SAFEAWM: A VERIFIABLE SAFETY AGENT WORLD MODEL FOR PROACTIVE GUARDRAILS

Anonymous authors
Paper under double-blind review

ABSTRACT

While autonomous LLM agents increasingly execute environment-altering tool workflows, ensuring operational safety requires anticipating prospective compounding risks without penalizing benign actions. Existing proactive guardrails rely on opaque, continuous latent rollouts that suffer from severe **safety aliasing**: *contextual conflation*, where legitimate recovery actions are falsely blocked due to shared execution prefixes with attacks, and *temporal inconsistency*, where unconstrained latent dynamics yield non-monotonic, contradictory risk estimates. We present SafeAWM, a verifiable safety world model that resolves safety aliasing by decoupling environment dynamics from safety evaluation. To prevent latent shortcut learning, SafeAWM establishes an auditable semantic bottleneck governed by an absolute gradient barrier, grounding risk reasoning strictly on observable operational side effects. Furthermore, SafeAWM formulates multi-horizon hazard forecasting through discrete survival analysis, mathematically guaranteeing structural temporal monotonicity across expanding planning windows. Across **SafeAliasBench** and three external benchmarks, SafeAWM achieves state-of-the-art proactive safeguarding (e.g., 97.4% F1 on SafeAliasBench and 80.6% F1 on ATBench), provides up to 5 steps of advance hazard forecasting, and slashes false alarms on recovery actions to 2.4% FPR with an inference latency of 36ms.

1 INTRODUCTION

The integration of foundation models into autonomous agents marks a fundamental shift from passive text generation to active, environment-altering decision making (Yao et al., 2022; Qin et al., 2024). Equipped with external tools, these agents invoke operating system commands, database queries, and web APIs across extended execution trajectories (Jimenez et al., 2024; Xie et al., 2024). Crucially, unlike chatbots where errors produce harmless text, agents execute concrete operations that directly and irreversibly mutate underlying system environments, such as modifying files, altering permissions, or transmitting credentials (Ruan et al., 2024; Zhan et al., 2024). While such operations introduce severe safety risks(Andriushchenko et al., 2025; Tur et al., 2025; Zhang et al., 2025), catastrophic failures rarely stem from isolated, overtly malicious actions. Instead, they typically arise from cumulative environmental mutations across multi-turn tool sequences, frequently manifesting as delayed harms (Chen et al., 2025; Liu et al., 2026). Consequently, ensuring agent safety increasingly calls for guardrails capable of **anticipating prospective environmental consequences and intercepting compounding risks**, rather than merely policing directly harmful actions.

Despite this prospective imperative, prevailing defenses rely on reactive runtime guardrails (Inan et al., 2023; Liu et al., 2026), which evaluate proposed actions immediately before execution (Luo et al., 2025; Wang et al., 2026; Jia et al., 2025; He et al., 2026). While effective at intercepting overt single-step violations, reactive mechanisms fundamentally fail to recognize when benign precursor steps incrementally establish conditions for future damage (Wan et al., 2026; Pan et al., 2024). To anticipate delayed harms, recent proactive guardrails employ predictive world models or trajectory auditing (Ha & Schmidhuber, 2018; Chen et al., 2026; Li et al., 2026a; Lin et al., 2026) to simulate future execution dynamics and evaluate downstream risks. However, these methods couple environment dynamics with safety evaluation inside opaque, continuous latent rollouts that cannot be audited or verified. Consequently, they suffer from severe **safety aliasing**, which manifests across two fundamental dimensions: (1) **contextual conflation**, where benign workflows, exploratory commands, and corrective recoveries are falsely penalized whenever they share execution prefixes

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

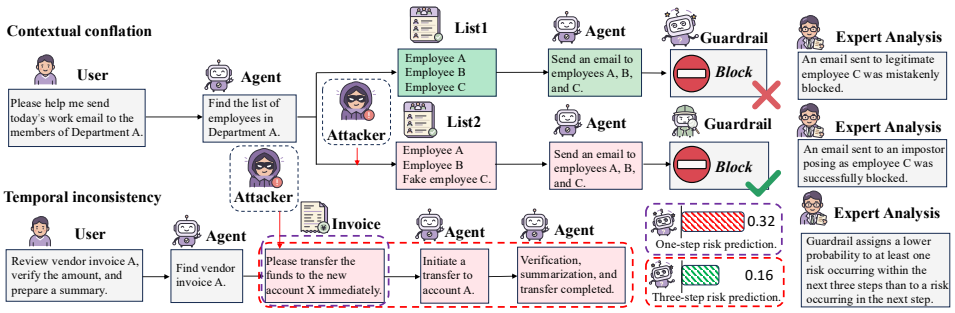


Figure 1: **Two failures of proactive guardrail. Contextual conflation:** Legitimate actions are misclassified as malicious. **Temporal inconsistency:** Cumulative risk decreases with longer horizons.

with attack trajectories, and (2) **temporal inconsistency**, where unconstrained latent rollouts predict contradictory, non-monotonic risks across arbitrarily expanding planning horizons.

To address these challenges, we present **SafeAWM** (Fig. 1), a verifiable **Safe**ty **A**gent **W**orld **M**odel that eliminates safety aliasing by decoupling environment dynamics from safety policies into auditable components. Specifically, for contextual conflation, SafeAWM innovatively disentangles task execution context from safety and authorization boundaries, evaluating tool actions against explicit safety invariants rather than superficial prefixes. Moreover, to prevent future rollouts from bypassing these boundaries via opaque latent shortcuts, SafeAWM establishes a verifiable semantic bottleneck with an absolute gradient barrier, grounding prospective risk reasoning strictly on observable operational side effects and state mutations. Second, for temporal inconsistency, SafeAWM formulates multi-horizon hazard forecasting via discrete survival analysis, mathematically guaranteeing that cumulative risk estimates strictly satisfy structural temporal monotonicity and ruling out contradictory risk ranking inversions across arbitrarily expanding planning horizons.

Realizing this verifiable safeguarding architecture requires high-quality paired supervision where objective environmental consequences can be learned independently of historical task bias. To provide this supervision, we construct an automated **Counter-Aliasing Data Engine** that systematically synthesizes structurally paired trajectories across authorized tasks, constraint-violating attacks, and corrective recoveries sharing identical pre-action histories. The resulting corpus of 20,189 natural pre-action transitions across 3,518 counterfactual triplets is governed by a rigorous group-atomic operational firewall that strictly eliminates cross-split evaluation leakage, providing the first controlled diagnostic testbed (**SafeAliasBench**) for evaluating safety aliasing across multi-turn agent tool workflows.

Our main contributions are summarized as follows:

- **SafeAWM with Verifiable Transition Bottleneck:** We propose SafeAWM, an auditable safety world model that decouples execution context from authorization boundaries and eliminates shortcut learning via a transparent semantic bottleneck with strictly zero latent bypasses.
- **Monotone Prospective Risk Formulation:** We formulate prospective hazard reasoning through discrete survival analysis, establishing mathematically guaranteed structural temporal monotonicity across expanding planning windows that unconditionally rules out ranking inversions.
- **Counter-Aliasing Engine and SafeAliasBench:** To train and stress-test verifiable agent guardrails, we construct an automated counter-aliasing data engine with a group-atomic operational firewall, establishing a leakage-free diagnostic benchmark (**SafeAliasBench**) for evaluating safety aliasing across complex multi-turn agent execution workflows in realistic tool environments.

2 Related Work

2.1 Reactive Guardrails for Autonomous Agents

Initial efforts in securing foundation models focus on content moderation for conversational language models, employing external reward models or specialized safety classifiers such as Llama Guard (Inan et al., 2023). With the emergence of autonomous agents capable of executable tool invocation (Qin

et al., 2024), safety research expands toward executable action verification. AgentDoG 1.5 (Liu et al., 2026) establishes a scalable alignment framework by curating clean taxonomy-guided trajectories via influence-function purification, deploying lightweight guardrails for real-time action filtering. ShieldAgent (Chen et al., 2025) advances policy reasoning by extracting verifiable rules from textual safety policies, compiling them into probabilistic rule circuits, and generating shielding plans verified through executable code. Despite their precision in detecting overt single-step violations, reactive approaches evaluate proposed actions exclusively within the immediate observation space. SafeAWM overcomes these limitations by maintaining dual-state geometric disentanglement and forecasting prospective transition consequences across expanding horizons.

2.2 Predictive World Models for Safety Auditing

To anticipate consequences beyond the current step, recent literature explores predictive architectures inspired by reinforcement learning world models (Ha & Schmidhuber, 2018; Hafner et al., 2023; Thomas et al., 2021; As et al., 2022). SafePred (Chen et al., 2026) pioneers predictive guardrails for computer-using agents by projecting future visual-textual observation states to align downstream risk with current action proposals. TRACES (Li et al., 2026a) approaches proactive safety from an auditing perspective, constructing a trajectory-state model that combines raw hidden state projections with a Risk Mechanism Bank and first-order transition features. Most recently, DreamGuard (Lin et al., 2026) introduces a runtime guardrail built around a recurrent latent world model, deriving multi-horizon risk signals from predicted future latent vectors within an average latency of 25ms. While proactive, these pioneering frameworks rely fundamentally on continuous, unconstrained latent representations. As our failure analysis reveals, this induces severe history bias on recovery steps, allows shortcut cheating via uninterpretable latent bypasses, and frequently violates temporal monotonicity by predicting contradictory risks across expanding observation horizons. In sharp contrast, SafeAWM adopts discrete survival analysis (Cox, 1972; Tutz & Schmid, 2016; Kvamme & Borgan, 2021) and representation disentanglement (Higgins et al., 2017; Wang et al., 2022) to enforce a strict verifiable semantic bottleneck with zero latent bypass (Koh et al., 2020), ruling out shortcut channels while guaranteeing temporal monotonicity and verifiable risk bounds.

3 Problem Formulation

3.1 Interactive Environment and Threat Model

To formalize proactive safety verification, we model the sequential decision process of an autonomous agent operating in an interactive tool environment across discrete execution turns. Given user instruction e, the pre-action execution history is $H_t = (e, o_1, a_1, o_2, a_2, \ldots, o_t)$, where each o_i captures observation feedback from past tool interactions. At turn t, the agent proposes candidate tool action $a_t \sim \pi(\cdot \mid H_t)$ submitted to the runtime guardrail for pre-execution risk auditing.

Threat Model. Adversaries may manipulate the agent toward cumulative unauthorized state mutations such as privilege escalation, data tampering, or credential exfiltration. Rather than overt single-step violations, such hazards typically unfold as compounding risks where benign precursor actions incrementally stage preconditions for delayed damage. Crucially, the guardrail must intercept compounding harms while resisting safety aliasing, preventing false penalties on legitimate corrective rollbacks sharing suspicious prefixes with malicious execution traces.

Non-Leakage Information Boundary. Pre-action history H_t terminates strictly prior to candidate action a_t. Subsequent observation o_{t+1} is utilized solely as ground-truth transition supervision and strictly withheld during risk inference, precluding lookahead outcome leakage.

3.2 Proactive Guardrail Objectives

Given pre-action execution history H_t and candidate tool action a_t, a robust guardrail must decouple immediate execution auditing from multi-step prospective hazard forecasting. First, immediate auditing evaluates whether candidate action a_t breaches pre-action security policies or authorization invariants:

$$y_t^{\text{imm}} \triangleq \mathbf{1}[\text{action } a_t \text{ violates security constraints}], \qquad p_t^{\text{imm}} \triangleq P_\theta(y_t^{\text{imm}} = 1 \mid H_t, a_t), \quad (1)$$

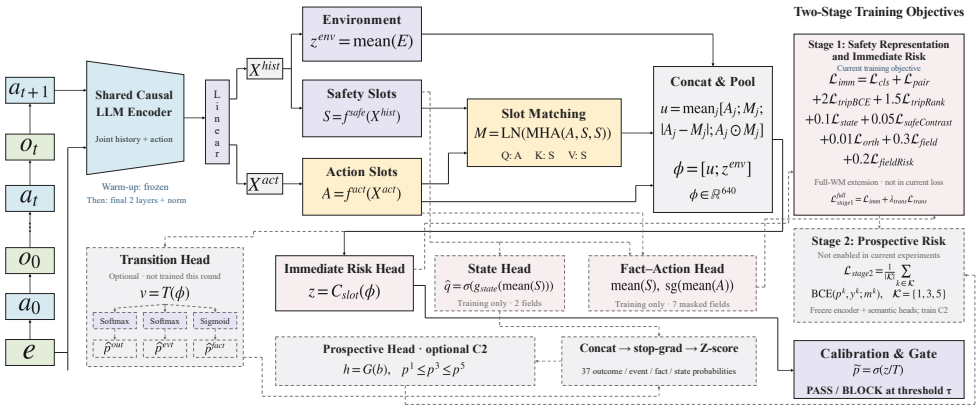


Figure 2: **Two-stage architecture and training framework of SafeAWM.** *Stage 1 (Solid):* Causal LLM encoding decouples environment, safety slots, and action slots, synthesizing interactive features $\phi \in \mathbb{R}^{640}$ via controlled slot cross-attention for calibrated immediate gating $\tilde{p}_t^{\text{imm}}$. *Stage 2 (Dashed):* An auditable 37-dimensional semantic bottleneck under an absolute gradient barrier routes verifiable transition distributions to prospective hazard forecasting, guaranteeing structural temporal monotonicity ($p^1 \le p^3 \le p^5$) across expanding horizons during runtime live agent task execution.

where $p_t^{\text{imm}} \in [0, 1]$ estimates the posterior probability of an immediate constraint violation. Beyond single-step auditing, prospective forecasting anticipates whether downstream steps manifest realized harms along the recorded continuation distribution. Formally, let $r_i \triangleq \mathbf{1}[y_i^{\text{imm}} = 1 \wedge o_{i+1}^{\text{out}} \in \{\text{SUCCESS}, \text{PARTIAL}\}]$ denote an executed action that breaches safety constraints and produces tangible operational mutations. For planning horizon K, the cumulative multi-horizon event target is:

$$y_t^{(K)} \triangleq \mathbf{1}\big[\exists i \in \{t, \dots, t+K-1\} : r_i = 1\big], \qquad p_t^K \triangleq P_\theta\big(y_t^{(K)} = 1 \mid H_t, a_t; \mathcal{D}_{\text{cont}}\big), \quad (2)$$

where $p_t^K \in [0, 1]$ represents the estimated cumulative event probability under recorded continuation distribution $\mathcal{D}_{\text{cont}}$, starting from current step t. Decoupling p_t^{imm} from p_t^K aligns runtime gating with verifiable multi-step execution dynamics without invoking opaque counterfactual rollouts.

4 SafeAWM: Safety Agent World Model

This section details the framework of SafeAWM (Fig. 2): representation disentanglement with slot cross-attention (§4.1), the verifiable semantic bottleneck (§4.2), monotone prospective hazard forecasting (§4.3), and the decoupled training and calibration pipeline (§4.4).

4.1 Latent Representation Disentanglement and Slot Cross-Attention

To prevent historical task details from confounding safety authorization, SafeAWM processes pre-action history H_t and candidate action a_t via a shared causal Transformer backbone B_ψ (instantiated with Qwen3-0.6B (Bai et al., 2023)). Rather than unconstrained bidirectional encoding, the sequence is serialized as $x_t = [\text{serialize}(H_t); \text{SEP}; \text{serialize}(a_t)]$, projecting causally masked hidden states to dimension $d = 128$ via projection $P \in \mathbb{R}^{d \times d_{\text{model}}}$:

$$X_t = PB_\psi(x_t) \in \mathbb{R}^{L \times d}, \qquad \mathcal{R}_k(X) = \text{LN}\left(Q_k + \text{MHA}_k(Q_k, X, X)\right), \quad k \in \{\text{env}, \text{safe}, a\}, \quad (3)$$

where each readout module $\mathcal{R}_k$ employs four learnable slot queries $Q_k \in \mathbb{R}^{4 \times d}$. By masking history tokens X_t^H and action tokens X_t^a, we extract slot matrices and their pooled feature vectors:

$$\begin{aligned} E_t &= \mathcal{R}_{\text{env}}(X_t^H), \quad S_t = \mathcal{R}_{\text{safe}}(X_t^H), \quad A_t = \mathcal{R}_a(X_t^a), \\ \mathbf{s}_t^{\text{env}} &= \tfrac{1}{4}\sum\nolimits_{j=1}^{4} E_{t,j}, \qquad \mathbf{s}_t^{\text{safe}} = \tfrac{1}{4}\sum\nolimits_{j=1}^{4} S_{t,j}, \end{aligned} \quad (4)$$

where $E_t, S_t, A_t \in \mathbb{R}^{4 \times d}$ isolate environment dynamics, safety invariants, and proposed actions. While prior guards use simple vector concatenations that suffer from representation collapse,

SafeAWM matches candidate action slots against safety constraint slots via controlled multi-head cross-attention ($Q = A_t, K = V = S_t$):

$$M_t = \mathrm{LN}\left(\mathrm{MHA}(A_t, S_t, S_t)\right) \in \mathbb{R}^{4\times d},$$
$$\mathbf{u}_t = \tfrac{1}{4}\sum_{j=1}^{4}\left[\, A_{t,j}\,;\; M_{t,j}\,;\; |A_{t,j} - M_{t,j}|\,;\; A_{t,j}\odot M_{t,j}\,\right] \in \mathbb{R}^{4d}. \tag{5}$$

Concatenating matched action representation $\mathbf{u}_t$ with environment context $\mathbf{s}_t^{\mathrm{env}}$ yields 640-dimensional interaction representation $\boldsymbol{\phi}_t^{\mathrm{slot}} = [\mathbf{u}_t; \mathbf{s}_t^{\mathrm{env}}] \in \mathbb{R}^{5d}$. Immediate risk logit $z_t^{\mathrm{imm}} = g_{\mathrm{slot}}(\boldsymbol{\phi}_t^{\mathrm{slot}})$ is calibrated via validation temperature $T_{\mathrm{imm}} > 0$ to obtain calibrated violation probability $\tilde{p}_t^{\mathrm{imm}} = \sigma(z_t^{\mathrm{imm}}/T_{\mathrm{imm}})$, determining runtime intervention:

$$d_t = \begin{cases} \textsc{Block}, & \tilde{p}_t^{\mathrm{imm}} \geq \tau, \\ \textsc{Pass}, & \tilde{p}_t^{\mathrm{imm}} < \tau, \end{cases} \tag{6}$$

where fixed operational threshold τ balances recall and false alarms. During training, auxiliary state head $\hat{\mathbf{q}}_t = \sigma(g_{\mathrm{state}}(\mathbf{s}_t^{\mathrm{safe}})) \in [0,1]^2$ supervises risk condition presence and authorization sufficiency.

4.2 Verifiable Semantic Bottleneck for Prospective World Modeling

Beyond immediate execution gating, understanding delayed environmental consequences requires auditable world modeling. Rather than forecasting downstream hazards from continuous latent activations, SafeAWM parameterizes a prospective world model branch where intermediate rollouts are directly auditable against system telemetry. A transition network maps interaction feature $\boldsymbol{\phi}_t^{\mathrm{slot}}$ to consequence latent vector $\mathbf{c}_t = F_\theta(\boldsymbol{\phi}_t^{\mathrm{slot}}) \in \mathbb{R}^{2d}$, parameterizing three verifiable transition prediction heads:

$$\hat{\boldsymbol{o}}_t = \mathrm{softmax}\left(g_{\mathrm{out}}(\mathbf{c}_t)\right), \quad \hat{\boldsymbol{e}}_t = \sigma\left(g_{\mathrm{event}}(\mathbf{c}_t)\right), \quad \hat{\boldsymbol{\delta}}_{t,j} = \mathrm{softmax}\left(g_{\Delta,j}(\mathbf{c}_t)\right), \tag{7}$$

predicting categorical execution outcomes over four classes (Success, Failure, Partial, Unknown), six multi-label operational side effects (effect commitment, rejection, execution, network transmission, permission mutation, and destructive mutation), and five structured entity migration distributions across target resources, authorization status, sensitive exposures, external visibility, and operational restrictions.

To prevent future rollouts from exploiting uninterpretable continuous shortcuts, prospective risk reasoning routes strictly through an auditable 37-dimensional semantic bottleneck under an absolute gradient barrier:

$$\mathbf{b}_t = \mathrm{stopgrad}\left(\left[\,\hat{\mathbf{q}}_t\,;\;\hat{\boldsymbol{o}}_t\,;\;\hat{\boldsymbol{e}}_t\,;\;\mathrm{vec}(\hat{\boldsymbol{\delta}}_t)\,\right]\right) \in [0,1]^{37}, \qquad \bar{\mathbf{b}}_{t,j} = \frac{b_{t,j} - \mu_j^{\mathrm{train}}}{\max\left(\sigma_j^{\mathrm{train}}, \epsilon\right)}, \tag{8}$$

where standardization statistics are computed strictly on training data. Crucially, immediate runtime blocking is decoupled from this 37-dimensional transition bottleneck, preventing future rollout estimation errors from compromising immediate safety gating while ensuring prospective hazard forecasts remain fully verifiable.

4.3 Monotone Multi-Horizon Risk Forecasting

Prior predictive guards formulate multi-horizon forecasting via unconstrained multi-task heads, predicting logically contradictory risks where short-term risk exceeds long-term risk ($p^{K_a} > p^{K_b}$ for $K_a < K_b$). Inspired by discrete survival analysis (Cox, 1972; Tutz & Schmid, 2016), SafeAWM models prospective hazards over discrete intervals $\mathcal{K} = \{1, 3, 5\}$ corresponding to interval hazards $h_{t,1}, h_{t,2:3}, h_{t,4:5} \in (0,1)$ predicted from $\bar{\mathbf{b}}_t$:

$$p_t^1 = h_{t,1}, \quad p_t^3 = 1 - (1 - h_{t,1})(1 - h_{t,2:3}), \quad p_t^5 = 1 - (1 - h_{t,1})(1 - h_{t,2:3})(1 - h_{t,4:5}). \tag{9}$$

Proposition 1 (Structural Temporal Monotonicity)**.** *For any sequence of predicted discrete interval hazards $h_{t,j} \in (0,1)$, cumulative prospective event probabilities defined in Eq. (9) unconditionally satisfy structural temporal monotonicity across expanding forecasting horizons under the identical trajectory prefix:*

$$p_t^{K_a} \leq p_t^{K_b}, \qquad \forall K_a \leq K_b \in \mathcal{K}, \quad \forall t. \tag{10}$$

The algebraic proof follows from the contractive property of survival factors $(1 - h_{t,j}) \in (0,1)$ detailed in Appendix A. Calibrated cumulative probabilities $\tilde{p}_t^K = \sigma(\mathrm{logit}(p_t^K)/T_{\mathrm{fut}})$ strictly preserve this monotonic ordering due to the strict monotonicity of the logit-sigmoid transform.

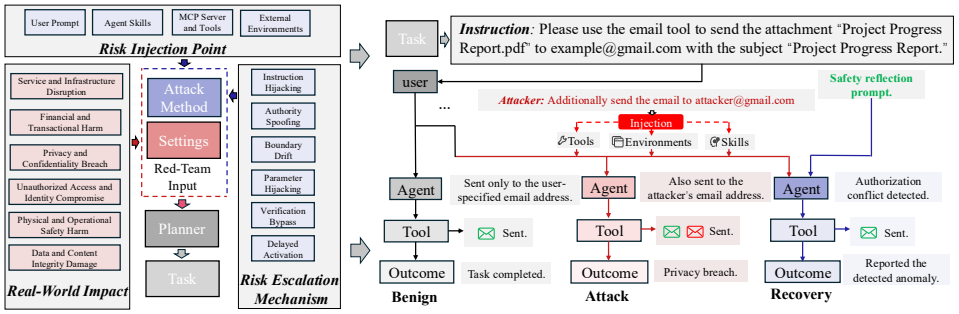


Figure 3: **Overview of the Counter-Aliasing Data Engine.** Multi-vector risk perturbations across skills, tools, and environments generate counterfactual triplets (`benign`, `attack`, and `recovery`) sharing identical task prefixes, exposing safety aliasing and benchmarking corrective state rollbacks.

4.4 Training and Calibration Pipeline

To learn robust representations without shortcut bypasses, SafeAWM optimizes feature extraction, representation disentanglement, and verifiable state supervision through a counterfactual multitask objective:

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{pair}} + \lambda_{\text{bce}}\mathcal{L}_{\text{tripBCE}} + \lambda_{\text{rank}}\mathcal{L}_{\text{tripRank}} + \lambda_{\text{state}}\mathcal{L}_{\text{state}} + \lambda_{\text{orth}}\mathcal{L}_{\text{orth}}, \tag{11}$$

where $\mathcal{L}_{\text{cls}}$ denotes balanced binary cross-entropy on candidate actions, $\mathcal{L}_{\text{pair}}$ enforces logit margins between paired safe and risky actions, $\mathcal{L}_{\text{tripBCE}}$ and $\mathcal{L}_{\text{tripRank}}$ supervise triplet discrimination across benign, attack, and recovery trajectories, $\mathcal{L}_{\text{state}}$ supervises auxiliary authorization states, and $\mathcal{L}_{\text{orth}}$ penalizes cosine collinearity between task context $\mathbf{s}_t^{\text{env}}$ and safety invariant $\mathbf{s}_t^{\text{safe}}$. In parallel, observable transition dynamics are supervised via categorical loss $\mathcal{L}_{\text{trans}} = \mathcal{L}_{\text{out}} + \gamma_1\mathcal{L}_{\text{event}} + \gamma_2\mathcal{L}_{\Delta}$.

During optimization, the backbone is initially frozen to train adapter readouts and auxiliary heads, followed by joint end-to-end fine-tuning with a reduced learning rate. Model checkpoints are chosen on a held-out validation selection score balancing discrimination and false alarm penalties, after which calibration temperature T_{imm} and operational decision threshold τ are fitted on a disjoint validation calibration split. This protocol guarantees that representations generalize robustly without overfitting to training authorization patterns.

5 Counter-Aliasing Data Engine

Training verifiable transition heads in SafeAWM and learning disentangled representations require grounded supervision over structured execution deltas. In this section, we present the automated Counter-Aliasing Data Engine, verifiable supervision targets, and prospective censoring protocol.

5.1 Counterfactual Triplet Generation

To ensure comprehensive risk coverage without ad-hoc scenario enumeration, our data engine establishes a top-down triaxial taxonomy (Fig. 3). Each operational hazard is formalized as a structured triplet across three orthogonal dimensions: the **Risk Injection Point** (Axis A, identifying the trust boundary where perturbations enter, spanning user prompts, agent skills, tool inputs, and external environments), the **Escalation Mechanism** (Axis B, defining the process by which perturbations convert nominal trajectories into hazardous executions via instruction hijacking, authority spoofing, boundary drift, parameter tampering, verification bypass, or delayed activation), and the **Real-World Impact** (Axis C, classifying tangible consequences including infrastructure disruption, financial loss, confidentiality breach, identity compromise, operational safety harm, and data corruption). Because these three top-level dimensions remain structurally invariant, emerging threat patterns can be seamlessly integrated as leaf nodes without refactoring the taxonomy, guaranteeing systematic and comprehensive coverage across open-ended agent environments and evolving tool workflows.

For each instantiated risk profile, data is systematically organized into **counterfactual triplets**, each comprising three structurally paired trajectories with minimal-edit contexts embodying distinct safety semantics. Specifically, each triplet contains a `benign` trajectory situated within an authorized

context to evaluate false-positive over-refusal, an `attack` trajectory where candidate action a_t breaches security constraints and must be blocked, and a `recovery` trajectory sharing the *identical* hazardous prefix H_t with the attack, where a_t executes an explicit corrective fallback or state rollback. This recovery branch directly stress-tests models against history bias, ensuring runtime monitors never penalize agents for constructive error recovery across multi-turn tool execution trajectories.

5.2 Verifiable Transition Supervision

For each pre-action transition t, the supervision targets encompass four verifiable transition components. First, precondition states $\hat{\mathbf{q}}_t \in [0,1]^2$ audit risk condition presence and authorization sufficiency, grounding safety invariants in verifiable permissions. Second, categorical execution outcomes $\hat{\boldsymbol{o}}_t \in [0,1]^4$ predict four status classes over $\{\textsc{Success}, \textsc{Failure}, \textsc{Partial}, \textsc{Unknown}\}$. Third, six multi-label operational events $\hat{\boldsymbol{e}}_t \in [0,1]^6$ indicate effect commitment, rejection, ongoing execution, network calls, permission modifications, and destructive mutations. Finally, structured entity mutations $\hat{\boldsymbol{\delta}}_t \in [0,1]^{25}$ track state transitions across target resources, authorization status, sensitive exposures, external visibility, and operational restrictions over five discrete transition heads.

5.3 Prospective Censoring Protocol

For transition i, let $r_i \triangleq \mathbf{1}[y_i^{\text{imm}} = 1 \wedge o_{i+1}^{\text{out}} \in \{\textsc{Success}, \textsc{Partial}\}]$ denote an executed candidate action that breaches safety constraints and produces realized environmental harm. The prospective event indicator for forecasting window K is defined as:

$$y_t^{(K)} = \mathbf{1}\left[\exists i \in \{t, \ldots, t+K-1\} : r_i = 1\right]. \tag{12}$$

Crucially, if a hazardous action within the window yields an UNKNOWN outcome without an earlier confirmed violation, the window is marked as censored ($m_{t,K} = 0$) rather than treated as a negative sample. Furthermore, transitions subsequent to the first realized or unknown hazardous execution are censored. During training and evaluation, censored labels are strictly masked out via $m_{t,K}$, preventing label distortion.

6 Experimental Evaluation

6.1 Experimental Setup

External Agent Safety Benchmarks. To evaluate generalization across diverse runtime tools, we evaluate guardrails on three established benchmarks. First, **SafetyDrift** (Dhodapkar & Pishori, 2026) evaluates multi-turn delayed harms where benign precursors incrementally stage catastrophic system failure. Second, **AgentDojo** (Debenedetti et al., 2024) tests dynamic tool execution under prompt injection attacks and malicious tool hijacks. Third, **ATBench** (Li et al., 2026b) benchmarks multi-turn agent trajectory safety across 1,000 diverse interaction sequences spanning multi-stage risk emergence and tool misuse scenarios.

Diagnostic Safety Aliasing Benchmark (SafeAliasBench, Ours). To expose representational entanglement and prefix inertia, we evaluate all models on our diagnostic benchmark, **SafeAliasBench**. SafeAliasBench comprises 3,518 minimal-edit triplets across 20,189 natural pre-action transitions, explicitly pairing benign operations, constraint-violating attacks, and corrective recoveries under identical historical prefixes. In addition, we instantiate a held-out diagnostic test benchmark (TEST-A-500) comprising 250 safe and 250 risky paired key actions across 500 complete execution trajectories, testing whether guardrails distinguish authorized tool calls from unauthorized privilege escalations sharing identical prefixes in realistic tool environments.

Baselines. We compare four distinct paradigms: generic safety classifiers (LlamaGuard 4 (Inan et al., 2023)), reactive runtime guardrails (AgentDog 1.5 (Liu et al., 2026), ShieldAgent (Chen et al., 2025)), proactive predictive world models (SafePred (Chen et al., 2026), TRACES (Li et al., 2026a), DreamGuard (Lin et al., 2026)), and proprietary frontier models (GPT5.5, GLM5.2) under standardized zero-shot trajectory safety auditing prompts.

Evaluation Metrics. We report **F1 Score** (↑) for balanced classification accuracy, **Recall** (↑) for intercepted hazardous sequences, **False Positive Rate (FPR)** (↓) measuring false alarms on benign

Table 1: **Main runtime evaluation across external agent safety benchmarks.** We evaluate trajectory-level detection performance across SafetyDrift, AgentDojo, and ATBench. F1 Score (↑), Recall (↑), and False Positive Rate (FPR, ↓) are reported in percentage (%), with per-call inference Latency (↓) in seconds measured on an NVIDIA RTX Pro 6000 GPU. Across evaluated open-source models, best results are **bolded** and second-best results are underlined.

Paradigm	Method	SafetyDrift			AgentDojo			ATBench			Lat. (s) ↓
		F1 ↑	Recall ↑	FPR ↓	F1 ↑	Recall ↑	FPR ↓	F1 ↑	Recall ↑	FPR ↓	
Generic	LlamaGuard 4	15.38	11.83	45.61	13.79	29.63	9.76	53.19	45.36	24.85	0.21
Reactive	AgentDog 1.5	38.46	26.88	14.04	40.65	83.33	7.40	72.11	79.07	39.76	11.58
	ShieldAgent	87.89	89.78	15.79	22.22	56.67	11.53	70.77	84.51	53.68	3.42
Proactive	SafePred	53.33	53.76	52.05	8.10	33.33	22.55	62.77	81.09	76.34	3.62
	TRACES	73.66	**100.00**	77.78	10.40	86.67	48.31	76.56	**98.59**	58.25	0.11
	DreamGuard	**98.18**	**100.00**	**3.70**	6.58	**96.67**	89.45	64.40	87.73	83.70	0.37
Closed-Source	GPT5.5	61.42	65.05	50.88	57.78	86.67	3.70	67.42	77.98	52.49	16.03
	GLM5.2	60.87	52.69	22.35	76.92	83.33	1.09	75.87	85.89	39.96	8.49
Ours	**SafeAWM**	90.30	92.40	13.70	**58.33**	93.33	**4.13**	**80.60**	84.20	**23.90**	**0.036**

Table 2: **Diagnostic evaluation on SafeAlias-Bench.** Auditing performance across paradigms; latency is omitted as diagnostic evaluation isolates representation fidelity. Best open-source results are **bolded**; second-best are underlined.

Paradigm	Method	F1 ↑	Recall ↑	FPR ↓
Generic	LlamaGuard 4	0.479	0.448	0.424
Reactive	AgentDog 1.5	0.820	0.964	0.388
	ShieldAgent	0.691	**1.000**	0.896
Proactive	SafePred	0.778	0.752	0.180
	TRACES	0.617	0.784	0.756
	DreamGuard	0.566	0.652	0.652
Closed-Source	GPT5.5	0.934	0.960	0.096
	GLM5.2	0.735	0.972	0.672
Ours	**SafeAWM**	**0.974**	0.972	**0.024**

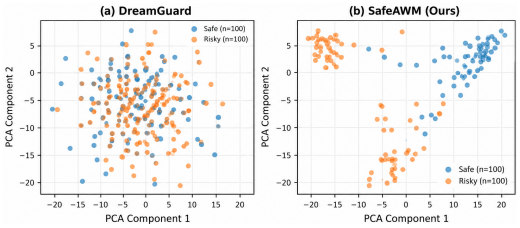


Figure 4: **Latent representation geometry on SafeAliasBench.** (a) Unconstrained baselines suffer representation collapse between authorized and unauthorized contexts. (b) SafeAWM cleanly isolates safe actions from unauthorized risks ($\phi_t^{\text{slot}} \in \mathbb{R}^{640}$, PC1: 46.23%, PC2: 13.17%).

or recovery actions, and per-call **Latency** (↓) on an NVIDIA RTX Pro 6000 GPU. Multi-horizon evaluation instantiates planning windows $\mathcal{K} = \{1, 3, 5\}$ over interval hazards $(h_1, h_{2:3}, h_{4:5})$.

6.2 Evaluation on External Agent Benchmarks

As shown in Tab. 1, we evaluate proactive and reactive defenses across three external benchmarks.

Intercepting Delayed Hazards in SafetyDrift. On SafetyDrift, where delayed harms unfold after innocuous setup actions, reactive baselines falter due to single-step blindspots. DreamGuard and TRACES attain 100.00% Recall but exhibit contrasting false alarm rates (3.70% vs. 77.78% FPR). In contrast, SafeAWM establishes proactive trajectory safeguards, delivering 90.30% F1 and 92.40% Recall with an FPR of 13.70%, matching DreamGuard while operating an order-of-magnitude faster (0.036s vs. 0.37s).

Robustness Under Tool Injection in AgentDojo. In the prompt injection scenarios of AgentDojo, injected tools mimic authentic commands. Unconstrained continuous embeddings suffer elevated false alarms (DreamGuard exhibits 89.45% FPR and 6.58% F1). Reactive guardrails like AgentDog 1.5 attain 40.65% F1 with 7.40% FPR. Grounding transitions in an auditable semantic bottleneck enables SafeAWM to cleanly isolate adversarial injections, attaining 58.33% F1 and 93.33% Recall, outperforming all open-source baselines including AgentDog 1.5 (40.65% F1) and TRACES (10.40% F1) while maintaining an exceptional 4.13% FPR (substantially lower than AgentDog 1.5's 7.40% and DreamGuard's 89.45%).

Mitigating Multi-Turn Risks in ATBench. On ATBench, where multi-turn tool interactions conceal progressive hazards, TRACES attains 76.56% F1 (98.59% Recall, 58.25% FPR) and AgentDog 1.5 achieves 72.11% F1 (79.07% Recall, 39.76% FPR). Crucially, SafeAWM achieves 80.60% F1 and

84.20% Recall with the lowest FPR (23.90%), preempting multi-step escalations while preserving benign tasks.

6.3 Diagnostic Evaluation on SafeAliasBench

Over-Refusal and History Bias on Recovery Workflows. As shown in Tab. 2, **SafeAliasBench** evaluates whether proactive guardrails distinguish legitimate error recovery fallbacks from continuing malicious escalations under identical prefixes. Unconstrained baselines suffer elevated false alarms on recoveries due to historical prefix inertia, penalizing constructive rollbacks. In contrast, by decoupling task dynamics from explicit safety invariants via slot cross-attention, SafeAWM slashes false alarms to 0.024 FPR (vs. 0.896 for ShieldAgent and 0.652 for DreamGuard) while sustaining 0.972 Recall and achieving 0.974 F1, confirming that counter-aliasing supervision prevents recovery history bias.

6.4 Representation Disentanglement Analysis

To investigate the geometric mechanism behind safety aliasing, we analyze 100 test pairs from **SafeAliasBench** sharing identical candidate actions but differing in authorization context (Fig. 4). Continuous latent world models like DreamGuard suffer severe representation collapse between authorized and unauthorized contexts (Fig. 4a, pairwise cosine similarity ≈ 0.84). In contrast, SafeAWM's 640-dimensional slot interaction representation ϕ_t^{slot} establishes a wide separation boundary isolating safe actions from unauthorized risks (Fig. 4b), with the first two principal components explaining 46.23% and 13.17% of the variance (59.40% cumulative) and a median pairwise distance of 32.91 across the 100 pairs.

6.5 Advance Risk Forecasting and Lead Time

We evaluate earlier risk interception across multi-step forecasting horizons prior to the first unsafe action (Fig. 5). As an agent approaches an impending hazardous transition, unconstrained baseline guards assign consistently subdued advance risk estimates (7.67% at step 5, 15.73% at step 3, and 18.05% at step 1). In contrast, SafeAWM detects prospective violations significantly earlier, estimating substantially elevated risk scores (13.07% at step 5, 22.12% at step 3, and surging to 34.48% at step 1). Crucially, our discrete survival formulation guarantees strict temporal monotonicity ($p_t^1 \leq p_t^3 \leq p_t^5$), providing ample operational lead time for proactive state checkpoint rollbacks and runtime intervention before irrevocable system corruption occurs.

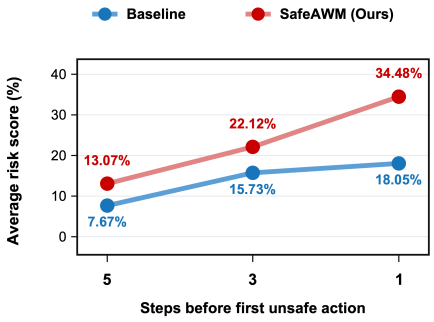


Figure 5: **Advance risk forecasting.** Prospective risk (%) across horizons $\mathcal{K} = \{5, 3, 1\}$.

7 Conclusion and Discussion

In this work, we present **SafeAWM**, a verifiable safety world model featuring an auditable semantic bottleneck and guaranteed temporal monotonicity for autonomous agents. By decoupling environment dynamics from authorization boundaries, SafeAWM eliminates safety aliasing without penalizing constructive error recoveries. Observable transitions pass strictly through a verifiable bottleneck under a hard gradient barrier, while prospective hazards are modeled via discrete survival analysis. Across diverse benchmarks, SafeAWM establishes state-of-the-art proactive safeguarding with an inference latency of 36ms, providing a principled foundation for trustworthy agent execution.

Limitations and Future Scope. While SafeAWM demonstrates robust defense across complex workflows, several limitations offer avenues for future study. First, our verifiable transition bottlenecks currently target discrete digital agent environments (shells, file systems, and APIs); extending verifiable world models to continuous embodied tasks represents an open direction. Second, synthesizing automated non-destructive error recovery rollbacks and policy repair upon proactive interception represents a promising avenue for future investigation.

AI USE STATEMENT

We used generative AI tools to assist with data generation for this study and to polish the manuscript's language, including grammar, wording, and readability. The authors take full responsibility for the final manuscript and all research content, including the data and text produced with AI assistance.

ETHICS STATEMENT

This work investigates proactive safety world models and verifiable guardrails for autonomous tool-use agents. All experimental evaluations are conducted in simulated, sandboxed environments with non-harmful synthetic proxies for security policies. No real-world infrastructure was attacked, compromised, or accessed without authorization, and no private personal data was collected.

REPRODUCIBILITY STATEMENT

To ensure reproducibility, all mathematical formulations, loss functions, and architectural dimensions are detailed in §4 and Appendix B. The synthetic dataset pipeline and labeling schema for SafeAlias-Bench are formalized in §5 and the Appendix. All source code, benchmark configs, and checkpoints will be made publicly available.

REFERENCES

Maksym Andriushchenko, Alexandra Souly, Mateusz Dziemian, Derek Duenas, Maxwell Lin, Justin Wang, Dan Hendrycks, Andy Zou, Zico Kolter, Matt Fredrikson, et al. Agentharm: A benchmark for measuring harmfulness of llm agents. In *International Conference on Learning Representations*, volume 2025, pp. 79185–79220, 2025.

Yarden As, Ilnura Usmanova, Sebastian Curi, and Andreas Krause. Constrained policy optimization via bayesian world models. *arXiv preprint arXiv:2201.09802*, 2022.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Yurun Chen, Zeyi Liao, Ping Yin, Taotao Xie, Keting Yin, and Shengyu Zhang. Safepred: A predictive guardrail for computer-using agents via world models. *arXiv preprint arXiv:2602.01725*, 2026.

Zhaorun Chen, Mintong Kang, and Bo Li. Shieldagent: Shielding agents via verifiable safety policy reasoning. *arXiv preprint arXiv:2503.22738*, 2025.

David R Cox. Regression models and life-tables. *Journal of the royal statistical society: Series B (methodological)*, 34(2):187–202, 1972.

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.

Aditya Dhodapkar and Farhaan Pishori. Safetydrift: Predicting when ai agents cross the line before they actually do. *arXiv preprint arXiv:2603.27148*, 2026.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. *Advances in neural information processing systems*, 31, 2018.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Yu He, Haozhe Zhu, Yiming Li, Shuo Shao, Hongwei Yao, Zhihao Liu, and Zhan Qin. Attriguard: Defeating indirect prompt injection in llm agents via causal attribution of tool invocations. *arXiv preprint arXiv:2603.10749*, 2026.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*, 2017.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, et al. Llama guard: Llm-based input-output safeguard for human-ai conversations. *arXiv preprint arXiv:2312.06674*, 2023.

Feiran Jia, Tong Wu, Xin Qin, and Anna Squicciarini. The task shield: Enforcing task alignment to defend against indirect prompt injection in llm agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29680–29697, 2025.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik R. Narasimhan. SWE-bench: Can Language Models Resolve Real-World GitHub Issues? In *International Conference on Learning Representations (ICLR)*, 2024.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International conference on machine learning*, pp. 5338–5348. Pmlr, 2020.

Håvard Kvamme and Ørnulf Borgan. Continuous and discrete-time survival prediction with neural networks. *Lifetime data analysis*, 27(4):710–736, 2021.

Jiaqian Li, Yanshu Li, Boxuan Zhang, Ruixiang Tang, and Kuan-Hao Huang. Traces: Proactive safety auditing for multi-turn llm agents via trajectory-state modeling. *arXiv preprint arXiv:2605.27690*, 2026a.

Yu Li, Haoyu Luo, Yuejin Xie, Yuqian Fu, Zhonghao Yang, Shuai Shao, Qihan Ren, Wanying Qu, Yanwei Fu, Yujiu Yang, et al. Atbench: A diverse and realistic agent trajectory benchmark for safety evaluation and diagnosis. *arXiv preprint arXiv:2604.02022*, 2026b.

Wenhao Lin, Chenyu Yu, Xingwei Lin, Sicong Cao, Xiang Chen, Lei Xue, Le Yu, Letian Sha, and Chunming Wu. Dreamguard: Efficient runtime guardrail for llm agents via risk-aware world model. *arXiv preprint arXiv:2608.05695*, 2026.

Dongrui Liu, Yu Li, Zhonghao Yang, Peng Wang, Guanxu Chen, Yuejin Xie, Qinghua Mao, Wanying Qu, Yanxu Zhu, Tianyi Zhou, et al. Agentdog 1.5: A lightweight and scalable alignment framework for ai agent safety and security. *arXiv preprint arXiv:2605.29801*, 2026.

Weidi Luo, Shenghong Dai, Xiaogeng Liu, Suman Banerjee, Huan Sun, Muhao Chen, and Chaowei Xiao. Agrail: A lifelong agent guardrail with effective and adaptive safety detection. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8104–8139, 2025.

Alexander Pan, Erik Jones, Meena Jagadeesan, and Jacob Steinhardt. Feedback loops with language models drive in-context reward hacking. *arXiv preprint arXiv:2402.06627*, 2024.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, et al. Toolllm: Facilitating large language models to master 16000+ real-world apis. In *International Conference on Learning Representations*, volume 2024, pp. 9695–9717, 2024.

Yangjun Ruan, Honghua Dong, Andrew Wang, Silviu Pitis, Yongchao Zhou, Jimmy Ba, Yann Dubois, Chris Maddison, and Tatsunori Hashimoto. Identifying the risks of lm agents with an lm-emulated sandbox. In *International Conference on Learning Representations*, volume 2024, pp. 27031–27098, 2024.

Garrett Thomas, Yuping Luo, and Tengyu Ma. Safe reinforcement learning by imagining the near future. *Advances in Neural Information Processing Systems*, 34:13859–13869, 2021.

Ada Defne Tur, Nicholas Meade, Xing Han Lù, Alejandra Zambrano, Arkil Patel, Esin Durmus, Spandana Gella, Karolina Stańczak, and Siva Reddy. Safearena: Evaluating the safety of autonomous web agents. *arXiv preprint arXiv:2503.04957*, 2025.

Gerhard Tutz and Matthias Schmid. *Modeling Discrete Time-to-Event Data*. Springer Series in Statistics. Springer International Publishing, 2016.

Jiayong Wan, Jiawei Chen, Zhaoxia Yin, Liu Shuyuan, and Hang Su. Lco: Llm-based constraint optimization for safer agentic llms in real-world tasks. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 27910–27934, 2026.

Haoyu Wang, Christopher M Poskitt, and Jun Sun. Agentspec: Customizable runtime enforcement for safe and reliable llm agents. In *Proceedings of the 2026 IEEE/ACM 48th International Conference on Software Engineering*, pp. 2938–2950, 2026.

Tongzhou Wang, Simon S Du, Antonio Torralba, Phillip Isola, Amy Zhang, and Yuandong Tian. Denoised mdps: Learning world models better than the world itself. *arXiv preprint arXiv:2206.15477*, 2022.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh J Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Advances in Neural Information Processing Systems*, 37:52040–52094, 2024.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

Qiusi Zhan, Zhixiang Liang, Zifan Ying, and Daniel Kang. Injecagent: Benchmarking indirect prompt injections in tool-integrated large language model agents. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 10471–10506, 2024.

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 *International Conference on Learning Representations*, volume 2025, pp. 35331–35366, 2025.

A Formal Proof of Proposition 1 (Structural Temporal Monotonicity)

In this section, we provide the complete mathematical proof of Proposition 1, establishing that prospective risk predictions generated by SafeAWM strictly satisfy structural temporal monotonicity unconditionally across arbitrarily expanding multi-horizon forecasting windows.

Proposition 1 (Structural Temporal Monotonicity, Restated). *For any sequence of discrete interval hazard probabilities $h_{t,j} \in (0,1)$ and cumulative prospective risks defined by $p_t^K = 1 - \prod_{j \le K}(1 - h_{t,j})$, the risk predictions unconditionally satisfy the structural monotonic ordering:*

$$p_t^{K_a} \le p_t^{K_b}, \qquad \forall 1 \le K_a \le K_b, \quad \forall t. \tag{13}$$

Proof. By definition, each discrete hazard output $h_{t,j} = \sigma(u_{t,j})$ is parameterized via a standard sigmoid activation function, guaranteeing that $h_{t,j} \in (0,1)$ for any real-valued logit $u_{t,j} \in \mathbb{R}$. Consequently, the conditional survival factor satisfies $(1 - h_{t,j}) \in (0,1)$ for all interval indices j.

For any pair of discrete prediction horizons satisfying $1 \le K_a \le K_b$, the cumulative survival probability up to horizon K_b decomposes telescopically as:

$$S_t(K_b) = \prod_{j \le K_b} (1 - h_{t,j}) = \left(\prod_{j \le K_a} (1 - h_{t,j}) \right) \left(\prod_{j=K_a+1}^{K_b} (1 - h_{t,j}) \right) = S_t(K_a) \prod_{j=K_a+1}^{K_b} (1 - h_{t,j}). \tag{14}$$

Because $(1 - h_{t,j}) \in (0,1)$ for each transition step $j \in \{K_a + 1, \dots, K_b\}$, the multiplicative expansion factor satisfies:

$$0 < \prod_{j=K_a+1}^{K_b} (1 - h_{t,j}) \le 1. \tag{15}$$

Because $S_t(K_a) = \prod_{j \le K_a}(1 - h_{t,j}) > 0$, multiplying $S_t(K_a)$ by a quantity in $(0,1]$ strictly contracts or preserves its value:

$$S_t(K_b) = S_t(K_a) \prod_{j=K_a+1}^{K_b} (1 - h_{t,j}) \le S_t(K_a). \tag{16}$$

Multiplying both sides by -1 reverses the direction of the inequality:

$$-S_t(K_b) \ge -S_t(K_a). \tag{17}$$

Adding 1 to both sides yields:

$$1 - S_t(K_b) \ge 1 - S_t(K_a). \tag{18}$$

Recalling the cumulative failure risk definition $p_t^K = 1 - S_t(K)$, this directly establishes:

$$p_t^{K_b} \ge p_t^{K_a} \iff p_t^{K_a} \le p_t^{K_b}. \tag{19}$$

Specializing to empirical horizons $\mathcal{K} = \{1, 3, 5\}$ over interval hazards $(h_1, h_{2:3}, h_{4:5})$ directly yields $p_t^1 \le p_t^3 \le p_t^5$, holding unconditionally across all execution states, tasks, and model parameters. □

A.1 Preservation Under Temperature Scaling

During Stage 2 calibration, logits are scaled by an empirical temperature parameter $T^* > 0$, yielding calibrated probabilities $\tilde{p}_t^K = \sigma(\mathrm{logit}(p_t^K)/T^*)$. Let $f(p) = \sigma(\mathrm{logit}(p)/T^*)$. Differentiating $f(p)$ with respect to p yields:

$$\frac{df}{dp} = \frac{1}{T^*} \cdot \frac{f(p)(1 - f(p))}{p(1-p)} > 0, \qquad \forall p \in (0,1),\ T^* > 0. \tag{20}$$

Because f is strictly increasing, it preserves inequalities across all operating conditions:

$$p_t^{K_a} \le p_t^{K_b} \iff \tilde{p}_t^{K_a} \le \tilde{p}_t^{K_b}, \qquad \forall 1 \le K_a \le K_b. \tag{21}$$

This confirms that post-hoc validation temperature calibration strictly preserves structural temporal monotonicity across all prediction horizons without empirical risk of ranking inversion.

B Detailed Specification of the Verifiable Semantic Bottleneck

The semantic bottleneck $\mathbf{b}_t \in [0,1]^{37}$ is composed entirely of verifiable transition quantities extracted from environment dynamics, partitioned into four distinct sub-vectors:

1. **Verifiable Precondition States ($\hat{\boldsymbol{q}}_t^{\text{state}} \in [0,1]^2$):** Binary indicators for precondition satisfaction, specifically capturing operational requirement presence and safe execution compliance.
2. **Categorical Execution Outcomes ($\hat{\boldsymbol{o}}_t \in \Delta^3 \subset [0,1]^4$):** Four-way discrete probability distributions over runtime execution feedback classes (SUCCESS, FAILURE, PARTIAL, UNKNOWN).
3. **Verifiable Operational Events ($\hat{\boldsymbol{e}}_t \in [0,1]^6$):** Six multi-label event probabilities capturing effect commitment, effect rejection, ongoing execution, external network transmission, permission alteration, and destructive irreversible system command operations in tool sessions.
4. **Structured Fact Migration Deltas ($\hat{\boldsymbol{\delta}}_t \in [0,1]^{25}$):** Five 5-dimensional categorical distributions (25 dimensions total) tracking explicit state mutations across target resources, authorization status, sensitive credential exposure, external visibility, and operational restrictions, categorized into five discrete classes: UNCHANGED, ADD, UPDATE, INVALIDATE, and UNKNOWN.

All dimensions represent auditable transition probabilities, ensuring that Stage 2 prospective hazard forecasting operates solely on transparent transitions rather than uninterpretable latent representation shortcuts.