lifelong adaptation	Sec. 6	From a fixed context window to layered, procedural, cross-session memory.	Which experience is worth keeping, and does it generalize or merely leak?
Temporal world models	Sec. 7	From point prediction to simulating and intervening on an environment.	What is the minimal world model that supports counterfactuals?
Decision-centric evaluation	Sec. 9	From forecast error to action quality under delayed feedback.	How to score prediction, uncertainty, and action jointly?
Standardized temporal tooling	Sec. 10	From ad hoc, per-paper tools to a typed, self-describing protocol.	What operations and metadata must a temporal tool expose?

shift is driven by non-stationarity, delayed feedback, multimodal evidence, and operational risk: trust must be judged over the whole loop rather than by a single prediction or fluent explanation (Cheng et al., 2026a; Dai et al., 2026a; Ning et al., 2025a; Xia et al., 2026). We summarize the resulting agenda through six frontiers tied to the closed-loop object in Sec. 2: verifiable temporal reasoning, end-to-end agentic training, memory and lifelong adaptation, temporal world models, decision-centric evaluation, and standardized temporal tooling. Tab. 12 maps these frontiers, and Sec. 11.7 states the survey's overall position.

11.1 Verifiable Temporal Reasoning

The first frontier is to make temporal reasoning auditable at the step level. Benchmarks show that models may pass one temporal ability while failing another, and review-based methods show that intermediate evidence is often more informative than the final answer alone (Cai et al., 2024; Weng et al., 2026; Ye et al., 2025; Guan et al., 2026; Su et al., 2025; Messica et al., 2026; Liu et al., 2025d). A mature reasoner should bind claims about trend, lag, anomaly, or cause to executable checks, expose uncertainty when evidence is short or confounded, and learn from verified traces rather than only prompted rationales (Zhou et al., 2026; Guan et al., 2025). The open question for Sec. 4 is how to make such step-grounding the default contract.

11.2 End-to-End Agentic Training

Most current agents are assembled from prompts, tools, memory calls, and reflection rules. Recent work suggests that parts of the loop can instead be learned through behavior cloning, reinforcement learning, tool-use trajectories, decision feedback, and process-verifiable trace synthesis (Wu et al., 2026b; Tao et al., 2026b; Guan et al., 2025; Xiao et al., 2025b; Zhou et al., 2026). The frontier for Sec. 5 is joint optimization: perception, planning, memory, and action should be trained against long-horizon objectives with delayed credit assignment and multi-objective rewards covering accuracy, calibration, efficiency, and safety.

11.3 Memory and Lifelong Adaptation

Context windows are not enough for non-stationary time series. Useful experience may be older, sparser, regime-specific, procedural, or failure-related. Declarative memory systems store patterns, rules, regimes, and confidence signals, while procedural memory systems preserve how an agent solved a task so competence can transfer across sessions (Tao et al., 2026a; Yu et al., 2025b; Shi et al., 2026a; Pan et al., 2026; Xu et al., 2026f; Cheng & Chin, 2024). The frontier for Sec. 6 is governed memory: agents must decide what to store, retrieve, update, and forget, and memory value should be measured across sessions under shift rather than within one context window (Jalori et al., 2025).

11.4 Temporal World Models

A predictor can extrapolate a trajectory, but an agent needs a model of how the environment might change under different conditions or actions. Early routes include executable agricultural environments, demand-and-inventory decision settings, and multimodal temporal simulation systems such as Chronicle (Zhang et al., 2026b; Yang et al., 2025g; Quinlan et al., 2026). The frontier for Sec. 7 is to build world models generative over futures, queryable under counterfactual interventions, and verifiable through explicit assumptions, simulator feedback, or tool traces. Because universal temporal simulators are unrealistic, the most credible near-term path lies in partially simulatable domains such as agriculture, traffic, energy, and supply chains.

11.5 Decision-Centric Evaluation

Forecast error is insufficient when an agent acts. Evaluation must follow the decision: engineering agents should be judged by working submissions, forecasting agents by settlement and calibration, trading agents by return and drawdown, and control agents by delay, throughput, and safety outcomes (Cai et al., 2025b; Nechepurenko & Shuvalov, 2026; Xiao et al., 2024; Xu et al., 2025c; Da et al., 2024). The frontier for Sec. 9 is an evaluation contract that scores prediction, uncertainty, process, and action jointly under delayed feedback, partial observability, distribution shift, and operational cost.

11.6 Standardized Temporal Tooling

Current systems repeatedly rebuild similar tools for diagnostics, forecasting, retrieval, validation, logging, and reporting (Zhao et al., 2025; Garza & Rosillo, 2025; Li et al., 2026b; Ye et al., 2026). A shared temporal tooling protocol should expose typed inputs, frequency, units, window semantics, missingness policy, covariate alignment, model version, confidence, provenance, and verifier-ready outputs. Early domain protocols point in this direction (Zhang et al., 2026b), while tool-grounded systems show why common operators such as decomposition, change-point detection, causal-lag discovery, and rolling validation need standardized execution contracts (Wu et al., 2026b; Ye et al., 2026; Liu et al., 2025d). This would turn per-paper engineering across Sec. 10 into reusable infrastructure.

11.7 Position: Time Series Agents as Temporal Decision Systems

This survey takes the position that agentic time series systems should be evaluated as temporal decision systems. A model-centric survey asks whether a model reduces error; an agentic survey asks whether a system can perceive heterogeneous evidence, reason with grounded steps, choose tools and actions, remember useful experience, verify its conclusions, and adapt when the environment changes. The frontier is therefore not a single larger foundation model, but the closed-loop object $\mathcal{A}_{TS}$ of Sec. 2 made reliable across layers. Time series agents are not NLP agents applied to numbers; they are systems for acting under temporal uncertainty, and the task ahead is to make that loop trustworthy, auditable, and capable of improvement.

12 Conclusion

This survey has framed agentic time series systems as a transition from model-centric prediction to closed-loop temporal intelligence. By distinguishing backbone models, LLM-based translators, temporal reasoners, and time series agents, we clarify that agency depends on stateful interaction among temporal evidence, tools or actions, feedback, and adaptation. The proposed five-layer architecture further organizes the field around perception, reasoning, planning and action, memory and knowledge, and world models, while treating reliability and trustworthiness as cross-cutting requirements rather than downstream metrics.

The central challenge is to make this autonomy reliable, auditable, and useful in dynamic environments where decisions have consequences. Progress will require stronger temporal reasoning, memory-centric adaptation, decision-aware evaluation, standardized tool protocols, temporal world models, and data agents that support the full lifecycle of temporal data. These directions can move time series agents beyond forecasting assistants toward trustworthy temporal analysts and adaptive decision-support systems grounded in domain constraints.

References

Md Atik Ahamed, Mihir Parmar, Palash Goyal, Yiwen Song, Long T Le, Qiang Cheng, Chun-Liang Li, Hamid Palangi, Jinsung Yoon, and Tomas Pfister. Tfrbench: A reasoning benchmark for evaluating forecasting systems. *arXiv preprint arXiv:2604.05364*, 2026.

Taha Aksu, Gerald Woo, Juncheng Liu, Xu Liu, Chenghao Liu, Silvio Savarese, Caiming Xiong, and Doyen Sahoo. Gift-eval: A benchmark for general time series forecasting model evaluation. *arXiv preprint arXiv:2410.10393*, 2024.

Samaneh Aminikhanghahi and Diane J Cook. A survey of methods for time series change point detection. *Knowledge and information systems*, 51(2):339–367, 2017.

Yihao Ang, Yifan Bao, Lei Jiang, Jiajie Tao, Anthony KH Tung, Lukasz Szpurch, and Hao Ni. Structured agentic workflows for financial time-series modelling with llms and reflective feedback. In *Proceedings of the 6th ACM International Conference on AI in Finance*, pp. 924–932, 2025.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, et al. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv:2510.15821*, 2025.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba, Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. 2025. URL `https://arxiv.org/abs/2506.09985`.

Yifan Bao, Xinyu Xi, Xinyu Liu, Wen Ge, Lei Jiang, Kevin Zhang, Raad Khraishi, Yihao Ang, Anthony KH Tung, Lukasz Szpruch, et al. Mosaic: Modular orchestration for structured agentic intelligence and composition. *arXiv preprint arXiv:2606.00708*, 2026.

Kaifeng Bi, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, and Qi Tian. Pangu-weather: A 3d high-resolution model for fast and accurate global weather forecast. *arXiv preprint arXiv:2211.02556*, 2022.

George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

Jake Bruce, Michael Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Bechtle, Feryal Behbahani, Stephanie Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando de Freitas, Satinder Singh, and Tim Rocktäschel. Genie: Generative interactive environments. 2024. URL `https://arxiv.org/abs/2402.15391`.

William Gouvêa Buratto, Gabriel Villarrubia Gonzalez, Ademir Nied, Laio Oriel Seman, Anne Carolina Rodrigues Klaar, and Stefano Frizzo Stefenon. An autonomous large language model-agent framework for transparent and local time series forecasting. *Advanced Intelligent Discovery*, pp. e202500236, 2026.

Yifu Cai, Arjun Choudhry, Mononito Goswami, and Artur Dubrawski. Timeseriesexam: A time series understanding exam. *arXiv preprint arXiv:2410.14752*, 2024.

Yifu Cai, Xinyu Li, Mononito Goswami, Michał Wiliński, Gus Welter, and Artur Dubrawski. Llm agents struggle at time series machine learning engineering. In *1st ICML Workshop on foundation models for structured data*, 2025a.

Yifu Cai, Xinyu Li, Mononito Goswami, Michał Wiliński, Gus Welter, and Artur Dubrawski. Timeseriesgym: A scalable benchmark for (time series) machine learning engineering agents. *arXiv preprint arXiv:2505.13291*, 2025b.

Defu Cao, Furong Jia, Sercan O Arik, Tomas Pfister, Yixiang Zheng, Wen Ye, and Yan Liu. Tempo: Prompt-based generative pre-trained transformer for time series forecasting. *arXiv preprint arXiv:2310.04948*, 2023.

Defu Cao, Michael Gee, Jinbo Liu, Hengxuan Wang, Wei Yang, Rui Wang, and Yan Liu. Conversational time series foundation models: Towards explainable and effective forecasting. *arXiv preprint arXiv:2512.16022*, 2025.

Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):1–58, 2009.

Ching Chang, Yidan Shi, Defu Cao, Wei Yang, Jeehyun Hwang, Haixin Wang, Jiacheng Pang, Wei Wang, Yan Liu, Wen-Chih Peng, et al. A survey of reasoning and agentic systems in time series with large language models. *arXiv preprint arXiv:2509.11575*, 2025.

Ching Chang, Jeehyun Hwang, Yidan Shi, Haixin Wang, Wei Wang, Wen-Chih Peng, and Tien-Fu Chen. Time-imm: A dataset and benchmark for irregular multimodal multivariate time series. *Advances in Neural Information Processing Systems*, 38, 2026.

Jialin Chen, Aosong Feng, Ziyu Zhao, Juan Garza, Gaukhar Nurbek, Cheng Qin, Ali Maatouk, Leandros Tassiulas, Yifeng Gao, and Rex Ying. MTBench: A multimodal time series benchmark for temporal reasoning and question answering. *arXiv preprint arXiv:2503.16858*, 2025a.

Jiaxiang Chen, Mingxi Zou, Zhuo Wang, Qifan Wang, Danny Dongning Sun, Zhang Chi, and Zenglin Xu. FinHEAR: Human expertise and adaptive risk-aware temporal reasoning for financial decision-making. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 1648–1672, Suzhou, China, November 2025b. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.87. URL https://aclanthology.org/2025.findings-emnlp.87/.

Peng Chen, Siyuan Wang, Shiyan Hu, Xingjian Wu, Yang Shu, Zhongwen Rao, Meng Wang, Yijie Li, Bin Yang, and Chenjuan Guo. Towards a multimodal foundation model for time series analysis, 2026. URL https://openreview.net/forum?id=dcfHcCAG3C.

Zihan Chen, Lei Nico Zheng, Cheng Lu, Jialu Yuan, and Di Zhu. Chatgpt informed graph neural network for stock movement prediction. *arXiv preprint arXiv:2306.03763*, 2023.

Dawei Cheng, Fangzhou Yang, Sheng Xiang, and Jin Liu. Financial time series forecasting with multi-modality graph neural network. *Pattern Recognition*, 121:108218, 2022.

Junyan Cheng and Peter Chin. Sociodojo: Building lifelong analytical agents with real-world text and time series. In *The Twelfth International Conference on Learning Representations*, 2024.

Mingyue Cheng, Xiaoyu Tao, Qi Liu, Ze Guo, and Enhong Chen. Position: Beyond model-centric prediction–agentic time series forecasting. *arXiv preprint arXiv:2602.01776*, 2026a.

Mingyue Cheng, Jiahao Wang, Daoyu Wang, Xiaoyu Tao, Qi Liu, and Enhong Chen. Can slow-thinking llms reason over time? empirical studies in time series forecasting. In *Proceedings of the Nineteenth ACM International Conference on Web Search and Data Mining*, pp. 99–110, 2026b.

Leonardo Christino and Fernando V. Paulovich. Chatkg: Visualizing time-series patterns aided by intelligent agents and a knowledge graph. *Computers & Graphics*, 124:104092, 2024. ISSN 0097-8493. doi: https://doi.org/10.1016/j.cag.2024.104092. URL https://www.sciencedirect.com/science/article/pii/S0097849324002279.

Jarosław A. Chudziak and Michał Wawer. Elliottagents: A natural language-driven multi-agent system for stock market analysis and prediction. *arXiv preprint arXiv:2507.03435*, 2025.

Hsing-Huan Chung, Shijun Li, Yoav Wald, Xing Han, Suchi Saria, and Joydeep Ghosh. Milm: Large language models for multimodal irregular time series with informative sampling. *arXiv preprint arXiv:2605.13711*, 2026.

Robert B Cleveland, William S Cleveland, Jean E McRae, Irma Terpenning, et al. Stl: A seasonal-trend decomposition. *J. off. Stat*, 6(1):3–73, 1990.

Zhiqing Cui, Binwu Wang, Qingxiang Liu, Yeqiang Wang, Zhengyang Zhou, Yuxuan Liang, and Yang Wang. Augur: Modeling covariate causal associations in time series via large language models. *arXiv preprint arXiv:2510.07858*, 2025.

Longchao Da, Kuanru Liou, Tiejin Chen, Xuesong Zhou, Xiangyong Luo, Yezhou Yang, and Hua Wei. Open-ti: Open traffic intelligence with augmented language model. *International Journal of Machine Learning and Cybernetics*, 15(10):4761–4786, 2024.

Xilin Dai, Ruidi Zhou, Jinhao Zhang, Keyi He, Fanfan Lin, and Hao Ma. Socnet: A physics-guided neural network for battery state-of-charge estimation robust to temperature variations and sensor noises. *IEEE Transactions on Transportation Electrification*, 2025a.

Xilin Dai, Ruidi Zhou, Jinhao Zhang, Fanfan Lin, Weifeng Zhang, and Hao Ma. Socgate: Physics-gated neural network for stable multicycle estimation of battery state-of-charge. *IEEE Transactions on Industrial Electronics*, 2025b.

Xilin Dai, Wanxu Cai, Zhijian Xu, and Qiang Xu. Position: Universal time series foundation models rest on a category error. *arXiv preprint arXiv:2602.05287*, 2026a.

Xilin Dai, Zhijian Xu, Wanxu Cai, and Qiang Xu. From samples to scenarios: A new paradigm for probabilistic forecasting. *arXiv preprint arXiv:2509.19975*, 2026b.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. *arXiv preprint arXiv:2310.10688*, 2023.

Sarkar Snigdha Sarathi Das, Palash Goyal, Mihir Parmar, Nanyun Peng, Vishy Tirumalashetty, Chun-Liang Li, Rui Zhang, Jinsung Yoon, and Tomas Pfister. Nexus: An agentic framework for time series forecasting. *arXiv preprint arXiv:2605.14389*, 2026.

Edoardo Debenedetti, Jie Zhang, Mislav Balunovic, Luca Beurer-Kellner, Marc Fischer, and Florian Tramèr. Agentdojo: A dynamic environment to evaluate prompt injection attacks and defenses for llm agents. *Advances in Neural Information Processing Systems*, 37:82895–82920, 2024.

Ashok Devireddy and Shunping Huang. Calm: A framework for continuous, adaptive, and llm-mediated anomaly detection in time-series streams. *arXiv preprint arXiv:2508.21273*, 2025.

Kuiye Ding, Fanda Fan, Yao Wang, Ruijie Jian, Xiaorui Wang, Luqi Gong, Yishan Jiang, Chunjie Luo, and Jianfeng Zhan. Dualsg: A dual-stream explicit semantic-guided multivariate time series forecasting framework. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 508–517, 2025.

Yuhao Ding, Youngsuk Park, Karthick Gopalswamy, Hilaf Hasson, Bernie Wang, and Jun Huan. Dynamic ensemble for probabilistic time-series forecasting via deep reinforcement learning, 2023. URL `https://openreview.net/forum?id=a6NvoZ5DLoe`.

Zihan Dong, Xinyu Fan, and Zhiyuan Peng. Fnspid: A comprehensive financial news dataset in time series. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 4918–4927, 2024.

Wei Fan, Shun Zheng, Pengyang Wang, Rui Xie, Jiang Bian, and Yanjie Fu. Addressing distribution shift in time series forecasting with instance normalization flows. *arXiv preprint arXiv:2401.16777*, 14(8), 2024.

Syeda Sitara Wishal Fatima and Afshin Rahimi. A review of time-series forecasting algorithms for industrial manufacturing systems. *Machines*, 2024.

Jie Feng, Tianhui Liu, Yuwei Du, Siqi Guo, Yuming Lin, and Yong Li. Citygpt: Empowering urban spatial cognition of large language models. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 591–602, 2025.

Kun Feng, Ziwei Shan, Yuchen Fang, Yiyang Tan, Sihan Lu, Shuqi Gu, Lintao Ma, Xingyu Lu, and Kan Ren. Kairosagent: Agentic time series forecasting with fused semantic reasoning. *arXiv preprint arXiv:2605.30002*, 2026.

Mohamed Amine Ferrag, Norbert Tihanyi, Djallel Hamouda, Leandros Maglaras, Abderrahmane Lakas, and Merouane Debbah. From prompt injections to protocol exploits: Threats in llm-powered ai agents workflows. *ICT Express*, 2025.

Azul Garza and Renée Rosillo. Timecopilot. *arXiv preprint arXiv:2509.00616*, 2025.

Yunfeng Ge, Ming Jin, Yiji Zhao, Hongyan Li, Bo Du, Chang Xu, and Shirui Pan. Eventtsf: Event-aware non-stationary time series forecasting. *arXiv preprint arXiv:2508.13434*, 2025.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: A family of open time-series foundation models. *arXiv preprint arXiv:2402.03885*, 2024.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. Large language models are zero-shot time series forecasters. *Advances in neural information processing systems*, 36:19622–19635, 2023.

Shuqi Gu, Yongxiang Zhao, Baoyu Jing, and Kan Ren. Counterfactual time series forecasting with textual conditions, 2026. URL `https://openreview.net/forum?id=YRp4xqTs3n`.

Yile Gu, Yifan Xiong, Jonathan Mace, Yuting Jiang, Yigong Hu, Baris Kasikci, and Peng Cheng. Argos: Agentic time-series anomaly detection with autonomous rule generation via large language models. *arXiv preprint arXiv:2501.14170*, 2025.

Tong Guan, Zijie Meng, Dianqi Li, Shiyu Wang, Chao-Han Huck Yang, Qingsong Wen, Zuozhu Liu, Sabato Marco Siniscalchi, Ming Jin, and Shirui Pan. Timeomni-1: Incentivizing complex reasoning with time series in large language models. *arXiv preprint arXiv:2509.24803*, 2025.

Tong Guan, Sheng Pan, Johan Barthelemy, Zhao Li, Yujun Cai, Cesare Alippi, Ming Jin, and Shirui Pan. Timeomni-vl: Unified models for time series understanding and generation. *arXiv preprint arXiv:2602.17149*, 2026.

Malgorzata Gwiazda, Yifu Cai, Mononito Goswami, Arjun Choudhry, and Artur Dubrawski. Timeseriesexamagent: Creating time series reasoning benchmarks at scale. In *The Fourteenth International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=DewXWSvQPH`.

Yifan Hu, Yuante Li, Peiyuan Liu, Yuxia Zhu, Naiqi Li, Tao Dai, Shu-tao Xia, Dawei Cheng, and Changjun Jiang. Fintsb: A comprehensive and practical benchmark for financial time series forecasting. *arXiv preprint arXiv:2502.18834*, 2025a.

Yifan Hu, Peiyuan Liu, Yuante Li, Dawei Cheng, Naiqi Li, Tao Dai, Jigang Bao, and Shu-Tao Xia. Finmamba: Market-aware graph enhanced multi-level mamba for stock movement prediction. *arXiv preprint arXiv:2502.06707*, 2025b.

Yifan Hu, Peiyuan Liu, Peng Zhu, Dawei Cheng, and Tao Dai. Adaptive multi-scale decomposition framework for time series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(16):17359–17367, 2025c.

Yifan Hu, Guibin Zhang, Peiyuan Liu, Disen Lan, Naiqi Li, Dawei Cheng, Tao Dai, Shu-Tao Xia, and Shirui Pan. TimeFilter: Patch-specific spatial-temporal graph filtration for time series forecasting. In *International Conference on Machine Learning*, 2025d.

Yifan Hu, Jie Yang, Tian Zhou, Peiyuan Liu, Yujin Tang, Rong Jin, and Liang Sun. Bridging past and future: Distribution-aware alignment for time series forecasting. In *The Fourteenth International Conference on Learning Representations*, 2026.

Yusha Hu and Yi Man. Energy consumption and carbon emissions forecasting for industrial processes: Status, challenges and perspectives. *Renewable and Sustainable Energy Reviews*, 2023.

Qihe Huang, Zhengyang Zhou, Kuo Yang, Gengyu Lin, Zhongchao Yi, and Yang Wang. Leret: Language-empowered retentive network for time series forecasting. In *IJCAI*, pp. 4165–4173, 2024.

Qihe Huang, Zhengyang Zhou, Yangze Li, Kuo Yang, Binwu Wang, and Yang Wang. Many minds, one goal: Time series forecasting via sub-task specialization and inter-agent cooperation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Gunjan Jalori, Preetika Verma, and Sercan O Arik. FLAIRR-TS - forecasting LLM-agents with iterative refinement and retrieval for time series. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 15427–15437, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.834. URL `https://aclanthology.org/2025.findings-emnlp.834/`.

Yushan Jiang, Wenchao Yu, Geon Lee, Dongjin Song, Kijung Shin, Wei Cheng, Yanchi Liu, and Haifeng Chen. Timexl: Explainable multi-modal time series prediction with llm-in-the-loop. *Advances in Neural Information Processing Systems*, 38:135431–135467, 2026.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. In *International conference on learning representations*, volume 2024, pp. 23857–23880, 2024.

Woojeong Jin, Rahul Khanna, Suji Kim, Dong-Ho Lee, Fred Morstatter, Aram Galstyan, and Xiang Ren. Forecastqa: A question answering challenge for event forecasting with temporal text data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4636–4650, 2021.

Baoyu Jing, Sanhorn Chen, Lecheng Zheng, Boyu Liu, Zihao Li, Jiaru Zou, Tianxin Wei, Zhining Liu, Zhichen Zeng, Ruizhong Qiu, et al. Tsaqa: Time series analysis question and answering benchmark. *arXiv preprint arXiv:2601.23204*, 2026.

Hyeongwon Kang, Jeongseob Kim, Jinwoo Park, and Pilsung Kang. Detecting time series anomalies like an expert: A multi-agent llm framework with specialized analyzers. *arXiv preprint arXiv:2605.05725*, 2026.

Ezra Karger, Houtan Bastani, Chen Yueh-Han, Zachary Jacobs, Danny Halawi, Fred Zhang, and Philip E Tetlock. Forecastbench: A dynamic benchmark of ai forecasting capabilities. In *International Conference on Learning Representations*, volume 2025, pp. 93943–93980, 2025.

Emaad Khwaja, Chris Lettieri, Gerald Woo, Eden Belouadah, Marc Cenac, Guillaume Jarry, Enguerrand Paquin, Xunyi Zhao, Viktoriya Zhukov, Othmane Abou-Amal, et al. Toto 2.0: Time series forecasting enters the scaling era. *arXiv preprint arXiv:2605.20119*, 2026.

Rebecca Killick, Paul Fearnhead, and Idris A Eckley. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107(500):1590–1598, 2012.

Kai Kim, Howard Tsai, Rajat Sen, Abhimanyu Das, Zihao Zhou, Abhishek Tanpure, Mathew Luo, and Rose Yu. Multi-modal forecaster: Jointly predicting time series and textual data. *arXiv preprint arXiv:2411.06735*, 2024.

Anton Kolonin, Alexey Glushchenko, Evgeny Bochkov, and Abhishek Saxena. Time series augmented generation for financial applications. *arXiv preprint arXiv:2604.19633*, 2026.

Yaxuan Kong, Yiyuan Yang, Yoontae Hwang, Wenjie Du, Stefan Zohren, Zhangyang Wang, Ming Jin, and Qingsong Wen. Time-mqa: Time series multi-task question answering with context enhancement. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025a.

Yaxuan Kong, Yiyuan Yang, Shiyu Wang, Chenghao Liu, Yuxuan Liang, Ming Jin, Stefan Zohren, Dan Pei, Yan Liu, and Qingsong Wen. Achieving time series reasoning requires rethinking model design, tasks formulation, and evaluation. *arXiv preprint arXiv:2502.01477*, 2025b.

Yaxuan Kong, Qingren Yao, Yuqi Nie, Yichen Li, Yilei Shao, Stefan Zohren, Anna Vettoruzzo, Joaquin Vanschoren, Ming Jin, and Qingsong Wen. Timesage-mt: A multi-turn benchmark for evaluating agentic time series reasoning. *arXiv preprint arXiv:2606.01498*, 2026.

Siva Rama Krishna Kottapalli, Karthik Hubli, Sandeep Chandrashekhara, Garima Jain, Sunayana Hubli, Gayathri Botla, and Ramesh Doddaiah. Foundation models for time series: A survey. *arXiv preprint arXiv:2504.04011*, 2025.

Jiyong Kwon, Ujin Jeon, Sooji Lee, and Guang Lin. Aivv: Neuro-symbolic llm agent-integrated verification and validation for trustworthy autonomous systems. *arXiv preprint arXiv:2604.02478*, 2026.

Tian Lan, Hao Duong Le, Jinbo Li, Wenjun He, Meng Wang, Chenghao Liu, and Chen Zhang. Axis: Explainable time series anomaly detection with large language models. *arXiv preprint arXiv:2509.24378*, 2025.

Xiang Lan, Feng Wu, Kai He, Qinghao Zhao, Shenda Hong, and Mengling Feng. Gem: Empowering mllm for grounded ecg understanding with time series and images. *Advances in Neural Information Processing Systems*, 38:94421–94455, 2026.

Patrick Langer, Thomas Kaar, Max Rosenblattl, Maxwell A Xu, Winnie Chow, Martin Maritsch, Robert Jakob, Ning Wang, Juncheng Liu, Aradhana Verma, et al. Opentslm: Time-series language models for reasoning over multivariate medical text-and time-series data. *arXiv preprint arXiv:2510.02410*, 2025.

Geon Lee, Wenchao Yu, Kijung Shin, Wei Cheng, and Haifeng Chen. Timecap: Learning to contextualize, augment, and predict time series events with large language model agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 18082–18090, 2025.

Jaehoon Lee, Suhwan Park, Tae Yoon Lim, Seunghan Lee, Jun Seo, Dongwan Kang, Hwanil Choi, Minjae Kim, Sungdong Yoo, SoonYoung Lee, et al. Fintexts: Financial text-paired time-series dataset via semantic-based and multi-level pairing. *arXiv preprint arXiv:2603.02702*, 2026.

Peixuan Lei, Jie Song, Yuzhe Hao, Tao Chen, Yuxuan Zhang, LEI JIA, Yuanxiang Li, et al. Itformer: Bridging time series and natural language for multi-modal qa with large-scale multitask dataset. In *Forty-second International Conference on Machine Learning*, 2025.

Hao Li, Yu-Hao Huang, Chang Xu, Viktor Schlegel, Renhe Jiang, Riza Batista-Navarro, Goran Nenadic, and Jiang Bian. Bridge: Bootstrapping text to control time-series generation via multi-agent iterative optimization and diffusion modeling. *arXiv preprint arXiv:2503.02445*, 2025.

Lin Li, Jiawei Huang, Qihao Quan, Dan Li, Boxin Li, Xiao Zhang, Erli Meng, Wenjie Feng, Jian Lou, and See-Kiong Ng. Empowering vlms for few-shot multimodal time series classification via tailored agentic reasoning. *arXiv preprint arXiv:2605.09395*, 2026a.

Shuhao Li, Wanfu Zheng, Mingcheng Li, Shanshuo Xing, and Zhe Wang. Autolfm: A multi-agent llm framework for automated building load forecasting. In *Building Simulation*, pp. 1–24. Springer, 2026b.

Sirui Li, Shuhan Xiao, Mihir Joshi, Ahmed Metwally, Daniel McDuff, Wei Wang, and Yuzhe Yang. Hearts: Benchmarking llm reasoning on health time series. *arXiv preprint arXiv:2603.06638*, 2026c.

Yuan Li, Bingqiao Luo, Qian Wang, Nuo Chen, Xu Liu, and Bingsheng He. CryptoTrade: A reflective LLM-based agent to guide zero-shot cryptocurrency trading. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2024.

Yuante Li, Xu Yang, Xiao Yang, Xisen Wang, Weiqing Liu, and Jiang Bian. R&d-agent-quant: a multi-agent framework for data-centric factors and model joint optimization. *Advances in Neural Information Processing Systems*, 38, 2026d.

Zechen Li, Baiyu Chen, Hao Xue, and Flora D. Salim. Zara: Training-free motion time-series reasoning via evidence-grounded llm agents. *arXiv preprint arXiv:2508.04038*, 2026e.

Minhua Lin, Zhengzhang Chen, Yanchi Liu, Xujiang Zhao, Zongyu Wu, Junxiang Wang, Xiang Zhang, Suhang Wang, and Haifeng Chen. Decoding time series with llms: A multi-agent framework for cross-domain annotation. In *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 6244–6281, 2026.

Chenghao Liu, Taha Aksu, Juncheng Liu, Xu Liu, Hanshu Yan, Quang Pham, Silvio Savarese, Doyen Sahoo, Caiming Xiong, and Junnan Li. Moirai 2.0: When less is more for time series forecasting. *arXiv preprint arXiv:2511.11698*, 2025a.

Haoxin Liu, Harshavardhan Kamarthi, Lingkai Kong, Zhiyuan Zhao, Chao Zhang, and B Aditya Prakash. Time-series forecasting for out-of-distribution generalization using invariant learning. In *International Conference on Machine Learning*, pp. 31312–31325. PMLR, 2024a.

Haoxin Liu, Shangqing Xu, Zhiyuan Zhao, Lingkai Kong, Harshavardhan Kamarthi, Aditya B Sasanur, Megha Sharma, Jiaming Cui, Qingsong Wen, Chao Zhang, et al. Time-mmd: Multi-domain multimodal dataset for time series analysis. *Advances in Neural Information Processing Systems*, 2024b.

Jiashuo Liu, Siyuan Chen, Zaiyuan Wang, Zhiyuan Zeng, Jiacheng Guo, Liang Hu, Lingyue Yin, Suozhi Huang, Wenxin Hao, Yang Yang, et al. Futurex-pro: Extending future prediction to high-value vertical domains. *arXiv preprint arXiv:2601.12259*, 2026a.

Lei Liu, Yuxing Tian, Chinmay Chakraborty, Jie Feng, Qingqi Pei, Li Zhen, and Keping Yu. Multilevel federated learning-based intelligent traffic flow forecasting for transportation network management. *IEEE Transactions on Network and Service Management*, 2023.

Peiyuan Liu, Hang Guo, Tao Dai, Naiqi Li, Jigang Bao, Xudong Ren, Yong Jiang, and Shu-Tao Xia. Calf: Aligning llms for time series forecasting via cross-modal fine-tuning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025b.

Peiyuan Liu, Beiliang Wu, Yifan Hu, Naiqi Li, Tao Dai, Jigang Bao, and Shu-Tao Xia. Timebridge: Non-stationarity matters for long-term time series forecasting. In *International Conference on Machine Learning*. PMLR, 2025c.

Penghang Liu, Elizabeth Fons, Svitlana Vyetrenko, Daniel Borrajo, Vamsi K Potluru, and Manuela Veloso. Ts-agent: A time series reasoning agent with iterative statistical insight gathering. In *First Workshop on Foundations of Reasoning in Language Models*, 2025d.

Qingxiang Liu, Zhiqing Cui, Xiaoliang Luo, Yuqian Wu, Zhuoyang Jiang, Huaiyu Wan, Sheng Sun, Lvchun Wang, Wei Yu, and Yuxuan Liang. Rationale-grounded in-context learning for time series reasoning with multimodal large language models. *arXiv preprint arXiv:2601.02968*, 2026b.

Shixu Liu. Optimizing multi-agent weather captioning via text gradient descent: A training-free approach with consensus-aware gradient fusion. *arXiv preprint arXiv:2603.21673*, 2026.

Yiding Liu, Yifan Hu, Hongjie Xia, Peiyuan Liu, Hongzhou Chen, Xilin Dai, Zewei Dong, and Jiang-Ming Yang. Falcon-x: A time series foundation model for heterogeneous multivariate modeling. *arXiv preprint arXiv:2605.27286*, 2026c.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations*, 2024c.

Yong Liu, Guo Qin, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Autotimes: Autoregressive time series forecasters via large language models. *Advances in Neural Information Processing Systems*, 37: 122154–122184, 2024d.

Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Timer: Generative pre-trained transformers are large time series models. *arXiv preprint arXiv:2402.02368*, 2024e.

Qingqing Long, Zheng Fang, Chen Fang, Chong Chen, Pengfei Wang, and Yuanchun Zhou. Unveiling delay effects in traffic forecasting: a perspective from spatial-temporal delay differential equations. In *Proceedings of the ACM web conference 2024*, pp. 1035–1044, 2024.

Junkai Lu, Peng Chen, Xingjian Wu, Yang Shu, Chenjuan Guo, Christian S Jensen, and Bin Yang. PATRA: Pattern-aware alignment and balanced reasoning for time series question answering. *arXiv preprint arXiv:2602.23161*, 2026.

Junyu Luo, Weizhi Zhang, Ye Yuan, Yusheng Zhao, Junwei Yang, Yiyang Gu, Bohan Wu, Binqi Chen, Ziyue Qiao, Qingqing Long, et al. Large language model agent: A survey on methodology, applications and challenges. *arXiv preprint arXiv:2503.21460*, 2025a.

Yucong Luo, Yitong Zhou, Mingyue Cheng, Jiahao Wang, Daoyu Wang, Tingyue Pan, and Jintao Zhang. Time series forecasting as reasoning: A slow-thinking approach with reinforced LLMs. *arXiv preprint arXiv:2506.10630*, 2025b.

Sisuo Lyu, Siru Zhong, Tiegang Chen, Weilin Ruan, Qingxiang Liu, Taiqiang Lv, Qingsong Wen, Raymond Chi-Wing Wong, and Yuxuan Liang. Ts-memory: Plug-and-play memory for time series foundation models. *arXiv preprint arXiv:2602.11550*, 2026.

Mike A Merrill, Mingtian Tan, Vinayak Gupta, Thomas Hartvigsen, and Tim Althoff. Language models still struggle to zero-shot reason about time series. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 3512–3533, 2024.

Shvat Messica, Jiawen Zhang, Kevin Li, Theodoros Tsiligkaridis, and Marinka Zitnik. Adaptive time series reasoning via segment selection. *arXiv preprint arXiv:2602.18645*, 2026.

Hamza Mostafa, Om Shastri, and Dennis Lee. Timeseek: Temporal reliability of agentic forecasters. *arXiv preprint arXiv:2604.04220*, 2026.

Maksym Nechepurenko and Pavel Shuvalov. Foresight arena: An on-chain benchmark for evaluating ai forecasting agents. *arXiv preprint arXiv:2605.00420*, 2026.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

Kanghui Ning, Zijie Pan, Yushan Jiang, Anderson Schneider, Yuriy Nevmyvaka, and Dongjin Song. Towards interpretable and trustworthy time series reasoning: A bluesky vision. *arXiv preprint arXiv:2510.16980*, 2025a.

Kanghui Ning, Zijie Pan, Yu Liu, Yushan Jiang, James Yiming Zhang, Kashif Rasul, Anderson Schneider, Lintao Ma, Yuriy Nevmyvaka, and Dongjin Song. Ts-rag: Retrieval-augmented generation based time series foundation models are stronger zero-shot forecaster. *arXiv preprint arXiv:2503.07649*, 2025b.

Peisong Niu, Ziqing Ma, Tian Zhou, Weiqi Chen, Lefei Shen, Rong Jin, and Liang Sun. Utilizing strategic pre-training to reduce overfitting: Baguan-a pre-trained weather forecasting model. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, 2025.

NVIDIA, :, Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, Daniel Dworakowski, Jiaojiao Fan, Michele Fenzi, Francesco Ferroni, Sanja Fidler, Dieter Fox, Songwei Ge, Yunhao Ge, Jinwei Gu, Siddharth Gururani, Ethan He, Jiahui Huang, Jacob Huffman, Pooya Jannaty, Jingyi Jin, Seung Wook Kim, Gergely Klár, Grace Lam, Shiyi Lan, Laura Leal-Taixe, Anqi Li, Zhaoshuo Li, Chen-Hsuan Lin, Tsung-Yi Lin, Huan Ling, Ming-Yu Liu, Xian Liu, Alice Luo, Qianli Ma, Hanzi Mao, Kaichun Mo, Arsalan Mousavian, Seungjun Nah, Sriharsha Niverty, David Page, Despoina Paschalidou, Zeeshan Patel, Lindsey Pavao, Morteza Ramezanali, Fitsum Reda, Xiaowei Ren, Vasanth Rao Naik Sabavat, Ed Schmerling, Stella Shi, Bartosz Stefaniak, Shitao Tang, Lyne Tchapmi, Przemek Tredak, Wei-Cheng Tseng, Jibin Varghese, Hao Wang, Haoxiang Wang, Heng Wang, Ting-Chun Wang, Fangyin Wei, Xinyue Wei, Jay Zhangjie Wu, Jiashu Xu, Wei Yang, Lin Yen-Chen, Xiaohui Zeng, Yu Zeng, Jing Zhang, Qinsheng Zhang, Yuxuan Zhang, Qingqing Zhao, and Artur Zolkowski. Cosmos world foundation model platform for physical ai. 2025. URL `https://arxiv.org/abs/2501.03575`.

Bokai Pan, Mingyue Cheng, Zhiding Liu, Shuo Yu, Xiaoyu Tao, Yuchong Wu, Qi Liu, Defu Lian, and Enhong Chen. Castflow: Learning role-specialized agentic workflows for time series forecasting. *arXiv preprint arXiv:2604.27840*, 2026.

Charidimos Papadakis, Angeliki Dimitriou, Giorgos Filandrianos, Maria Lymperaiou, Konstantinos Thomas, and Giorgos Stamou. Atlas: Adaptive trading with llm agents through dynamic prompt optimization and multi-agent coordination. *arXiv preprint arXiv:2510.15949*, 2025.

Gijun Park. Bridging temporal and textual modalities: A multimodal framework for automated cloud failure root cause analysis. *arXiv preprint arXiv:2601.04709*, 2026.

Felix Parker, Nimeesha Chan, Chi Zhang, and Kimia Ghobadi. Eliciting chain-of-thought reasoning for time series analysis using reinforcement learning. *arXiv preprint arXiv:2510.01116*, 2025.

Isha Patel and Iman Rahimi. Ai-based demand forecasting and load balancing for optimising energy use in healthcare systems: A real case study. *Systems*, 2026.

Oleeviya Babu Poikarayil, Cédric Schockaert, Abdulrahman Nahhas, Christian Daase, Mursal Dawodi, and Jawid Ahmad Baktash. Genautoml: An agentic framework for dynamic architecture generation and optimization in time-series analysis. *arXiv preprint arXiv:2606.05860*, 2026.

Willa Potosnak, Cristian Challu, Mononito Goswami, Kin G Olivares, Michał Wiliński, Nina Żukowska, and Artur Dubrawski. Investigating compositional reasoning in time series foundation models. *arXiv preprint arXiv:2502.06037*, 2025.

Ye Qiao, Haoxiang Zhang, Linfeng Wang, Cheng Chen, Tong Niu, Xueru Bai, and Yang Xiang. Multi-agent adversarial time series forecasting, 2025. URL `https://openreview.net/forum?id=lj1qwU6Gxs`.

Zhongzheng Qiao, Sheng Pan, Anni Wang, Viktoriya Zhukova, Yong Liu, Xudong Jiang, Qingsong Wen, Mingsheng Long, Ming Jin, and Chenghao Liu. It's time: Towards the next generation of time series forecasting benchmarks. *arXiv preprint arXiv:2602.12147*, 2026.

Zhenkai Qin, Qining Luo, Xunyi Nong, Xiaolong Chen, Hongfeng Zhang, and Cora Un In Wong. Mas-lstm: A multi-agent lstm-based approach for scalable anomaly detection in iiot networks. *Processes*, 13(3):753, 2025.

Paul Quinlan, Jeremy Levasseur, Qingguo Li, and Xiaodan Zhu. Chronicle: A multimodal foundation model for joint language and time series understanding. *arXiv preprint arXiv:2605.20268*, 2026.

Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Hena Ghonia, Rishika Bhagwatkar, Arian Khorasani, Mohammad Javad Darvishi Bayazi, George Adamopoulos, Roland Riachi, Nadhir Hassen, et al. Lag-llama: Towards foundation models for probabilistic time series forecasting. *arXiv preprint arXiv:2310.08278*, 2023.

Chidaksh Ravuru, Sagar Srinivas Sakhinana, and Venkataramana Runkana. Agentic retrieval-augmented generation for time series analysis. *arXiv preprint arXiv:2408.14484*, 2024.

Weilin Ruan and Yuxuan Liang. Visual reasoning over time series via multi-agent system. *arXiv preprint arXiv:2602.03026*, 2026.

Oisín Ryan, Jonas MB Haslbeck, and Lourens J Waldorp. Non-stationarity in time-series analysis: Modeling stochastic and deterministic trends. *Multivariate Behavioral Research*, 2025.

Hoon Sagong, Heesu Kim, and Hanbeen Hong. Hi-darts: Hierarchical dynamically adapting reinforcement trading system. *arXiv preprint arXiv:2509.12048*, 2025.

Oleksandr Shchur, Abdul Fatir Ansari, Caner Turkmen, Lorenzo Stella, Nick Erickson, Pablo Guerron, Michael Bohlke-Schneider, and Yuyang Wang. fev-bench: A realistic benchmark for time series forecasting. *arXiv preprint arXiv:2509.26468*, 2025.

Feifei Shi, Xueyan Yin, Kang Wang, Wanyu Tu, Qifu Sun, and Huansheng Ning. Large language models for time series analysis: Techniques, applications, and challenges. *arXiv preprint arXiv:2506.11040*, 2025a.

Haojie Shi, Yong Wang, and Lin Zhou. Sage: Spatiotemporal feature fusion for anomaly detection in multivariate time series. In *International Conference on Information Security and Cryptology*, pp. 143–156. Springer, 2025b.

Jiawen Shi, Zenghui Yuan, Guiyao Tie, Pan Zhou, Neil Zhenqiang Gong, and Lichao Sun. Prompt injection attack to tool selection in llm agents. *arXiv preprint arXiv:2504.19793*, 2025c.

Jingzhe Shi, Qinwei Ma, Huan Ma, and Lei Li. Scaling law for time series forecasting. *Advances in Neural Information Processing Systems*, 37:83314–83344, 2024.

Runze Shi, Shengyu Yan, Yuecheng Cai, and Chengxi Lv. Hubble: An llm-driven agentic framework for safe, diverse, and reproducible alpha factor discovery. *arXiv preprint arXiv:2604.09601*, 2026a.

Yuhan Shi, Yuanyuan Yao, Lu Chen, Mourad Khayati, and Tianyi Li. A hierarchical agent system with reinforcement learning for multivariate time series data cleaning. *arXiv preprint arXiv:2605.04902*, 2026b.

Geri Skenderi, Christian Joppi, Matteo Denitto, and Marco Cristani. Well googled is half done: Multimodal forecasting of new fashion product sales with image-based google trends. *Journal of Forecasting*, 43(6): 1982–1997, 2024.

Ireneous N Soyiri and Daniel D Reidpath. An overview of health forecasting. *Environmental health and preventive medicine*, 18(1):1–9, 2013.

Chen Su, Yuanhe Tian, and Yan Song. Chain-of-thought reviewing and correction for time series question answering. *arXiv preprint arXiv:2512.22627*, 2025.

Songyuan Sui, Zihang Xu, Yu-Neng Chuang, Kwei-Herng Lai, and Xia Hu. Training-free time series classification via in-context reasoning with llm agents. *arXiv preprint arXiv:2510.05950*, 2025.

Daniil Sukhorukov, Andrei Zakharov, Nikita Glazkov, Katsiaryna Yanchanka, Vladimir Kirilin, Maxim Dubovitsky, Roman Sultimov, Yury Maximov, and Ilya Makarov. Hierarchical ai-meteorologist: Llm-agent system for multi-scale and explainable weather forecast reporting. In *AAAI-26 AI for Environmental Science Workshop*, 2026.

Chenxi Sun, Hongyan Li, Yaliang Li, and Shenda Hong. Test: Text prototype aligned embedding to activate llm's ability for time series. In *International Conference on Learning Representations*, volume 2024, pp. 37854–37881, 2024.

Hui Sun, Chang Xu, Haonan Xie, Hao Li, Yuhao Huang, Chuheng Zhang, Ming Jin, Xiaoguang Liu, Gang Wang, and Jiang Bian. Chatad: Reasoning-enhanced time-series anomaly detection with multi-turn instruction evolution. *arXiv preprint arXiv:2601.13546*, 2026.

Siming Sun, Kai Zhang, Xuejun Jiang, Wenchao Meng, and Qinmin Yang. Enhancing llms for time series forecasting via structure-guided cross-modal alignment. *arXiv preprint arXiv:2505.13175*, 2025.

Zhao Tan, Yiji Zhao, Shiyu Wang, Chang Xu, Yuxuan Liang, Xiping Liu, Shirui Pan, and Ming Jin. Sonar-ts: Search-then-verify natural language querying for time series databases. *arXiv preprint arXiv:2602.17001*, 2026.

Yihong Tang, Andrew Robert Williams, Arjun Ashok, Vincent Zhihao Zheng, Lijun Sun, Alexandre Drouin, Issam H Laradji, Étienne Marcotte, and Valentina Zantedeschi. Dr-cik: A testbed for foresight-driven agents. *arXiv preprint arXiv:2605.27904*, 2026.

Xiaoyu Tao, Mingyue Cheng, Ze Guo, Shuo Yu, Yaguo Liu, Qi Liu, and Shijin Wang. Memcast: Memory-driven time series forecasting with experience-conditioned reasoning. *arXiv preprint arXiv:2602.03164*, 2026a.

Xiaoyu Tao, Mingyue Cheng, Chuang Jiang, Tian Gao, Huanjian Zhang, and Yaguo Liu. Cast-r1: Learning tool-augmented sequential decision policies for time series forecasting. *arXiv preprint arXiv:2602.13802*, 2026b.

Xiaoyu Tao, Yuchong Wu, Mingyue Cheng, Ze Guo, and Tian Gao. Anomamind: Agentic time series anomaly detection with tool-augmented reasoning. *arXiv preprint arXiv:2602.13807*, 2026c.

PAN Team, Jiannan Xiang, Yi Gu, Zihan Liu, Zeyu Feng, Qiyue Gao, Yiyan Hu, Benhao Huang, Guangyi Liu, Yichi Yang, Kun Zhou, Davit Abrahamyan, Arif Ahmad, Ganesh Bannur, Junrong Chen, Kimi Chen, Mingkai Deng, Ruobing Han, Xinqi Huang, Haoqiang Kang, Zheqi Liu, Enze Ma, Hector Ren, Yashowardhan Shinde, Rohan Shingre, Ramsundar Tanikella, Kaiming Tao, Dequan Yang, Xinle Yu, Cong Zeng, Binglin Zhou, Zhengzhong Liu, Zhiting Hu, and Eric P. Xing. Pan: A world model for general, interactable, and long-horizon world simulation. 2025. URL `https://arxiv.org/abs/2511.09057`.

Malik Tiomoko, Hamza Cherkaoui, Giuseppe Paolo, Zhang Yili, Yu Meng, Zhang Keli, and Hafiz Tiomoko Ali. Post-training corrections for improved time-series forecasting. *arXiv preprint arXiv:2505.15354*, 2025.

Patara Trirat and Jae-Gil Lee. Monaq: Multi-objective neural architecture querying for time-series analysis on resource-constrained devices. In *The 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 16922–16950. Association for Computational Linguistics (ACL), 2025.

Patara Trirat, Jin Myung Kwak, Jay Heo, Heejun Lee, and Sung Ju Hwang. Ts-debate: Multimodal collaborative debate for zero-shot time series reasoning. *arXiv preprint arXiv:2601.19151*, 2026.

Mariano Vargas-Santiago, Diana Assaely León-Velasco, Christian Efraín Maldonado-Sifuentes, and Liliana Chanona-Hernandez. A state-of-the-art review of artificial intelligence (ai) applications in healthcare: advances in diabetes, cancer, epidemiology, and mortality prediction. *Computers*, 2025.

Sky Chenwei Wan, Tianjun Hou, Yifei Wang, Xiqing Chang, and Aymeric Jan. Grammar of the wave: Towards explainable multivariate time series event detection via neuro-symbolic vlm agents. *arXiv preprint arXiv:2603.11479*, 2026.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345, 2024a.

Xinlei Wang, Maike Feng, Jing Qiu, Jinjin Gu, and Junhua Zhao. From news to forecast: Integrating event analysis in llm-based time series forecasting with reflection. *Advances in Neural Information Processing Systems*, 37:58118–58153, 2024b.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Yong Liu, Chen Wang, Mingsheng Long, and Jianmin Wang. Deep time series models: A comprehensive survey and benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026a.

Zhen Wang, Xi Zhou, Yating Yang, Bo Ma, Lei Wang, Rui Dong, and Azmat Anwar. Openforecast: A large-scale open-ended event forecasting dataset. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 5273–5294, 2025a.

Zhensheng Wang, Xiaole Liu, Wenmian Yang, Kun Zhou, Yiquan Zhang, and Weijia Jia. Odtqa-fore: An open-domain tabular question answering dataset for future data forecasting and reasoning. *arXiv preprint arXiv:2606.02433*, 2026b.

Zixiang Wang, Yinghao Zhu, Huiya Zhao, Xiaochen Zheng, Dehao Sui, Tianlong Wang, Wen Tang, Yasha Wang, Ewen Harrison, Chengwei Pan, et al. Colacare: Enhancing electronic health record modeling through large language model-driven multi-agent collaboration. In *Proceedings of the ACM on Web Conference 2025*, pp. 2250–2261, 2025b.

Michał Wawer and Jarosław Chudziak. Integrating traditional technical analysis with ai: A multi-agent llm-based approach to stock market forecasting. In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence*, pp. 100–111. SCITEPRESS - Science and Technology Publications, 2025. doi: 10.5220/0013191200003890. URL `http://dx.doi.org/10.5220/0013191200003890`.

Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun. Transformers in time series: A survey. *arXiv preprint arXiv:2202.07125*, 2022.

Muyan Weng, Defu Cao, Wei Yang, Yashaswi Sharma, and Yan Liu. Temporalbench: A benchmark for evaluating llm-based agents on contextual and event-informed time series tasks. *arXiv preprint arXiv:2602.13272*, 2026.

Andrew Robert Williams, Arjun Ashok, Étienne Marcotte, Valentina Zantedeschi, Jithendaraa Subramanian, Roland Riachi, James Requeima, Alexandre Lacoste, Irina Rish, Nicolas Chapados, et al. Context is key: A benchmark for forecasting with essential textual information. *arXiv preprint arXiv:2410.18959*, 2024.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Forty-first International Conference on Machine Learning*, 2024.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, volume 34, pp. 22419–22430, 2021.

Huizhe Wu, Wei Zhang, Weiwei Shen, and Jun Wang. Hybrid deep sequential modeling for social text-driven stock prediction. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pp. 1627–1630, 2018.

Jing Wu, Yang Liu, Lin Zhang, Junbo Zeng, Jiabin Wang, Zi Ye, Guowen Li, Shilei Cao, Jiashun Cheng, Fang Wang, et al. Agcd: Agent-guided cross-modal decoding for weather forecasting. *arXiv preprint arXiv:2603.15260*, 2026a.

Xingjian Wu, Junkai Lu, Zhengyu Li, Xiangfei Qiu, Jilin Hu, Chenjuan Guo, Christian S Jensen, and Bin Yang. Timeart: Towards agentic time series reasoning via tool-augmentation. *arXiv preprint arXiv:2601.13653*, 2026b.

Yihan Xia, Panpan You, Taotao Wang, Fang Liu, Han Qi, Xiaoxiao Wu, and Shengli Zhang. Agentic trading: When llm agents meet financial markets. *arXiv preprint arXiv:2605.19337*, 2026.

Mengxi Xiao, Zihao Jiang, Lingfei Qian, Zhengyu Chen, Yueru He, Yijing Xu, Yuecheng Jiang, Dong Li, Ruey-Ling Weng, Min Peng, Jimin Huang, Sophia Ananiadou, and Qianqian Xie. Retrieval-augmented large language models for financial time series forecasting. *arXiv preprint arXiv:2502.05878*, 2025a.

Yijia Xiao, Edward Sun, Di Luo, and Wei Wang. Tradingagents: Multi-agents llm financial trading framework. *arXiv preprint arXiv:2412.20138*, 2024.

Yijia Xiao, Edward Sun, Tong Chen, Fang Wu, Di Luo, and Wei Wang. Trading-r1: Financial trading with llm reasoning via reinforcement learning. *arXiv preprint arXiv:2509.11420*, 2025b.

Stephan Xie, Ben Cohen, Mononito Goswami, Junhong Shen, Emaad Khwaja, Chenghao Liu, David Asker, Othmane Abou-Amal, and Ameet Talwalkar. Arfbench: Benchmarking time series question answering ability for software incident response. *arXiv preprint arXiv:2604.21199*, 2026.

Zhe Xie, Zeyan Li, Xiao He, Longlong Xu, Xidao Wen, Tieying Zhang, Jianjun Chen, Rui Shi, and Dan Pei. Chatts: Aligning time series with llms via synthetic data for enhanced understanding and reasoning. *arXiv preprint arXiv:2412.03104*, 2024.

Binyan Xu, Xilin Dai, Di Tang, and Kehuan Zhang. One surrogate to fool them all: Universal, transferable, and targeted adversarial attacks with clip. In *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security*, pp. 3087–3101, 2025a.

Binyan Xu, Fan Yang, Xilin Dai, Di Tang, and Kehuan Zhang. Clip-guided backdoor defense through entropy-based poisoned dataset separation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 7415–7423, 2025b.

Binyan Xu, Xilin Dai, Fan Yang, and Kehuan Zhang. When agent automation becomes profitable: Quantifying and insuring autonomous ai risk through trace-economic underwriting. *arXiv preprint arXiv:2606.16465*, 2026a.

Binyan Xu, Xilin Dai, and Kehuan Zhang. Contextual agentic memory is a memo, not true memory. *arXiv preprint arXiv:2604.27707*, 2026b.

Binyan Xu, Dong Fang, Haitao Li, and Kehuan Zhang. From multi-agent to single-agent: When is skill distillation beneficial? *arXiv preprint arXiv:2604.01608*, 2026c.

Binyan Xu, Fan Yang, Xilin Dai, Di Tang, and Kehuan Zhang. From internal diagnosis to external auditing: A vlm-driven paradigm for online test-time backdoor defense. *arXiv preprint arXiv:2601.19448*, 2026d.

Binyan Xu, Fan Yang, Di Tang, Xilin Dai, and Kehuan Zhang. Breaking the stealth-potency trade-off in clean-image backdoors with generative trigger optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 27197–27205, 2026e.

Congluo Xu, Zhaobin Liu, and Ziyang Li. Finarena: A human-agent collaboration framework for financial market analysis and forecasting. *arXiv preprint arXiv:2503.02692*, 2025c.

Longkun Xu, Xiaochun Zhang, Qiantu Tuo, and Rui Li. Sea-ts: Self-evolving agent for autonomous code generation of time series forecasting algorithms. *arXiv preprint arXiv:2603.04873*, 2026f.

Wenyan Xu, Dawei Xiang, Yue Liu, Xiyu Wang, Yanxiang Ma, Liang Zhang, Shu Hu, Chang Xu, and Jiaheng Zhang. Finmultitime: A four-modal bilingual dataset for financial time-series analysis. *arXiv preprint arXiv:2506.05019*, 2025d.

Xiongxiao Xu, Haoran Wang, Yueqing Liang, Philip S Yu, Yue Zhao, and Kai Shu. Can multimodal llms perform time series anomaly detection? In *Proceedings of the ACM Web Conference 2026*, pp. 5392–5403, 2026g.

Yumo Xu and Shay B Cohen. Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1970–1979, 2018.

Zhijian Xu, Wanxu Cai, Xilin Dai, Zhaorong Deng, and Qiang Xu. Fidel-ts: A high-fidelity benchmark for multimodal time series forecasting. *arXiv preprint arXiv:2509.24789*, 2025e.

Hao Xue and Flora D Salim. Promptcast: A new prompt-based learning paradigm for time series forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 36(11):6851–6864, 2023.

Qi Yan, Raihan Seraj, Jiawei He, Lili Meng, and Tristan Sylvain. Autocast++: Enhancing world event prediction with zero-shot ranking-based context retrieval. In *International Conference on Learning Representations*, volume 2024, pp. 34523–34539, 2024.

Jie Yang, Yifan Hu, Kexin Zhang, Luyang Niu, Philip S Yu, and Kaize Ding. Revisiting multivariate time series forecasting with missing values. *arXiv preprint arXiv:2509.23494*, 2025a.

Jie Yang, Rui Zhang, Ziyang Cheng, Dawei Cheng, Guang Yang, and Bo Wang. Grad: Guided relation diffusion generation for graph augmentation in graph fraud detection. In *Proceedings of the ACM on Web Conference 2025*, pp. 5308–5319, 2025b.

Jie Yang, Yifan Hu, Yuante Li, Kexin Zhang, Kaize Ding, and Philip S Yu. From observations to states: Latent time series forecasting. *arXiv preprint arXiv:2602.00297*, 2026a.

Jie Yang, Kexin Zhang, Guibin Zhang, Philip S Yu, and Kaize Ding. Glocal information bottleneck for time series imputation. *Advances in Neural Information Processing Systems*, 38:104452–104484, 2026b.

Qingchuan Yang, Simon Mahns, Sida Li, Anri Gu, Jibang Wu, and Haifeng Xu. Llm-as-a-prophet: Understanding predictive intelligence with prophet arena. *arXiv preprint arXiv:2510.17638*, 2025c.

Tiankai Yang, Junjun Liu, Michael Siu, Jiahang Wang, Zhuangzhuang Qian, Chanjuan Song, Cheng Cheng, Xiyang Hu, and Yue Zhao. Ad-agent: A multi-agent framework for end-to-end anomaly detection. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pp. 191–205, 2025d.

Xu Yang, Xiao Yang, Shikai Fang, Yifei Zhang, Jian Wang, Bowen Xian, Qizheng Li, Jingyuan Li, Minrui Xu, Yuante Li, et al. R&d-agent: An llm-agent framework towards autonomous data science. *arXiv preprint arXiv:2505.14738*, 2025e.

Yiyuan Yang, Zichuan Liu, Lei Song, Kai Ying, Zhiguang Wang, Tom Bamford, Svitlana Vyetrenko, Jiang Bian, and Qingsong Wen. Time-ra: Towards time series reasoning for anomaly diagnosis with llm feedback. *arXiv preprint arXiv:2507.15066*, 2025f.

Yongbin Yang, Mengdie Wang, Jiyuan Wang, Pan Li, and Mengjie Zhou. Multi-agent deep reinforcement learning for integrated demand forecasting and inventory optimization in sensor-enabled retail supply chains. *Sensors*, 25(8):2428, 2025g.

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

Wen Ye, Wei Yang, Defu Cao, Yizhou Zhang, Lumingyuan Tang, Jie Cai, and Yan Liu. Ts-reasoner: Domain-oriented time series inference agents for reasoning and automated analysis. *Transactions on Machine Learning Research*, 2026.

Chin-Chia Michael Yeh, Vivian Lai, Uday Singh Saini, Xiran Fan, Yujie Fan, Junpeng Wang, Xin Dai, and Yan Zheng. Dcats: Empowering time series forecasting with LLM-agents. *arXiv preprint arXiv:2508.04231*, 2025.

Yao Yin, Zhenyu Xiao, Musheng Li, Yiwen Liu, Sutong Nan, Yiting He, Ruiqi Wang, Zhenwei Zhang, Qingmin Liao, and Yuantao Gu. Mmts-bench: A comprehensive benchmark for time series understanding and reasoning. *arXiv preprint arXiv:2602.08588*, 2026.

Fangxu Yu, Hongyu Zhao, and Tianyi Zhou. Ts-reasoner: Aligning time series foundation models with llm reasoning. *arXiv preprint arXiv:2510.03519*, 2025a.

Fangxu Yu, Xingang Guo, Lingzhi Yuan, Haoqiang Kang, Hongyu Zhao, Lianhui Qin, Furong Huang, Bin Hu, and Tianyi Zhou. Tsrbench: A comprehensive multi-task multi-modal time series reasoning benchmark for generalist models. *arXiv preprint arXiv:2601.18744*, 2026a.

Xiaoyun Yu, Li fan, Xiangfei Qiu, Nanqing Dong, Yonggui Huang, Honggang Qi, Geguang Pu, Wanli Ouyang, Xi Chen, and Jilin Hu. Memts: Internalizing domain knowledge via parameterized memory for retrieval-free domain adaptation of time series foundation models. *arXiv preprint arXiv:2602.13783*, 2026b.

Yangyang Yu, Zhiyuan Yao, Haohang Li, Zhiyang Deng, Yuechen Jiang, Yupeng Cao, Zhi Chen, Jordan W Suchow, Zhenyu Cui, Rong Liu, et al. Fincon: A synthesized llm multi-agent system with conceptual verbal reinforcement for enhanced financial decision making. *Advances in Neural Information Processing Systems*, 37:137010–137045, 2024.

Yangyang Yu, Haohang Li, Zhi Chen, Yuechen Jiang, Yang Li, Jordan W Suchow, Denghui Zhang, and Khaldoun Khashanah. Finmem: A performance-enhanced llm trading agent with layered memory and character design. *IEEE Transactions on Big Data*, 2025b.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 11121–11128, 2023.

Zhiyuan Zeng, Jiashuo Liu, Siyuan Chen, Tianci He, Yali Liao, Yixiao Tian, Jinpeng Wang, Zaiyuan Wang, Yang Yang, Lingyue Yin, et al. Futurex: An advanced live benchmark for llm agents in future prediction. *arXiv preprint arXiv:2508.11987*, 2025.

Tianxiang Zhan, Xiaobao Song, Tong Guan, Shirui Pan, and Ming Jin. Aion: Next-generation tasks and practical harness for time series. *arXiv preprint arXiv:2605.25045*, 2026.

Hanrong Zhang, Jingyuan Huang, Kai Mei, Yifei Yao, Zhenting Wang, Chenlu Zhan, Hongwei Wang, and Yongfeng Zhang. Agent security bench (ASB): Formalizing and benchmarking attacks and defenses in LLM-based agents. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL `https://openreview.net/forum?id=V4y0CpX4hK`.

Peng Zhang, Yapeng Ji, Qi Cheng, Yunji Liang, and Zhiwen Yu. Stockagent: A multi-agent collaborative framework for financial time series prediction. *2025 IEEE International Conference on Social Computing and Networking (SocialCom)*, pp. 8–17, 2025b.

Regina Zhang, Si Qi Goh, Xingsheng Chen, Zongru Li, Honggang Wen, Jiale Guo, Menglin Yang, Hongzhi Yin, Qiang Yang, SM Yiu, et al. From prompts to agents: A comprehensive survey of llm-driven time series analysis. *Available at SSRN 6614598*, 2025c.

Xiaohan Zhang, Tian Gao, Mingyue Cheng, Bokai Pan, Ze Guo, Yaguo Liu, Xiaoyu Tao, and Qi Liu. Alphacast: A human wisdom-llm intelligence co-reasoning framework for interactive time series forecasting. *arXiv preprint arXiv:2511.08947*, 2026a.

Xiyuan Zhang, Ranak Roy Chowdhury, Rajesh K Gupta, and Jingbo Shang. Large language models for time series: A survey. *arXiv preprint arXiv:2402.01801*, 2024.

Zhixing Zhang, Jesen Zhang, Hao Liu, Qinhan Lv, Jing Yang, Kaitong Cai, and Keze Wang. Agriworld: A world tools protocol framework for verifiable agricultural reasoning with code-executing llm agents. *arXiv preprint arXiv:2602.15325*, 2026b.

Zijian Zhang, Yujie Sun, Zepu Wang, Yuqi Nie, Xiaobo Ma, Ruolin Li, Peng Sun, and Xuegang Ban. Large language models for mobility analysis in transportation systems: A survey on forecasting tasks. *Transportation Research Record*, 2026c.

Haokun Zhao, Xiang Zhang, Jiaqi Wei, Yiwei Xu, Yuting He, Siqi Sun, and Chenyu You. Timeseriesscientist: A general-purpose ai agent for time series analysis. *arXiv preprint arXiv:2510.01538*, 2025.

Lecheng Zheng, Zhengzhang Chen, Dongjie Wang, Chengyuan Deng, Reon Matsuoka, and Haifeng Chen. Lemma-rca: A large multi-modal multi-domain dataset for root cause analysis. *arXiv preprint arXiv:2406.05375*, 2024.

Siru Zhong, Yuxuan Liang, Weilin Ruan, Ming Jin, Huan Li, and Qingsong Wen. Time-vlm: Exploring multimodal vision-language models for augmented time series forecasting. In *AI for Time Series*, pp. 173–203. CRC Press, 2025.

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

Jiahui Zhou, Dan Li, Boxin Li, Xiao Zhang, Erli Meng, Lin Li, Zhuomin Chen, Jian Lou, and See-Kiong Ng. Time series reasoning via process-verifiable thinking data synthesis and scheduling for tailored llm reasoning. *arXiv preprint arXiv:2602.07830*, 2026.

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

Shu Zhou, Yunyang Xuan, Yuxuan Ao, Xin Wang, Tao Fan, and Hao Wang. MERIT: Multi-agent collaboration for unsupervised time series representation learning. In *Findings of the Association for Computational Linguistics: ACL 2025*. Association for Computational Linguistics, 2025b.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International Conference on Machine Learning*, pp. 27268–27286. PMLR, 2022.

Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. One fits all: Power general time series analysis by pretrained lm. *Advances in neural information processing systems*, 36:43322–43355, 2023.

Xin Zhou, Weiqing Wang, Francisco J Baldán, Wray Buntine, and Christoph Bergmeir. Motime: A dataset suite for multimodal time series forecasting. *arXiv preprint arXiv:2505.15072*, 2025c.

Zihao Zhou and Rose Yu. Can llms understand time series anomalies? In *International Conference on Learning Representations*, volume 2025, pp. 1858–1896, 2025.

Peng Zhu, Yuante Li, Yifan Hu, Qinyuan Liu, Dawei Cheng, and Yuqi Liang. Lsr-igru: Stock trend prediction based on long short-term relationships and improved gru. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pp. 5135–5142, 2024.

Andy Zou, Tristan Xiao, Ryan Jia, Joe Kwon, Mantas Mazeika, Richard Li, Dawn Song, Jacob Steinhardt, Owain Evans, and Dan Hendrycks. Forecasting future world events with neural networks. *Advances in Neural Information Processing Systems*, 35:27293–27305, 2022.