FOREWARN: RISK-ADAPTIVE INFERENCE FOR PROACTIVE SAFETY WARNING IN STREAMING VIDEO

Anonymous authors
Paper under double-blind review

ABSTRACT

Streaming video models now describe what is happening in front of a camera, and proactive safety warning asks them to act on what is about to happen. However, existing warning methods decide from the current state, and confirming a hazard once it is visible consumes part of the time left to warn. Our diagnostics reveal that serial confirmation filters out early candidates and that a later confirmation recognizes more but returns too late to warn. In this paper, we propose ForeWarn, which lets the near-future risk predicted by a small vision-language model decide both when to warn and when to verify, so that a forecast acts before confirmation where time is short and waits for it where time allows. The same risk read at two horizons drives an immediate path that does not wait for a verifier and an advisory path that combines predicted risk with a returned state judgment; a risk-dependent schedule decides when to query the verifier. Furthermore, calibration on normal footage sets the advisory thresholds per deployment, and a one-step analysis ties the value of verification to confirmation quality and timely return. Extensive experiments on five public domains, self-recorded live trials, and three Jetson edge devices demonstrate that ForeWarn raises macro pre-incident recall to 1.4× that of the strongest of four recent baselines at similar verifier cost, with gains of up to 47.7 points on a single domain and up to 110× less verifier time per clip.

1 INTRODUCTION

Vision-language models (VLMs) have moved from describing recorded clips to answering while a stream is still running (Chen et al., 2024a; Qian et al., 2025; Yang et al., 2025; Wang et al., 2025a; Li et al., 2025; Zhang et al., 2026). Proactive safety warning extends this ability from reporting what is happening to anticipating what is about to happen, so that a home camera, a sports feed, or a wearable says that a fall or a collision is imminent while there is still time to intervene (Yuan et al., 2025; Zhao et al., 2026a; Qiu et al., 2026; Panpatil et al., 2026; Zhao et al., 2025). Such a warning rests on a forecast made before the incident and on a decision to act on it. However, the hazard is not yet visible when the warning is due, so the evidence that would settle the decision arrives only as the time to act runs out (Li et al., 2020; Eyzaguirre et al., 2024; Ma et al., 2026). As a result, deciding when to act on a forecast has become the central problem of proactive safety warning.

Existing warning methods mostly decide from the current state (Zhao et al., 2025; Panpatil et al., 2026; Yuan et al., 2025). A VLM is asked whether what is visible now already counts as a hazard, and its answer is reliable once the hazard is in view. However, by then most of the time left to warn is gone, and the confirmation itself consumes part of what remains. A forecast-driven alternative anticipates the incident from an incomplete precursor (Epstein et al., 2020; Liu et al., 2026; Qiu et al., 2026). Selective computation then decides when a stronger model is worth its cost (Chen et al., 2024b; Gupta et al., 2024; Geifman & El-Yaniv, 2017; Jitkrittum et al., 2023; Snell et al., 2025; Yue et al., 2024). Acting on the forecast lets such systems warn before the hazard is visible, and calling the stronger model selectively keeps verification cheap.

The closest systems combine a forecast with selective confirmation. StreamSense escalates from lightweight perception to a VLM selectively and defers when evidence is insufficient (Wang et al., 2026a). However, its first stage only decides whether to escalate, and every warning waits for the VLM. PaSBench-Video, SPRINT, and EgoSafetyBench study proactive safety warnings with their own confirmation rules (Zhao et al., 2026a; Qiu et al., 2026; Panpatil et al., 2026). However, every

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.

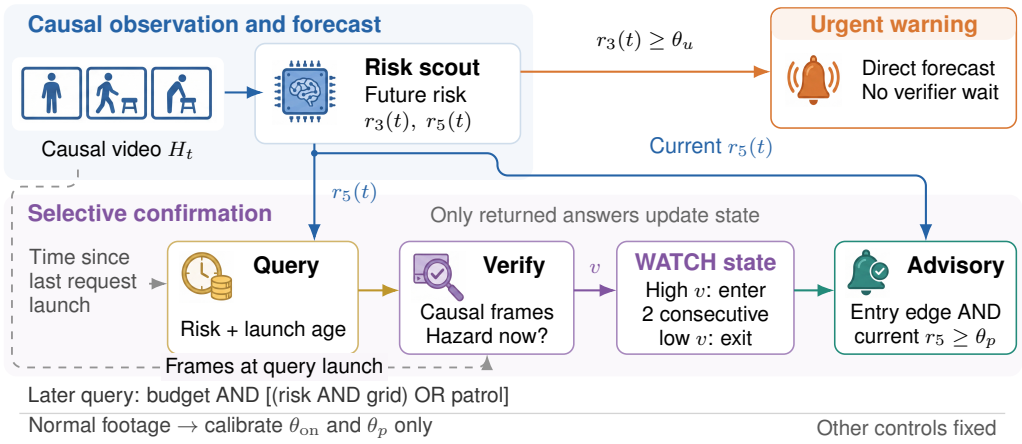


Figure 1: ForeWarn's two warning paths: an immediate warning needs no verifier; an advisory needs a verifier-confirmed hazardous state (a WATCH entry) and high predicted risk at the same time.

warning in them waits for the verifier's answer, so the time the forecast bought is spent confirming what it predicted. Our diagnostics in Section 3.2 show that the verifier confirms only 73.1% of the windows flagged 2–8 seconds before the incident against 94.0% of those flagged in its final half-second, and that the evidence it gains by waiting arrives after the moment to act.

Despite this promise, three challenges remain unresolved once the forecast is to decide when to warn and when to verify. First, a forecast and a confirmation do different jobs. Gating every forecast behind a verifier filters out the early candidates that the forecast is meant to find, while warning on every forecast floods the deployment with false alerts (Geifman & El-Yaniv, 2017; Jitkrittum et al., 2023). Thus, the first challenge is to decide which warnings may bypass confirmation. Second, each verification costs time and compute, and its answer is useful only while time remains (Frazier & Yu, 2007; Banerjee & Veeravalli, 2012). A periodic schedule spends most of its calls on quiet footage, a call launched once the hazard is visible returns too late, and a returned answer ages as the scene changes. Therefore, the second challenge is to decide when to query the verifier and how often to repeat the query. Third, a threshold that suits one camera floods another with false alerts, and a new site offers only routine normal footage, which carries no hazard labels to tune against (Vovk et al., 2005; Angelopoulos & Bates, 2023). Thus, the third challenge is to set the warning thresholds per deployment without hazard labels.

In this paper, we propose ForeWarn, a risk-adaptive inference method for proactive safety warning in streaming video. The core idea is to read the near-future risk that a small VLM predicts at two horizons and to give each reading its own job, so that a forecast warns on its own where confirmation would return too late and asks for confirmation where time remains. Specifically, as shown in Figure 1, high short-horizon risk activates an immediate warning path whose decision cannot be vetoed by the verifier, while a complementary advisory path warns only when the longer-horizon risk and a returned hazardous-state judgment agree. The verifier is queried on a schedule set by the same longer-horizon risk and the age of the last answer. Calibration on normal footage alone then sets the advisory thresholds for each deployment. We evaluate ForeWarn against four recent baselines (Zhao et al., 2026a; Qiu et al., 2026; Wang et al., 2026a; Panpatil et al., 2026) on five public domains (Schneider et al., 2026; Zhang et al., 2014; Epstein et al., 2020; Zhao et al., 2025), in self-recorded live trials, and on three Jetson tiers. Under measured latency the immediate path carries the fast incidents and the advisory path the slow home hazards, and verifier time stays close to the cheapest baseline's. The main contributions are summarized as follows.

- We measure two failure modes of confirmation-gated warning on public footage: the verifier filters out early forecasts, and the confident verdicts it reaches by waiting return too late.
- We propose ForeWarn, in which one forecast read at two horizons controls the verification schedule, an immediate warning path that needs no confirmation, and an advisory path gated by a returned state judgment.

- We give a one-step analysis of when a verification call is worth its delay, and a per-deployment calibration that sets the advisory thresholds from normal footage alone.
- On five public domains and live trials ForeWarn raises macro pre-incident recall from 0.478 for the strongest of four recent baselines to 0.670 at similar verifier cost, and on three Jetson tiers accumulated lag roughly halves the timely hits that replay credits.

2 Related Work

Streaming video understanding and proactive safety. VideoLLM-online studies temporally aligned dialogue over continuous video (Chen et al., 2024a). Dispider, LiveStar, and Thinking-QwenVL make evidence acquisition and response timing explicit through asynchronous processing, silence decisions, or evidence-sufficiency modeling (Qian et al., 2025; Yang et al., 2025; Zhang et al., 2026). StreamSense combines a lightweight encoder, selective VLM escalation, and deferral when evidence is insufficient (Wang et al., 2026a). On the safety side, PaSBench studies proactive risk awareness, and PaSBench-Video evaluates causal warnings between annotated risk onset and accident boundaries, including safe-video controls and deployment constraints (Yuan et al., 2025; Zhao et al., 2026a). SPRINT probes early physical-risk cues in sports, and EgoSafetyBench examines embodied safety guards with matched situational controls (Qiu et al., 2026; Panpatil et al., 2026). Observe Before You Alert casts driver alerting as a choice among staying silent, observing longer, and alerting (Wang et al., 2026c). Its observe action delays an uncertain alert, and the choice is still made from the current judgment. However, all of them decide from a judgment about the current state, and none lets the forecast itself decide when to warn.

Selective computation and sequential decisions. FrugalGPT and Language Model Cascades allocate inference across models, and confidence-based deferral and selective classification analyze conditional errors and coverage tradeoffs (Chen et al., 2024b; Gupta et al., 2024; Jitkrittum et al., 2023; Geifman & El-Yaniv, 2017). Sequential testing under stochastic deadlines and observation-controlled detection connect sampling cost, delay, and false alarms (Frazier & Yu, 2007; Banerjee & Veeravalli, 2012). Streaming perception evaluates predictions at the time they become available (Li et al., 2020). Model routing and cascading decide when a more expensive model is needed and how many samples to draw (Yue et al., 2024; Ong et al., 2025; Chuang et al., 2025; Dekoninck et al., 2025; Ding et al., 2025). However, in all of these analyses the decision to defer is made from the current evidence and its cost; none takes predicted future risk as an input to that decision. Conformal prediction and conformal risk control give marginal guarantees under exchangeability, and their online and non-exchangeable variants weaken that assumption at a cost in what is guaranteed (Vovk et al., 2005; Angelopoulos & Bates, 2023; Angelopoulos et al., 2024a;b; Farinhas et al., 2024). The fixed-score form of our per-deployment calibration is an exchangeable-rank argument of this kind; the joint search that the experiments use has no such guarantee and is assessed empirically on the benchmark domains of Section 4.

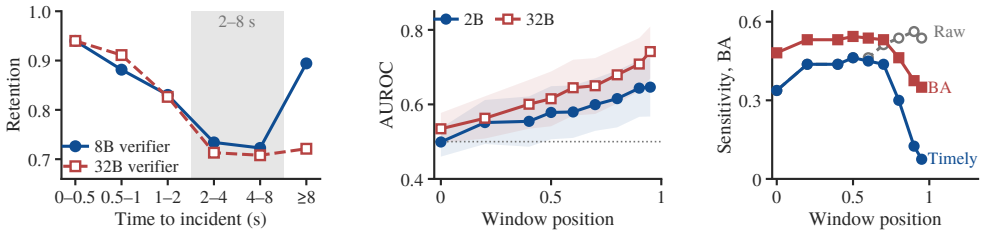
3 Risk-Adaptive Vision-Language Inference

3.1 Problem Setup

Causal observations and usable actions. Let H_t contain frames available up to video time t. A policy observes successive H_t, optionally requests verification, and emits immediate warnings or advisories. Hazardous video i has an annotated incident time a_i, unknown to the policy. Let d_i be the time its first warning becomes available after the method's own latency, or $+\infty$ if none occurs. We evaluate pre-incident coverage and the normal-video false-alert rate (FAR) together:

$$R_\tau = \frac{1}{N_+} \sum_{i:Y_i=1} \mathbf{1}\{d_i < a_i - \tau\}, \qquad F = \frac{1}{N_-} \sum_{i:Y_i=0} \mathbf{1}\{\text{at least one warning in } i\}. \tag{1}$$

The primary result uses $\tau = 0$, accompanied by verification calls and inference costs. Lead time $a_i - d_i$ is reported only for timely hits. Diagnostic subsets additionally annotate first visible evidence e_i and evaluate warnings in $[e_i, a_i)$.



(a) Serial confirmation (b) Evidence quality (c) Confirmation timing

Figure 2: The timing cost of confirmation: (a) serial confirmation keeps fewer early candidates; (b) single-call AUROC by window position; (c) single-confirmation timing scan with the 2B model.

Horizon risks as the control interface. The scout is a small VLM fine-tuned to classify time to a harmful incident from a causal frame sequence. Its five classes are time-to-incident buckets, from no incident within the forecast horizon down to an imminent one; the reference prompt fixes their boundaries. Denoting their probabilities by $p_0(H_t), \ldots, p_4(H_t)$ from the longest horizon to the shortest, the policy uses

$$r_5(t) = \sum_{k=1}^{4} p_k(H_t), \qquad r_3(t) = \sum_{k=2}^{4} p_k(H_t). \tag{2}$$

The subscripts name the reference horizons in seconds. The shorter horizon drives immediate action; the longer horizon drives verification and advisory eligibility. The reference scout is Qwen3-VL-2B adapted with a multi-domain low-rank adaptation (LoRA) using bucket-label cross-entropy and then frozen (Qwen Team, 2025; Hu et al., 2022). Its training videos come from staged-fall, OOPS, driving, and home sources and are separated from the evaluation videos by subject, source, and file identity; the 8B verifier remains zero-shot. By construction $r_3(t) \leq r_5(t)$. Both are thresholded as scores, and neither is used as a calibrated probability.

3.2 The timing cost of confirmation

Serial confirmation discards early candidates. In a serial cascade, the verifier confirms each window the scout flags before any warning is issued. Figure 2a shows what this costs. In the final half-second before the incident the 8B verifier confirms 94.0% of the flagged windows, but 2–8 seconds before it confirms only 73.1%; the 32B verifier behaves the same, at 94.0% and 71.2%. The cascade thus keeps the candidates that arrive too late to be useful and drops a quarter of the early ones. Early forecasts therefore need a route to a warning that does not pass through confirmation. Confirmation keeps its role for candidates that can wait, since not every candidate the cascade drops is a correct forecast.

Later evidence recognizes more but returns too late. Two diagnostics separate what looking later adds to recognition from the time it leaves to act (Figure 2b,c). In the first-evidence study, the scout's single-call area under the ROC curve (AUROC) rises from 0.499 at the annotated early boundary to 0.646 near the incident. In the 2B single-confirmation scan, moving the confirmation from the middle of the precursor window ($\tau = 0.5$) to its end ($\tau = 0.95$) raises raw sensitivity from 0.463 to 0.538. Once the call's own latency is charged, timely sensitivity drops to 0.075, and timely balanced accuracy (BA) falls from 0.544 to 0.350; BA is the mean of sensitivity and specificity. More evidence improves recognition, but the answer still returns too late to warn.

A 32B verifier separates hazards from normal windows better than the 2B model only near the incident, an AUROC gain of 0.095 against 0.035 at the early boundary, so scale gains accuracy where the least time remains. Under a 3 s anticipation budget the 2B model can still use 77.5% of the budgeted window once its round trip is charged, and the 32B verifier none of it, because its round trip alone already takes 5.4 s.

3.3 FOREWARN

ForeWarn keeps both routes to a warning (Figure 1). A short-horizon rule warns without waiting, a longer-horizon rule gates advisories and verification requests, and per-deployment calibration sets the advisory thresholds from normal footage.

Immediate forecast warning. When $r_3(t) \geq \theta_u$, ForeWarn emits an URGENT alarm without waiting for verification; this is the immediate warning. A verifier answer that arrives later cannot cancel it, because within a short horizon that answer may come back after the moment to act has passed; a "no" that arrives then prevents nothing and could only take back a warning that has already gone out to the user.

Advisory from risk and returned state evidence. The zero-shot 8B verifier judges whether the current causal window shows a hazardous state or activity. Its single-token probability v updates a persistent WATCH state: enter when $v \geq \theta_{\text{on}}$ and exit once n_{off} consecutive returned answers have fallen below θ_{off}.

An advisory fires at the moment the clip enters WATCH, and only if the scout's risk at that moment satisfies $r_5(t) \geq \theta_p$. The state judgment supplies confirmation, and the forecast decides whether that confirmation supports a warning now. Staying in WATCH does not repeat the advisory, and an entry whose risk check fails at that moment does not fire later when r_5 rises.

Risk-driven verification requests. The controller allocates additional inference using predicted risk and the age of its last request. After the initial causal window is complete, let $\Delta q = t - t_{\text{last query}}$ denote time since the previous request was launched. On the query grid, a request is eligible when

$$r_5(t) \geq \theta_{\text{arm}}(\Delta q), \qquad \theta_{\text{arm}}(\Delta q) = \max(\theta_{\min}, \theta_{\max} - \kappa \Delta q). \tag{3}$$

Rising risk triggers a request at once, and the falling threshold allows a refresh as the previous query ages. High risk deserves a fresh judgment now, and an old judgment loses value as the scene changes, so the bar for refreshing it falls with its age; the floor $\theta_{\min}$ keeps low-risk footage from being queried on age alone. A fixed patrol interval adds a refresh trigger, and a per-episode budget limits verification calls. An exhausted budget leaves the immediate path available. Requests run asynchronously, so the scout keeps its decision step while verification adapts, and only returned answers update WATCH. Algorithm 1 lists one decision step.

Algorithm 1 One decision step.

Require: window H_t; state w, c_{low}, $t_{\text{last query}}$, n_q
$r_3, r_5 \leftarrow$ scout scores of H_t; $w_{\text{prev}} \leftarrow w$
if a verifier answer v has returned **then**
  if $v \geq \theta_{\text{on}}$ **then** $w \leftarrow 1$; $c_{\text{low}} \leftarrow 0$
  else if $v < \theta_{\text{off}}$ **then** $c_{\text{low}} \leftarrow c_{\text{low}} + 1$
  else $c_{\text{low}} \leftarrow 0$
  if $c_{\text{low}} \geq n_{\text{off}}$ **then** $w \leftarrow 0$
end if
if Eq. (3) or patrol is due, within budget **then**
  launch verification of H_t asynchronously
  $t_{\text{last query}} \leftarrow t$; $n_q \leftarrow n_q + 1$
end if
$u \leftarrow \mathbf{1}\{r_3 \geq \theta_u\}$
$a \leftarrow \mathbf{1}\{w = 1 \wedge w_{\text{prev}} = 0 \wedge r_5 \geq \theta_p\}$
emit URGENT if $u = 1$; emit an advisory if $a = 1$

3.4 SETTING THE ADVISORY THRESHOLDS PER DEPLOYMENT

Per-deployment calibration uses routine normal footage to select the entry and prompt thresholds, leaving the scout and the other control constants fixed. The joint grid search replays each threshold pair on the calibration clips and keeps the pairs whose clip-level alert FAR stays within the budget; immediate warnings count toward that FAR. Among those pairs it picks the most sensitive one, measured by how often the advisory path fires on normal clips. Normal footage measures the alert burden but says nothing about hazard recall.

With the entry rule fixed, a quantile rule on the scout risk at entry bounds the false-advisory rate on exchangeable clips (Appendix C); the experiments use the joint search and assess it empirically.

3.5 VALUE OF A VERIFICATION CALL

Consider a one-step decision conditioned on the available history. Let p be the probability of a true hazard candidate, q_0 the false-confirmation probability on a benign candidate, and r_1 the joint

probability that a true candidate is confirmed and that the answer returns before the incident. A timely true alarm earns 1 and a false alarm costs $\lambda > 0$. Waiting earns nothing here. Comparing warning now, verifying first, and waiting gives

$$U_A = p - (1-p)\lambda, \qquad U_V = pr_1 - (1-p)\lambda q_0, \qquad U_W = 0. \tag{4}$$

Proposition 1 (Verification advantage)**.** *For $0 < r_1 < 1$, verification strictly improves on both alternatives on a nonempty interval of odds $o = p/(1-p)$ if and only if $r_1 > q_0$. The interval is*

$$\frac{\lambda q_0}{r_1} < o < \frac{\lambda(1-q_0)}{1-r_1}. \tag{5}$$

Pairwise comparison gives both bounds (Appendix C). The result ties confirmation quality to usable time. A more reliable judgment helps only while it can still return in time to shape the warning.

The domain-level check in Section 4.3 approximates r_1 by the product $q_1\rho$ of the confirmation rate on true candidates and the timely-return rate, each taken at its conservative recorded end.

4 Experiments

4.1 Experimental setup

Datasets. The benchmark covers five domains: staged falls from CAUCAFall and GMDCSA-24 through OmniFall (Eraso Guerrero et al., 2022; Alam et al., 2024; Schneider et al., 2026), EDF and its occluded counterpart OCCU (Zhang et al., 2014), OOPS accidents (Epstein et al., 2020), and two disjoint hazardous pools from SmartHome-Bench (Zhao et al., 2025; 2026a). It holds 299 hazardous and 183 held-out normal videos, and a separate 183 normal videos serve per-deployment calibration. Home combines 54 hazards with a long runway and 55 with a short one, where the runway is the time from first visible evidence to the incident; mechanism studies use the former, Home-L. The live trials (R65) come from one home and hold 35 hazards, 15 held-out normals, and 15 calibration normals.

Models. The scout is Qwen3-VL-2B with the multi-domain LoRA of Section 3.1 and stays frozen in every experiment; the verifier is zero-shot Qwen3-VL-8B, the diagnostics additionally use a 32B verifier, and the device tiers replace the 8B verifier with 4B and 2B quantized variants. ForeWarn runs at the operating point that calibration on each domain's normal footage returns for a nominal FAR budget of 0.5.

Baselines. PaSBench uses its released evaluator (Zhao et al., 2026a). SPRINT-style probing retains its anchored window and prompt, with queries on an online grid (Qiu et al., 2026). StreamSense uses small- and large-model scores with deferred reads included in its call accounting (Wang et al., 2026a). The EgoSafetyBench guard (Guard in the tables) uses ten duration-dependent blocks (Panpatil et al., 2026); because the blocks are defined from clip length, its lead times are reported but not compared. All methods share source videos and an 8B service but keep their native prompts, windows, generation lengths, and operating points, so recall, FAR, and balanced accuracy are all reported in Table 1.

Metrics. We report three quantities: delivered recall R, the normal-clip false-alert rate F, and verifier calls per normal clip C. Delivered recall is the share of hazardous videos warned before the incident after each method's own latency is charged, as measured on one server, and R_0 is the same recall with latency ignored. Appendix D lists these latency constants together with the populations, the models, and the metrics.

4.2 Main results

Overall comparison. ForeWarn's macro recall over the five domains is 0.670 (Table 1). StreamSense is the strongest baseline; it reaches 0.478 at similar verifier calls per normal clip and a lower FAR. Re-thresholding StreamSense to ForeWarn's FAR in every domain raises its macro recall to 0.541 at twice the verifier calls, a lead of 12.9 points at matched FAR. The gain comes from the two warning paths acting on different events. The immediate path keeps the short-runway falls and accidents that a verifier would confirm too late, and the advisory path adds the long-runway home

Table 1: Delivered recall / normal-clip FAR per domain and five-domain macro averages (BA $= (R + 1 - F)/2$; T: verifier seconds per clip; †: four-domain macro; ‡: re-thresholded to ForeWarn's FAR; Zero-shot: ForeWarn with the unadapted scout).

Method	OmniFall	EDF	OCCU	OOPS	Home	$R\uparrow$	$F\downarrow$	BA$\uparrow$	$C\downarrow$	$T\downarrow$
PaSBench	0.000/0.125	0.000/0.095	0.000/0.000	0.013/0.324	0.220/0.443	0.047	0.197	0.425	39.64	209
SPRINT	0.065/0.000	0.000/0.000	0.000/0.000	0.587/0.514	0.330/0.239	0.196	0.151	0.523	24.08	41
StreamSense	0.645/0.062	0.452/0.095	0.048/0.048	**0.933**/0.784	0.312/0.466	0.478	0.291	0.594	3.81	1.84
StreamSense‡	0.742/0.125	0.500/0.143	0.095/0.238	0.853/0.676	**0.514**/0.580	0.541	0.352	0.594	8.03	—
Guard	0.194/0.000	0.000/0.000	0.024/0.000	0.520/0.541	0.312/0.216	0.210	0.151	0.529	10.00	5.9
Zero-shot	0.774/0.062	0.667/0.143	**0.286**/0.381	0.827/0.541	—	0.639†	0.282†	0.678†	2.56†	—
ForeWarn	**0.935**/0.125	**0.929**/0.143	0.119/0.286	0.880/0.676	0.486/0.580	**0.670**	0.362	**0.654**	3.90	1.89

events that need confirmation. The cost is specificity, one minus FAR. On OCCU and Home, FAR exceeds recall, so ForeWarn's balanced accuracy falls below the never-warning 0.500, and SPRINT and the guard score higher on Home (Table 9). Calls count verification effort, not total computation. At native latency, ForeWarn's verifier time per clip is close to StreamSense's and 22 and 110 times below SPRINT's and PaSBench's (T in Table 1).

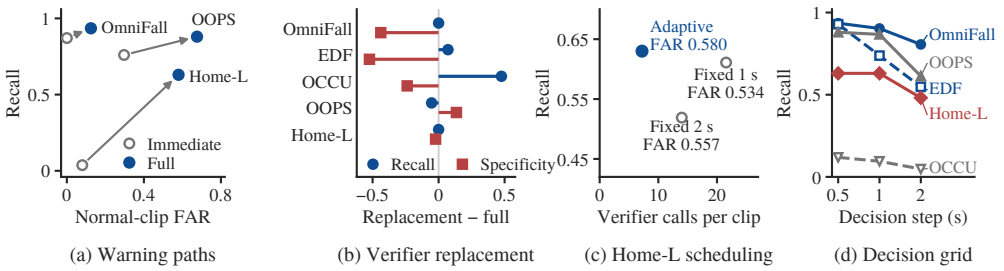
Per-domain behavior. On the staged falls and EDF, ForeWarn recalls 0.935 and 0.929 where StreamSense reaches 0.645 and 0.452, because the scout's short-horizon risk fires on the fall itself and needs no confirmation. Re-thresholding StreamSense to ForeWarn's FAR on the staged falls lifts its recall only to 0.742, at three times its verifier calls. PaSBench recalls a similar share there before inference is charged, but nothing once its 5.28 s exchange is included. On OOPS, StreamSense recalls 0.933 against ForeWarn's 0.880, but it warns on most normal clips and several times per clip, so its warnings barely separate hazardous from normal footage. Brought down to ForeWarn's FAR, it recalls 0.853. SPRINT and the guard reach zero FAR on EDF and OCCU by not warning at all, so their low false alerts there reflect abstention, not judgment about the scene.

Home. On combined Home, ForeWarn leads SPRINT by 0.156 recall at higher FAR while using about one ninth of its verifier calls. The lead comes from the advisory path, which covers the long-runway home events that the immediate path alone misses. SPRINT queries on a fixed grid, so its calls land on quiet footage as often as on the precursor, and it has no forecast to bring a call forward. ForeWarn still generates 95 false alerts per hour on held-out home normals, fewer than SPRINT and StreamSense but more than the guard, so the lead costs interruptions. It has the lowest per-decision false-alert rate of the four. Home is also the one domain where StreamSense edges ahead once re-thresholded to the same FAR. It recalls 0.514 against 0.486 at 11.06 verifier calls per normal clip against 7.51. Clip FAR counts a normal episode as disturbed once any warning fires, so neither measure substitutes for the other.

4.3 Ablation studies

Advisory path. The two paths cover different events (Figure 3a). Removing the advisory path costs little on the staged falls and OOPS, where the immediate path alone already recalls 0.871 and 0.760. On Home-L it collapses recall from 0.630 to 0.037. The runway length explains the difference. Falls and accidents push the scout's short-horizon risk over the immediate threshold within seconds, whereas home hazards develop slowly and that risk stays below it, so only a WATCH entry with concurrent risk turns them into warnings. The full policy keeps the immediate path's short-event coverage and adds a channel for long-runway events, which raises FAR in every domain.

Verifier. The advisory path matters more than which model confirms it. Replacing it with a scout threshold and recalibrating each arm leaves Home-L recall unchanged and moves specificity by 0.023. The home advisories are carried by the risk concurrence rule, and the verifier mainly decides when to fire, not whether. On the staged falls and EDF the replacement instead loses 0.438 and 0.524 specificity, because there the verifier rejects the scout's false candidates. Under occlusion it gains 0.476 recall at a 0.238 specificity cost, because the verifier withholds confirmation on views it cannot



(a) Warning paths (b) Verifier replacement (c) Home-L scheduling (d) Decision grid

Figure 3: Ablation studies with frozen models: (a) removing the advisory path; (b) replacing the verifier with a scout threshold, each arm recalibrated; (c) verification cadence on Home-L; (d) decision step of the scout.

Table 2: Scheduling replay (R / S / Q: recall / specificity / queries per replayed clip).

Domain	Adaptive	Fixed 1 s	Fixed 2 s
OmniFall	0.935/0.875/2.73	0.935/0.812/6.00	0.903/0.938/3.24
EDF	0.929/0.857/3.25	0.929/0.857/5.40	0.929/0.905/2.88
OCCU	0.119/0.714/2.67	0.119/0.714/4.67	0.071/0.762/2.40
OOPS	0.880/0.324/3.91	0.867/0.324/5.14	0.880/0.405/2.91
Home-L	0.630/0.420/7.22	0.611/0.466/21.55	0.519/0.443/14.00

Table 3: Cross-fitted false-advisory rate at target α.

Domain	0.10	0.20	0.30	0.50
OmniFall	0.094	0.125	0.188	0.188
EDF	0.095	0.119	0.119	0.119
OCCU	0.119	0.119	0.238	0.238
OOPS	0.054	0.176	0.284	0.554
Home-L	0.080	0.210	0.290	0.517

read (Figure 3b). Semantic verification pays where the scout over-fires and hurts where the verifier cannot see the scene.

One-step criterion. Section 3.5 gives a criterion for when a verifier call is worth making; applied after the fact to the logged queries, it agrees with the ablation on three domains. On the staged falls and EDF it says the verifier is worth it, and removing the verifier does hurt; on OCCU it says the verifier is not, and removing it helps. On OOPS and Home-L it also says the verifier is worth it, yet removing it changes little. Table 11 lists the rates behind this check.

Risk-guided schedule. On Home-L, risk-guided scheduling matches the recall of a fixed 1 s cadence with about one third of its verifier calls, 7.22 against 21.55 per replayed clip. It also beats a fixed 2 s cadence by 0.111 recall with half the calls at nearly unchanged FAR (Figure 3c). The difference is where the calls go. Requests triggered by rising risk land in the windows before an incident, whereas a fixed grid spends most of its budget on quiet footage. The benefit is therefore largest on clips that mix long quiet stretches with sustained risk. The schedule is not tuned to one point; on development replays, perturbing its three constants by $\pm 50\%$ keeps every variant at or above the fixed cadence. The benefit is domain-dependent (Table 2). On OOPS and EDF a fixed 2 s grid is cheaper at unchanged recall, and on OCCU it lowers recall from 0.119 to 0.071, so no cadence is uniformly best.

Per-deployment calibration. Relative to thresholds transferred from the development footage, calibration raises Home-L recall by 0.204 at a 0.125 specificity cost and OCCU recall by 0.095 at unchanged specificity, because each deployment's normal footage sets how much advisory coverage the FAR budget can afford. A separate check uses the fixed-score form with cross-fitting. Its measured false-advisory rate stays at or below the target in 16 of 20 cells (Table 3). On Home-L it settles at a lower-recall, higher-specificity operating point than the joint search.

Scout adaptation. Adapting the scout changes the full policy's operating point (zero-shot row of Table 1). Adaptation raises recall on the staged falls by 0.161 and on EDF by 0.262, because the adapted scout fires earlier and more often on the fall precursors it was trained on. The price is more false alerts on OmniFall. OCCU recall instead falls from 0.286 to 0.119; occlusion remains a failure condition for the adapted scout.

Table 4: Home-L budget sweep (Q: queries per replayed clip).

Budget	R	R_0	F	Q
30	0.630	0.648	0.580	7.22
20	0.630	0.648	0.580	7.03
10	0.593	0.611	0.568	5.80
5	0.519	0.537	0.443	3.94
2	0.315	0.315	0.307	1.94

Table 5: Recorded execution on three device tiers (R_{replay}, R_{wall}: public recall without and with accumulated lag; S: specificity; $\text{BA}_{\text{wall}} = (R_{\text{wall}} + S)/2$; R65 under direct wall timing).

Tier	Public				R65, direct wall	
	R_{replay}	R_{wall}	S	BA_{wall}	R	S
2B/8B, 64 GB	0.391	0.187	0.760	0.473	0.029	1.000
2B/4B, 32 GB	0.535	0.281	0.667	0.474	0.029	1.000
2B/2B, 16 GB	0.773	0.385	0.290	0.337	0.314	0.467

4.4 Sensitivity analysis

Decision step. The decision step changes when the scout can act, not what it predicts (Figure 3d). Coarsening it from 0.5 to 2 s costs 0.381 recall on EDF and 0.149 on Home-L, with clip FAR falling as well. This loss is mechanical, not predictive. A short precursor that falls between two decision points is never scored, so even frozen predictions lose warning opportunities whenever the decision grid is coarser than the precursor.

Query budget. The query budget bounds verification and shifts the operating point. On Home-L, cutting it from 30 to 2 requests roughly halves both recall and FAR (Table 4), because fewer opportunities to verify remove false alerts and useful advisories together. On the other four domains the same cut moves recall by at most 0.024, because those clips are short and end within a few verifier calls, so the cap rarely decides a warning.

Verifier input. More verifier frames do not always help. On Home-L, going from 8 to 16 frames lowers recall from 0.741 to 0.593 while also lowering FAR, a longer window that trades early sensitivity for fewer false confirmations.

4.5 Execution on edge devices

The runtime executes on all three tiers, but sustained processing can fall behind the video. On the device the scout and verifier share one GPU and memory, so the immediate path is independent in logic only, and overruns accumulate across a clip until decisions on both paths arrive with that lag. Table 5 reports recall with and without this accumulated lag.

Charging accumulated lag roughly halves the public timely hits on every tier, from 0.391, 0.535, and 0.773 to 0.187, 0.281, and 0.385 on the 64, 32, and 16 GB tiers. In long home clips the lag typically reaches 4.3 and 4.7 s on the two larger tiers and 18.4 s on the smallest. R65 shows the same picture under direct wall timing, where only the smallest tier delivers 0.314 of the hazards in time; it does so because it warns most often, at the lowest specificity on both populations, 0.290 on the public clips and 0.467 on R65. On that tier the verifier is a 4-bit 2B model in place of the 8B. Gains measured in replay therefore do not by themselves become timely warnings under sustained load, and sustained runs of 72–73 minutes show no thermal throttling, so the deficit is throughput, not heat.

5 Conclusion

This paper treated proactive safety warning as two coupled decisions and let predicted near-future risk make both of them: when to verify and when to warn. Our diagnostics showed that gating every forecast behind a verifier discards early candidates and that later evidence returns after the moment to act. ForeWarn addressed this with two warning paths driven by horizon risks, a risk-dependent verification schedule, and normal-footage calibration of the advisory thresholds. The one-step analysis showed that a verifier call pays only while its confirmation is reliable and still returns in time. Across five public domains ForeWarn raised macro pre-incident recall from 0.478 to 0.670 at similar verifier cost; the gains came from the short-horizon path on fast incidents and from the advisory path on slow ones. The device runs showed that accumulated processing lag roughly halved the timely hits that the benchmark credits. The next step is to close that gap under sustained load by scheduling the scout itself against the lag that builds up on the device.

AI USE STATEMENT

In this work, we used generative AI tools (OpenAI Codex, Anthropic Claude, and Google Gemini) to provide ingredients for proving the mathematical claims and to assist in writing their proofs, to implement the method, to assist with translation between the authors' working language and English, and to interpret the archived results. We have not used generative AI tools to generate synthetic data sets, to develop the theoretical model or conceptual framework, to formulate the mathematical claims, to propose or refine hypotheses, to design or give feedback on the research methodology or experiments, or to clean and reformat datasets, and qualitative or thematic data analysis is not applicable to this work. Additionally, we used generative AI tools to edit the paper for readability, to identify, search for, and summarize related literature, to format references, and to create and modify the figures and the LaTeX and visualization code. We have reviewed all AI-assisted work: AI-assisted proofs, text, and numbers were checked by the authors against the available source code, the experimental records, and the cited literature, and AI-assisted code was executed and its output inspected. No experimental observation reported in this paper was produced by a generative AI tool, and no new model training was run during manuscript preparation. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This study concerns physical-risk warning and includes public human-activity footage and a small collection of controlled recordings. Video monitoring can expose private behavior, and missed hazards and false alarms fall unevenly on different people. The evaluation therefore reports false alerts alongside warning coverage and separates recorded performance from safety guarantees. The method does not establish that a warning prevents injury or that any deployment is suitable for unattended safety-critical use. Dataset access and redistribution must respect the conditions of the original providers; footage and identifying information should not be released merely to demonstrate an alert. The controlled recordings comprise a limited set of scripted volunteer enactments and are reported as a scene-specific implementation evaluation. The available archive contains the recordings and annotations but does not include consent or institutional-review documentation.

REPRODUCIBILITY STATEMENT

Section 3 and Appendix A specify the controller, thresholds, prompts, and calibration profiles. Appendix C proves the stated propositions. Section 4 and Appendix D document the evaluation populations, timing conventions, model configurations, and scope of the archived measurements. Appendix F.4 distinguishes what the archive holds for cached predictions, the fine-tuned checkpoint, and long-stream video; replaying cached predictions and regenerating them from a model are two different levels of reproducibility.

REFERENCES

Ekram Alam, Abu Sufian, Paramartha Dutta, Marco Leo, and Ibrahim A. Hameed. GMDCSA-24: A dataset for human fall detection in videos. *Data in Brief*, 57:110892, 2024.

Anastasios Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal Risk Control. In *International Conference on Learning Representations (ICLR)*, pp. 55198–55218, 2024a.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023. doi: 10.1561/2200000101.

Anastasios Nikolas Angelopoulos, Rina Barber, and Stephen Bates. Online conformal prediction with decaying step sizes. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 1616–1630. PMLR, 2024b.

Taposh Banerjee and Venugopal V. Veeravalli. Data-efficient quickest change detection with on-off observation control. *Sequential Analysis*, 31(1):40–77, 2012.

Joya Chen, Zhaoyang Lv, Shiwei Wu, Kevin Qinghong Lin, Chenan Song, Difei Gao, Jia-Wei Liu, Ziteng Gao, Dongxing Mao, and Mike Zheng Shou. VideoLLM-online: Online video large language model for streaming video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18407–18418, 2024a.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024b.

Yilong Chen, Xiang Bai, Zhibin Wang, Chengyu Bai, Yuhan Dai, and Ming Lu. StreamKV: Streaming Video Question-Answering with Segment-based KV Cache Retrieval and Compression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 3120–3128, 2026. doi: 10.1609/aaai.v40i4.37305.

Yu-Neng Chuang, Prathusha Kameswara Sarma, Parikshit Gopalan, John Boccio, Sara Bolouki, Xia Hu, and Helen Zhou. Learning to Route LLMs with Confidence Tokens. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 10859–10878. PMLR, 2025.

Jasper Dekoninck, Maximilian Baader, and Martin Vechev. A Unified Approach to Routing and Cascading for LLMs. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 12987–13010. PMLR, 2025.

Shangzhe Di, Zhelun Yu, Guanghao Zhang, Haoyuan Li, Tao Zhong, Hao Cheng, Bolin Li, Wanggui He, Fangxun Shu, and Hao Jiang. Streaming Video Question-Answering with In-context Video KV-Cache Retrieval. In *International Conference on Learning Representations (ICLR)*, pp. 42115–42127, 2025.

Dujian Ding, Ankur Mallick, Shaokun Zhang, Chi Wang, Daniel Madrigal, Mirian Del Carmen Hipolito Garcia, Menglin Xia, Laks V. S. Lakshmanan, Qingyun Wu, and Victor Rühle. BEST-Route: Adaptive LLM Routing with Test-Time Optimal Compute. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 13870–13884. PMLR, 2025.

Dave Epstein, Boyuan Chen, and Carl Vondrick. Oops! predicting unintentional action in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 916–926, 2020.

José Camilo Eraso Guerrero, Elena Muñoz España, Doyra Mariela Muñoz-Añasco, and Jesús Emilio Pinto Lopera. Dataset for human fall recognition in an uncontrolled environment. *Data in Brief*, 45:108610, 2022.

Cristóbal Eyzaguirre, Eric Tang, Shyamal Buch, Adrien Gaidon, Jiajun Wu, and Juan C. Niebles. Streaming Detection of Queried Event Start. In *Advances in Neural Information Processing Systems*, volume 37, pp. 100698–100733, 2024. doi: 10.52202/079017-3194.

António Farinhas, Chrysoula Zerva, Dennis Ulmer, and Andre Martins. Non-Exchangeable Conformal Risk Control. In *International Conference on Learning Representations (ICLR)*, pp. 50952–50966, 2024.

Peter I. Frazier and Angela J. Yu. Sequential hypothesis testing under stochastic deadlines. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 20, 2007.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, and Xing Sun. Video-MME: The First-Ever Comprehensive Evaluation Benchmark of Multi-modal LLMs in Video Analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24108–24118, 2025.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Neha Gupta, Harikrishna Narasimhan, Wittawat Jitkrittum, Ankit Singh Rawat, Aditya Krishna Menon, and Sanjiv Kumar. Language model cascades: Token-level uncertainty and beyond. In *International Conference on Learning Representations (ICLR)*, 2024.

Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. MA-LMM: Memory-Augmented Large Multimodal Model for Long-Term Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13504–13514, 2024.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Zhenpeng Huang, Xinhao Li, Jiaqi Li, Jing Wang, Xiangyu Zeng, Cheng Liang, Tao Wu, Xi Chen, Liang Li, and Limin Wang. Online Video Understanding: OVBench and VideoChat-Online. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3328–3338, 2025.

Md Mohaiminul Islam, Tushar Nagarajan, Huiyu Wang, Gedas Bertasius, and Lorenzo Torresani. BIMBA: Selective-Scan Compression for Long-Range Video Question Answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29096–29107, 2025.

Wittawat Jitkrittum, Neha Gupta, Aditya K. Menon, Harikrishna Narasimhan, Ankit Rawat, and Sanjiv Kumar. When does confidence-based cascade deferral suffice? In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Mengtian Li, Yu-Xiong Wang, and Deva Ramanan. Towards streaming perception. In *European Conference on Computer Vision (ECCV)*, 2020. arXiv:2005.10420.

Wei Li, Bing Hu, Rui Shao, Leyang Shen, and Liqiang Nie. LION-FS: Fast & Slow Video-Language Thinker as Online Video Assistant. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3240–3251, 2025.

Shuming Liu, Chen Zhao, Tianqi Xu, and Bernard Ghanem. BOLT: Boost Large Vision-Language Model Without Training for Long-form Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3318–3327, 2025.

Xingcheng Liu, Bin Rao, Yanchen Guan, Chengyue Wang, Haicheng Liao, Jiaxun Zhang, Chengyu Lin, Meixin Zhu, and Zhenning Li. Predict and Resist: Long-Term Accident Anticipation Under Sensor Noise. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 782–790, 2026. doi: 10.1609/aaai.v40i1.37045.

Haocheng Lu, Nan Zhang, Wei Tao, Xiaoyang Qu, Guokuan Li, Jiguang Wan, and Jianzong Wang. Vista: Scene-Aware Optimization for Streaming Video Question Answering Under Post-Hoc Queries. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 7539–7547, 2026. doi: 10.1609/aaai.v40i9.37694.

Martin Q. Ma, Yuxiao Qu, Aditya Agrawal, Willis Guo, Paul Pu Liang, Ruslan Salakhutdinov, and Louis-Philippe Morency. Act2See: Emergent Active Visual Perception for Video Reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5455–5464, 2026.

Junbo Niu, Yifei Li, Ziyang Miao, Chunjiang Ge, Yuanhang Zhou, Qihao He, Xiaoyi Dong, Haodong Duan, Shuangrui Ding, Rui Qian, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, and Jiaqi Wang. OVO-Bench: How Far is Your Video-LLMs from Real-World Online Video Understanding? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18902–18913, 2025.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to Route LLMs from Preference Data. In *International Conference on Learning Representations (ICLR)*, pp. 34433–34448, 2025.

Siddhant Panpatil, Arth Singh, Mijin Koo, Chaeyun Kim, Haon Park, and Dasol Choi. EgoSafety-Bench: A diagnostic egocentric video benchmark for evaluating embodied VLMs as runtime safety guards. *arXiv preprint arXiv:2607.00218*, 2026.

Rui Qian, Shuangrui Ding, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Dahua Lin, and Jiaqi Wang. Dispider: Enabling video LLMs with active real-time interaction via disentangled perception, decision, and reaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24045–24055, 2025.

Jiawei Qiu, Yichen Xu, Jianzhe Ma, Mingyang Yu, Wenbin Zhu, Yang Han, Pinzheng Lv, and Wenxuan Wang. From sports to safety: Benchmarking proactive risk inference in MLLMs. *arXiv preprint arXiv:2608.05560*, 2026.

Qwen Team. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. TimeChat: A Time-sensitive Multimodal Large Language Model for Long Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14313–14323, 2024.

David Schneider, Zdravko Marinov, Moritz Mistol, Zeyun Zhong, Alexander Jaus, Rodi Düger, Rafael Baur, M. Saquib Sarfraz, and Rainer Stiefelhagen. OmniFall: From staged through synthetic to wild, a unified multi-domain dataset for robust fall detection. In *European Conference on Computer Vision (ECCV)*, pp. 112–129, 2026. doi: 10.1007/978-3-032-37023-5_7.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM Test-Time Compute Optimally Can be More Effective than Scaling Parameters for Reasoning. In *International Conference on Learning Representations (ICLR)*, pp. 10131–10165, 2025.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, Yan Lu, Jenq-Neng Hwang, and Gaoang Wang. MovieChat: From Dense Token to Sparse Memory for Long Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18221–18232, 2024.

Wei Song, Zhenya Huang, Cheng Cheng, Weibo Gao, Bihan Xu, GuanHao Zhao, Fei Wang, and Runze Wu. IRT-Router: Effective and Interpretable Multi-LLM Routing via Item Response Theory. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15629–15644, 2025. doi: 10.18653/v1/2025.acl-long.761.

Dennis Ulmer, Martin Gubri, Hwaran Lee, Sangdoo Yun, and Seong Oh. Calibrating Large Language Models Using Their Generations Only. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15440–15459, 2024. doi: 10.18653/v1/2024.acl-long.824.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, 2005. doi: 10.1007/b106715.

Haibo Wang, Bo Feng, Zhengfeng Lai, Mingze Xu, Shiyu Li, Weifeng Ge, Afshin Dehghan, Meng Cao, and Ping Huang. StreamBridge: Turning your offline video large language model into a proactive streaming assistant. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 132332–132359, 2025a.

Han Wang, Deyi Ji, Lanyun Zhu, Jiebo Luo, and Roy Ka-Wei Lee. StreamSense: Streaming social task detection with selective vision–language model routing. In *Proceedings of the ACM Web Conference 2026 (WWW)*, pp. 8897–8906, 2026a.

Qingni Wang, Tiantian Geng, Zhiyuan Wang, Teng Wang, Bo Fu, and Feng Zheng. Sample then Identify: A General Framework for Risk Control and Assessment in Multimodal Large Language Models. In *International Conference on Learning Representations (ICLR)*, pp. 64280–64297, 2025b.

Qingni Wang, Yue Fan, and Xin Wang. SAFER: Risk-Constrained Sample-then-Filter in Large Language Models. In *International Conference on Learning Representations (ICLR)*, pp. 53723–53747, 2026b.

Yuhang Wang, Lingyao Li, and Hao Zhou. Observe before you alert: Adaptive driver alerting with vision-language models. *arXiv preprint arXiv:2609.08130*, 2026c.

Zhiyuan Wang, Qingni Wang, Yue Zhang, Tianlong Chen, Xiaofeng Zhu, Xiaoshuang Shi, and Kaidi Xu. SConU: Selective Conformal Uncertainty in Large Language Models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 19052–19075, 2025c. doi: 10.18653/v1/2025.acl-long.934.

Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. VideoTree: Adaptive Tree-based Video Representation for LLM Reasoning on Long Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3272–3283, 2025d.

Yiweng Xie, Bo He, Junke Wang, Xiangyu Zheng, Ziyi Ye, and Zuxuan Wu. FluxMem: Adaptive Hierarchical Memory for Streaming Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31272–31282, 2026.

Zhenyu Yang, Kairui Zhang, Yuhang Hu, Bing Wang, Shengsheng Qian, Bin Wen, Fan Yang, Tingting Gao, Weiming Dong, and Changsheng Xu. LiveStar: Live streaming assistant for real-world online video understanding. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 31266–31304, 2025.

Youliang Yuan, Wenxiang Jiao, Yuejin Xie, Chihao Shen, Menghan Tian, Wenxuan Wang, Jen-Tse Huang, and Pinjia He. Towards Evaluating Proactive Risk Awareness of Multimodal Language Models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-0240.

Murong Yue, Jie Zhao, Min Zhang, Liang Du, and Ziyu Yao. Large Language Model Cascades with Mixture of Thought Representations for Cost-Efficient Reasoning. In *International Conference on Learning Representations (ICLR)*, pp. 21691–21728, 2024.

Xiangyu Zeng, Kefan Qiu, Qingyu Zhang, Xinhao Li, Jing Wang, Jiaxin Li, Ziang Yan, Kun Tian, Meng Tian, Xinhai Zhao, Yi Wang, and Limin Wang. StreamForest: Efficient Online Video Understanding with Persistent Event Memory. In *Advances in Neural Information Processing Systems*, volume 38, pp. 75804–75835, 2025. doi: 10.52202/085713-2547.

Haoji Zhang, Yiqin Wang, Yansong Tang, Yong Liu, Jiashi Feng, and Xiaojie Jin. Flash-VStream: Efficient Real-Time Understanding for Long Video Streams. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21059–21069, 2025.

Kecheng Zhang, Zongxin Yang, Mingfei Han, Haihong Hao, Yunzhi Zhuge, Changlin Li, Junhan Zhao, Zhihui Li, and Xiaojun Chang. Progressive online video understanding with evidence-aligned timing and transparent decisions. In *International Conference on Learning Representations (ICLR)*, 2026.

Zhong Zhang, Christopher Conly, and Vassilis Athitsos. Evaluating depth-based computer vision methods for fall detection under occlusions. In *Advances in Visual Computing (ISVC)*, pp. 196–207, 2014.

Xinyi Zhao, Congjing Zhang, Pei Guo, Wei Li, Lin Chen, Chaoyue Zhao, and Shuai Huang. SmartHome-Bench: A comprehensive benchmark for video anomaly detection in smart homes using multi-modal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 3975–3985, 2025.

Yusong Zhao, Yuejin Xie, Youliang Yuan, Junjie Hu, Jitian Guo, Yujiu Yang, and Pinjia He. PaSBench-Video: A streaming video benchmark for proactive safety warning. *arXiv preprint arXiv:2606.02443*, 2026a.

Zicheng Zhao, Kangyu Wang, Shijie Li, Rui Qian, Weiyao Lin, and Huabin Liu. CogStream: Context-guided Streaming Video Question Answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 13332–13341, 2026b. doi: 10.1609/aaai.v40i16.38336.

APPENDIX CONTENTS

A Controller and Prompts **15**
A.1 State and asynchronous execution . . . 15
A.2 Reference constants . . . 16
A.3 Prompts . . . 16
A.4 Joint calibration grid . . . 17

B Analysis Details **17**

C Proofs **18**
C.1 Verification advantage . . . 18
C.2 Fixed-score advisory control . . . 18

D Experimental Settings and Metrics **19**
D.1 Data populations . . . 19
D.2 Models and inference . . . 19
D.3 Metrics . . . 19

E Additional Results **20**
E.1 Full benchmark . . . 20
E.2 Verification criterion . . . 21
E.3 Scheduling and decision step . . . 22
E.4 Per-deployment calibration . . . 22
E.5 Scout adaptation . . . 23
E.6 Verifier candidates . . . 23
E.7 Device measurements . . . 23
E.8 Statistical summary . . . 24

F Discussion **24**
F.1 Guidelines for a new deployment . . . 24
F.2 Scope of the analysis . . . 24
F.3 Boundaries of the evaluation . . . 24
F.4 Experiment history and reproducibility scope . . . 25

G Extended Related Work **25**

A CONTROLLER AND PROMPTS

A.1 STATE AND ASYNCHRONOUS EXECUTION

The controller stores the previous query-launch time, the number of launched requests, a WATCH state, and a consecutive-low-answer counter; an intermediate verifier answer resets that counter.

These are decision states, not a language-based long-term memory. The scout reads recent frames, and the verifier is queried for the currently visible hazardous state, not for a chain of future events or an explanation that must finish before an alarm.

Algorithm 1 specifies the state transitions of the asynchronous runner. Video time determines the causal window and request schedule; wall time determines when computation finishes. A result is consumed once, after its request completes.

The offline score replay and the asynchronous device runner implement this interface differently: a replay reads the most recent cached verifier output at an eligible query time, whereas the device runner checks completion of the actual request.

A.2 Reference constants

Table 6: Reference policy constants (entry and prompt thresholds are adapted per deployment).

Quantity	Value / rule
Scout window	12 causal frames, up to 1.5 s
Nominal decision step	0.5 s
Immediate risk gate θ_u	0.90 on r_3
First query eligibility	$t \geq 1.5$ s
Query grid and tolerance	1.0 s; distance from grid below 0.26 s
Query threshold $(\theta_{\max}, \theta_{\min}, \kappa)$	$(0.60, 0.35, 0.04)$
Patrol interval	15 s from the previous query launch
Query budget	30 per episode
Default entry / exit thresholds	0.50 / 0.35
Exit rule	$n_{\text{off}} = 2$ consecutive returned scores strictly below exit threshold
Default scout advisory gate	$r_5 \geq 0.60$
Advisory emission	Only on a WATCH entry edge with risk concurrence

The schedule checks elapsed video time since the previous launch, not the age of the last answer, and after the first request the budget is checked before risk, grid, or patrol eligibility. Finite videos define episodes in the evaluated replays, whereas the original monitoring design maps an episode to a person-present interval; a different episode segmentation changes both the budget and clip-level FAR and is a protocol change.

A.3 Prompts

The scout classifies five forecast buckets; the verifier classifies the present state. Figure 4 reproduces the reference prompts. The scout placeholders give frame count, history length, and domain; frames are ordered from oldest to newest, and the verifier prompt never asks for the incident time.

SCOUT PROMPT

Inspect `{n}` causal video frames from oldest to newest, covering at most `{h}` seconds. Domain: `{domain}`. Judge only evidence visible by the newest frame. Classify the time until a harmful accident occurs: Class 0: no accident within 5 seconds. Class 1: accident in 3 to 5 seconds. Class 2: accident in 2 to 3 seconds. Class 3: accident in 1 to 2 seconds. Class 4: accident within 1 second. Controlled sitting, bending, lying, or ordinary motion belong to class 0. *Return exactly one digit: 0, 1, 2, 3, or 4.*

VERIFIER PROMPT

These chronological frames show the most recent short causal window from a monitoring stream. Judge only what is visible now. R means a person or child is currently in a visibly hazardous state or activity that could plausibly lead to physical harm soon: climbing or standing on furniture, railings, ladders or other elevated or unstable surfaces, balancing stunts, rough play near hard edges, precarious or slippery positions, or dangerous proximity to moving vehicles, deep water, fire or heavy unstable objects. S means no such hazardous state is visible: ordinary walking, sitting, standing, controlled activity, or nobody present. *Return exactly one letter: R or S.*

Figure 4: Scout and verifier prompts (braces: runtime placeholders).

Table 7: Cascade bins behind Figure 2a: N windows, n scout-positive, k also confirmed; retention $= k/n$; rows use $l \le a - t < u$.

Time to incident	N	n	k_{8B}	Retention	k_{32B}	Retention
$[0, 0.5)$ s	662	549	516	0.940	516	0.940
$[0.5, 1)$ s	537	405	357	0.881	369	0.911
$[1, 2)$ s	707	501	416	0.830	414	0.826
$[2, 4)$ s	961	624	458	0.734	445	0.713
$[4, 8)$ s	378	202	146	0.723	143	0.708
$[8, \infty)$ s	211	104	93	0.894	75	0.721
$[2, 8)$ s, pooled	1339	826	604	0.731	588	0.712

A.4 Joint calibration grid

The prompt grid is $\{0.20, 0.30, \ldots, 0.90, 1.01\}$ and the entry grid is $\{0.30, 0.40, 0.50, 0.60, 0.70\}$; the value 1.01 disables advisories, and no pair is feasible if the immediate path alone exceeds the budget. Normal clips are sorted by their manifest keys and split alternately for the default profile. Other archived profiles use their recorded split and must not be pooled with this one.

For each pair, replay produces both immediate and advisory outputs. The implementation computes total normal-clip FAR, retains pairs with calibration FAR at most α (nominally 0.5), and orders feasible candidates by the tuple (number of normal clips with an advisory, F, $-\theta_p$, $-\theta_{\text{on}}$). The search selects the largest tuple, which is the most permissive pair within the budget. This is a finite empirical operating-point search. Proposition 2 concerns a separate fixed-score procedure with a strict inequality; it says nothing about this joint grid search, which compares with inclusive inequalities.

B Analysis Details

The diagnostic studies use distinct populations and decision rules, separate from the 299-hazard benchmark. Where first-evidence time e and incident time a are annotated, normalized position is $\tau = (t - e)/(a - e)$. These retrospectively available times define diagnostic anchors; the runtime does not receive them. Within-video pre-risk anchors compare a developing risk with its own earlier state; those earlier states are not treated as interchangeable with held-out normal videos.

Evidence growth and model scale. The first-evidence comparison uses 80 test events. Qwen3-VL-2B AUROC rises from 0.499 at $\tau = 0$ to 0.646 at $\tau = 0.95$ (Figure 2b). The 32B-BF16 advantage over 2B is 0.035 at $\tau = 0$, 0.046 at $\tau = 0.4$, and 0.095 at $\tau = 0.95$; only the last interval excludes zero (Table 14). The recorded advantage is at the late position.

Single-confirmation timing. This scan uses Qwen3-VL-2B with no separate large expert and a measured latency of 0.522 s. Its gate is fixed on development events and evaluated on held-out events from driving, daily-life, healthcare, and industry footage. At $\tau = 0.5$, raw and timely sensitivity are both 0.463 and timely BA is 0.544. At $\tau = 0.95$, raw sensitivity rises to 0.538 while timely sensitivity falls to 0.075 and BA to 0.350 (Figure 2c). In the recorded bootstrap, 0.993 of the selected maxima lie within $\tau = 0.2$–0.8, but only 0.431 lie within 0.3–0.5, so the result supports a broad intermediate region, not a universal best timestamp.

Serial cascade and candidate retention. The PaSBench-full cascade diagnostic uses an event-level split of 20% validation and 80% test events: 72 hazardous and 52 normal validation events, and 288 hazardous and 207 normal test events. The scout threshold is 0.737024. Expert thresholds maximize validation specificity while preserving at least 95% of scout event-level true positives, giving 0.000708718 for 8B and 0.003167999 for 32B. The diagnostic applies scout AND expert; its prompts, thresholds, and update rule are distinct from ForeWarn's WATCH-entry policy.

Table 7 reports conditional filtering, without subtracting inference latency or declaring every rejected window a correct forecast. Windows can share a video, so they are not treated as independent events for confidence intervals. The 8B curve rises again beyond 8 s; the result is not globally monotone. At

the diagnostic's event-level operating point, the scout detects 97.6% of hazards and rejects 6.8% of normals; specificity rises to 29.0% with 8B and 33.3% with 32B. The recorded serial pipeline latency rises from 0.58 to 2.51 s.

A separate geometry-of-time diagnostic considers 481 PaSBench-Video risk events with a 3 s anticipation budget. Its measured 32B exchange of 5.4 s exceeds that budget. Charging each model its measured exchange latency shrinks the usable share of the budgeted window from 77.5% for the 2B model to zero for the 32B verifier. Without latency, the three model sizes' nominal advance times lie within 0.7 s of one another; balanced accuracy counted only on warnings that meet the deadline falls from 0.519 to 0.381 as the model grows. The diagnostic uses a single split and is read as a constraint on the time left to act.

C PROOFS

C.1 VERIFICATION ADVANTAGE

Proof of Proposition 1. For $p < 1$, divide utilities by $1 - p > 0$. The resulting values for alarm, verify, and wait are $o - \lambda$, $or_1 - \lambda q_0$, and zero. Verification exceeds waiting exactly when $o > \lambda q_0 / r_1$, and it exceeds alarm exactly when $o < \lambda(1 - q_0)/(1 - r_1)$. Since the denominators are positive, the lower bound is below the upper bound exactly when

$$q_0(1 - r_1) < r_1(1 - q_0),$$

which simplifies to $q_0 < r_1$. This proves both directions. If $r_1 \le q_0$, no posterior can give verification a strict advantage over both alternatives. At $p = 1$, an immediately available true alarm earns 1 and verification earns $r_1 \le 1$, so it adds no strict advantage. □

In the limiting cases, $r_1 = 0$ means verification never produces a useful true warning, $r_1 = q_0$ makes the two bounds meet with no strict-advantage region, and $r_1 = 1$ with $q_0 < 1$ makes verification strictly dominate both alternatives when $o > \lambda q_0$. Under the product approximation $r_1 = q_1 \rho$, a lower ρ raises the lower odds threshold and lowers the upper one, so the verification region contracts as timely return becomes less likely. This holds confirmation behavior fixed; hardware, model size, or a change in the candidates the policy selects may alter it.

C.2 FIXED-SCORE ADVISORY CONTROL

Fix the query/state rule independently of the calibration sample. For normal clip i, let s_i be its maximum scout risk at a candidate WATCH entry, or $-\infty$ if no candidate occurs. With n calibration clips sorted as $s_{(1)} \le \cdots \le s_{(n)}$, define $k = \lceil (n+1)(1-\alpha) \rceil$ and $\theta_p = s_{(k)}$, taking $\theta_p = +\infty$ if $k > n$. Emit an advisory only for a score strictly above this threshold.

Proposition 2 (Marginal advisory control)**.** *If the n calibration clips and a future normal clip are exchangeable under the fixed scoring rule, then*

$$\Pr(\textit{a future normal clip produces an advisory}) \le \alpha. \tag{6}$$

The rank argument is the split-conformal construction (Vovk et al., 2005; Angelopoulos & Bates, 2023); ties with a strict exceedance rule are conservative. The bound is marginal, restricted to advisories, and conditional on fixing the scoring procedure; it applies to the fixed-score rule on exchangeable clips, and the joint grid search and the immediate path are evaluated empirically in the experiments of Section 4.

Proof of Proposition 2. Let the fixed clip-scoring map produce $S_1, \ldots, S_n, S_{n+1}$ for the calibration and future clips. Exchangeability makes the future score's randomized rank uniform among $n + 1$ positions. For $k \le n$, the event $S_{n+1} > S_{(k)}$ implies that its rank is above k; ties can only reduce the probability of strict exceedance. Therefore

$$\Pr(S_{n+1} > S_{(k)}) \le \frac{n + 1 - k}{n + 1} \le \alpha.$$

For $k = n + 1$, the threshold is $+\infty$ and the exceedance probability is zero. An advisory on the future clip requires an entry candidate above this threshold, equivalently its maximum entry score exceeding it. This establishes the episode-level bound. □

The probability averages over calibration and future clips. Selecting the entry rule with the same data used for its quantile breaks the fixed-map premise, and abnormal content in a supposedly normal calibration pool changes the interpretation.

D Experimental Settings and Metrics

D.1 Data populations

Table 8 distinguishes model evaluation from normal-only calibration. OmniFall combines staged fall recordings from CAUCAFall and GMDCSA-24, using subjects disjoint from training. EDF and OCCU provide separate external-capture evaluations, with OCCU retaining the occluded views. OOPS supplies internet accidents with source channels separated from the training material. The home collection combines two disjoint hazardous pools with a shared normal pool. Home-L contains 54 long-runway SmartHome-Bench clips whose incident timestamps come from the proactive-safety benchmark; 67% concern animal intrusions, with the remainder including falls, children, and intruders. The other 55 hazards form the shorter home pool. Their union has 109 hazardous clips; the normal pool is deduplicated before being split into 88 calibration and 88 evaluation clips. Because both hazardous pools share this one normal pool, the pooled home result is not an average of two independently sampled normal sets.

Table 8: Videos by role (Home-L and Home-S share the normal pool; R65 not in public averages).

Domain	Hazardous evaluation	Normal evaluation	Normal calibration
OmniFall	31	16	16
EDF	42	21	21
OCCU	42	21	21
OOPS	75	37	37
Home	109	88	88
Home-L	54	88	88
Home-S	55	88	88
Public total	299	183	183
R65	35	15	15

Public videos are decoded at 10 fps and cut at the incident plus 0.5 s; the scout reads 12 frames spanning 1.5 s at a 0.5 s decision step, and the live prototype captures at 8 fps. A live episode is a continuous person-present interval, which also defines the query-budget reset. The 65 self-recorded videos (R65) include 15 normal calibration clips and 50 scored clips, of which 35 show hazards and 15 are normal footage.

D.2 Models and inference

The scout's training inventory of approximately 382 videos includes 85 staged-fall and 34 OOPS videos with additional driving and home sources, separated from the evaluation sources.

The schedule constants were selected on 59 long-runway home development clips used in scout training and disjoint from the 54 Home-L evaluation hazards. The inference services use deterministic single-token decoding. The scout obtains horizon probabilities from the token log-probabilities; the verifier provides a current-state yes/no score. Each baseline runs at its own operating point: PaSBench's released evaluator, the prompt answers of SPRINT and the guard, and StreamSense's routing thresholds; none was re-thresholded to a common FAR for the main table. Only the entry and prompt thresholds change with calibration. Both models remain resident, verification is asynchronous, and explanatory language is outside the warning path. The latest-question queue and WATCH entry-edge semantics are part of the runtime, not additional learned components.

D.3 Metrics

Pre-incident recall R_τ and normal-clip FAR F are defined in Equation (1); R is delivered recall at $\tau = 0$ with each method's inference latency charged, and R_0 ignores that latency. Balanced accuracy

is $\mathrm{BA} = (R + 1 - F)/2$, so a policy that never warns scores 0.500. FA/h counts false alerts per hour of normal footage, and the per-decision false-alert rate divides false alerts by the number of decision opportunities on normal clips. C is the number of verifier calls per normal clip and Q the number of queries per replayed clip. Benchmark replay charges native-payload latency measured on an NVIDIA DGX Spark with 128 GB of memory: 0.342 s for the scout, 0.483 s for verification and StreamSense, 0.59 s for the guard, 1.72 s for SPRINT, and 5.28 s for PaSBench. Lead time $a_i - d_i$ is reported only over timely hits. On the device tiers, S is the specificity over the recorded normal flags, R_{replay} omits accumulated lag, R_{wall} reconstructs it, and $\mathrm{BA}_{\mathrm{wall}} = (R_{\mathrm{wall}} + S)/2$. In the diagnostics of Appendix B, $\tau = (t - e)/(a - e)$ is the normalized position between first visible evidence and the incident. Single-call AUROC separates hazardous from normal windows at one position. Timely sensitivity and timely BA count only confirmations that still return before the incident after the call's own latency is charged.

The main metric is anchored at the incident timestamp because onset annotations are available only for the diagnostic subsets; it measures pre-incident coverage. Two always-warn references alarm on every clip, one at $t = 0$ and one at $t = 3$ s; their balanced accuracies of 0.500 and 0.465 show why this measure must be reported with normal-video errors.

E Additional Results

E.1 Full benchmark

Table 9 preserves all five-domain method rows. The main-table averages weight domains equally. The counts 192 of 299 for ForeWarn and 147 of 299 for StreamSense weight hazardous clips equally, and the call comparison of 5.3 for ForeWarn against 64.5 for PaSBench weights the 183 normal clips equally. On OOPS, StreamSense issues 7.11 false alerts per normal clip against 1.00 for ForeWarn. At native latency, verifier time per clip is 1.89 s for ForeWarn, 1.84 s for StreamSense, 5.9 s for the guard, 41 s for SPRINT, and 209 s for the released PaSBench protocol; PaSBench generates a longer response, and timing it as a single-token verifier call would change the implementation being tested.

The 0.156 combined-home recall lead over SPRINT is the only benchmark comparison whose intervals do not overlap (Table 14). On Home-L, the corresponding recall is 0.630, while the zero-shot scout reaches 0.556 with the same clip FAR of 0.580 and 11.69 against 7.51 verifier calls per normal clip; Table 9 carries these Home-L values in its zero-shot Home row. The zero-shot combined-home cell was not evaluated. Successful home warnings have different lead distributions in the two hazardous pools: the combined analysis reports conditional medians of 9.83 s on Home-L and 2.15 s on Home-S. Lead times are computed over each method's own hits, so they do not compare the same events across methods.

Table 10 re-thresholds StreamSense to ForeWarn's clip FAR in each domain. Its router labels a step positive when the probability exceeds 0.5. A monotone remap of the cached scout and verifier scores moves that threshold and leaves the router's deferral rules and constants unchanged. The threshold is chosen on the test normals to reach ForeWarn's FAR, which favors the baseline. At the threshold reproducing Table 1 the replay returns every StreamSense row of Table 9. Raising its FAR on the staged falls and EDF lifts recall to 0.742 and 0.500 at two to three times the verifier calls, far below ForeWarn's 0.935 and 0.929. On OCCU no threshold reaches ForeWarn's FAR of 0.286; at its highest reachable FAR of 0.238, StreamSense recalls 0.095 against 0.119. On OOPS, lowering its FAR to 0.676 reduces recall from 0.933 to 0.853, below ForeWarn's 0.880. Home is the exception. At FAR 0.580 StreamSense recalls 0.514 against 0.486, using 11.06 verifier calls per normal clip against 7.51. Averaged over domains, the re-thresholded StreamSense reaches recall 0.541 at FAR 0.352 with 8.03 calls per normal clip, against 0.670 at 0.362 with 3.90 for ForeWarn. The 19.2-point lead of Table 1 thus becomes 12.9 points at matched FAR, at twice the baseline's verifier cost.

No conclusion changes when the scout latency constant is swept over 0.53–0.80 s or the PaSBench constant over 2.14–6.70 s on OmniFall and OOPS; the scout range lies above the 0.342 s actually charged. On held-out home normals, ForeWarn generates 95 false alerts per hour, compared with 361 for SPRINT, 186 for StreamSense, and 49 for the guard. Its per-decision false-alert rate is 0.013, compared with 0.045–0.108 among the other methods. These observations coexist with its higher clip FAR because a clip can contain many decision opportunities and multiple warnings. Neither the

Table 9: Full benchmark. R_0: latency-ignored recall; R: delivered recall; F: clip FAR; BA $= (R + 1 - F)/2$; FA/h: false alerts per hour; C: verifier calls per normal clip. Dash: not evaluated; the zero-shot Home row is Home-L only.

Domain	Method	R_0	R	F	BA	FA/h	C
OmniFall	PaSBench	0.645	0.000	0.125	0.438	246	32.56
	SPRINT	0.226	0.065	0.000	0.532	0	19.69
	StreamSense	0.677	0.645	0.062	0.791	45	3.50
	Guard	0.226	0.194	0.000	0.597	0	10.00
	Zero-shot scout	0.774	0.774	0.062	0.856	22	1.94
	ForeWarn	1.000	0.935	0.125	0.905	67	2.88
EDF	PaSBench	0.857	0.000	0.095	0.452	377	16.00
	SPRINT	0.000	0.000	0.000	0.500	0	10.00
	StreamSense	0.452	0.452	0.095	0.679	171	3.62
	Guard	0.000	0.000	0.000	0.500	0	10.00
	Zero-shot scout	0.714	0.667	0.143	0.762	103	1.52
	ForeWarn	1.000	0.929	0.143	0.893	103	2.43
OCCU	PaSBench	0.143	0.000	0.000	0.500	0	16.00
	SPRINT	0.000	0.000	0.000	0.500	0	10.00
	StreamSense	0.071	0.048	0.048	0.500	137	3.57
	Guard	0.048	0.024	0.000	0.512	0	10.00
	Zero-shot scout	0.452	0.286	0.381	0.452	309	1.57
	ForeWarn	0.143	0.119	0.286	0.417	206	2.19
OOPS	PaSBench	0.773	0.013	0.324	0.345	1348	22.54
	SPRINT	0.827	0.587	0.514	0.537	2209	13.81
	StreamSense	0.973	0.933	0.784	0.575	3654	4.57
	Guard	0.587	0.520	0.541	0.490	611	10.00
	Zero-shot scout	0.907	0.827	0.541	0.643	709	5.19
	ForeWarn	0.947	0.880	0.676	0.602	514	4.51
Home	PaSBench	0.303	0.220	0.443	0.389	1286	111.08
	SPRINT	0.413	0.330	0.239	0.546	361	66.89
	StreamSense	0.312	0.312	0.466	0.423	186	3.81
	Guard	0.312	0.312	0.216	0.548	49	10.00
	Zero-shot scout (Home-L only)	—	0.556	0.580	0.488	—	11.69
	ForeWarn	0.514	0.486	0.580	0.453	95	7.51

Table 10: StreamSense re-thresholded to ForeWarn's clip FAR in each domain (R / F / C as in Table 9; on OCCU its lowest threshold reaches only $F = 0.238$).

Domain	StreamSense, native	StreamSense, FAR matched	ForeWarn
OmniFall	0.645 / 0.062 / 3.50	0.742 / 0.125 / 10.19	0.935 / 0.125 / 2.88
EDF	0.452 / 0.095 / 3.62	0.500 / 0.143 / 6.81	0.929 / 0.143 / 2.43
OCCU	0.048 / 0.048 / 3.57	0.095 / 0.238 / 7.33	0.119 / 0.286 / 2.19
OOPS	0.933 / 0.784 / 4.57	0.853 / 0.676 / 4.76	0.880 / 0.676 / 4.51
Home	0.312 / 0.466 / 3.81	0.514 / 0.580 / 11.06	0.486 / 0.580 / 7.51
Macro	0.478 / 0.291 / 3.81	0.541 / 0.352 / 8.03	0.670 / 0.362 / 3.90

lower per-decision error nor the lower event count cancels the disturbance to the 0.580 fraction of normal clips that contain an alert.

E.2 Verification criterion

Table 11 lists the recorded confirmation rates and the replacement effects behind Figure 3b. For each domain, the hazard-side rate counts confirmations among queries made within five seconds before the incident, and the normal-side rate counts confirmations among queried normal candidates. There are 240–2,743 logged queries per side and domain. The original empirical criterion uses the product approximation and a conservative endpoint comparison $q_1^{\mathrm{lo}}\rho > q_0^{\mathrm{hi}}$ with ρ held at its domain-level

Table 11: Verification statistics and replacement effects ($\Delta R/\Delta S$: recall/specificity, replacement minus full, each arm recalibrated).

Domain	$q_1\rho$	q_0	ΔR	ΔS
OmniFall	0.314	0.035	0.000	-0.438
EDF	0.121	0.048	$+0.071$	-0.524
OCCU	0.028	0.057	$+0.476$	-0.238
OOPS	0.655	0.427	-0.053	$+0.135$
Home-L	0.235	0.155	0.000	-0.023

point estimate; the logged timeliness factors span 0.886–0.960. The main text's joint-probability analysis states the more general condition, without assuming that confirmation and timely return are independent. Existing measurements are reported under the formulation they were made with, not converted to the other. These rates are measured on the queries the policy made at its calibrated thresholds on each domain.

E.3 Scheduling and decision step

Table 2 gives the scheduling replay behind Figure 3c on all five domains, and Table 12 the decision-step replay behind Figure 3d. On development-side home replays, the adaptive schedule raised the probability of triggering an advisory while cutting verifier queries on normal clips by 45%. Perturbing its three constants by $\pm 50\%$ kept every variant at or above the fixed cadence, so the schedule is not tuned to one point. The benefit of adaptive scheduling is strongest when a clip contains substantial quiet footage and sustained risk. On Home-L, the recall difference against the fixed 1 s cadence has an interval that includes zero, whereas the 0.111 lead over a fixed 2 s cadence does not (Table 14). Table 2 also retains domains where fixed 2 s probing is cheaper. Reducing budget from 30 to 2 leaves OmniFall and OCCU recall unchanged in the budget replay, changes OOPS from 0.880 to 0.867 and EDF from 0.929 to 0.905, and changes Home-L from 0.630 to 0.315. On Home-L, a budget of 20 gives recall 0.630 and specificity 0.420, a budget of 10 gives 0.593 and 0.432, and a budget of 5 gives 0.519 and 0.557. Reducing verification cost can therefore remove false alerts and useful warnings at the same time, as the budget sweep shows.

Table 12: Decision-step replay (recall / clip FAR / queries per replayed clip), matching Figure 3d.

Domain	0.5 s	1 s	2 s
OmniFall	0.935/0.125/2.73	0.903/0.125/2.73	0.806/0.188/1.57
EDF	0.929/0.143/3.25	0.738/0.143/3.25	0.548/0.048/1.89
OCCU	0.119/0.286/2.67	0.095/0.238/2.67	0.048/0.048/1.73
OOPS	0.880/0.676/3.91	0.867/0.622/3.91	0.613/0.568/1.92
Home-L	0.630/0.580/7.22	0.630/0.580/7.22	0.481/0.500/4.78

The decision-step sweep evaluates steps of 0.5, 1, and 2 s. A coarse grid may miss a short interval even without changing either model. The separate verifier-window study uses 8, 12, and 16 frames. On Home-L it reports recall 0.741, 0.630, and 0.593 with specificity 0.386, 0.420, and 0.443; on OOPS recall is 0.893, 0.880, and 0.907 with unchanged specificity 0.324. A matched-concurrency probe on 50 windows measures p50 latency of 0.580, 1.080, and 1.004 s and p95 latency of 1.297, 1.844, and 1.942 s for the three window sizes.

E.4 Per-deployment calibration

Table 3 reports the cross-fitted false-advisory rates behind the calibration paragraph of Section 4.3. The calibration target caps false advisories only, so immediate warnings add to the total FAR outside it; the numbers in that paragraph show the range of operating points that calibration reaches, not a comparison at the same FAR. The main joint-search profile instead calibrates entry and prompt thresholds together; the archived pairs $(\theta_{\text{prompt}}, \theta_{\text{on}})$ are (0.20,0.30) on OmniFall, EDF, and OCCU, (0.60,0.60) on OOPS, and (0.40,0.40) on Home-L. Their calibration FARs are 0.312 on OmniFall,

0.095 on EDF, 0.286 on OCCU, 0.486 on OOPS, and 0.500 on Home-L. The fixed-score result covers one scoring rule; the joint search over several rules on the same footage is evaluated empirically.

On Home-L, the cross-fit at target $\alpha = 0.3$ yields mean recall 0.306 and specificity 0.682, compared with 0.630 and 0.420 for the main joint-search profile; the procedures differ in candidate rules, threshold selection, and normal evaluation pools, 176 cross-fitted against 88 held-out clips. These are two operating profiles with different balances of warning coverage and normal-episode interruptions.

An additional offline stream experiment uses a trailing window of 300 clips after a 30-clip warm-up and five arrival orders. The settled gate approaches the one-shot operating point on four of five domains. Removing hazardous clips from those simulated streams changes the settled threshold by up to 0.16, recall by at most 0.018, and false alerts by at most 14 per hour, with no consistent direction. The implemented device profile keeps the fixed gate from its calibration on normal footage.

E.5 Scout adaptation

Table 13: Scout adaptation ($R/F/C$, full controller; FARs not matched).

Domain	Zero-shot	Adapted
OmniFall	0.774/0.062/1.94	0.935/0.125/2.88
EDF	0.667/0.143/1.52	0.929/0.143/2.43
OCCU	0.286/0.381/1.57	0.119/0.286/2.19
OOPS	0.827/0.541/5.19	0.880/0.676/4.51

Table 13 reports all four domains with paired zero-shot and adapted results. Adaptation improves recall on OmniFall and EDF by 0.161 and 0.262, but reduces OCCU recall by 0.167, and it raises OmniFall false alerts per hour from 22 to 67 (Table 9). Combined Home has adapted-scout recall 0.486, FAR 0.580, and 7.51 calls, but no zero-shot evaluation. The recorded zero-shot macro average over four domains is recall 0.639, FAR 0.282, and 2.56 calls, whereas the adapted-scout macro average over five domains is 0.670, 0.362, and 3.90; the two macro averages cover different domain sets. On the same four domains, macro balanced accuracy is 0.678 zero-shot and 0.704 adapted.

E.6 Verifier candidates

A candidate study replayed ten verifiers through the same controller family on Home-L and OOPS, each with a separately calibrated gate; the default Qwen3-VL-8B remained the model used in the main experiment. The recorded caches are incomplete for one candidate and contain collapsed or exactly neutral scores for several others, so the study is read as a description of those pipelines.

E.7 Device measurements

The 64 GB AGX Orin hosts a 2B scout and an 8B verifier, both in 16-bit brain floating point (bf16). The 32 GB AGX Orin retains the bf16 scout and uses a 4B verifier quantized with activation-aware weight quantization (AWQ). The 16 GB Orin NX uses two 2B instances with 4-bit weights and 16-bit activations (W4A16). Both models share device resources. The public replay processes every 0.5 s video step, recording lag as work accumulates. Its replay clock for the immediate path is video time plus scout inference, and its replay clock for advisories is the video tick at which a returned answer is read. Adding the logged accumulated lag and scout work approximates when each decision became available. R65 instead provides an explicit decision-wall timestamp. Where that field was not recorded, wall-clock timing is not reported for that footage.

Median accumulated lag in long home clips is 4.3 s on the 64 GB tier, 4.7 s on the 32 GB tier, and 18.4 s on the 16 GB tier. Low median call latency can coexist with lag, because occasional overruns and sustained throughput deficits accumulate across a clip. Sustained 72–73 minute measurements showed no thermal throttling. Temperatures settle after 15–20 minutes, near 73 °C on the AGX tiers and 86 °C on the NX, peaking at 89.1 °C. Mean power is 41.1 and 40.0 W on the two AGX tiers using three-rail accounting, and 23.9 W on the NX input rail. The rail definitions differ across tiers.

ForeWarn's direct-wall recall is 1 of 35 hazards on each of the two larger tiers and 11 of 35 on the 16 GB tier; the normal specificities are 1 on the two larger tiers and 7 of 15 on the 16 GB tier.

Volunteer enactments exercise the physical capture and warning path. On every tier the two models share the hardware, so the immediate path's independence in logic does not carry over to timing.

E.8 Statistical summary

Uncertainty uses Wilson intervals for proportions and paired bootstrap with 10^4 resamples for differences. Table 14 collects the intervals behind the comparisons in the main text and in this appendix; the main text reports the point estimates only. Three comparisons also have recorded p values: 0.0048 for the combined-home recall lead over SPRINT, 0.7314 for the replacement specificity difference, and 0.0032 for the lead over a fixed 2 s cadence.

Table 14: Uncertainty of the recorded comparisons (95% intervals; –: not recorded).

Comparison	Quantity	Estimate	Interval
Combined Home, ForeWarn	recall	0.486	[0.394, 0.579]
Combined Home, SPRINT	recall	0.330	[0.249, 0.423]
Home-L, replacement minus full	specificity	-0.023	–
Home-L, adaptive minus fixed 1 s	recall	0.019	$[-0.056, 0.111]$
Home-L, adaptive minus fixed 2 s	recall	0.111	[0.037, 0.204]
32B minus 2B, $\tau = 0$	AUROC	0.035	$[-0.011, 0.087]$
32B minus 2B, $\tau = 0.4$	AUROC	0.046	$[-0.015, 0.106]$
32B minus 2B, $\tau = 0.95$	AUROC	0.095	[0.032, 0.159]

F Discussion

F.1 Guidelines for a new deployment

A new deployment of ForeWarn needs normal footage only. Record routine footage, split it as in Appendix A, replay the joint grid at the chosen FAR budget, and keep the pair the search returns; the immediate threshold, the schedule constants, the patrol interval, and the budget stay at their reference values (Table 6). Where verification is expensive, Tables 2 and 12 show which controls cost recall. On long-runway home footage the budget and the decision step both matter, whereas on the short clips a budget of 2 changes recall by at most 0.024. In replay, two settings lower sustained load at no measured Home-L recall cost: a 1 s decision step that halves the scout's decision rate, and a budget of 10. No device run at these settings was recorded, so they are candidates, not validated defaults.

F.2 Scope of the analysis

The one-step model values a single verification decision and assigns no future value to continued observation; the schedule, budget, state hysteresis, and thresholds are evaluated empirically over whole clips, and the runtime does not compute the interval of Proposition 1. The non-veto rule protects the action logic; resource sharing is a runtime matter (Section 4.5).

Before-event warning is a timing objective, and an event with no visible precursor in the causal frames is counted as a miss for every policy.

F.3 Boundaries of the evaluation

Three boundaries from the original development are retained. Conditional on hits, ForeWarn's median lead time on the home tier is the shortest among the methods at comparable recall; OCCU is the one domain where its median lead is longest, but it has too few hits on every method to support a comparison. Intent-type events such as break-ins carry semantic cues without a physical clock. In an earlier evaluation round, even an oracle tuned on the held-out test set reached only 0.50 recall on this class. In a long-footage home subset audited earlier, roughly three quarters of falls showed no visible precursor in the camera's causal past, capping every policy at an oracle recall of 0.25; human raters on the same frames reached the same ceiling.

A separate UR-Fall collection of 70 sequences was excluded from the main comparison because its median pre-incident footage is 1.88 s and only five sequences retain at least 3 s before the incident.

F.4 Experiment history and reproducibility scope

The study accumulated several evaluation revisions. In particular, the requirement that an advisory coincide with scout risk was introduced after inspecting the initial OmniFall and OOPS results. EDF, OCCU, and the home evaluation followed that correction, so OmniFall and OOPS are not first-pass held-out tests. OmniFall and OOPS were evaluated before the concurrence rule was fixed and are reported as such. StreamSense call counts additionally include a correction for deferred reads, increasing their accounting by 1.19–1.36 times. The dataset lists and final score artifacts define the reported populations. No new training, inference, or test population was added for this paper.

The preserved research materials include the controller and replay scripts, data-role manifests, cached risk and verifier scores, and the recorded execution logs supporting the reported analyses. The manuscript source bundle contains its LaTeX files, vector figure definitions, and tabulated result values. The archive supports replaying the reported decisions from stored scores; the fine-tuned scout checkpoint, the optimizer settings, and the long-stream video are not in it, so training and score regeneration are outside what it supports.

G Extended Related Work

This section extends Section 2 with the lines of work the main text could only name; the comparisons concern task formulation and inference design, not additional experiments.

G.1 Online interaction, event starts, and proactive risk

Streaming video interaction already supports decisions about when to speak. VideoLLM-online learns temporally aligned dialogue on continuous video (Chen et al., 2024a); Dispider separates asynchronous perception, decisions, and reactions (Qian et al., 2025); and LiveStar models response and silence while accumulating context (Yang et al., 2025). StreamBridge combines a streaming memory buffer with activation decisions (Wang et al., 2025a), while Thinking-QwenVL ties response timing to sufficient visual evidence (Zhang et al., 2026). LION-FS is a close architectural antecedent: its fast path determines whether a response is needed, and its slow path augments keyframes for detailed generation (Li et al., 2025). ForeWarn applies such complementary paths to horizon-specific risk estimates, and its contribution is the resulting warning policy and its measured behavior, not the two processing speeds themselves.

Online benchmarks make several dimensions of this task explicit. OVO-Bench includes backward tracing, real-time perception, and forward active responding, including the need to wait for sufficient evidence (Niu et al., 2025). OVBench organizes online understanding around past, present, and future information and accompanies this evaluation with VideoChat-Online (Huang et al., 2025). Streaming Detection of Queried Event Start asks models to detect a language-specified event's beginning from causal frames with low latency (Eyzaguirre et al., 2024). This is close to timely alerts, although the queried event start and the pre-accident warning interval define different success conditions. Video-MME provides broad evaluation across video durations, domains, and modalities (Fu et al., 2025). Such capability measurements inform model selection, but their aggregate question-answering scores cannot determine an operating point for before-event coverage and safe-stream false alerts.

Safety-specific benchmarks add the cost of inappropriate intervention. The original PaSBench evaluates proactive risk awareness using image sequences and textual scenarios, identifying unstable risk reasoning across repeated trials (Yuan et al., 2025). PaSBench-Video then evaluates warnings on causal video prefixes within annotated intervals and checks warning content (Zhao et al., 2026a). SPRINT examines early physical-risk cues and causal inference in sports, including safe controls (Qiu et al., 2026); EgoSafetyBench evaluates embodied runtime safety judgments under situational and visual-text perturbations (Panpatil et al., 2026). Accident anticipation has an explicit early-warning tradition: Predict and Resist addresses long-term anticipation under sensor noise and evaluates both warning timeliness and false alarms (Liu et al., 2026). ForeWarn adds an inference policy; the experiments charge its computation and warning delivery under the reported timing protocols.

G.2 Visual context and memory for long streams

Long-video methods address the growing cost of retaining visual history. MA-LMM maintains a memory bank while processing video online (He et al., 2024), and MovieChat converts dense visual tokens into a sparse memory representation (Song et al., 2024). ReKV uses sliding-window video processing, stores video key–value caches, and retrieves query-relevant caches for answering (Di et al., 2025). Flash-VStream separates frame processing from question handling and maintains compact contextual and detailed visual information (Zhang et al., 2025). StreamForest organizes persistent context as event memory together with recent observations (Zeng et al., 2025). These approaches make long histories usable without repeatedly passing every observed frame to the language model.

Recent work further adapts the retained representation to scene structure. FluxMem compresses temporal redundancy across adjacent frames and spatial redundancy within frames, with compression rates determined by scene statistics (Xie et al., 2026). StreamKV combines semantic segmentation with cache retrieval and compression for streaming video question answering (Chen et al., 2026). Vista uses scene-aware optimization for video questions that arrive after the corresponding observations (Lu et al., 2026). CogStream's CogReasoner baseline combines visual compression with retrieval guided by historical dialogue context (Zhao et al., 2026b). ForeWarn's controller retains only compact decision state, and a better context mechanism could serve its scout or verifier. The policy would still face the same question, whether one more answer is worth its computation before the event.

G.3 Selecting visual evidence and allocating inference effort

Adaptive visual processing provides another way to improve the information obtained per unit of computation. TimeChat combines timestamp-aware frame encoding with a sliding video Q-Former for temporally grounded long-video understanding (Ren et al., 2024). VideoTree constructs query-adaptive hierarchical video representations (Wang et al., 2025d), while BOLT uses training-free, query-guided sampling to select informative frames (Liu et al., 2025). BIMBA compresses long-range video representations through selective scans (Islam et al., 2025). Act2See learns to interleave reasoning with calls that retrieve observed frames or generate hypothetical visual evidence (Ma et al., 2026). These methods offer different mechanisms for choosing, compressing, or requesting visual information. ForeWarn keeps its frame and window protocol and does not incorporate these methods, since a streaming decision can only use observations already available and generated hypothetical evidence is not an observation of a later real event.

Language-model routing addresses a complementary allocation problem. Mixture-of-thought cascades use consistency among answer representations to decide when a more expensive model is needed (Yue et al., 2024). RouteLLM learns routing from preference data (Ong et al., 2025); Self-REF trains confidence tokens for routing and rejection (Chuang et al., 2025); and IRT-Router models query difficulty and model capability through item response theory (Song et al., 2025). A unified treatment of routing and cascading studies their joint quality–cost objective (Dekoninck et al., 2025). BEST-Route additionally selects the number of samples as well as the model (Ding et al., 2025). Work on test-time scaling shows that the benefit of extra reasoning computation varies with prompt difficulty (Snell et al., 2025). In risk warning, the input changes as frames arrive, and a better prediction loses its value if delivered too late. Our one-step analysis captures this dependence on timely return without importing routing optimality results as guarantees for the full controller.

G.4 Confidence estimation and the scope of calibration guarantees

Practical confidence estimation and statistical calibration address related but distinct questions. APRICOT estimates confidence from generated text and the input through an auxiliary predictor, making calibration possible without internal model logits (Ulmer et al., 2024). SConU studies selective conformal uncertainty for language models, including deviations between calibration and test examples (Wang et al., 2025c). TRON constructs and filters sampled response sets under two risk levels, with experiments on video question answering (Wang et al., 2025b). SAFER separately calibrates a sampling budget and a filtering threshold, permitting abstention when the sampling risk cannot be met within a cap (Wang et al., 2026b). These methods illustrate why confidence signals, sampling failure, and answer-filtering errors should not be treated as interchangeable quantities.

Conformal risk control extends the calibration framework to bounded monotone losses (Angelopoulos et al., 2024a). Non-exchangeable conformal risk control uses weighted calibration to address differences between calibration and test observations (Farinhas et al., 2024). Online conformal prediction with decaying step sizes distinguishes retrospective coverage on arbitrary sequences from quantile estimation when the distribution stabilizes (Angelopoulos et al., 2024b). Together with classical conformal prediction (Vovk et al., 2005; Angelopoulos & Bates, 2023), these results clarify the assumptions and units to which a guarantee applies.