IS A SCREENSHOT WORTH A THOUSAND TOKENS? LEARNING EXPERT WORKFLOWS FROM SOC INVESTIGATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Expert computer workflows contain organizational knowledge that is difficult to preserve and reuse. We investigate whether reconstructing these workflows requires visual inputs or whether compact textual representations can support comparable understanding at lower inference cost. We introduce SOCBench, a benchmark of 86 recorded security operations center (SOC) investigations with paired visual and DOM-based textual representations. Our evaluation framework separately assesses procedural fidelity and summary quality, with reliability examined through cross-judge comparisons and domain-expert assessment. We compare two model families under visual and textual inputs, teacher-supervised fine-tuning, and alternative training-task mixtures. Fine-tuning improves both model families, *Qwen-3.5* and *Gemma-4*, on held-out investigations. For the selected configurations, DOM provides similar observed summary quality while Frames achieves higher procedural fidelity. DOM also reduces mean report-generation latency by 59% on the validation cohort and consumes fewer input tokens. Compact text can therefore support efficient workflow documentation, while visual input retains value for preserving procedural detail.

1 INTRODUCTION

Expert knowledge is embedded in how people use computers: the questions they pursue, the evidence they inspect, and the decisions connecting successive actions. Such knowledge often spans applications and depends on the context in which an action was taken. Recording these trajectories can transform implicit expertise into reusable organizational knowledge, supporting documentation and knowledge transfer, as well as future agents that retrieve and learn from prior workflows. This opportunity extends across domains such as finance and cybersecurity. Realizing this opportunity requires representations that preserve task semantics while remaining practical to process at scale.

For models learning from computer interactions, is a picture worth a thousand words (or, in our case, tokens)? Screenshots retain visual cues, spatial relationships, and interface context, but repeatedly processing frames can incur substantial computational and token costs, even when most content remains unchanged. Textual representations offer another design choice: describe the current interface state, retain reusable context, and encode subsequent changes. Their value, however, depends on whether this abstraction preserves the information required to recover the workflow. Aggressive compression can discard important observations, while excessive detail can undermine the intended efficiency gains. This tradeoff also depends on interaction dynamics. A localized update, navigation to a different application, and a rapidly changing display may require different strategies for refreshing the represented state. We therefore consider when incremental descriptions are sufficient and when the full interface state needs to be refreshed. Prior benchmarks examine browser navigation, representation choice, and workflow documentation Lu et al. (2024); Wornow et al. (2024). However, these results do not establish when compact text can serve as an effective alternative to visual input for reconstructing complete workflows, or how the choice of representation affects both output quality and resource use.

We study this question through security operations center (SOC) investigations, where analysts navigate complex interfaces, assemble evidence across tools, and document conclusions. We introduce

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.

a benchmark of recorded investigations with paired screenshots and structured textual descriptions derived from browser states. We compare learning from visual and textual representations across model sizes, fine-tuning configurations, and training task cocktails. Ablations examine the contribution of individual training signals and whether combining complementary auxiliary tasks improves generalization. In particular, we investigate whether domain-adapted small language models can recover investigative knowledge and which auxiliary training tasks contribute to reconstructing an entire investigation from its constituent interactions.

Evaluating these outputs requires more than matching reference wording. A fluent description can omit a decisive observation, introduce an unsupported action, or misrepresent the order of investigative steps. Our evaluation framework therefore separates procedural fidelity from summary quality, grounding assessments in the recorded actions, evidence, and sequence. We quantify agreement between LLM judges and domain experts and examine robustness across evaluation settings to assess the reliability of automated scoring explicitly.

Our contributions are as follows: (1) a benchmark of recorded SOC investigations with paired visual and structured textual representations; (2) an evaluation framework for investigation summaries, with expert validation of LLM judgments and statistical robustness tests; and (3) a comparison of vision-to-text and text-to-text learning, using SOC investigations as a case study to assess tradeoffs across output quality, token usage, latency, compute, and end-to-end cost.

2 Learning from User-Computer-Interaction Trajectories

Recorded computer interactions can provide supervision for learning how tasks are performed and turning expert behavior into reusable knowledge. This requires decisions about data collection, representation, training, and evaluation. The following subsections introduce the background, define the learning problem, and describe a general methodology for this process.

2.1 Background and Related Work

Recorded computer interactions provide demonstrations of how users perform tasks: which interfaces they visit, what information they inspect, and how their actions unfold over time. Such trajectories preserve intermediate observations and decisions that may be absent from the final task outcome. Learning from these trajectories requires interpreting interface content, integrating observations across time, and selecting information relevant to the task. Visual and textual representations provide different forms of access to this information. WebLINX (Lu et al., 2024) compares fine-tuned text-only, screenshot-based, and multimodal models for browser action prediction, finding that text-only models can outperform larger multimodal baselines. WebGUM (Furuta et al., 2024) compares HTML-only and HTML-plus-screenshot training, demonstrating benefits from visual context and temporal information. Wonderbread (Wornow et al., 2024) extends the study of recorded workflows to documentation, knowledge transfer, and process improvement, including validation of LLM judgments against human assessments. In cybersecurity, Singh et al. (2026) examine SOC analysts' use of LLMs for sensemaking and technical communication, while Cybench (Zhang et al., 2025) evaluates agents on professional capture-the-flag tasks.

Evaluating generated workflow descriptions introduces a complementary challenge: fluent text may omit important actions or misrepresent supporting evidence. G-Eval (Liu et al., 2023) uses structured criteria to assess generated text, while FActScore (Min et al., 2023) evaluates source support for individual factual claims. In an expert domain, SECQUE (BenYoash et al., 2025) aggregates repeated LLM assessments and calibrates scoring thresholds against financial experts' judgments. However, these studies leave open whether compact textual representations can preserve both summary quality and procedural detail when reconstructing complex expert workflows at lower inference cost. Addressing this question requires examining representation and training choices alongside an evaluation framework that distinguishes procedural fidelity from summary quality and measures agreement with domain experts. This motivates the methodology introduced next and its subsequent application to SOC investigations.

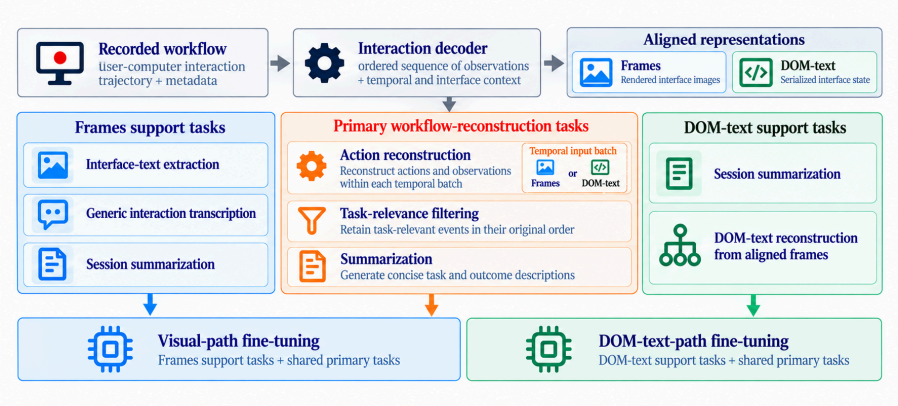


Figure 1: General training pipeline for learning from recorded user-computer-interaction trajectories. A recorded workflow is decoded into aligned visual and DOM-text representations. Representation-specific support tasks are combined with shared workflow-reconstruction tasks to fine-tune the visual and DOM-text paths.

2.2 METHODOLOGY

Our methodology follows the workflow from recording and representation through training and inference.

Data collection. Users are recorded while performing tasks, capturing the progression of their interactions with the interface. In browser-based applications, recordings can include rendered frames, accessibility states, temporal and interface metadata, and snapshots of the Document Object Model (DOM). The DOM is a hierarchical representation of a web page's structure and content, including its elements, attributes, and text, providing a basis for textual interface descriptions and tracking changes across successive states. An interaction decoder transforms the captured activity into an ordered sequence of observations. Additional recordings of general computer use can supply auxiliary training examples. Source recordings are assigned to training, validation, and test partitions before expansion into task examples, keeping examples from the same recording within one partition. Temporal batching, quality filtering, and length filtering then produce the inputs used for learning.

Visual and textual representations. Interaction trajectories can be represented visually through rendered frames or textually through structured descriptions of interface content. Where DOM information is captured, textual representations can be constructed deterministically from the recorded interface structure. An initial full-state description can be followed by state changes, allowing unchanged content to be reused. The two representations can be aligned to the same recorded activity, although they may retain different information: frames preserve rendered appearance and spatial relationships, while text exposes the content and structure available through the capture mechanism. Input modality and model architecture are specified separately; a system receiving only text may still use a multimodal pretrained checkpoint.

Fine-tuning and the cocktail effect. Given either representation, a final summary provides limited supervision for the intermediate capabilities needed to understand a long workflow. These capabilities can be supervised through a cocktail of complementary training tasks. This is motivated by the cocktail effect: combining related tasks during fine-tuning can improve target-task performance compared with single-task training, as demonstrated in finance Brief et al. (2024). Task mixtures for workflow reconstruction are therefore selected and evaluated empirically.

Figure 1 summarizes the training pipeline. Recorded workflows are decoded into temporally ordered interface states, represented as rendered frames or serialized DOM, and divided into temporal batches. Both paths share three primary tasks that follow the dependencies of workflow reconstruction. *Action reconstruction* maps each batch and a cumulative summary of preceding activity to meaningful actions and observations within that batch. *Task-relevance filtering* receives the concatenated action trace and focal task context, retaining relevant events in their original order, including negative findings, unsuccessful attempts, and changes in direction when they explain the workflow. *Summarization* converts the filtered trace into concise descriptions of the task and its outcome.

These primary tasks are combined with representation-specific support tasks to form candidate training cocktails. Support tasks may include interface-text extraction, generic interaction transcription, session summarization, and DOM-text reconstruction from aligned frames. A teacher language model can generate training targets offline, while temporal ordering, representation construction, and alignment are determined by the data pipeline. Where paired frames and structured states are available, DOM-reconstruction targets are obtained directly from the recorded states.

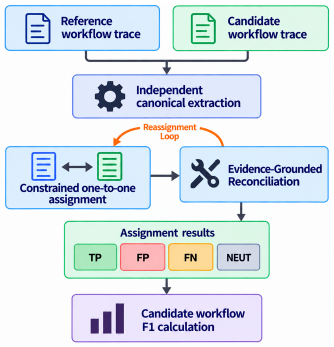


Figure 2: Procedural-fidelity evaluation using canonical procedure correspondence.

Within each path, a single model is fine-tuned on interleaved examples from the selected tasks. Causal language-modeling loss is computed over target response tokens, with instruction and input tokens masked. Task mixtures and training configurations are selected on validation recordings through ablations that assess each support task's contribution for a given representation and model.

At inference, the same model reconstructs actions sequentially across batches, then filters and summarizes the resulting trace. Only reconstruction receives representation-specific observations; subsequent stages consume text in both paths. Comparisons using different training mixtures assess their combined design choices; isolating the effect of input representation requires holding other experimental factors fixed.

Evaluation. The resulting workflow reports must explain the task and its outcome accurately while preserving the procedure followed by the user. These requirements are related but distinct: a fluent report may communicate the correct conclusion while omitting important actions, whereas a procedurally complete report may organize or explain the evidence poorly. Moreover, equivalent descriptions can differ in wording and granularity, making lexical similarity insufficient. To capture these complementary aspects, evaluation separates *procedural fidelity* from *summary quality* through two independent branches (Figures 2 and 3), whose scores are reported separately and are not combined into a single overall measure.

Procedural fidelity assesses which actions and targets are recovered from the recorded interaction. To make the reference and candidate traces comparable, they are independently mapped to ordered, evidence-grounded occurrences, each represented by a canonical action and its target: what was done and which object or entity it concerned. Canonicalization makes semantically equivalent descriptions comparable despite differences in wording. The reference trace is constructed and frozen before candidate evaluation, and every candidate occurrence must be supported by local evidence in the candidate trace.

Once the traces are canonicalized, correspondence is established through a constrained one-to-one assignment. One reference occurrence cannot provide credit to multiple candidate occurrences, while actions repeated at different points in the procedure remain distinct. A corresponding action–target pair receives full credit, a compatible but incomplete target correspondence receives graded credit, and an incompatible or missing correspondence receives no credit. For workflow w, let E_w denote earned correspondence credit, and let M_w and G_w denote the maximum candidate-side and reference-side credit, respectively. Precision, recall, and Credit F1 are defined as $P_w = \frac{E_w}{M_w}$, $R_w = \frac{E_w}{G_w}$, and $F_{1,w} = \frac{2E_w}{M_w+G_w}$, respectively.

The resulting procedural-fidelity score is macro Credit F1 across workflows, giving each workflow equal weight. Complementary measures capture aspects not reflected directly in Credit F1: compactness measures trace-length similarity, while capture-order Kendall's τ_b assesses the ordering of matched action–target occurrences. Both are reported separately and do not alter Credit F1.

Summary quality captures the complementary question of whether a report explains the workflow clearly, faithfully, and with appropriate scope. Procedural correspondence alone does not establish these properties. The second branch therefore evaluates structured report units using an anchored rubric covering content recall, factual grounding, entity roles, scope, outcome faithfulness, workflow progression, and surface integrity. Each criterion is assessed against fixed score anchors and the corresponding reference unit, with evidence retained for the assigned score. An independent

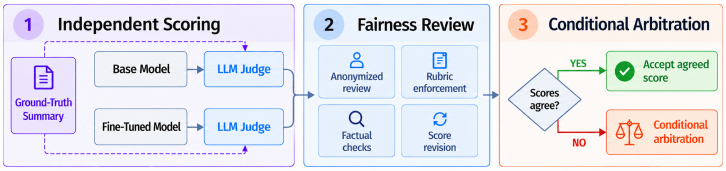


Figure 3: Summary quality scoring with fairness review and conditional arbitration.

LLM judge produces the initial criterion-level decisions, followed by an anonymized fairness review that checks consistent application of the anchors. Disagreements are conditionally arbitrated, while agreement between the initial judgment and fairness review bypasses arbitration. This design makes decisions traceable to specific criteria and supporting evidence.

Finally, the automatic evaluation is validated through domain-expert assessment and robustness checks across judge configurations. Agreement on individual workflows and overall model rankings is examined separately, with paired comparisons and uncertainty estimates supporting interpretation of performance differences. Output quality is considered alongside computational cost and latency, with measurement boundaries stated explicitly. The next section presents the SOC-specific instantiation of this framework, with the complete procedure taxonomy, rubric anchors, and worked examples provided in the appendix.

3 SOCBench: Learning from SOC Investigations

3.1 Task Definition and Data Collection

A security operations center (SOC) monitors and investigates potential security threats to an organization. During an investigation, analysts examine alerts, inspect affected entities and supporting evidence, and determine the incident's scope, significance, and appropriate disposition. This process involves navigating between interface views, following investigative leads, and interpreting successive observations. Building on the general framework presented in Section 2.2, we introduce SOCBench, a framework for reconstructing and summarizing these expert workflows from recorded SOC investigations, with paired visual and structured textual representations.

Given a recorded investigation and its incident context, the task is to generate three complementary report artifacts that preserve both the analyst's procedure and the investigation's findings. The *SOC Timeline* provides a chronological account of the analyst's actions and observations. The *Narrative Summary* explains the investigative process, evidence, findings, and conclusions. The *Executive Summary* provides a concise account organized into four fields: title, incident description, detected evidence, and conclusion, including the final disposition and supporting rationale. These artifacts support the separate assessment of procedural fidelity and summary quality defined in Section 2.2.

Data Collection. We recorded five SOC analysts while they investigated security incidents in Microsoft Defender. The resulting corpus contains 86 investigations, each capturing one analyst's interaction with the Defender interface throughout an investigation. We refer to this data collection as *SOCBench* data. Across the corpus, investigations lasted 28.8 minutes on average (median 14.3) and contained 262.7 rendered frames on average (median 190.5). To support the auxiliary training tasks, we additionally use 128 general computer-use recordings from the public Paradigm Shift AI Computer Use dataset (Howland et al., 2025) as support data. We split SOCBench data at the investigation level into 53 training, 17 validation, and 16 held-out test investigations. The split is fixed before recordings are expanded into task-specific examples, so all examples derived from one investigation remain in the same partition. The validation investigations are used for model selection, and the test investigations are reserved for final evaluation.

3.2 Interaction Representation and Model Training

Each recorded investigation is processed by an interaction decoder that converts the recording into a temporally ordered sequence of observations. We construct two aligned representations of the same sequence: the *Visual* representation consists of rendered interface frames, while the *DOM* representation serializes the corresponding DOM states as an initial full interface state followed by

Table 1: Final training examples and token-length statistics for the selected Visual and DOM task mixtures. A dash indicates that a task is not included in the selected mixture. Counts exclude validation and test examples. Token counts are computed over the finalized training examples after preprocessing and filtering, including exclusion of examples longer than 50,000 tokens.

	Visual			DOM		
Task	n	Mean tokens	Median	n	Mean tokens	Median
T1: Generic OCR	2,545	2,601.38	2,575	–	–	–
T2: Generic Transcription	921	2,230.62	2,127	–	–	–
T3: Generic Summarization	118	10,899.66	6,719	–	–	–
T4: Frame-to-DOM Reconstruction	–	–	–	–	–	–
T5: Defender Transcription	530	42,167.45	41,049	381	31,468.68	32,394
T6: Defender Filtering	53	7,646.02	6,498	53	8,394.70	6,808
T7: Defender Summarization	53	8,064.89	6,805	53	9,253.68	8,010
Total	4,220	–	–	487	–	–

subsequent state changes. The ordered observations are then divided chronologically into temporal batches of up to 35 observations, which provide the representation-specific inputs used during training. The resulting temporal batches provide the visual or textual inputs for reconstructing investigative actions. Following Section 2.2, training combines two task categories: general support tasks that provide supervision for understanding interfaces and interactions, and SOC-specific tasks that reconstruct, filter, and summarize each investigation.

General support tasks. Four tasks provide auxiliary supervision: *T1 (Generic OCR)* extracts visible UI text and interface state from a short frame sequence; *T2 (Generic Transcription)* converts general computer-use frames into a step-by-step interaction description; *T3 (Generic Summarization)* converts an ordered interaction trace into a session summary; and *T4 (Frame-to-DOM Reconstruction)* maps a single rendered frame to its corresponding DOM state. *T1–T3* use examples from the public computer-use dataset, while *T4* uses aligned frames and recorded DOM states from SOC training investigations. Therefore, tasks *T1–T3* are used for the Visual path and *T3* and *T4* for the DOM path.

SOC-specific tasks. Three tasks reconstruct and summarize investigations. *T5 (Defender Transcription)* takes a cumulative Previous Summary and a temporal batch of up to 35 ordered frames or corresponding DOM states, producing the Step-by-Step of the Current Batch. *T6 (Defender Filtering)* takes the concatenated batch traces and investigated incident, retaining incident-relevant events in their original order. *T7 (Defender Summarization)* converts the filtered trace into the Executive Summary and Narrative Summary. Both paths use these tasks: *T5* receives representation-specific observations, whereas *T6* and *T7* consume text.

Training examples. We used GPT-5.4 to generate output targets for all above-mentioned *T1-T5* tasks. Each batch contributes one *T5* example before length filtering, while *T6* and *T7* each contribute one example per investigation. A shared 50,000-token limit removes more DOM batches because corresponding batches can have different token lengths, leaving 530 Visual and 381 DOM *T5* examples. Examples from selected tasks are interleaved during training. Appendix A, Table 4 provides the complete task definitions and representation-specific applicability.

3.3 Experimental Setup

We evaluate two model families under Frames and DOM inputs: Qwen, using the Qwen/Qwen3.5-4B checkpoint Qwen Team (2026), and Gemma, using google/gemma-4-E2B-it Gemma Team et al. (2026). Each fine-tuned model is compared with its corresponding unadapted checkpoint. We additionally evaluate GPT-5.4, which supplies offline training supervision, as a teacher/control system without task-specific fine-tuning. Both student families support multimodal inputs; the Frames–DOM comparison concerns the observations supplied to the model.

We evaluate configurations selected independently for each model family and representation to assess their practical quality–efficiency trade-offs. Selection considers task composition, fine-tuning strategy (LoRA or full-parameter fine-tuning), learning rate, and training epochs. All experiments use the fixed split of 53 training, 17 validation, and 16 held-out investigations described in Section 3.1. Configurations are selected using validation results and fixed before held-out evaluation.

The selected Frames configurations combine T1–T3 with T5–T7, whereas the selected DOM configurations use only T5–T7. These mixtures contain 4,220 and 487 training examples, respectively, including 530 Frames and 381 DOM transcription examples after length filtering. Table 1 summarizes these independently selected mixtures; Appendix A documents the search space and selected settings. Their comparison measures the combined effects of representation and configuration choices, while the controlled experiment isolates representation under matched training conditions.

The controlled models and the four independently selected configurations use full-parameter supervised fine-tuning for three epochs with AdamW, a learning rate of 2×10^{-5}, weight decay of 0.01, a linear learning-rate schedule, and a warm-up ratio of 0.03. Training uses bfloat16 precision on NVIDIA H100 GPUs. Instruction, input-text, and media tokens are masked from the loss, which is computed only over target response tokens.

3.4 EVALUATION

From either the Visual or DOM representation described in Section 3.2, the models generate a *SOC Timeline*, *Narrative Summary*, and *Executive Summary*, as defined in Section 3.1 (see Appendix B.1). Following the two-branch framework in Section 2.2, we separately evaluate *procedural fidelity* on the SOC Timeline and *summary quality* on the Narrative Summary and four Executive Summary fields. LLM-based evaluation uses both GPT-5.4 and GPT-6 to mitigate potential bias from evaluating with the same model used as the teacher during data generation.

Procedural Fidelity through Canonical Correspondence. We evaluate procedural fidelity by establishing one-to-one correspondence between procedure-step–target occurrences in generated and ground-truth SOC Timelines. To account for differences in wording, each timeline is mapped to a sequence of canonical occurrences, each containing a procedure step, typed target, supporting evidence, and available position and recording metadata.

Here, the entry is mapped to `update_status`, with the incident ID as its typed target, following the taxonomy in Appendix B.2, Table 7; the procedural-fidelity pipeline (Figure 2) then combines LLM-based extraction with deterministic checks of taxonomy membership, target type, lifecycle compatibility, and source support.

Original: *06:20:38: Changed incident status to In Progress in the Manage incident pane.*

⇓

Canonical: (**procedure step:** `update_status`, **target (incident_id):** `89053`)

The ground-truth trace is constructed and frozen before candidate evaluation through extraction, review, canonicalization, and validation; the generated trace is extracted independently using only the canonical ground-truth label set. A global monotonic one-to-one assignment then establishes correspondence, using source order, frame/page metadata, and capture time to distinguish repeated operations while enforcing compatible procedure and target semantics. Supporting steps receive no additional credit, and unmatched occurrences are reviewed as unsupported additions, reference omissions, incorrect reference targets, or inconclusive cases. Procedural fidelity is reported as macro-averaged **PF-F1**: exact duplicates are removed before scoring but still affect compactness, while verified reference omissions and incorrect reference targets are neutralized in the reported score. Unadjusted PF-F1 and sequence-order Kendall's τ_b are retained as diagnostics; Figures 5 and 6 in Appendix B.2 illustrate the process.

Summary-Quality Evaluation. Summary quality is evaluated with an anchored rubric of 17 criteria across five report units: title, incident description, detected evidence, conclusion, and narrative. The criteria cover recall, factual grounding, entity roles, scope, outcome faithfulness, workflow progression, and surface integrity; the full rubric is provided in Appendix B.3, Table 8. Following the three-stage procedure in Section 2.3, an independent judge scores each report unit against fixed anchors and its corresponding ground-truth subsection. An anonymized fairness review then checks consistent rubric application, with disagreements sent to conditional arbitration. Criterion-level scores, revisions, rationales, and final decisions are retained for audit. To keep omission judgments localized, coverage is assessed only against the matching ground-truth subsection, although the complete report may be consulted to verify whether additional content is supported elsewhere. This allows cross-section evidence to be validated without creating new omission requirements. Finally, deterministic post-processing prevents cascading penalties from the same underlying error while preserving independent deductions; both original and adjusted scores are retained. The final summary-quality score is the mean of the 17 criterion scores after normalization by their declared

Table 2: Validation and held-out test results (%) by input modality, reported according to the metrics defined in Section 3.4. Exact model versions are given in Section 3.3.

		Validation set					Held-out test set				
Input	System	n	Rubric	PF-F1	Comp.	τ	n	Rubric	PF-F1	Comp.	τ
Frames	GPT teacher	17	**90.96**	**89.95**	**96.03**	86.68	16	**88.01**	**87.67**	**98.39**	94.44
	Qwen-4B tuned	17	86.59	89.63	84.36	91.23	16	84.24	87.39	92.18	**97.15**
	Qwen-4B base (enriched)	17	78.62	85.20	18.17	79.55	16	75.86	76.84	33.26	80.77
	Gemma tuned	17	70.29	80.18	62.48	**95.66**	16	74.82	75.60	79.59	94.15
	Gemma base	17	69.85	49.83	52.95	91.94	16	68.47	58.22	43.78	85.65
DOM	Qwen-4B tuned	17	87.89	**86.66**	**99.29**	**95.65**	16	**84.97**	80.88	**95.41**	94.14
	GPT teacher/control	17	**89.62**	80.86	97.00	82.30	16	84.70	**83.56**	25.88	93.42
	Qwen-4B base	17	82.83	78.93	20.89	93.85	16	79.23	78.92	24.30	93.97
	Gemma tuned	17	79.33	83.25	86.22	95.05	16	76.01	77.77	63.37	**95.28**
	Gemma base	17	58.00	53.04	17.72	92.01	16	53.63	55.65	24.25	84.39

Table 3: Inference efficiency of selected Frames and DOM Qwen-4B models on the 17-investigation validation cohort. Duration reflects the retained source-capture span, while token counts are cumulative per report. Both Frames and DOM values average five complete runs.

Input	Mean / median investigation duration (s)	Mean summary-generation latency (s/report)	Mean input tokens/report	Mean output tokens/report
Frames	1,175.34 / 986.25	55.14	509,947	5,143
DOM	1,175.34 / 986.25	22.63	277,166	5,000

maxima, giving each criterion equal weight. It is reported separately from PF-F1, and Appendix B.3 provides a worked example.

4 RESULTS

Table 2 reports validation results in the left block and held-out test results in the right. Frames and DOM results are reported separately because their source representations and context construction differ; we do not combine them into a cross-modality grand score.

Interpreting teacher performance. The ground-truth reports were seeded by an LLM, then iteratively reviewed and refined through human validation before being frozen as the evaluation reference. Relative to this human-validated reference, the GPT teachers provide strong supervision, although abstractive differences in emphasis, step grouping, target granularity, or contextual interpretation can reduce rubric or PF-F1 agreement despite aligned incident conclusions. The GPT models nevertheless achieve high aggregate agreement, approximately 90% for the leading Frames teacher, and rank at or near the top of their respective leaderboard cohorts. We therefore treat absolute scores as agreement with the reference rather than direct estimates of report correctness, and use relative system ordering for model selection.

Distillation gains. GPT-generated Frames and DOM summaries supervised the corresponding Qwen and Gemma students. Fine-tuning improves every base-model comparison on validation, with all gains persisting on the held-out test set. Although the held-out set contains more challenging investigations and yields lower absolute scores for the leading systems, the broad model hierarchy remains stable: the Frames ordering is preserved across both evaluation branches, while the DOM ordering changes only in the middle of the procedural-coverage ranking. Thus, the main model-selection conclusions persist under the more demanding test distribution.

Qwen performs better overall, while Gemma shows larger relative gains. Gemma's weaker base performance, particularly under Frames, may reflect long-context demands: Frames requests commonly contain 50k–100k input tokens, approaching Gemma-4 E2B's 128K context window, where its model card also reports limited long-context performance Gemma Team et al. (2026); Qwen3.5-4B, by contrast, supports 262K tokens (Qwen Team, 2026). Because the reported evaluations use different benchmarks, we treat this as a plausible contributing factor rather than a causal explanation.

Under Frames, fine-tuned Qwen nearly closes the teacher gap on PF-F1 while retaining a larger gap in rubric-based summary quality, suggesting stronger transfer of investigated actions and targets. Under DOM, transfer is less uniform: fine-tuned Qwen exceeds the GPT control on validation PF-F1 and reaches comparable held-out rubric quality, but remains below GPT on held-out PF-F1. Taken together, teacher-supervised fine-tuning improves both model families across representations, with gains varying by representation and evaluation branch.

4.1 Inference Efficiency

Representation trade-off. The selected Qwen configurations reveal a clear deployment trade-off between summary quality, procedural correspondence, and inference efficiency. DOM achieves higher rubric scores, whereas Frames achieves higher PF-F1 on both splits. In the controlled inference benchmark (Table 3), DOM generates a report in 22.63 seconds versus 55.14 seconds for Frames, a 59.0% reduction in summary-generation latency, while processing approximately 45.6% fewer cumulative input tokens (277,166 versus 509,947). Because both configurations use the same model scale and hardware allocation, this reduction corresponds to lower inference-time cost under a fixed hardware-hour tariff. Importantly, both representations are constructed from the same investigation traces after preprocessing removes irrelevant lingering. The efficiency difference therefore reflects the representation and context-construction pipelines under the intended operating regime, rather than redundant idle interaction in the underlying investigations. Taken together, DOM provides higher summary quality and lower inference cost, whereas Frames provides stronger procedure-faithful correspondence.

Robustness and expert alignment. Across judge architectures, absolute rubric scores vary, but the broad system ordering remains stable, with procedural fidelity showing even greater stability across judge families (Appendix Section C.1). SOC experts independently rank the Frames GPT teacher first and prefer the fine-tuned Frames Qwen model among deployable SLMs (Appendix Section C.2). Together, these results suggest that the observed gains are not artifacts of a single evaluator and are directionally consistent with analyst preferences. Although some variation remains across comparisons, the overall ranking is broadly stable and consistent with expert judgment.

5 Discussion and Conclusions

The results show a practical trade-off between representations. DOM provides similar observed summary quality to Frames, but Frames achieves higher procedural fidelity for the selected Qwen systems. This difference reinforces the need to evaluate report quality and procedural correspondence separately: a coherent summary may preserve an investigation's conclusions while omitting parts of its execution. DOM also reduces mean report-generation latency by 59% on the validation cohort, while processing 45.6% fewer cumulative input tokens at the same model scale and hardware allocation. Rapid navigation across pages or systems may reduce this advantage by requiring full DOM descriptions instead of incremental updates. These measurements exclude decoding and representation construction. However, both can run online during user activity, substantially reducing their contribution to post-session latency for either representation.

Teacher-supervised fine-tuning improves both model families across representations, with gains persisting on held-out investigations. Frames Qwen approaches its teacher's procedural fidelity while retaining a larger summary-quality gap, indicating uneven transfer across capabilities.

Evaluation robustness is strongest for broad system ordering. Absolute rubric scores vary across judges, while procedural scores are more stable. Experts descriptively rank the Frames teacher first and prefer Frames Qwen among the evaluated SLMs, but paired human differences do not survive multiplicity correction. The evidence supports broad alignment with expert preferences, with scores interpreted as reference agreement.

By combining paired interaction representations, task-based fine-tuning, and separate evaluations of procedural fidelity and summary quality, SOCBench provides a foundation for learning from recorded expert workflows. The results support structured text as an efficient input for workflow documentation while highlighting the value of visual observations for procedural reconstruction. This work advances the transformation of expert interactions into reusable organizational knowledge and a foundation for future agents.

AI USE STATEMENT

In this work, we used generative AI tools to support research execution and code development, to serve as LLM judges within the evaluation methodology described in the paper, and to aid and polish writing. All AI-generated or AI-assisted research artifacts were reviewed and validated by the authors, including testing code for correctness and verifying experimental and evaluation outputs. All AI-assisted writing was reviewed and edited by the authors for accuracy, clarity, and consistency with the underlying results.

ETHICS STATEMENT

This work does not use data from real customers, organizations, or security incidents. The experiments were conducted using a live/public demonstration environment built around a fictional customer organization and designed for demonstrating and testing security scenarios across products. Consequently, the data used in this study does not contain real customer information or personally identifiable information.

REPRODUCIBILITY STATEMENT

To support reproducibility, we will release the code and data developed in this work. This includes the implementation of the interaction decoding, representation construction, training, inference, and evaluation pipelines, as well as the SOCBench data of recorded SOC investigations. We will provide the code for transforming recorded user-computer-interaction trajectories into the Visual and structured DOM representations, constructing temporal batches, and generating the training and inference examples used by the workflow-reconstruction tasks. While our experiments instantiate this pipeline on SOC investigations, the general methodology is designed for recorded computer workflows and can be adapted to other domains with corresponding recorded interaction data. We will also release the scripts and configurations required to reproduce the procedural-fidelity and summary-quality evaluations reported in this work. Together, these resources will support independent reproduction, verification, and extension of our results.

REFERENCES

Noga BenYoash, Menachem Brief, Oded Ovadia, Gil Shenderovitz, Moshik Mishaeli, Rachel Lemberg, and Eitam Sheetrit. Secque: A benchmark for evaluating real-world financial analysis capabilities. In *Proceedings of the Fourth Workshop on Generation, Evaluation and Metrics (GEM^2)*, pp. 212–230, 2025.

Meni Brief, Oded Ovadia, Gil Shenderovitz, Noga Ben Yoash, Rachel Lemberg, and Eitam Sheetrit. Mixing it up: The cocktail effect of multi-task fine-tuning on llm performance – a case study in finance, 2024. URL https://arxiv.org/abs/2410.01109.

Hiroki Furuta, Kuang-Huei Lee, Ofir Nachum, Yutaka Matsuo, Aleksandra Faust, Shixiang Gu, and Izzeddin Gur. Multimodal web navigation with instruction-finetuned foundation models. In *International Conference on Learning Representations*, volume 2024, pp. 29528–29558, 2024.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, et al. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026.

Anais Howland, Ashwin Thinnappan, and Jameel Shahid Mohammed. Computer use data - paradigm shift ai, 2025. URL https://huggingface.co/datasets/anaisleila/computer-use-data-psai.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 2511–2522, 2023.

Xing Han Lu, Zdeněk Kasner, and Siva Reddy. WebLINX: Real-world website navigation with multi-turn dialogue. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria

Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 33007–33056. PMLR, 21–27 Jul 2024.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 12076–12100, 2023.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Ronal Singh, Shahroz Tariq, Fatemeh Jalalvand, Mohan Baruwal Chhetri, Surya Nepal, Cecile Paris, and Martin Lochner. Llms in the soc: An empirical study of human-ai collaboration in security operations centres. In *2026 IEEE Symposium on Security and Privacy (SP)*, pp. 4262–4281. IEEE, 2026.

Michael Wornow, Avanika Narayan, Ben Viggiano, Ishan S Khare, Tathagat Verma, Tibor Thompson, Miguel A Hernandez, Sudharsan Sundar, Chloe Trujillo, Krrish Chawla, et al. Wonderbread: A benchmark for evaluating multimodal foundation models on business process management tasks. *Advances in Neural Information Processing Systems*, 37:115963–116021, 2024.

Andy K Zhang, Neil Perry, Riya Dulepet, Joey Ji, Celeste Menders, Justin Lin, Eliot Jones, Gashon Hussein, Samantha Liu, Donovan Jasper, Pura Peetathawatchai, Ari Glenn, Vikram Sivashankar, Daniel Zamoshchin, Leo Glikbarg, Derek Askaryar, Haoxiang Yang, Aolin Zhang, Rishi Alluri, Nathan Tran, Rinnara Sangpisit, Kenny Oseleononmen, Dan Boneh, Daniel Ho, and Percy Liang. Cybench: A framework for evaluating cybersecurity capabilities and risks of language models. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 25094–25243, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/3e9412a9c1d93810ef3ef7825115016b-Paper-Conference.pdf.

A Ablation Study and Model Selection Details

For completeness, Table 4 provides the full input and output definition of every primary and support task considered during model development, together with the representation paths for which each task was considered.

All T4 examples used in model-selection experiments were constructed only from SOC investigations in the training split; validation and test investigations did not contribute T4 supervision.

Across the model-selection experiments, we varied the training-task mixture, fine-tuning strategy, and optimization configuration. All candidates used the same 53-investigation SOC training cohort for the Defender tasks and were compared on the same 17-investigation validation cohort. The 16 held-out test investigations were not used during model selection. Not every combination of the search dimensions was evaluated for every model family and representation.

Table 5 summarizes the main dimensions explored. Task-composition experiments varied the inclusion of Defender Filtering (T6) and support tasks. DOM experiments also considered different T4 treatments and temporal batch sizes.

After model selection, one configuration was fixed for each model family and representation before held-out evaluation. Table 6 reports the four selected fine-tuned systems used in the main results.

The validation and held-out performance of these selected systems is reported in Table 2.

Table 4: Training tasks considered for the Visual and DOM paths during model development. The three Defender tasks instantiate the primary workflow tasks in Section 2.2; support tasks are varied during model selection.

Task	Input	Output	Visual	DOM
Support tasks				
T1: Generic OCR	Short sequence of rendered frames	Visible UI text and interface state	✓	–
T2: Generic Transcription	Short sequence of general computer-use frames	Step-by-step interaction description	✓	–
T3: Generic Summarization	Ordered interaction trace	Session-level summary	✓	✓
T4: Frame-to-DOM Reconstruction	One rendered frame	Corresponding DOM state	–	✓
Primary Defender tasks				
T5: Defender Transcription	Cumulative Previous Summary + current batch of up to 35 ordered rendered frames (Visual) or corresponding ordered DOM states	Step-by-Step of the Current Batch	✓	✓
T6: Defender Filtering	Concatenated Step-by-Step outputs from all batches + investigated incident	Filtered Concatenated Step-by-Step	✓	✓
T7: Defender Summarization	Filtered Concatenated Step-by-Step from T6	Executive Summary + Narrative Summary	✓	✓

Table 5: Main dimensions varied during model selection. The alternatives shown do not imply that every Cartesian combination was evaluated for every model family and representation.

Dimension	Type	Alternatives considered
Training-task mixture	Task composition	Alternative combinations of the primary Defender tasks and support tasks; T6 was evaluated both included and excluded.
T4 treatment	Task composition	No T4, proportional T4 sampling, and full T4 sampling in DOM candidates.
DOM T5 batch size	Input construction	15 or 35 DOM states per temporal batch.
Fine-tuning method	Training strategy	Full-parameter supervised fine-tuning and LoRA.
Training epochs	Optimization	1 or 3 epochs.
Learning rate	Optimization	5×10^{-6} for refinement runs and 2×10^{-5} for scratch runs.

Table 6: Selected fine-tuning configurations used in the main evaluation.

Setting	Qwen Visual	Qwen DOM	Gemma Visual	Gemma DOM
Base checkpoint	Qwen3.5-4B	Qwen3.5-4B	Gemma-4-E2B-it	Gemma-4-E2B-it
Task mixture	T1, T2, T3, T5, T6, T7	T5, T6, T7	T1, T2, T3, T5, T6, T7	T5, T6, T7
T5 input	≤ 35 rendered frames	≤ 35 DOM states	≤ 35 rendered frames	≤ 35 DOM states
Fine-tuning	Full-parameter SFT	Full-parameter SFT	Full-parameter SFT	Full-parameter SFT
Epochs	3	3	3	3
Learning rate	2×10^{-5}	2×10^{-5}	2×10^{-5}	2×10^{-5}
Weight decay	0.01	0.01	0.01	0.01
Warmup ratio	0.03	0.03	0.03	0.03
Precision	bfloat16	bfloat16	bfloat16	bfloat16

B SOC INVESTIGATION PROCEDURE

B.1 SOC INVESTIGATION REPORT EXAMPLE

Executive Summary

Title
Command and control incident involving multiple users

What the incident is about
Cloud-identity compromise activity affecting multiple users, characterized by successful sign-ins and cloud app access from suspicious IP addresses, with no corroborating endpoint process or network activity. Scope included at least 15 accounts in the incident, with seven accounts directly confirmed via identity telemetry.

What was detected

- High-severity incident #89053 opened in Microsoft Defender, aggregating multiple correlated alerts and merges from other incidents into a single case.
- Key suspicious IPs: `108.142.230.59` and `20.107.46.209` (ISP Microsoft Corporation, ASN 8075; location: Netherlands/Schiphol); IP `20.107.46.209` showed "979 active alerts in 38 incidents," reputation Suspicious (1/100).
- Alerting included multiple "TI Map IP Entity to SigninLogs" detections linking `20.107.46.209` to users such as `u8573@int.zava-corp.com`, `u4165@int.zava-corp.com`, and `u2517@int.zava-corp.com`.
- Advanced hunting (Identity telemetry, 7-day lookback) found successful `IdentityLogonEvents` from the suspicious IPs for seven users:
  - `u1362@int.zava-corp.com` (7 events)
  - `u623@int.zava-corp.com` (6)
  - `u22938@int.zava-corp.com` (4)
  - `u12944@int.zava-corp.com` (3)
  - `u21720@int.zava-corp.com` (1)
  - `u13548@int.zava-corp.com` (1)
  - `yabershitz@microsoft.com` (1)
- Cloud app access associated with these IPs within 7 days: Microsoft 365 (21 events), Microsoft 365 Copilot Chat (4 events from `20.107.46.209`), Microsoft Dynamics 365 (1 event from `20.107.46.209`).
- Advanced hunting found no endpoint corroboration:
  - `DeviceProcessEvents` filtering `InitiatingProcessRemoteSessionIP`/`ProcessRemoteSessionIP` for the two IPs returned 0 results (7-day and 180-day checks).
  - `DeviceNetworkEvents` for the two IPs returned 0 results (7-day).
- Assets > Devices listed 0 impacted devices; incident Assets included 15 accounts, indicating primarily cloud-identity impact.

Conclusion
Verdict: True Positive - Resolved
Reasoning: Multiple "TI Map IP to SigninLogs" alerts and `IdentityLogonEvents` confirm successful logons and cloud app usage from suspicious IPs across several accounts, consistent with compromised cloud identities. The absence of `DeviceProcessEvents`/`DeviceNetworkEvents` indicates no endpoint foothold, constraining scope to cloud access. Alerts and the incident were classified "True positive - Compromised account" and resolved after bulk handling of 80 related alerts.

Narrative Summary

The investigation of command-and-control incident #89053 centered on suspicious sign-in activity tied to two external IPs (108.142.230.59 and 20.107.46.209). The incident was high severity and consolidated multiple correlated alerts, ultimately scoped to compromised accounts rather than endpoint-driven C2 activity.

Key entities included IPs 108.142.230.59 and 20.107.46.209 (ASN 8075, Microsoft; geolocated to Schiphol, NL; 20.107.46.209 flagged as suspicious with 979 active alerts in 38 incidents globally) and at least 15 impacted accounts. Identity-focused hunting identified seven organization accounts with recent logons from these IPs: u1362@int.zava-corp.com, u623@int.zava-corp.com, u22938@int.zava-corp.com, u12944@int.zava-corp.com, u21720@int.zava-corp.com, u13548@int.zava-corp.com, and u8573@int.zava-corp.com; one external identity (yabershitz@microsoft.com) also appeared once. No devices were implicated.

Methodology combined alert correlation, entity enrichment (IP reputation/geo/ASN), and advanced hunting across DeviceProcessEvents, DeviceNetworkEvents, and IdentityLogonEvents. KQL queries against device telemetry returned no process or network events linked to the IPs over 7–180 days, narrowing scope to cloud identity. A pivotal IdentityLogonEvents query confirmed recent sign-ins from the suspect IPs across seven users and cloud app access (Microsoft 365; Copilot Chat; Dynamics 365).

Critical findings: multiple user accounts exhibited successful sign-ins from the two suspect IPs with no corroborating endpoint activity, indicating an identity-focused compromise. The alerts "TI Map IP Entity to SigninLogs" drove correlation across affected users.

Investigation flow: opened and triaged incident, enriched IPs, ruled out endpoint involvement via hunting, confirmed identity-based access via IdentityLogonEvents, and bulk-classified 80 alerts as True positive - Compromised account. Incident determination was set to True alert and resolved. Several incidents (e.g., 91079, 91039, 91002, 90964, 90823, 90775, 90677) were merged into #89053 as part of automated correlation to reflect a single multi-user compromise event.

SOC Analysis Timeline - Step-by-Step Log

- **06:20:19:** Opened main incident #89053 (Command and control incident involving multiple users) in Microsoft Defender - severity High, state Active, classification Unclassified.
- **06:20:29:** Opened Manage incident and assigned the incident to u21256@int.zava-corp.com (Assign to me).
- **06:20:38:** Changed incident status to In Progress in the Manage incident pane.
- **06:20:41:** Saved incident updates - confirmation banner "Incident updated successfully" (6:20 AM).
- **06:20:58:** Navigated to Evidence and Response for incident #89053 and reviewed evidence items including a cloud logon session and suspicious IP addresses 108.142.230.59 and 20.107.46.209; impacted assets shown include u2652@int.zava-corp.com and 15 Accounts.
- **06:29:53:** Selected IP address 108.142.230.59 and opened its details - observed "5 active alerts in 5 incidents" and organization/geo information (Postal code 1171).
- **06:29:59:** Opened details for IP address 20.107.46.209 - observed "979 active alerts in 38 incidents" (majority Medium, 1 High).
- **06:30:16:** Closed the IP details side pane and returned to the evidence list.
- **06:30:56:** Switched to the alerts/activities view and reviewed entries showing automated correlations, playbook executions, and multiple incident merges into #89053 (e.g., incidents 91079, 91039, 91002, 90964, 90823, 90775, 90677).
- **06:31:03:** Opened Activities for main incident #89053 to review active alerts and impacted assets list.
- **06:31:11:** Navigated to Evidence and Response for incident #89053 to review consolidated evidence.
- **06:31:49:** Selected evidence item and opened IP details for 108.142.230.59 - observed ISP microsoft corporation (ASN 8075), location netherlands (schiphol), and note showing 5 active alerts in 5 incidents.
- **06:32:29:** Opened IP details for 20.107.46.209 - ISP microsoft corporation (ASN 8075), location netherlands (schiphol), with detection summary 979 active alerts in 38 incidents.

- **06:31:37:** Opened Advanced hunting workspace from incident #89053.
- **06:32:04:** Used Copilot to ask for a query to check process creation events from IPs `108.142.230.59` and `20.107.46.209` and summarize counts by account.
- **06:33:37:** Executed `DeviceEvents` query filtering last 7 days for `RemoteIP` `108.142.230.59` and `20.107.46.209`, summarizing event counts by `AccountName` (See query 1 below).
- **06:33:43:** Query returned 0 results - no process creation events found for IPs `108.142.230.59` or `20.107.46.209` in the last 7 days.
- **06:34:40:** Continued work in Advanced hunting for main incident #89053; opened the editor containing the prior IP-focused process query context for `108.142.230.59` and `20.107.46.209` (no results message visible).
- **06:34:42:** Began composing a new process-creation-oriented query using schema suggestions for Create/ProcessCreate datasets.
- **06:35:02:** Pivoted to the Devices schema and targeted the `DeviceProcessEvents` table to look for process activity tied to IPs `108.142.230.59` and `20.107.46.209`.
- **06:35:05:** Editor displayed an error indicating the column `RemoteIP` does not exist in `DeviceProcessEvents` ("The name 'RemoteIP' does not refer to any known column").
- **06:35:09:** Updated the filter logic to use an `in~` list containing the IPs `108.142.230.59` and `20.107.46.209` while revising the query (no execution recorded).
- **06:35:16:** Inspected available fields in `DeviceProcessEvents` (e.g., `DeviceName`, `AccountUpn`, `InitiatingProcessAccountName`) to identify correct attributes for filtering and summarization.
- **06:35:24:** Located the field `InitiatingProcessRemoteSessionIP` in the `DeviceProcessEvents` schema, indicating the likely correct column to use for IP-based filtering in subsequent queries.
- **06:35:25:** Executed `DeviceProcessEvents` query for last 7 days filtering `InitiatingProcessRemoteSessionIP in~` ("108.142.230.59", "20.107.46.209") and summarizing counts by `AccountName` to check for process activity tied to these IPs for main incident #89053 (See query 2 below).
- **06:35:41:** Observed query results for the above `DeviceProcessEvents` search - no results returned (0 items, "No results found in the specified time frame").
- **06:35:39:** Viewed schema help for column `InitiatingProcessRemoteSessionIP` ("IP address of the remote device from which the initiating process's RDP session was initiated") to validate correct field usage.
- **06:36:12:** Scrolled through `DeviceProcessEvents` field list (e.g., `ProcessRemoteSessionIP`, `InitiatingProcessAccountName`, `ProcessCreationTime`) to identify additional attributes for refining subsequent hunts related to IPs `108.142.230.59` and `20.107.46.209`.
- **06:36:22:** Executed `DeviceProcessEvents` query to investigate process activity where `ProcessRemoteSessionIP` matches `108.142.230.59` or `20.107.46.209` in the last 7 days, summarizing counts by `AccountName` (See query 3 below).
- **06:36:38:** Executed `DeviceNetworkEvents` query to check network connections to/from `108.142.230.59` and `20.107.46.209` in the last 7 days, summarizing `EventCount` and distinct `InitiatingProcessAccountName` values by `RemoteIP` (See query 4 below).
- **06:36:47:** Viewed results for the `DeviceNetworkEvents` query - 0 items returned; Copilot summarized no network connections from `108.142.230.59` or `20.107.46.209` in the last 7 days.
- **06:36:57:** Used Copilot to search for security alerts involving IPs `108.142.230.59` and `20.107.46.209` in the last 7 days - response indicated no alerts found.
- **06:37:26:** Opened the IP overview page for `20.107.46.209` - observed 979 active alerts across 38 incidents, prevalence 0 devices in organization (30 Days), and reputation Suspicious (1/100).
- **06:37:46:** Navigated to incident #89053 Alerts tab and reviewed multiple "TI Map IP Entity to SigninLogs" alerts with impacted assets including `u8573`, `u4165`, and `u2517`, showing correlation reasons such as Similar IP counts.

- **06:38:00:** Returned to Advanced hunting and confirmed both queries for IPs `108.142.230.59` and `20.107.46.209` showed no results (0 items) for the last 7 days.
- **06:38:43:** Executed `DeviceProcessEvents` query over last 180 days to search for process activity where `ProcessRemoteSessionIP` matches `108.142.230.59` or `20.107.46.209`, summarizing counts by `AccountName` (results pending) for main incident #89053 (See query 5 below).
- **06:38:56:** Opened main incident #89053 (Command and control incident involving multiple users reported by multiple sources) and viewed the Alerts tab with 80 alerts listed.
- **06:39:14:** Executed `IdentityLogonEvents` query for logons from IP addresses `108.142.230.59` and `20.107.46.209` in the last 7 days, summarizing `EventCount` per `AccountDomain` and `AccountName` and returning `AccountSid`/`AccountUpn`/`ObjectId` for context (See query 6 below).
- **06:39:38:** Viewed query results for `IdentityLogonEvents` - 7 accounts with logons from the targeted IPs, including `u1362@int.zava-corp.com` (7 events), `u623@int.zava-corp.com` (6), `u22938@int.zava-corp.com` (4), `u12944@int.zava-corp.com` (3), `u21720@int.zava-corp.com` (1), `u13548@int.zava-corp.com` (1), and `yabershitz@microsoft.com` (1).
- **06:39:57:** Navigated within incident #89053 to the Assets (15) tab to review impacted assets.
- **06:40:26:** Reviewed Advanced hunting results for `IdentityLogonEvents` from IPs `108.142.230.59` and `20.107.46.209` in the last 7 days - 7 accounts returned with event counts: `u1362@int.zava-corp.com` (7), `u623@int.zava-corp.com` (6), `u22938@int.zava-corp.com` (4), `u12944@int.zava-corp.com` (3), `u21720@int.zava-corp.com` (1), `u13548@int.zava-corp.com` (1), `yabershitz@microsoft.com` (1); Copilot displayed observations, visualization grouped by `AccountName`, and recommended actions (e.g., force password reset for 6 users).
- **06:40:19:** Opened main incident #89053 (Command and control incident involving multiple users) - viewed Assets > Devices showing 0 items (status High, In Progress; owner `u21256@int.zava-corp.com`).
- **06:40:41:** Used Copilot action to check geolocation of source IP addresses for the `IdentityLogonEvents` summary - Copilot started generating a KQL query.
- **06:40:48:** Executed `IdentityLogonEvents` query to list logon events from `108.142.230.59` and `20.107.46.209` (7-day lookback) with account and location fields for geolocation analysis - query execution failed with an error message (no results returned) (See query 7 below).
- **06:41:06:** In Advanced hunting, executed a query to geolocate source IP addresses `108.142.230.59` and `20.107.46.209` using identity logon data (7-day lookback) - execution failed with semantic error: `extract(): failed to cast argument 2 to scalar constant.`
- **06:41:23:** Opened main incident #89053 and reviewed Assets > Devices - 0 items listed.
- **06:41:45:** Opened alert "TI Map IP Entity to SigninLogs" (Alert ID `snd7e278d6-e68c-47ce-874c-8323a3e603eb`) linked to incident #89053 - viewed query results showing SigninLogs match for IP `20.107.46.209` associated with user `u8573@int.zava-corp.com` (severity Medium; status In progress; workspace CyberSOC).
- **06:42:28:** On incident #89053 Alerts tab, loaded the full list of 80 alerts with filter Status: New, In progress.
- **06:42:31:** Selected all 80 alerts and opened Manage alerts.
- **06:42:43:** Returned to Advanced hunting and reviewed Copilot summary of cloud app access from IPs `108.142.230.59` and `20.107.46.209` in the last 7 days - Microsoft 365 (21 events), Microsoft 365 Copilot Chat (4 events from `20.107.46.209`), and Microsoft Dynamics 365 (1 event from `20.107.46.209`).
- **06:42:43:** In the Manage alerts pane for incident #89053, prepared bulk update for 80 selected alerts - Status set to In progress, Assign to `u21256@int.zava-corp.com`, Classification Unclassified (no save action recorded).
- **06:43:17:** In Advanced hunting, reviewed Copilot conversation summarizing activity for IPs `108.142.230.59` and `20.107.46.209` - no process or network events or alerts in the last 7 days, but `IdentityLogonEvents` showed 7 accounts with logons (e.g., `u1362@int.zava-corp.co`

m, u623@int.zava-corp.com) and cloud apps accessed (Microsoft 365, Microsoft 365 Copilot Chat, Microsoft Dynamics 365).

- **06:43:25:** Returned to incident #89053 Alerts tab and opened Manage alerts for 80 selected alerts (status In progress; assigned to u21256@int.zava-corp.com) to perform a bulk update.
- **06:43:27:** In the Manage alerts pane for incident #89053, changed Classification for the 80 selected alerts to "True positive - Compromised account."
- **06:44:30:** Entered a bulk comment in the Manage alerts pane for the 80 selected alerts (no save action captured).
- **06:46:08:** Saved bulk Manage alerts update for 80 alerts in incident #89053 - set Status to In progress, assigned to u21256@int.zava-corp.com, and Classification to True positive - Compromised account - confirmation banner "Alerts updated successfully" at 6:46 AM.
- **06:46:34:** In Manage incident for #89053, updated incident determination to "True alert" and saved - banner "Incident updated successfully" (last update time Jun 10, 2026 6:46 AM).
- **06:46:49:** In Manage incident for #89053, changed Status to "Resolved" and began filling "Explain why you resolved this incident" (no save confirmation shown).
- **06:46:51:** Opened Manage incident for main incident #89053 and prepared resolution - assigned to u21256@int.zava-corp.com, set Status to Resolved, Classification to True positive - Compromised account, and entered a resolution explanation.
- **06:47:07:** Saved Manage incident updates for incident #89053 - banner showed "Incident updated successfully" and Last update time: Jun 10, 2026 6:47 AM.
- **06:47:11:** Clicked the More actions menu on the incident #89053 page.

Figure 4: SOC investigation report example for incident #89053. The report contains the Executive Summary, Narrative Summary, and complete SOC Analysis Timeline for the investigation of suspicious cloud-identity activity affecting multiple user accounts.

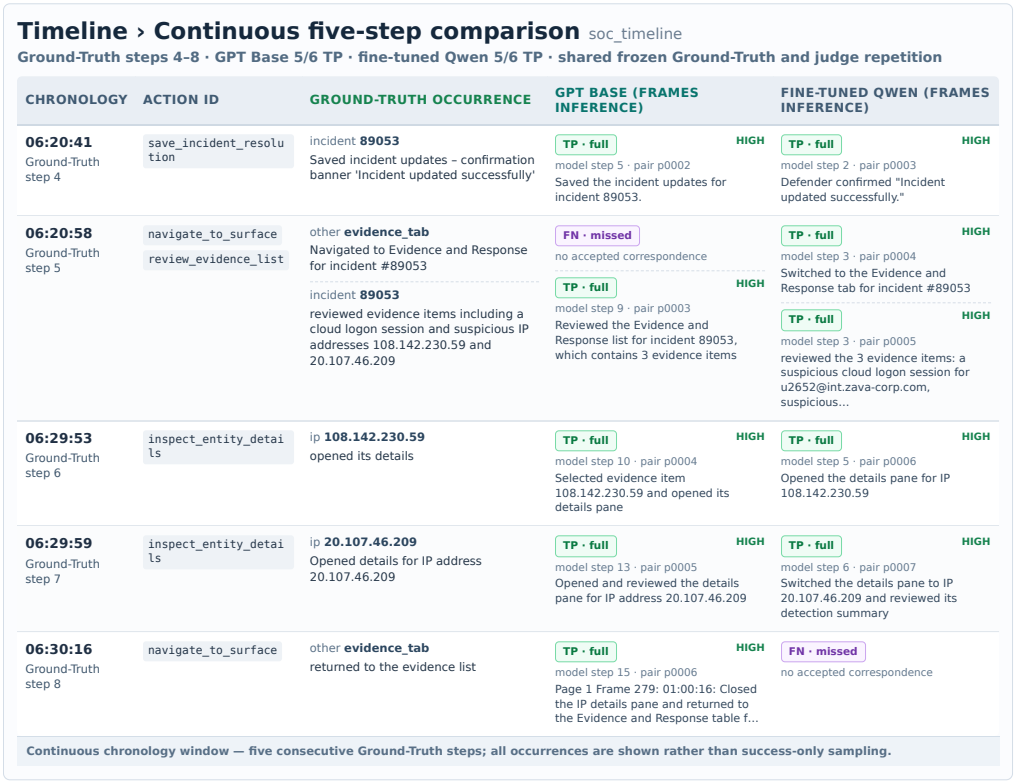
B.2 Procedural Fidelity via Canonical Procedure Correspondence

Table 7: **SOC Investigation Procedure Taxonomy for procedural-fidelity evaluation.** The 43 canonical procedure-step types define scoreable occurrences used to compute PF-F1. Eight additional residue-only labels are retained to audit specialized UI cases but cannot contribute true positives or false negatives. The family groupings are explanatory and do not affect correspondence or scoring.

Family	Canonical procedure steps	Operational scope
Entry and navigation	`open_incident`, `return_to_incident`, `navigate_to_surface`, `open_advanced_hunting`, `open_email_explorer`, `open_alert_details`, `open_custom_detection_rule`, `pivot_to_entity_page`	Entering, returning to, or switching among investigation surfaces and objects.
Triage and review	`review_asset_list`, `review_alert_list`, `review_activities_log`, `review_attack_story_graph`, `review_evidence_list`, `review_alert_timeline`, `inspect_alert_query_results`, `inspect_query_record`	Inspecting incident-level collections, histories, graphs, timelines, and telemetry records.
Entity and evidence	`inspect_entity_details`, `inspect_file_reputation`, `inspect_process_tree`, `review_aad_alert_history`, `review_device_timeline`, `review_email_components`, `review_ti_indicator`, `review_user_risk`, `review_zap_actions`	Examining users, devices, processes, files, messages, indicators, and their histories or risk state.
Hunting and detection	`compose_kql_query`, `execute_hunt_query`, `review_query_results`, `review_rule_query`, `review_triggered_alerts`	Constructing and executing hunts, inspecting results, and reviewing detection logic or related alerts.
Copilot assistance	`copilot_request_query`, `copilot_generate_query`, `copilot_summarize`, `generate_analyst_notes`	Submitting assistant requests and consuming generated queries, summaries, or notes.
Response and lifecycle	`add_custom_indicator`, `assign_incident`, `bulk_classify_alerts`, `classify_alert`, `classify_incident`, `contain_entity`, `copy_indicator_value`, `save_incident_resolution`, `update_status`	Committing classification, containment, assignment, indicator, status, and resolution operations.
Target schema	`user`, `device`, `mailbox`, `app_or_process`, `ip`, `file`, `hash`, `url`, `domain`, `email_message`, `query`, `alert`, `incident`, `rule`, `policy`, `other`, `none`	The object on which an action operates; `none` is reserved for targetless navigation.
Structured context	`incident_id`, `alert_family`, `alert_count`, `query_table`, `status`, `classification`, `ui_surface`, `entities[]`	Source-grounded bindings used to distinguish otherwise similar actions and targets.



Figure 5: **Example canonical procedure correspondence between a ground-truth timeline and a generated timeline.** Each source step is decomposed into one or more canonical procedure-step–target occurrences, shown beneath the original timeline text. Green edges denote accepted one-to-one correspondences between occurrences despite differences in wording. The unmatched navigation occurrence illustrates a ground-truth procedure step that is not recovered in the generated timeline and therefore contributes a false negative to PF-F1.



Timeline › Continuous five-step comparison soc_timeline

Ground-Truth steps 4–8 · GPT Base 5/6 TP · fine-tuned Qwen 5/6 TP · shared frozen Ground-Truth and judge repetition

CHRONOLOGY	ACTION ID	GROUND-TRUTH OCCURRENCE	GPT BASE (FRAMES INFERENCE)	FINE-TUNED QWEN (FRAMES INFERENCE)
06:20:41 Ground-Truth step 4	`save_incident_resolution`	incident **89053** Saved incident updates – confirmation banner 'Incident updated successfully'	TP · full HIGH model step 5 · pair p0002 Saved the incident updates for incident 89053.	TP · full HIGH model step 2 · pair p0003 Defender confirmed "Incident updated successfully."
06:20:58 Ground-Truth step 5	`navigate_to_surface` `review_evidence_list`	other **evidence_tab** Navigated to Evidence and Response for incident #89053 / incident **89053** reviewed evidence items including a cloud logon session and suspicious IP addresses 108.142.230.59 and 20.107.46.209	FN · missed no accepted correspondence / TP · full HIGH model step 9 · pair p0003 Reviewed the Evidence and Response list for incident 89053, which contains 3 evidence items	TP · full HIGH model step 3 · pair p0004 Switched to the Evidence and Response tab for incident #89053 / TP · full HIGH model step 3 · pair p0005 reviewed the 3 evidence items: a suspicious cloud logon session for u2652@int.zava-corp.com, suspicious…
06:29:53 Ground-Truth step 6	`inspect_entity_details`	ip **108.142.230.59** opened its details	TP · full HIGH model step 10 · pair p0004 Selected evidence item 108.142.230.59 and opened its details pane	TP · full HIGH model step 5 · pair p0006 Opened the details pane for IP 108.142.230.59
06:29:59 Ground-Truth step 7	`inspect_entity_details`	ip **20.107.46.209** Opened details for IP address 20.107.46.209	TP · full HIGH model step 13 · pair p0005 Opened and reviewed the details pane for IP address 20.107.46.209	TP · full HIGH model step 6 · pair p0007 Switched the details pane to IP 20.107.46.209 and reviewed its detection summary
06:30:16 Ground-Truth step 8	`navigate_to_surface`	other **evidence_tab** returned to the evidence list	TP · full HIGH model step 15 · pair p0006 Page 1 Frame 279: 01:00:16: Closed the IP details pane and returned to the Evidence and Response table f…	FN · missed no accepted correspondence

Continuous chronology window — five consecutive Ground-Truth steps; all occurrences are shown rather than success-only sampling.

Figure 6: **Occurrence-level procedural-fidelity decisions for a continuous timeline excerpt.** Five consecutive ground-truth chronology steps are compared with the GPT Base (frames inference) and Fine-Tuned Qwen 3.5 VLM (frames inference) using the same frozen reference trace. Each model column reports whether a ground-truth occurrence received an accepted one-to-one correspondence (TP) or remained unmatched (FN), together with the corresponding generated step and pair identifier. All occurrences in the excerpt are shown; the table is not restricted to successful correspondences.

B.3 SUMMARY-QUALITY RUBRIC

Table 8: **Anchored summary-quality rubric.** All 17 criteria receive equal weight after normalization by their declared maximum scores. Criteria D-2, N-1, N-2, N-4, and N-5 each comprise four independently scored components and have a raw maximum of eight. Every other criterion has a raw maximum of two, with possible scores of 0, 1, or 2.

Report unit	Criterion	Evaluation construct
Title	**T-1: Topic alignment**	Correct and sufficiently specific threat or detection topic; naming a primary entity is beneficial but not required.
Title	**T-2: Surface integrity**	Readable, well-formed title without language drift, encoding artifacts, broken markup, or template collapse.
Incident description	**A-1: Primary entity correctness**	Correct primary user, device, account, or indicator in its reference-supported role; semantic wording variants are accepted.
Incident description	**A-2: Faithfulness and scope**	Grounded threat type, stage, severity, and affected scope without fabricated behavior or unsupported over-claiming.
Incident description	**A-3: Surface integrity**	Clear, coherent, and well-formed incident description without language, encoding, or template artifacts.
Detected evidence	**D-1: Detection-item recall**	Coverage of salient alerts, techniques, process chains, evidence, indicators, and hunting findings in the matching reference subsection.
Detected evidence	**D-2: Factual precision**	Component score for primary-claim grounding, secondary-detail grounding, role/fact compatibility, and absence of invented phases or malformed entities.
Detected evidence	**D-3: Scope discipline**	Detection bullets remain focused on the focal incident without unrelated incidents or peripheral padding.
Detected evidence	**D-4: Bullet integrity**	Detection bullets are readable, consistently structured, and free of language, encoding, markup, or template defects.
Conclusion	**C-1: Verdict alignment**	Correct normalized verdict, reference-required operational subcategory, and incident or entity scope.
Conclusion	**C-2: Reasoning faithfulness**	Verdict rationale matches the reference basis, cites genuine telemetry, and introduces no fabricated correlates.
Narrative	**N-1: Story arc**	Component score for reference-present phases, methodology and findings, relative order and causal flow, and absence of an invented major phase.
Narrative	**N-2: Findings substance**	Component score for observation accuracy, interpretation accuracy, limiting context, and absence of material contradiction.
Narrative	**N-3: Entity coverage**	Reference-supported users, devices, processes, indicators, alerts, and incidents appear in the narrative in their correct roles.
Narrative	**N-4: Narrative faithfulness**	Component score for primary and secondary claim grounding, role/fact compatibility, and absence of invented pivots or malformed entities.
Narrative	**N-5: Outcome accuracy**	Component score for reference-explicit status, classification, determination, and linkage to the correct outcome scope.
Narrative	**N-6: Surface integrity**	Coherent, grammatical, and readable long-form prose without substantial language, encoding, formatting, or template defects.



Figure 7: **Summary-quality evaluation for the "What Was Detected" report unit.** The candidate and base reports are initially scored independently against the corresponding ground-truth subsection and fixed rubric anchors. Their decisions are then cross-audited under anonymization to check consistent rubric application. Conditional arbitration is invoked only when an initial score and its fairness-reviewed score disagree.

C Evaluation Robustness and Expert Validation

C.1 Cross-Judge Stability

We assess evaluator repeatability by holding generated reports and references fixed and rerunning the judging pipeline with isolated caches. Table 9 covers all ten summary-generating systems using GPT-5.4 and GPT-6 judges. All models are evaluated on the same 17 investigations for both judges.

The observed SDs range from 0.09 to 1.77 percentage points, supporting repeatable cohort-level scoring under repeated judging. Moreover, both judges preserve the same model ordering within each representation for both rubric-based summary quality and procedural fidelity (PF-F1). This ranking agreement persists despite differences in absolute score calibration, including lower GPT-6 rubric scores and larger procedural-score differences for the Gemma base models. These results support the stability of the observed within-representation rankings across the two judges. Because repetition coverage varies, the comparisons remain descriptive.

Table 9: Automatic-judge repeatability across ten summary-generating models. Low variability in cohort-level scores across repeated evaluations supports the framework's repeatability, while broad ranking trends are consistent with the separate SOC-expert assessment. Scores (%) are means ± SD across completed full-cohort repetition means, with generated reports and references held fixed. Repetition coverage varies.

		GPT-5.4 automatic judge		GPT-6 automatic judge	
Input	Summary-generating model	Rubric	PF-F1	Rubric	PF-F1
Frames	GPT Base	90.54 ± 0.27	89.35 ± 0.70	85.15 ± 0.62	87.26 ± 0.50
Frames	Qwen-4B tuned	86.64 ± 0.09	89.31 ± 0.65	81.53 ± 0.37	89.29 ± 0.56
Frames	Qwen-4B base	78.70 ± 0.41	81.01 ± 0.62	70.14 ± 0.54	79.81 ± 0.11
Frames	Gemma tuned	72.65 ± 0.46	79.59 ± 1.01	68.65 ± 0.37	79.73 ± 0.42
Frames	Gemma base	70.60 ± 0.51	50.16 ± 0.57	66.88 ± 0.35	41.15 ± 1.37
DOM	Qwen-4B tuned	87.90 ± 0.56	85.75 ± 0.67	81.41 ± 0.25	87.39 ± 0.18
DOM	GPT Base	89.85 ± 0.38	80.14 ± 0.87	84.17 ± 0.29	80.83 ± 0.19
DOM	Qwen-4B base	82.72 ± 0.80	78.33 ± 0.59	75.87 ± 0.48	77.99 ± 0.43
DOM	Gemma tuned	79.33 ± 0.15	81.15 ± 1.51	74.39 ± 1.04	81.01 ± 0.39
DOM	Gemma base	57.54 ± 0.34	52.55 ± 0.74	53.94 ± 1.77	43.49 ± 1.13

C.2 SOC-Expert Validation

Five SOC subject-matter experts independently assessed blinded and randomized reports from three archival model configurations. The study covers 11 investigations: eight reviewed by three experts and three by two, yielding 30 reviewer - investigation ballots and 90 summary ratings, with 30 ratings per model. Table 10 reports rating-weighted mean scores and descriptive preferences. Experts ranked the Frames GPT teacher highest (94.17), followed by Frames Qwen tuned (91.17) and archival DOM Qwen LoRA (87.50). Among the reviewed SLMs, Frames Qwen received more approvals without edits (19 versus 13) and fewer "most risky" votes (5 versus 11).

On the same 11 investigations and archival model outputs, both GPT-5.4 and GPT-6 summary-quality evaluations reproduce the aggregate human ordering ($\rho = \tau_b = 1$ across three model-level means). This agreement supports alignment with expert preferences on the reviewed subset. For statistical comparisons of human scores, reviewer ratings are first averaged within each investigation, and the resulting 11 investigation means receive equal weight. None of the three pairwise system comparisons remains statistically significant after Holm correction for multiple comparisons. The observed ordering therefore supports descriptive alignment without establishing superiority or equivalence. Its scope is limited to the reviewed systems: the archival DOM LoRA model differs from the selected full-weight DOM checkpoint used in the current ten-model evaluation.

Separately, experts judged 1,251 of 1,252 reference-action-step ratings correct and identified nine gaps. These counts include repeated reviewer judgments and support the face validity of the reviewed reference traces. They do not independently validate PF-F1 scores assigned to generated reports.

Table 10: SOC-expert assessment by five SMEs across 11 investigations, with 30 ratings per system. The aggregate human ranking agrees with both automatic judges on the matched archival subset. Mean scores are rating-weighted; scores and critical-issue rates are percentages, while approvals and votes are counts. The DOM result refers to the archival LoRA model.

Summary-generating model	Mean score	Approve as is	Best overall	Most risky	Critical-issue rate
Frames GPT teacher	94.17	22	13	4	6.67
Frames Qwen-4B tuned	91.17	19	10	5	20.00
DOM Qwen (archival LoRA)	87.50	13	7	11	20.00