National College of Ireland

Evaluating Reasoning vs. No-Reasoning Models on Insider-Threat Datasets

MSc Research Project

Cyber Security Online

Lihao Sun
Student ID: 23330449

School of Computing
National College of Ireland

Supervisor: Raza Ul Mustafa

National College of Ireland
MSc Project Submission Sheet

School of Computing

Student Name:	Lihao Sun
Student ID:	23330449
Program:	MSc Cyber Security Online **Year:** 2024
Module:	MSc Research Project
Lecturer:	Raza Ul Mustafa
Submission Due Date:	11-Dec-2025
Project Title:	Evaluating Reasoning vs. No-Reasoning Models on Insider-Threat Datasets
Word Count:	5157 **Page Count:** 19

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Lihao Sun

Date: 08-Dec-2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Evaluating Reasoning vs. No-Reasoning Models on Insider-Threat Datasets

Lihao Sun
23330449

Abstract

Security operations centres (SOCs) are tasked with analyzing enormous volumes of asymmetrical log data that is unevenly distributed. Actual insider threat attacks are extremely hard to detect. This thesis asks one question: do “reasoning-oriented” LLMs (used as RL-post-trained / RL-distilled model families) work better than non-reasoning models for insider-threat detection on the CERT r6.2 dataset? We reformat raw log events into concise 4W session summaries (When, Where, What, Which), train models with careful attention to class imbalance data, and evaluate performance using AUPRC and AUROC. To keep training comparable, we adapt the 7B and 8B parameter models using LoRA with INT4 quantisation, fitting them onto consumer-grade GPUs typical in SOCs.

Our findings indicate that the ‘reasoning’ model—DeepSeek R1 Distill Qwen 7B with LoRA—achieves higher AUROC than the ‘no-reasoning’ model – LlaMA 3.1 8B Instruct with LoRA—on a 200k validation slice. However, baseline encoder models were evaluated on the full 1.41M-session validation set in the initial draft, which limits direct cross-model ranking. Therefore, conclusions are framed as protocol-specific until an apples-to-apples re-evaluation is completed on a fully harmonised evaluation set.

1. Introduction

1.1 Motivation

Modern SOCs face the huge task of extracting actionable intelligence from the flood of network, host, and system logs. A challenge is the severe class imbalance, where true insider threats are exceedingly rare. Traditional detection models, trained on historical data, are vulnerable to any concept drift. The attackers could use generative tools to change their tactics, signature-based models degrade until they are retrained. At the same time, early deployments of LLMs in related areas—such as IEC-61850 digital substations—suggest that GenAI-assisted detectors can reduce errors and scale better than purely manual, human-in-the-loop (HITL) workflows, while adapting more quickly to novel patterns (Zaboli et al., 2024).

A new generation of RL-post-trained LLMs shows step-by-step (chain-of-thought) reasoning and is trained as a policy to maximise a task reward; in our setting we

hypothesise this helps aggregate time-ordered evidence across a session rather than treat each log line alone (DeepSeek-AI, 2025; Ouyang et al., 2022; Schulman et al., 2017). When combined with out-of-distribution (OOD) monitoring and escalation protocols, these systems can flag unfamiliar patterns and limit human analysts, therefore reducing over-all-confident and false alarms (Samaddar et al., 2024). Advances in efficient inference as Native Sparse Attention and lightweight adaptation (LoRA with INT4 quantisation), together with work on improving the quality of small training sets, suggests that compact, reasoning-capable models can be run effectively on consumer-grade GPUs typical of SOC deployments (Yuan et al., 2025).

The central problem, however, is that despite this promise of the reasoning LLMs, a question remains unclear: Do reasoning-aligned LLMs improve performance on insider-threat logs when compared with strong non-reasoning baselines? We test this under severe class imbalance and realistic compute limits.

1.2 Research Question

“Do reasoning-aligned LLMs such as DeepSeek-R1 family outperform conventional non-reasoning models on session-based classification of insider threats in the CERT r6.2 dataset, under budget-aware training conditions?”

1.3 Research Statement and Objectives

Research Statement. Our goal is to see if the reasoning ability of certain large language models provides a real advantage for detecting malicious sessions. This evaluation uses the CERT r6.2 dataset and operates under strict computational limits and severe class imbalance data.

Primary Objective. To conduct a controlled comparison between reasoning and non-reasoning models on the CERT r6.2 dataset using a standardized 4W session-based representation.

1.4 Key Characteristics: Reasoning vs. Non-Reasoning Models

Table 1 outlines key differences reported in previous studies between reasoning-enabled models and non-reasoning models, focusing on learning behaviour, response to new threats, decision-making, and deployment practicality (Li & Shan, 2023; Samaddar et al., 2024; DeepSeek-AI, 2025; Yuan et al., 2025). It is a synthesis of prior work, not results from our experiments. These characteristics directly motivate this thesis: We standardise all logs into a 4W token format, employ metrics that account for severe class imbalance, calibrate prediction scores, and track computational costs like latency and memory usage.

Aspect	Reasoning AI	Non-Reasoning AI

Learning ability	Adaptable via RL/policy optimisation or distillation.	Weights remain frozen after initial training; no further adaptation.
Response to new threats	Can respond out of training data, including zero-shot detection of novel and some of AI-generated attack patterns.	Often blind to attacks outside training data.
Decision-making	Uses step-by-step reasoning to connect related log events and explain why a session is flagged.	Performs single-step pattern matching without contextual reasoning.
Practicality	Needs some optimisation (for example, LoRA and quantisation), but can be deployed on standard GPUs if tuned carefully.	Lightweight by default, but limited in adaptability and detection depth.

Table 1: Literature synthesis of key characteristics of reasoning vs. non-reasoning models (summary of prior work; not results from our experiments).

Scope: This study is confined to log-based user-behaviour detection within an enterprise-style setting, using the CERT r6.2 dataset as the primary evaluation corpus.

2. Literature Review

2.1 Gen AI for Anomaly Detection

In smart-grid networks, messages like GOOSE and Sampled Values follow strict rules, so unusual sequences can be a sign of an attack or fault. Zaboli et al. (2024) rewrite these substation rules into a task-oriented dialogue and use a conversational GenAI agent to check traffic against the expected behaviour. They report that detection improves as the model gets stronger and as more human feedback is included.

This matters because their system does not judge packets in isolation. It checks the relationship between messages over time, which is key for spotting abnormal behaviour. For example, it checks that packet numbers are in order, or that a state number consistently increases. This analysis of context is what makes the system effective at securing and stabilizing the smart grid.

Building on this, we preprocess contextual logs into sequences (behaviour-aware 4W - When, Where, What, Which) so the model can use context across events. Our goal is to give the model similar cross-event reasoning capabilities.

2.2 Hardware-Aligned Sparse Attention

Yuan et al. introduced Native Sparse Attention (NSA), a sparsity pattern and training method that is natively trainable and optimised for commodity accelerators, offering favorable speed/throughput trade-offs without sacrificing accuracy (Yuan et al., 2025).

Hardware-aligned model efficiency is critical for real-world deployment and NSA integrates algorithmic innovations with hardware-aligned optimizations to achieve efficient long-context modeling. The paper specifically explains the need for hardware-friendly algorithm design to mitigate memory access and hardware scheduling bottlenecks.

We prioritize model efficiency to ensure our systems can be deployed in the real world. Although our method does not implement Yuan et al.'s Native Sparse Attention (NSA) directly, their findings help our focus on efficiency. For our 7B-scale models, this means using practical techniques like INT4 quantization and LoRA to maximize speed and throughput without compromising accuracy.

2.3 Eliciting Reasoning through Reinforcement Learning

We extend the approach of DeepSeek-R1, which demonstrated that pure reinforcement learning (RL) can evoke step-by-step reasoning in smaller, compact models (1.5B–7B parameters) (DeepSeek-AI, 2025). Their success on math and logic benchmarks leads us to a central question: Can this RL-elicited reasoning transfer from clean benchmarks to the messy, real-world task of classifying insider threats? We are testing this specifically on security logs that have been rendered as concise "who-what-when-where" (4W) narratives.

2.4 Managing Uncertainty in Cyber-Defense

Neurosymbolic cyber-defense agents that combine behavior trees with probabilistic OOD monitoring can detect distributional shifts and escalate decisions to humans rather than acting with misplaced confidence under uncertainty (Samaddar et al., 2024).

Our design calibrates confidence scores and flags low-confidence outputs for analyst review.

2.5 Data-Efficient Reinforcement Learning

We apply the data-efficiency principles outlined by Li, Zou & Liu (2025a) to our methodology for a resource-constrained home lab. They showed that a high-quality, small RL corpus of ~1.4k samples can outperform a much larger set by using careful selection and design (Li, Zou & Liu 2025a). For this reason, we focused on creating

clean data splits, rigorous calibration, and precise evaluation over simply maximizing our dataset size.

2.6 LLMs in Vulnerability Detection: Potential and Pitfalls

While research shows LLMs can help find vulnerabilities like stack overflows and triage results, it also warns of their tendency to hallucinate without rigorous prompts and verification (Li & Shan, 2023). We've integrated this lesson directly into our SOC analytics by pairing LLM scoring with calibration, conservative thresholds, and mandatory human verification in our evaluation protocol.

2.7 Dual-Modality Fusion for Insider Threats

Our thesis uses a 4W representation for logs, a format directly inspired by Kong et al. Their DMFI framework, which fuses a semantic and a behavioral (4W) view using two LoRA-tuned LLMs, showed strong results (Kong X. et al., 2025). By adopting this compatible format now, we can seamlessly integrate a similar dual-view approach in the future, once we've established the value of reasoning for the core task.

2.8 Cybersecurity Benchmarks for LLM Evaluation

Recent work has started to define benchmark suites that measure how good LLMs perform on practical CTI and cybersecurity tasks.

CTIBench evaluates LLMs on practical cyber threat intelligence tasks. These include CTI knowledge questions, mapping CVEs to CWEs, predicting CVSS severity vectors, attributing threats to actors, and extracting attack techniques. This gives a more relevant comparison for cybersecurity work than general NLP benchmarks (Alam et al., 2024).

CySecBench is a cybersecurity prompt dataset designed to test how LLMs behave when they receive malicious or unsafe requests. It focuses on safety and robustness, and shows that security skills and policy compliance should be tested directly rather than assumed from general training (Wahréus et al., 2025).

2.9 Domain-Specific Models and Retrieval-Augmented Generation

To reduce hallucination and improve terminology coverage, domain-adapted foundation models are being trained specifically for security.

Foundation Sec 8B Instruct is a security-focused instruction-tuned model in the FoundationAI-SecurityLLM family, built on Llama 3.1 (Kassianik et al., 2025; Weerawardhena et al., 2025).

CyberRAG takes a different approach. Instead of retraining the model, it retrieves relevant security information and feeds it into the LLM to support more accurate and better-structured incident reporting (Blefari et al., 2025).

2.10 LLMs for Log Processing and Quality

LLMs are increasingly explored as assistants that can summarise, normalise, and debug operational logs.

Ji and Luo present an LLM-driven workflow for intelligent log processing and autonomous debugging in cloud AI platforms, shows how LLMs can help parse logs and support downstream fault investigation (Ji and Luo, 2025).

Because logging quality directly affects what a SOC model sees, Defects4Log introduces a benchmark of logging-code defect patterns and evaluates whether LLMs can detect and reason about these issues, which is relevant when log representations are used for security analytics (Wang et al., 2025).

2.11 Reliability of Explicit Reasoning and Alternatives

Although chain-of-thought prompting often increase performance, it can also introduce new failure modes.

Li et al. (2025b) show that explicit reasoning can degrade instruction-following in some settings, implying that “thinking” traces should be evaluated as part of reliability, not only accuracy (Li et al., 2025b).

As an alternative to emitting long rationales, Geiping et al. propose latent-space recurrent reasoning that scales test-time compute by iterating a recurrent block, avoiding extra output tokens while still improving multi-step performance (Geiping et al., 2025).

Mechanistic work on chain-of-thought faithfulness such as feature-level intervention studies using sparse autoencoders helps, clarify when generated thoughts reflect internal reasoning and when they may be post-hoc rationalisations (Chen et al., 2025).

At decoding time, rationale-enhanced decoding (RED) treats rationales as a conditioning signal and improves reasoning by combining an image-conditional distribution with a rationale-conditional distribution, offering a middle ground between no-rationale and full chain-of-thought outputs (Yamaguchi et al., 2025).

For multi-step systems, Bouzenia and Pradel analyse thought-action-result trajectories of software engineering agents, offering a concrete audit lens for reasoning-driven behaviours and tool-use sequences that parallels our “thinking vs. no-thinking” comparison design (Bouzenia and Pradel, 2025).

2.12 Efficient Architectures and Scaling for Practical Deployment

Efficiency trends increasingly matter for deploying advanced models in constrained environments.

Nakamura et al. analyse how mixture-of-experts sparsity impacts reasoning tasks and provide guidance for selecting routing sparsity that balances compute and capability (Nakamura et al., 2025).

DeepSeek-AI's DeepSeek-V3 hardware analysis discusses scaling challenges and constraints in current accelerators, reinforcing the importance of hardware-aware design for training and serving high-capability models efficiently (Zhao et al., 2025).

For insider threat detection specifically, MambaITD proposes a cross-modal Mamba-based network for insider threat detection and highlights the importance of modelling temporal dynamics, providing a strong non-LLM baseline family alongside transformer-based approaches (Kong K. et. al., 2025).

Beyond technical efficiency, Governance-as-a-Service proposes a multi-agent framework for compliance and policy enforcement in agentic AI systems, supporting our emphasis on monitoring, escalation, and human-in-the-loop safeguards when deploying LLM-based cyber defence tools (Gaurav et al., 2025).

3. Methodology

3.1 Dataset & Sessionisation

The CERT Insider Threat Test Dataset is a synthetic collection for insider-threat research produced by CMU's CERT with ExactData under DARPA sponsorship, containing both normal background activity and planted malicious behaviour (Carnegie Mellon University, 2020).

The datasets are grouped by generator release versions (e.g., r3.1, r3.2), with later releases generally superseding earlier functionality; each bundle includes a README explaining that release's features. An accompanying archive, answers.tar.bz2, serves as an answer key describing the malicious scenarios and the synthetic user IDs involved. In this series, r6.2 is the latest publicly available release.

This study uses the CERT r6.2 dataset as its primary corpus. We create session-level instances by implementing a sliding 60-minute window with a 10-minute interval on events arranged in chronological order, grouped by user and host identity.

A session receives an abnormal label if it covered with any insider ground-truth interval specified in the dataset. All other sessions are considered normal.

3.2 4W-Guided Behaviour Abstraction

To manage the vast amount of log data, we represent all events using a compact 4W token scheme, based on the DMFI dual-modality framework (Kong X, et. al., 2025):

- When — hour-of-day, day-of-week, and short-range rate deltas.
- Where — user and host/computer identifiers, plus subnet/ASN buckets.
- What — actions such as auth_failed, logon_success, process_start, dns_query, file_copy.
- Which — resource IDs grouped by account, process family, port/service, or file-path family.

Each session is rendered as a single string of ≤512 tokens containing the ordered 4W elements along with light summary statistics (e.g., events per minute, counts of unique actions).

Dimension	**Pre-4W (raw)**	**Post-4W**
Schema (columns)	id, date, user, pc, to, cc, bcc, from, activity, size, attachments, content	user, host, start, text, label
Row count	646,148 rows	1,407,163 sessions
users	3,831 (user)	387 (user)

Table 2: CERT r6.2: Pre-4W (raw) vs Post-4W (sessionised)

3.3 Splits & Imbalance Handling

We partition the data into time-aware, identity-disjoint 70/10/20 train/validation/test splits using fixed random seeds, where identity is defined by the user and host pair. Because the “positive” cases are extremely rare (about 0.0187% of all sessions), a normal training loss would ignore them.

So, during training we use a loss that gives extra weight to the positive class (class-weighted cross-entropy) or focuses on hard, rare examples (focal loss). We then pick the best model using the validation AUPRC score. Which is a better measure than

accuracy when positives are very rare. This protocol minimizes temporal leakage and ensures a ‘fair’ comparison across models under severe class imbalance.

3.4 Models Compared

Our comparison pits strong non-reasoning baselines against a compact family of reasoning-capable models, all using the same 4W session data. The non-reasoning set includes a classical TF-IDF with a Logistic Regression baseline; DistilBERT-base and DistilRoBERTa-base encoders with a 512-token limit; Longformer-base-4096 as a long-context control and LLaMA 3.1 8B Instruct. Although LLaMA 3.1 8B Instruct uses SFT+RLHF for alignment, it is treated as a non-reasoning here because it is not explicitly post-trained to elicit stable multi-step deliberation.

The reasoning model uses DeepSeek R1 Distill Qwen 7B with LoRA and 4-bit quantisation. Encoder models use a simple classification head. The LLMs take an instruction prompt plus the 4W session text and then output a Normal/Abnormal decision.

The main LLM comparison is approximately size-matched (7B vs 8B), uses the same 4W representation (≤512 tokens), and applies a similar adapter-based fine-tuning approach (LoRA + INT4).

3.4.1 Operational definition of “reasoning” in this study

In this thesis, “reasoning” is defined operationally (by study-scoped, testable criteria) rather than as a claim that the model thinks like a human. A model is treated as reasoning-oriented if it belongs to a family explicitly post-trained to elicit stable multi-step behaviour under instruction prompts (e.g., RL-post-training / RL-distillation) and is used in this thesis to integrate evidence across multiple ordered events within a session.

Models are treated as non-reasoning baselines if they are standard instruction-tuned LLMs without an explicit RL-for-reasoning objective, or encoder-style classifiers that perform direct classification over the session text. Also instruction-tuned LLMs that are not explicitly post-trained with an RL for reasoning objective. In this study, LLaMA 3.1 8B Instruct is therefore treated as a non-reasoning.

3.5 Hardware & Practical Limits

All experiments ran on a single workstation with an Intel Core i9-12900KF CPU, 64 GB system RAM, and a 1 TB SSD. We started with an NVIDIA RTX 5080 (16 GB VRAM) and later changed it for an RTX 6000 Pro (96 GB VRAM).

Baselines Encoders (TF-IDF/LogReg, DistilBERt, DistilRoBERTa, Longformer) were trained and evaluated on the RTX 5080. The larger 7B model (DeepSeek R1 Distill

Qwen 7B + LoRA and LLaMA 3.1 8B Instruct + LoRA) were trained on the upgraded RTX 6000 Pro to comfortably fit longer sequences, larger batch sizes, and LoRA adapters.

3.6 Training & Regularisation

We follow the split and imbalance setup in §3.3 and the 512-token limit in §3.2. We used AdamW, dropout 0.1, weight decay 0.01, gradient clipping, and early stopping by validation AUPRC.

Baselines Encoders (RTX 5080):

We fine-tune DistilBERT, DistilRoBERTa and Longformer end-to-end on the CERT r6.2 train set with cross-entropy.

- DistilBERT (distilbert-base-uncased): max_ len 512, batch 16, 3 epochs, lr 2e-5, weight_decay 0.01, warmup_ratio 0.1, linear scheduler
- DistilRoBERTa (distilroberta-base): max_len 512, batch 16, 3 epochs, lr 2e-5, weight_decay 0.01.
- Longformer (longformer-base-4096): tokenizer model_max_length 4096, train at MAX_LEN 2048; batch 2 with grad_accum 8 (effective 16); 3 epochs; lr 2e-5; weight_decay 0.01.

All use seed=42 and identity-disjoint, time-aware splits. Checkpoints (when saved) are chosen by validation AUPRC.

7B and 8B LLMs (RTX 6000 Pro):

For DeepSeek R1 Distill Qwen 7B and LLaMA 3.1 8B Instruct we use the base in INT4 (Q4) and train only LoRA adapters plus a 2-class head.

- LoRA: q_proj, k_proj, v_proj, o_proj, gate_proj, up_proj, down_proj; r=16, alpha=32, dropout=0.05.
- Optimisation: AdamW (lr 1e-4), max_steps=200,batch 1–2 with gradient accumulation, max_len 512.
- Input: instruction header + 4W session text -> Normal/Abnormal.
- Selection: best validation AUPRC; inference in INT4.

3.7 Calibration

We fit temperature scaling on the validation split and report Expected Calibration Error (ECE) and Brier score. No fixed operating thresholds (e.g., high-precision or high-recall) are used in the results tables; comparisons focus on threshold-free metrics (AUPRC/AUROC).

3.8 Metrics & Statistics

Our evaluation primarily uses threshold-free metrics that do not require choosing a fixed decision threshold, with a focus on imbalance-aware measures.

AUPRC (Area Under the Precision–Recall Curve). This metric shows precision against recall across all decision thresholds. Under extreme class imbalance, AUPRC directly reflects the operational trade-off an analyst faces: if the system raises “k” alerts, how many are likely to be true positives? A higher AUPRC indicates better performance.

AUROC (Area Under the ROC Curve). The Receiver Operating Characteristic (ROC) curve show the true positive rate against the false positive rate. AUROC measures the model's overall ranking quality and is less sensitive to class prevalence. However, a high AUROC can be misleading under imbalance if precision remains poor, so we always report it alongside AUPRC.

3.9 Verification & Validation Protocol

All model training uses the time-aware, identity-disjoint training partition of CERT r6.2. Evaluation is performed on matched validation and test subsets within each evaluation protocol, ensuring comparability only among models evaluated under the same protocol. Calibration is fit on validation and then reused for testing; we do not tune operating thresholds on the test set.

For a controlled comparison, models must be evaluated on the same validation/test sets under the same protocol and same hardware. In the initial project runs, LLM experiments were evaluated on a 200k validation slice on same RTX 6000 PRO. While encoder baselines were evaluated on the full 1.41M-session validation set on same RTX 5080. These results should therefore be interpreted as protocol-specific and are not used for cross-protocol ranking.

3.10 Threats to Validity

Internal validity is strengthened by time-aware, identity-disjoint splits, fixed random seeds, and explicit configuration control. External validity is inherently limited by the semi-synthetic nature of the CERT r6.2 dataset; we partially mitigate this through calibration and by planning future work on external corpora. Construct validity reflects our focus on user and activity logs via the 4W representation; this scope excludes packet-centric intrusion detection, which would require different feature sets.

4. Results

4.1 Findings

This section shows results from experiments completed on November 3, 2025 . The following tables summarise the main results reported in this thesis. As a baseline, a random classifier's AUPRC is equal to the positive class prevalence(p): 0.000187 on the full validation set (263 positives out of 1,407,163 sessions) and 0.001315 on the 200k subset (263 positives out of 200,000 sessions).

Positive class prevalence:

Prevalence $p = \frac{positives}{Total\ examples}$

Full validation set: $p = \frac{263}{1{,}407{,}163} \approx 0.000187 = 0.0187\%$

200k subset: $p = \frac{263}{200{,}000} \approx 0.001315 = 0.1315\%$

Model / Checkpoint	**Train regime**	**Eval N / Pos**	**AUROC**	**AUPRC**
DeepSeek-R1-Distill-Qwen-7B + LoRA	Unbalanced, 200 steps	200,000 / 263	0.6386	0.002190
DeepSeek-R1-Distill-Qwen-7B + LoRA	**Balanced**, 3 epochs	200,000 / 263	0.5197	0.001414
LLaMA-3.1-8B-Instruct + LoRA	Unbalanced, max_steps=200	200,000 / 263	0.4999	0.001315

Table 3: Evaluated on the 200k validation slice.

Model / Run	**Train regime**	**Eval N / Pos**	**AUROC**	**AUPRC**

DistilBERT (class-weighted) *(runs/distilbert_weighted)*	Unbalanced	1,407,163 / 263	0.7257	0.000527
DistilRoBERTa-base *(runs/distilroberta_full)*	Unbalanced	1,407,163 / 263	0.6650	0.000450
Longformer-base-4096 *(runs/longformer_full)*	Unbalanced	1,407,163 / 263	0.6260	0.000269
TF-IDF + Logistic Regression *(runs/tfidf_baseline)*	Unbalanced	1,407,163 / 263	0.5887	0.000232

Table 4: Evaluated on the full validation set (1.41M)/

Hardware & precision note. 7B and 8B LoRA models were trained/evaluated on an RTX 6000 Pro (96 GB VRAM); the baselines were trained/evaluated on an RTX 5080 (16 GB VRAM). All 7B and 8B runs used INT4 (Q4) this time. And the baselines used framework defaults (FP16 or FP32). Precision can affect calibration and AUPRC.

4.2 Visual AUPRC

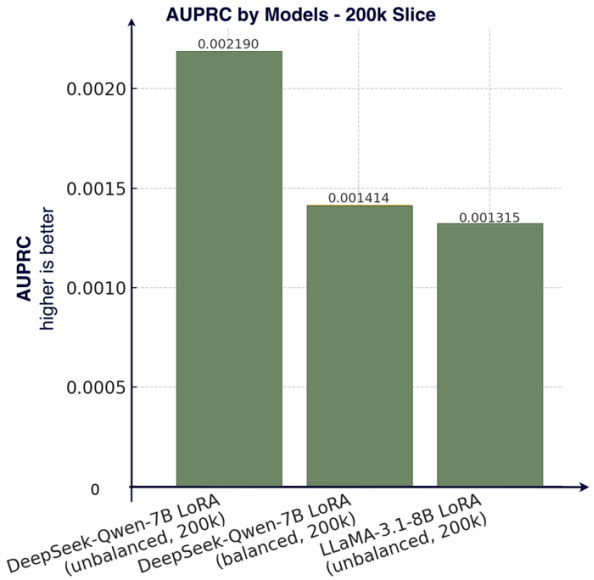

Figure 1: AUPRC- 200k validation. (Bars are comparable only within the 200k evaluation protocol)

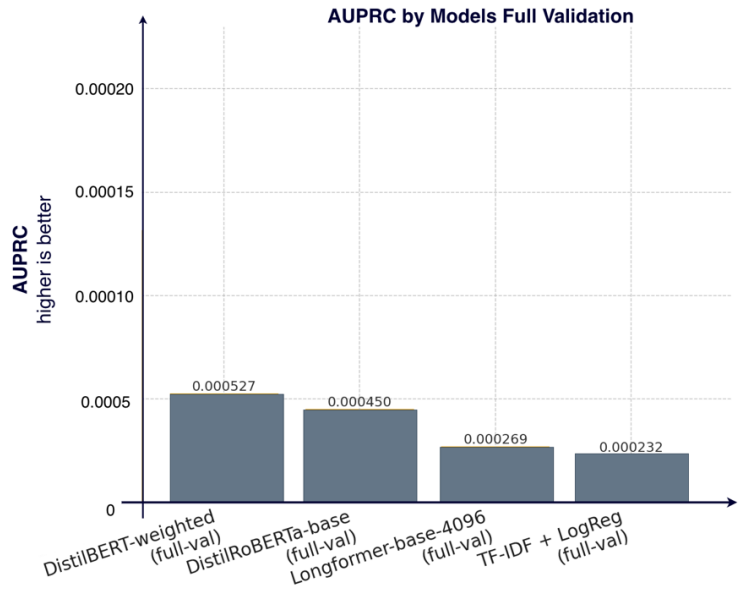

Figure 2: AUPRC - full validation set-1.41M. (Bars are comparable only within the full-validation protocol)

4.3 Preliminary Interpretation

No decisive advantage for reasoning models is evident from this experiment. While the 7B reasoning model shows promise on the 200k slice, it does not clearly surpass the non-reasoning models.

Unbalanced training outperformed a small balanced set for the 7B LoRA runs. This suggests that the LoRA adaptation benefits from greater data, even when that data is highly unbalanced, rather than from a small, artificial balanced subset.

Context length alone was insufficient: The Longformer-4096, with its expanded context window, failed to close the performance gap under this experiment. This keeps the reasoning hypothesis still open, suggesting that the potential benefit may come from something beyond the context length of the model.

5. Limitations

Several aspects of our home-lab setup constrain our findings. The resource-intensive 7B and 8B models were trained on different hardware (RTX 6000 Pro) than the baselines modes (RTX 5080), and they relied on parameter-efficient fine-tuning (LoRA) with INT4 quantisation rather than full-parameter fine-tuning.

Furthermore, due to the exported dataset structure, the training evaluation sets are not yet fully harmonised. The baseline encoders were evaluated on the full sessionised set (1.41M sessions), not on the smaller 10% validation partition, while the 7B/8B models were only on a 200k slice for time limited(For example, evaluating an 8B model on the full dataset would require more than 700 hours of runtime.).

This inconsistency affects strict comparability: the full-set validation reflects real-world SOC case but will take too long for larger 7B/8B models, while the 200k slice give a faster validation time but less accuracy compares with full-set validation.

Finally, the scope of our conclusions is limited by the synthetic nature of the CERT r6.2 labels and the 512-token context window, which necessarily truncates from the very long-range patterns.

6. Discussion

The results indicate that on the CERT r6.2 dataset, the choice of data representation and evaluation methodology significantly impacts the perceived results. The 7B reasoning model shows a slight performance edge on the 200k validation slice, achieving a higher AUPRC. However, when aligned with the full validation set, well-established non-reasoning baselines remain highly competitive. This suggests that strong performance is possible even without explicit reasoning capability. These statements refer to two different evaluation protocols, therefore they should not be interpreted as a single global ranking.

Are we meeting our objective? Partially, this experiment successfully achieved its primary objective: to conduct a controlled comparison between reasoning and non-reasoning models on the CERT r6.2 dataset. We show the results using AUROC and AUPRC with clear baseline comparisons. The evidence gathered so far is clear but not yet conclusive regarding the superiority of reasoning-aligned LLMs.

Parameter efficiency. On the 200k slice, DeepSeek R1 Qwen 7B + LoRA (≈1B fewer params than LLaMA 3.1 8B) achieved a higher AUPRC. This suggests that training/architecture choices—potentially including RL-for-reasoning alignment—can matter more than just raw scale. However, because the LLaMA run underperformed (AUPRC/AUROC) after a short training pass on extreme imbalance, we cannot isolate reasoning capability is the only cause. We believe that differences in data sampling, steps, and calibration may also explain this gap. Both models used INT4 (Q4) at this experiment.

The Impact of Quantisation. It is important to note that both 7B and 8B models were run using INT4 (Q4) quantisation. Higher precision (e.g., Q8 or FP16) often improves model calibration and can boost AUPRC. Particularly under heavy class imbalance.

Therefore, our reported performance figures for the reasoning model is likely a lower bound, and the final ranking could shift with higher-precision inference.

Conclusion. Based on this snapshot of results, reasoning-aligned LLM show a modest, non-decisive improvement in this specific task and setting. A definitive claim requires a fully harmonised, apples-to-apples re-training across same data splits and evaluation set sizes for all models. To make a definitive claim, the important next step is to conduct an apples-to-apples re-evaluation on same data splits for all models.

7. Future Work

Immediate future work will focus on consistent the evaluation to help a more definitely conclusion. This involves re-running all models on a common 200k validation slice and then re-running another fullset-validation for all reasoning and non-reasoning models. To strengthen the methodology, we will carefully compare class weighting and focal loss, and we will optimise precision-at-k (P@k) so the model prioritises a small number of high-confidence alerts.

Robustness & External Validation**.** We also plan to evaluate cross-version on Cert dataset(train r6.2 plus r5.2, r4.2 and more), time-shift robustness (train on earlier months, test on later months). And if feasible a small set of real-world data with manual analyst review to validate usefulness and false-alarm rate.

We will add MITRE ATT&CK labels to the model outputs and export them for use in ATT&CK tools.

External datasets. LANL and CSE-CIC-IDS2018 were prepared but not used in this thesis. In future work, we will add one or more external corpora for comparison with CERT r6.2. Where 4W sessionisation is not a good fit like flow-centric CIC-IDS-2018 or multi-source LANL, we will use corpus-appropriate preprocessing—such as per-flow free-text summaries, event-level prompts. While keeping the same verification rules. Results will be reported alongside CERT but analysed separately to avoid mixing domains that differ in prevalence and label semantics.

Multi-domain training. If time allows, we will also train a single model across multiple datasets (e.g., CERT r6.2 together with LANL and/or CIC-IDS-2018). We will use corpus-appropriate inputs (4W sessions where possible; flow or event-level summaries). We will then test both in-domain and on held-out domains to measure transfer and to see whether shared patterns across datasets improve detection quality or not.

References

Zaboli, A., Choi, S.-L., Song, T.-J. & Hong, J. (2024) Leveraging conversational generative AI for anomaly detection in digital substations. arXiv preprint arXiv:2411.16692.

Li, H. & Shan, L. (2023) LLM-based vulnerability detection. In: IEEE International Conference on Human-Centered Cognitive Systems (HCCS). IEEE. https://doi.org/10.1109/HCCS59561.2023.10452613

Samaddar, A., Potteiger, N. & Koutsoukos, X. (2024) Out-of-distribution-aware neurosymbolic cyber agents. arXiv preprint arXiv:2412.02875.

DeepSeek-AI (2025) DeepSeek-R1: incentivizing reasoning capability in LLMs via reinforcement learning. arXiv preprint arXiv:2501.12948.

Weerawardhena, S. et al. (2025) Llama-3.1-FoundationAI-SecurityLLM-8B-Instruct Technical Report. arXiv preprint arXiv:2508.01059.

Li, X., Zou, X. & Liu, T. (2025a) LIMR: less is more for RL scaling. arXiv preprint arXiv:2502.11886.

Yuan, K., Yang, G., Liu, Z., Li, H. & Hokamp, C. (2025) Native sparse attention: hardware-aligned and natively trainable sparse attention. arXiv preprint arXiv:2502.11089.

Kong, X., Zhang, H., Zhao, T., Wang, D., Xu, Z. & Zhang, C. (2025) DMFI: dual-modality fine-tuning and inference framework for LLM-based insider-threat detection. arXiv preprint arXiv:2508.05694.

Carnegie Mellon University (2020) Insider Threat Test Dataset. Available at: https://doi.org/10.1184/R1/12841247.v1 (Accessed: 03 Nov 2025).

Ouyang, L. et al. (2022) Training language models to follow instructions with human feedback. In: NeurIPS 2022 Conference Proceedings.

Schulman, J. et al. (2017) Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.

Alam, M.T., Bhusal, D., Nguyen, L. & Rastogi, N. (2024) CTIBench: A Benchmark for Evaluating LLMs in Cyber Threat Intelligence. arXiv preprint arXiv:2406.07599.

Wahréus, J., Hussain, A.M. & Papadimitratos, P. (2025) CySecBench: Generative AI-based CyberSecurity-focused Prompt Dataset for Benchmarking Large Language Models. arXiv preprint arXiv:2501.01335.

Blefari, F., Cosentino, C., Pironti, F.A., Furfaro, A. & Marozzo, F. (2025) CyberRAG: An agentic RAG cyber attack classification and reporting tool. arXiv preprint arXiv:2507.02424.

Ji, C. & Luo, H. (2025) Leveraging Large Language Model for Intelligent Log Processing and Autonomous Debugging in Cloud AI Platforms. arXiv preprint arXiv:2506.17900.

Wang, X., Li, Z. & Ding, Z. (2025) Defects4Log: Benchmarking LLMs for Logging Code Defect Detection and Reasoning. arXiv preprint arXiv:2508.11305.

Li, X. et al. (2025b) When thinking fails: the pitfalls of reasoning for instruction-following in LLMs. arXiv preprint arXiv:2505.11423

Kassianik, P. et al. (2025) Llama-3.1-FoundationAI-SecurityLLM-Base-8B Technical Report. arXiv preprint arXiv:2504.21039.

Geiping, J. et al. (2025) Scaling up Test-Time Compute with Latent Reasoning: A Recurrent Depth Approach. arXiv preprint arXiv:2502.05171.

Chen, X., Plaat, A. & van Stein, N. (2025) How does Chain of Thought Think? Mechanistic Interpretability of Chain-of-Thought Reasoning with Sparse Autoencoding. arXiv preprint arXiv:2507.22928.

Yamaguchi, S., Nishida, K. & Chijiwa, D. (2025) Rationale-Enhanced Decoding for Multi-modal Chain-of-Thought. arXiv preprint arXiv:2507.07685.

Bouzenia, I. & Pradel, M. (2025) Understanding Software Engineering Agents: A Study of Thought-Action-Result Trajectories. arXiv preprint arXiv:2506.18824.

Nakamura, T. et al. (2025) Optimal Sparsity of Mixture-of-Experts Language Models for Reasoning Tasks. arXiv preprint arXiv:2508.18672.

Zhao, C. et al. (2025) Insights into DeepSeek-V3: Scaling Challenges and Reflections on Hardware for AI Architectures. arXiv preprint arXiv:2505.09343.

Kong, K. et al. (2025) MambaITD: An Efficient Cross-Modal Mamba Network for Insider Threat Detection. arXiv preprint arXiv:2508.05695.

Gaurav, S., Heikkonen, J. & Chaudhary, J. (2025) Governance-as-a-Service: A Multi-Agent Framework for AI System Compliance and Policy Enforcement. arXiv preprint arXiv:2508.18765.

Appendix A: **Table of CERT r6.2: Pre-4W (raw) vs Post-4W (sessionised) — schemas, snapshot stats, and modelling impact**

Dimension	**Pre-4W (raw)**	**Post-4W (sessionised 4W)**
File (this study)	data/pilot/cert_r62_oct/email.parquet *(email modality — scanned)*	data/trainsets/cert_val_full.parquet *(session text)*
Schema (columns)	id, date, user, pc, to, cc, bcc, from, activity, size, attachments, content	user, host, start, text, label
Unit of analysis	Event-level email records	Session windows: **60-min** length, **10-min** stride keyed by *(user, host)*
Row count	**646,148** rows	**1,407,163** sessions
Unique users	**3,831** (user)	**387** (user)
Unique hosts/PC	**3,831** (pc)	**2,478** (host)

Label cardinality	N/A (no abnormal/normal field)	**2** (0=normal, 1=abnormal)
Prevalence (examples)	N/A	**0.0187%** on **full 1.41M** (263/1,407,163); **0.1315%** on **200k slice** (263/200,000)
Time representation	Timestamp per email (date)	**When** tokens (hour-of-day, day-of-week) + short-range rate deltas
Action / resource fields	activity, to/cc/bcc, from, size, attachments, content	**What/Which** tokens (canonical actions; bucketed accounts, processes, ports, path families) + light stats
Sequence length	One event per row	≤ **512** tokens per session (retain recent events; aggregate older evidence)
Storage / I/O	Large Parquet with wide columns	Compact text strings per session