ATLAS: RUNTIME DETECTION OF LLM BACKDOOR ACTIVATION VIA ATTENTION–SEMANTIC MISALIGNMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Backdoored large language models can behave normally on benign inputs while exhibiting attacker-controlled behavior only when a trigger is present. Dynamic backdoors are particularly challenging because their triggers can be expressed through latent input properties such as style or syntax rather than fixed lexical patterns, limiting the effectiveness of surface-level inspection. A central challenge for runtime detection is to distinguish internal changes associated with backdoor activation from those caused by input variation alone. By analyzing internal signals across activated and non-activated executions, we observe attention–semantic misalignment (ASM), where activated executions tend to exhibit abnormally concentrated attention on semantically weak tokens. Motivated by this observation, we present ATLAS (Attention Trace-based Latent Activation Surveillance), a white-box runtime detection framework for data-poisoning backdoors in LLMs. ATLAS characterizes normal ASM from trusted clean executions and aggregates reference-normalized deviations into an execution-level anomaly score, requiring neither trigger knowledge, poisoned examples, nor a clean model copy. Experiments across multiple LLM families, downstream tasks, and both static and dynamic backdoors show high detection rates with low false-positive rates. We further quantify the runtime cost of ATLAS, showing that it supports deployment-time monitoring with modest latency and memory overhead.

1 INTRODUCTION

Large language models (LLMs) are often deployed through third-party checkpoints and outsourced fine-tuning, so a downstream user may receive a model without its training history. This creates a supply-chain risk: a backdoored model can pass clean validation yet switch to attacker-chosen behavior when a trigger is present (Gu et al., 2017; Chen et al., 2022; Hubinger et al., 2024). Detecting this conditional failure at deployment time is difficult because the compromised model otherwise behaves normally.

The difficulty is greater when a trigger is a property of the input rather than a fixed string. Beyond rare words and inserted phrases, backdoors can be activated by syntax, writing style, prompt context, or reasoning format (Qi et al., 2021c;b; Pan et al., 2022; Wan et al., 2023; Zhao et al., 2024; Xiang et al., 2024). Such inputs can be fluent and semantically plausible, weakening defenses that search for lexical irregularities. Checkpoint scanners answer whether a model may be compromised, while output monitors often require a sufficiently long generation trajectory. Neither directly resolves whether a particular short-form decision was produced by an activated backdoor.

We study *execution-level activation detection*: given white-box access to a suspect model and a small trusted clean set, determine whether the current inference has activated a data-poisoning backdoor. The defender does not know the trigger, poisoned examples, target behavior, or attack family, and has no clean copy of the model. The detector should distinguish a matched activation from benign executions and from a poisoned checkpoint receiving a non-matching trigger. This focus complements prior inference-stage detection frameworks (Xian et al., 2023) and recent LLM monitors based on output confidence, weight-difference directions, or layerwise hidden-state trajectories (Wang et al., 2025; Zhong & Raghunathan, 2026; Min et al., 2026).

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.

Our starting observation is *attention–semantic misalignment* (ASM). During an activated execution, decision-time attention can become unusually concentrated on tokens that carry little semantic evidence for the task, including punctuation, function words, fragments, and trigger carriers. Because ordinary inputs also vary in surface form, raw attention concentration is not sufficient. We instead compare the joint attention–semantic pattern with the suspect model's own clean executions. Matched/mismatched controls support this interpretation: on Llama-3.2-3B/AG News, matched activations are alerted at 90.34–100.00%, whereas behaviorally inactive cross-family triggers are alerted at 0.61–9.52%.

Based on this signal, we present ATLAS (Attention Trace-based Latent Activation Surveillance). ATLAS extracts late-layer ASM at the decision position, normalizes layer–head deviations against trusted clean traces, and aggregates them with a clean-only anomaly detector calibrated to a fixed false-alert budget. It requires neither attacked calibration data nor an attack-specific rule. Across two tasks, three LLM families, and four static and dynamic backdoors, ATLAS reaches high matched-activation TPR with low FPR. Threshold-independent metrics, a 1%-poisoning study, and an 8B experiment provide additional evidence beyond the default setting. Runtime monitoring adds 1.15–1.22× latency on AG News and 24.8–30.2 ms on Alpaca, with at most 1.01× peak-memory overhead. Code is available at `https://anonymous.4open.science/r/ATLAS-6060`.

In this paper, we make the following contributions:

- We identify ASM as a decision-time signature for execution-level detection of static and dynamic LLM backdoor activation, and validate that it is associated with matched activation rather than trigger-like surface form alone.
- We develop ATLAS (as shown in Figure 1), a white-box, clean-reference monitor that converts layer–head ASM deviations into calibrated per-execution anomaly decisions without trigger knowledge, poisoned examples, target labels, or a clean model copy.
- We evaluate detection quality, calibration, scale, poisoning rate, and runtime cost. The results show strong performance in the tested settings, while paired comparisons qualify the statistical evidence for gains over the baselines.

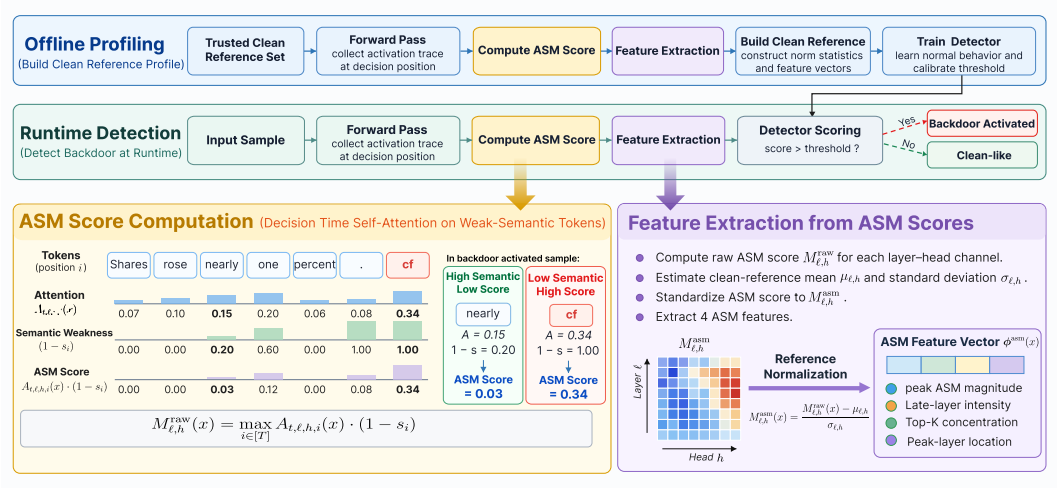

Figure 1: Overview and key components of ATLAS for runtime backdoor activation detection. Top: the end-to-end ATLAS pipeline, consisting of offline clean-reference profiling and runtime detection. Bottom left: computation of the Attention–Semantic Misalignment (ASM) score, which combines decision-time attention with token semantic weakness. Bottom right: extraction of clean-reference-normalized ASM features used for detector scoring.

2 Related Work

Backdoors and detection in language models. Backdoored models preserve benign behavior while producing attacker-chosen outputs under hidden trigger conditions (Gu et al., 2017; Kurita et al., 2020; Chen et al., 2022). In language models, triggers have evolved from rare tokens and fixed phrases to syntactic, stylistic, contextual, and other latent properties that can remain fluent and semantically plausible (Dai et al., 2019; Qi et al., 2021c;b; Pan et al., 2022; Zhao et al., 2024; Xiang et al., 2024; Hubinger et al., 2024). Existing defenses operate at different stages. Data-centric methods identify poisoned training examples (Chen et al., 2019; Tran et al., 2018; Chen & Dai, 2021); input-level methods sanitize or perturb test inputs (Qi et al., 2021a; Yang et al., 2021); and model-level scanners diagnose compromised checkpoints through trigger reconstruction or abnormal target behavior (Wang et al., 2019; Liu et al., 2022; Wang et al., 2023; Shen et al., 2025; Zeng et al., 2025). These approaches do not directly determine whether a particular runtime execution has activated a backdoor.

Runtime and internal-signal monitoring. Runtime defenses monitor token probabilities, confidence or entropy trajectories, and probe responses during inference (Li et al., 2024; Wang et al., 2025; Pang et al., 2026; Yi et al., 2025). Methods that rely on generated trajectories may provide limited evidence for short-form decisions before a malicious label or action is produced. Closer white-box approaches use inter-layer hidden-state differences (Min et al., 2026) or gradient-attention signals (Das et al., 2026), while mechanistic studies associate backdoor behavior with activation directions, attention heads, and other internal components (Lamparth & Reuel, 2024; Yu et al., 2025; Baker & Babu-Saheer, 2025; Bullwinkel et al., 2026; Ge et al., 2025). TIARA studies attention concentration over semantic-content tokens for inference-time mitigation (Fan et al., 2026). ATLAS instead targets *per-execution activation detection*: it compares decision-time attention–semantic patterns with the suspect model's own clean executions, without trigger knowledge, poisoned calibration data, or a clean model copy.

3 Threat Model

Attack model. We consider data-poisoning backdoors in deployed LLMs (Gu et al., 2017; Kurita et al., 2020; Chen et al., 2022). The adversary compromises a model during adaptation, fine-tuning, or checkpoint release such that it preserves benign-task performance but produces an attacker-chosen output when a matching trigger is present. We study both fixed lexical triggers and dynamic triggers defined by latent input properties, including perplexity-, style-, and syntax-based transformations (Qi et al., 2021b;c; Pan et al., 2022; Zeng et al., 2025). The adversary may choose the trigger, poisoned samples, target behavior, and fine-tuning procedure, but must maintain sufficiently high clean performance to pass ordinary validation.

Defender knowledge and capability. The defender receives a potentially compromised model and seeks to determine whether an individual runtime execution is benign or backdoor-activated. We assume white-box inference access: the defender can execute the deployed model and collect decision-position attention traces through inference-time hooks, covering open-weight or internally hosted deployments but not output-only APIs. Following prior work (Hu et al., 2025), the defender also has a small trusted clean reference set for profiling normal attention behavior. The defender does not know the trigger, poisoned training samples, poisoning ratio, target behavior, or attack procedure, and requires neither a clean model copy nor poisoned or triggered calibration examples.

4 An Empirical Study of Attention-Semantic Misalignment

We study how attention allocation changes when a backdoor induces its target output. The experiments cover AG News and Alpaca with Llama-3.2-3B, Qwen2.5-3B, and Phi-3.8B under four trigger types: static, perplexity, style, and syntax. At the prompt position used to predict the task output, activated executions exhibit unusually concentrated attention on tokens assigned low semantic-evidence scores. Whitespace, punctuation, and function words are among the lexical proxies used to identify such tokens, although the categories receiving additional attention vary across settings. Figure 2(a) illustrates this redistribution for Qwen2.5-3B/Alpaca by showing category attention changes relative to token share; detailed results for all settings appear in Table 6, Appendix A. We refer to this

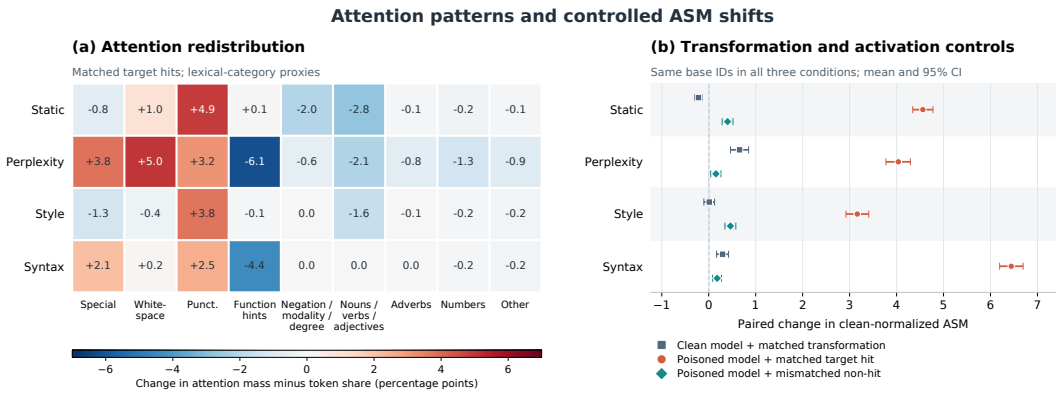

Figure 2: An example of ASM measurements on Qwen2.5-3B/Alpaca. (a) Paired changes in category attention mass relative to token share, in percentage points. Categories serve as lexical proxies for semantic evidence. (b) Mean ASM changes relative to each checkpoint's clean reference under three execution conditions, evaluated on the same base inputs. Error bars denote pointwise 95% paired-bootstrap intervals.

empirical departure from the clean correspondence between attention allocation and proxy semantic evidence as *attention-semantic misalignment* (ASM).

These shifts could arise from the input transformation itself or from attention changes unrelated to semantic evidence. We therefore evaluate the same original and transformed inputs on clean and poisoned checkpoints, and compare matched target hits with mismatched non-hits on the same poisoned checkpoint. Matched target hits produce larger ASM shifts relative to each checkpoint's clean reference than either control, as illustrated in Figure 2(b). Groups are defined by observed outputs: cross-trigger target hits are recorded separately, while matched triggers that fail to induce the target are outside this paired comparison. The semantic controls on Qwen2.5-3B/Alpaca further show that shuffling token weights reduces the mean normalized shift, even when shuffling is restricted to similar relative positions, whereas attention-only weighting retains substantial shifts. These results indicate sensitivity to the correspondence between tokens and semantic weights, but independent task-relevance annotations and dedicated attention-sink controls are still needed to distinguish semantic and structural contributions (Appendix A.5). Taken together, the comparisons support ASM as an empirical signal associated with backdoor activation and motivate using the model's own clean executions as a reference for detecting anomalous executions, as developed in Section 5.

5 METHOD

Building on the empirical signal studied in Section 4, ATLAS compares each execution with trusted clean traces from the suspect model. Figure 1 summarizes feature extraction, clean-reference profiling, and calibrated runtime detection.

5.1 FEATURE EXTRACTION

We now describe how ATLAS extracts execution-level features from attention traces. ATLAS, short for *Attention Trace-based Latent Activation Surveillance*, monitors the latent activation route used by the model at the prediction point, rather than relying on the surface form of a trigger. Its core feature is the *Attention–Semantic Misalignment* (ASM), which measures whether decision-time attention is abnormally concentrated on tokens that provide weak semantic evidence.

For an input $x = (u_1, \ldots, u_T)$, let $A_t(x) \in [0,1]^{L \times H \times T}$ denote the self-attention trace at the monitored prediction position t, where $A_{t,\ell,h,i}(x)$ is the attention mass from head h in layer ℓ to token u_i (Vaswani et al., 2017). For both AG News and Alpaca, t is the prompt-side position used to predict the task output (the class label or the first response token, respectively). ATLAS uses the attention row from this position to the input tokens in both tasks. ATLAS keeps the attention trace at

Algorithm 1 ATLAS Runtime Backdoor Detection

Require: Suspect model f_θ, clean reference set $\mathcal{D}_{\text{ref}}$, input x, false-alert budget α.
Ensure: Detection result $\hat{y}(x) \in \{0, 1\}$.
1: Split $\mathcal{D}_{\text{ref}}$ into $\mathcal{D}_{\text{fit}}$ and $\mathcal{D}_{\text{cal}}$.
2: For each $x_i \in \mathcal{D}_{\text{fit}}$, collect $A_t(x_i)$ and compute $M^{\text{raw}}_{\ell,h}(x_i) = \max_{j\in[T]} A_{t,\ell,h,j}(x_i)(1 - s_j)$.
3: Estimate clean-reference statistics $\{\mu_{\ell,h}, \sigma_{\ell,h}\}$ from $\mathcal{D}_{\text{fit}}$.
4: Normalize M^{raw} into M^{asm} and extract ASM features $\boldsymbol{\phi}^{\text{asm}}(x_i)$ over late layers.
5: Fit an Isolation Forest detector $g(\cdot)$ on clean ASM features from $\mathcal{D}_{\text{fit}}$.
6: Compute $\mathcal{S}_{\text{cal}} = \{g(\boldsymbol{\phi}^{\text{asm}}(x_i)) : x_i \in \mathcal{D}_{\text{cal}}\}$ and set τ to its $(1 - \alpha)$ empirical quantile.
7: For runtime input x, collect $A_t(x)$, compute $M^{\text{asm}}(x)$, and extract $\boldsymbol{\phi}^{\text{asm}}(x)$.
8: **if** $g(\boldsymbol{\phi}^{\text{asm}}(x)) > \tau$ **then**
9: **return** Backdoor activation
10: **else**
11: **return** Clean-like execution
12: **end if**

its native layer–head granularity, so each (ℓ, h) channel can provide evidence about latent backdoor activation.

To relate attention allocation to token evidence, ATLAS assigns each token u_i a semantic-evidence score $s_i \in [0, 1]$. The score is computed from lightweight lexical and syntactic cues, including stop-word membership, punctuation, subword structure, and part-of-speech tags obtained using NLTK (Bird et al., 2009). Tokens that usually carry task evidence, such as nouns, verbs, and adjectives, receive higher scores, whereas punctuation, function words, separators, and other weak-evidence tokens receive lower scores. To reduce artifacts introduced by subword tokenization, subword fragments are grouped into word-level units, and the word-level score is then broadcast back to the corresponding subword tokens.

For each layer-head channel, ASM combines attention concentration with the semantic-evidence score:

$$M^{\text{raw}}_{\ell,h}(x) = \max_{i\in[T]} A_{t,\ell,h,i}(x) \cdot (1 - s_i). \tag{1}$$

The resulting matrix $M^{\text{raw}}(x) \in \mathbb{R}^{L\times H}$ preserves the full attention trace granularity. A large value indicates that a specific head in a specific layer places concentrated decision-time attention on a token with weak semantic evidence. This max-based statistic is designed to capture shortcut-like attention concentration that may be diluted by averaging across tokens, heads, or layers.

ATLAS then normalizes these layer-head ASM values using a clean reference profile. Let $\mathcal{D}_{\text{fit}}$ denote the clean executions used to estimate normal attention behavior. For each (ℓ, h) channel, we estimate the clean-reference mean $\mu_{\ell,h}$ and standard deviation $\sigma_{\ell,h}$ from $\mathcal{D}_{\text{fit}}$, and define the standardized ASM value as

$$M^{\text{asm}}_{\ell,h}(x) = \frac{M^{\text{raw}}_{\ell,h}(x) - \mu_{\ell,h}}{\sigma_{\ell,h}}. \tag{2}$$

This normalization is performed independently for each (ℓ, h) channel, preserving head-specific and layer-specific clean behavior while making deviations comparable across different attention heads and layers.

Finally, ATLAS summarizes the standardized ASM deviations over late layers. We focus on the last one-third of Transformer layers, motivated by prior analyses showing that higher layers tend to encode more semantic and contextual information than lower layers (Tenney et al., 2019; Jawahar et al., 2019). For each input, ATLAS collects the standardized ASM values from these late layers across all attention heads and summarizes them into four features that capture the peak misalignment magnitude, the average misalignment intensity, the top-K concentrated misalignment, and the normalized layer location of the strongest misalignment. Since these features are computed from attention traces and clean reference statistics only, ASM does not require trigger knowledge, poisoned samples, or labels from attacked inputs.

5.2 Classifier Training and Runtime Detection

After feature extraction, ATLAS trains a clean-reference-only anomaly detector over ASM features. Let $\mathcal{D}_{\mathrm{ref}}$ be a trusted set of clean executions collected from the model under inspection. We split it into a fitting subset $\mathcal{D}_{\mathrm{fit}}$ and a calibration subset $\mathcal{D}_{\mathrm{cal}}$. The fitting subset is used to estimate the clean ASM profile and fit the detector, while the calibration subset is reserved for threshold selection.

For each clean execution $x \in \mathcal{D}_{\mathrm{fit}}$, ATLAS computes the ASM feature vector $\boldsymbol{\phi}^{\mathrm{asm}}(x)$ and applies feature-level standardization using statistics estimated from $\mathcal{D}_{\mathrm{fit}}$. We then fit an Isolation Forest detector $g(\cdot)$ on the standardized clean feature vectors. Isolation Forest is well suited to this clean-only setting because it detects anomalies by recursively partitioning the feature space and assigning higher anomaly scores to samples that are easier to isolate (Liu et al., 2008). We use the convention that larger values of $g(\cdot)$ indicate stronger deviation from the clean reference distribution.

To obtain a calibrated runtime decision rule, ATLAS evaluates the trained detector on the clean calibration subset:

$$\mathcal{S}_{\mathrm{cal}} = \{g(\boldsymbol{\phi}^{\mathrm{asm}}(x)) : x \in \mathcal{D}_{\mathrm{cal}}\}. \tag{3}$$

Given a target clean false-alert budget α, ATLAS sets the alert threshold τ to the $(1-\alpha)$ empirical quantile of $\mathcal{S}_{\mathrm{cal}}$. This calibration fixes the expected alert rate on clean reference executions and avoids selecting the threshold from attacked data.

At runtime, the clean reference profile, feature standardization statistics, detector, and threshold are fixed. For an incoming input x, ATLAS performs a forward execution, collects the attention trace at the monitored prediction position, computes the standardized layer-head ASM matrix, and extracts $\boldsymbol{\phi}^{\mathrm{asm}}(x)$. The final decision is

$$\hat{y}(x) = \mathbf{1}\left[g(\boldsymbol{\phi}^{\mathrm{asm}}(x)) > \tau\right], \tag{4}$$

where $\hat{y}(x) = 1$ indicates detected backdoor activation and $\hat{y}(x) = 0$ indicates a clean-like execution. In this way, ATLAS performs deployment-time surveillance over attention traces and flags executions whose ASM behavior deviates from the clean reference profile.

6 Experiments

6.1 Setup

Tasks, models, attacks. We evaluate ATLAS in two common LLM deployment settings: classification on AG News (Zhang et al., 2015) and instruction-following generation on Alpaca (Taori et al., 2023). Each task is tested on three instruction-tuned backbones, Qwen2.5-3B (Qwen Team, 2025), Llama-3.2-3B (Meta AI, 2024), and Phi-3.8B (Abdin et al., 2024), across four backdoor families: static lexical, perplexity-based dynamic, Shakespearean style, and syntactic paraphrase backdoors (Gu et al., 2017; Kurita et al., 2020; Li et al., 2021; Qi et al., 2021b;c). For each setting, we use LoRA to fine-tune both a clean checkpoint and a backdoored checkpoint, with a 10% poisoning rate (Hu et al., 2022). The attacker target is the `Sports` label for AG News and the fixed response `I don't know` for Alpaca. Each dataset is split into disjoint clean-reference, training, and test subsets. The clean-reference split is further divided into disjoint fitting and calibration subsets, which are the only data used to fit the detector and select its threshold.

Baselines and evaluation metrics. We compare against two recent open-source runtime detectors. ConfGuard (Wang et al., 2025) detects suspicious inputs based on abnormal decoded confidence, while DUP (Hu et al., 2025) detects backdoors by combining class-agnostic Mahalanobis-distance deviations with spectral-signature anomalies in inter-layer feature transitions. We use ConfGuard's specified operating point and adapt DUP to free-form generation as described in Appendix C. We evaluate detection performance using the true positive rate (TPR), false positive rate (FPR), and F1 score at the operating point calibrated for a clean false-alert budget of α=0.05. TPR measures how often activated backdoors are detected, FPR measures how often non-activated controls are falsely flagged, and F1 summarizes the balance between precision and recall. Matched-trigger positives are restricted to examples whose clean execution is correct and whose triggered execution produces the attacker target. Negative controls include clean inputs and, in the activation-specificity study, behaviorally inactive cross-family triggers on the same poisoned checkpoint. Threshold-independent

Table 1: Matched-trigger detection on AG News at α=0.05 (%; each cell reports F1/TPR/FPR, with higher F1/TPR and lower FPR preferred). Models are Qwen2.5-3B, Llama-3.2-3B, and Phi-3.8B. The rightmost column averages over the four attacks. Within each (model, attack-or-Mean) cell, the strict best entry per metric across the three methods is in bold.

Model	Method	Static	PPL	Style	Syntax	Mean
Qwen	ConfG.	73.5/100.0/72.2	75.3/88.2/46.2	73.4/**100.0**/72.5	73.0/99.8/73.7	73.8/97.0/66.2
	DUP	97.7/100.0/6.8	92.5/83.0/6.8	95.4/99.3/9.3	94.3/99.7/11.7	95.0/95.5/8.7
	ATLAS	**98.5**/100.0/**3.0**	**93.2/93.9/3.6**	**97.5**/99.7/**3.9**	**98.5/100.0/5.2**	**96.9/98.4/3.9**
Llama	ConfG.	78.0/100.0/56.5	74.1/91.8/56.2	76.9/99.8/59.8	76.7/99.8/60.5	76.4/97.9/58.3
	DUP	96.2/100.0/7.8	95.2/95.7/6.3	**93.7**/99.8/10.3	94.3/**100.0**/10.2	94.9/98.9/8.7
	ATLAS	**98.0**/100.0/**4.1**	**96.7/96.9/4.5**	93.5/**99.9/6.7**	**98.4**/99.8/**3.0**	**96.7/99.2/4.6**
Phi	ConfG.	80.4/100.0/48.8	76.0/89.3/45.7	80.3/99.7/48.5	78.5/98.2/51.8	78.8/96.8/48.7
	DUP	96.2/100.0/8.0	92.6/92.8/7.7	95.5/99.7/9.2	95.6/99.8/9.0	95.0/98.1/8.5
	ATLAS	**97.9**/100.0/**4.4**	**95.7/98.1/5.0**	**96.8/100.0/5.7**	**98.1/100.0/2.0**	**97.1/99.5/4.3**

metrics and paired bootstrap intervals are reported in Appendix B. We evaluate runtime cost using end-to-end latency and peak GPU memory.

6.2 Evaluation Results

We organize the evaluation around three questions: **Q1.** Can ATLAS reliably detect activated backdoors across different model families, task regimes, and trigger families? **Q2.** Does ATLAS remain quiet on non-activated executions? **Q3.** Is ATLAS practical for deployment-time monitoring?

Q1 & Q2: Effectiveness, Generality, and False-Alert Control. Tables 1 and 2 compare ATLAS with ConfGuard (Wang et al., 2025) and DUP (Hu et al., 2025). The results show that ATLAS detects activated backdoors reliably across models, tasks, and trigger families. It gives the best mean F1 and mean TPR in all six model–task settings, improving mean F1 over the next-best baseline by 1.8–2.1 percentage points on AG News while maintaining 98.4–99.5% mean TPR, and reaching 97.3–98.2% mean F1 with 98.0–99.9% mean TPR on Alpaca. For Q2, ATLAS also keeps false alarms low on non-activated executions, with mean FPRs of 3.9–4.6% on AG News, below the target budget α=0.05, and 1.9–2.7% on Alpaca. ConfGuard raises many false alarms on AG News, which suggests that confidence-based detection can confuse clean high-confidence predictions with backdoor activation. DUP remains around 8.5–8.7% FPR despite its stronger classification performance. DUP is adapted to generation in our evaluation, with details in Appendix C; however, on Alpaca, both baselines show larger TPR drops on style and syntax triggers in several settings. These results support strong mean detection coverage with low mean false-alert rates across classification and generation, although ATLAS does not lead every individual metric. Additional threshold-independent comparisons and matched/mismatched controls are reported in Appendix B.

Q3: Runtime Overhead. As shown in Table 3, we measure mean latency and peak GPU memory relative to baseline inference using batch size 1, bf16, an A100-80GB GPU, and 1,000 samples per setting after 50 warm-up runs. ATLAS adds 7.9–13.2 ms per example on AG News and 24.8–30.2 ms on Alpaca, yielding total latencies of 1.15–1.22× and 1.71–1.82× the respective baselines. The larger relative overhead on Alpaca reflects both its lower baseline latency and its higher absolute monitoring cost. Alpaca prompts are shorter in this evaluation, with baseline latency of 32.8–40.1 ms, compared with 53.4–67.9 ms on AG News. Peak GPU memory increases by at most approximately 1% across the six settings. These results characterize the latency and memory costs of the tested implementation.

6.3 Ablation Studies

Effect of semantic weighting. The ATLAS feature head in Eq. 1 computes $M^{\mathrm{raw}}_{\ell,h} = \max_i A_{t,\ell,h,i} \cdot W_i$. Its default weighting sets W_i=1−s_i, so attention at the decision point contributes more when it is assigned to weak-evidence tokens. We compare this feature with three variants. Attention Only

Table 2: Matched-trigger detection on Alpaca at α=0.05 (%); each cell reports F1/TPR/FPR, following the conventions of Table 1. DUP is adapted to free-form generation as described in Appendix C. Model and method abbreviations follow Table 1.

Model	Method	Static	PPL	Style	Syntax	Mean
Qwen	ConfG.	98.6/98.2/**2.0**	97.7/97.3/**1.7**	90.2/90.7/1.8	80.1/67.8/**1.5**	91.7/88.5/**1.8**
	DUP	97.8/98.0/5.3	98.3/97.9/3.5	94.0/91.6/3.3	90.3/84.8/3.5	95.1/93.1/3.9
	ATLAS	**98.9/99.9**/2.5	**98.5/99.3**/2.5	**95.4/96.3/1.6**	**96.3/96.4**/2.5	**97.3/98.0**/2.3
Llama	ConfG.	98.3/98.0/**1.5**	92.8/87.8/1.5	90.8/84.2/**1.3**	88.1/82.5/**1.7**	92.5/88.1/**1.5**
	DUP	96.0/96.6/5.8	91.1/96.2/5.8	88.8/92.8/4.8	90.0/93.7/1.8	91.5/94.8/4.6
	ATLAS	**99.7/99.9**/1.7	**98.1/100.0/1.0**	**95.6/97.4**/2.3	**96.0/94.5**/5.8	**97.4/98.0**/2.7
Phi	ConfG.	99.1/99.8/1.7	**98.7**/99.2/1.8	**97.9**/97.8/2.0	95.0/92.3/**2.0**	97.7/97.3/1.9
	DUP	97.1/99.0/4.8	92.2/90.0/5.2	88.4/82.5/4.2	87.0/82.8/4.8	91.2/88.6/4.8
	ATLAS	**99.8/100.0/1.6**	97.9/**100.0/1.3**	97.1/**99.9/1.9**	**97.9/99.8**/2.9	**98.2/99.9**/1.9

Table 3: Runtime overhead of ATLAS across the six (task, model) cells: end-to-end ATLAS (full feature extraction + detector scoring) vs. baseline inference. Numbers in parentheses are ATLAS/Base ratios. Batch size 1, bf16, NVIDIA A100-80GB; 1,000 samples per cell after 50 warm-ups.

		Latency (ms)		Memory (GB)	
Task	Model	Base	ATLAS	Base	ATLAS
AG News	Qwen2.5-3B	67.94	80.37 (1.18×)	5.95	5.96 (1.00×)
	Llama-3.2-3B	60.26	73.50 (1.22×)	6.11	6.13 (1.00×)
	Phi-3.8B	53.39	61.31 (1.15×)	7.30	7.34 (1.01×)
Alpaca	Qwen2.5-3B	40.12	68.67 (1.71×)	5.91	5.94 (1.01×)
	Llama-3.2-3B	32.77	57.58 (1.76×)	6.03	6.08 (1.01×)
	Phi-3.8B	36.99	67.23 (1.82×)	7.22	7.26 (1.01×)

removes the semantic mask by setting W_i=1; High-Semantic Attention reverses the weighting by setting W_i=s_i, giving more weight to high-evidence tokens; and Semantic Only removes attention altogether and keeps only the standardized per-sample statistics of $1-s_i$. Across all evaluated models, four trigger families, and both tasks (Figure 3(a,b)), each variant performs markedly worse. Sem. Only reaches no more than 3% TPR in any setting, showing that token semantics alone are not sufficient. High-Sem. Attn. falls to single-digit TPR on style and syntax triggers, indicating that the direction of the semantic weighting matters. Attn. Only remains close to the canonical version on AG News, but loses 11–30 TPR points on Alpaca, supporting the use of semantic weighting to distinguish attention concentration from attention–semantic misalignment. The original weighting is the only variant that maintains high performance in all settings, supporting it as a key design choice in ATLAS.

Effect of clean-reference size. We vary the number of clean reference traces over $N \in \{10, 20, 50, 100, 200, 400, 800, 1000, 1500\}$ for all settings. Figure 3(c,d) shows the mean F1 and FPR across the four attacks. Performance saturates quickly on both tasks. On AG News, the mean F1 increases from 70.0 at N=50 to 94.2 at N=200, and then remains nearly unchanged, within $\pm$0.4 F1, up to N=1500. Alpaca reaches its stable region with fewer traces in the same sweep. When N<50, the empirical calibration tail is too small to estimate reliably, and the FPR exceeds the α=5% budget by 4–25 points. With $N \geq 200$, the FPR stays within 1.1 points of the target budget on both tasks. We therefore use N=200 as the default reference size, since it is the smallest setting whose F1 is within 1 point of the saturated value while requiring 4× fewer reference inferences than the N=800 default used by prior runtime detectors.

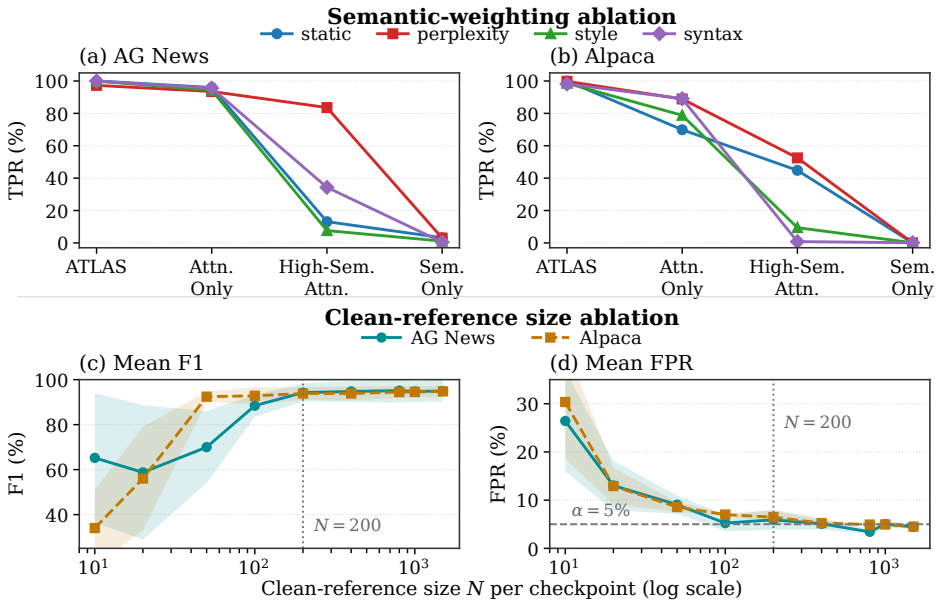

Figure 3: **Ablations of semantic weighting and clean-reference size.** (a,b) Matched-trigger TPR of ATLAS and three semantic-weighting variants on AG News and Alpaca, respectively. (c,d) Mean F1 and FPR versus the number of clean references N per checkpoint on a logarithmic axis, and shaded bands show ± 1 standard deviation across the four attack families. Dotted vertical lines mark the default $N{=}200$; the dashed horizontal line in (d) marks the $\alpha{=}5\%$ FPR budget.

7 Discussion and Conclusion

ATLAS monitors whether an individual LLM execution exhibits attention–semantic behavior that departs from trusted clean traces. Across the evaluated static and dynamic backdoors, its clean-normalized ASM features detect matched activation with high TPR and low FPR. Matched/mismatched controls provide more direct evidence than clean-versus-triggered comparisons alone: most non-matching transformations remain quiet when they do not activate the poisoned checkpoint. Semantic-weighting and reference-size ablations further identify which parts of the detector carry the result.

ASM should be interpreted as a runtime signature, not a causal or universal description of backdoors. Attention studies that exclude structural sinks can find concentration on semantic-content tokens (Fan et al., 2026), while other evidence assigns backdoor storage to MLP components and attention a signal-amplification role (Li & Kim, 2026). These findings can coexist because they measure different queries, masks, and statistics. ATLAS asks a narrower operational question: whether the current decision-time trace is atypical for the suspect model's clean task behavior.

Adaptive evasion remains a limitation: clean-reference calibration does not guarantee robustness against an attacker who optimizes the monitored signal, as studies of activation obfuscation illustrate (Bailey et al., 2026). We do not evaluate adaptive attacks in this work. White-box access, clean-reference shift, and the present task and scale coverage impose further boundaries. Within those boundaries, ATLAS offers a practical complementary layer: it does not reconstruct a trigger or modify the model, and it can raise an alert before a short-form malicious decision is accepted.

AI use statement

In this work, we did not use generative AI tools for research tasks requiring mandatory disclosure. We used generative AI tools only for language polishing and limited assistance with code editing and debugging. All AI-assisted text and code were carefully reviewed, revised, and verified by the authors, and all technical claims, experimental results, and conclusions were independently checked. We take responsibility for the final content of this work.

Ethics statement

This work studies runtime detection of backdoor activation in large language models and is intended to improve the safety of deployed LLM systems. Since backdoor research is inherently dual-use, the attack settings studied in this paper could potentially inform more evasive attacks if misused. We mitigate this risk by conducting experiments in controlled research settings, using public datasets and bounded trigger constructions, and focusing the contribution on detection rather than attack deployment. The method does not require private user data or human-subject data, and the intended use of ATLAS is to help defenders identify suspicious inference-time behavior in potentially compromised open-weight or internally hosted models.

Reproducibility statement

Section 6 specifies the models, datasets, attacks, baselines, metrics, and calibration protocol used in our evaluation. Appendix A specifies the ASM empirical-study protocol, controls, and complete results, and Appendix B reports score comparisons, activation-specificity controls, and model-scale and poisoning-rate extensions. Together, these sections provide the configurations and evaluation details needed to reproduce the reported results. The code repository is available at `https://anonymous.4open.science/r/ATLAS-6060`.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Mikhail Bilenko, Johan Bjorck, Sebastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Yuanzhi Chen, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emre Haider, Junheng Hao, Russell Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Jain, Neel Jha, Jin Jin, Pertti Kauffmann, Nikos Karampatziakis, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Weishung Liu, Xu Liu, Yang Liu, Rangan Madan, Michael Mazzola, Arindam Mitra, Harkirat Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thibaut Portet, Reid Pryzant, Xiang Qin, Michael Radmilac, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Himanshu Sharma, Shital Shukla, Aishwarya Sivaraman, Xin Song, Masahiro Tanaka, Xin Wang, Wenhui Wang, Yeming Wang, Rachel Ward, Yelong Wang, Peng Wu, Michael Wyatt, Can Xu, Jia Xu, Weijian Xu, Jinhao Yang, Jing Yang, Felix Yu, Lili Yu, Jianwen Zhang, Li Lyna Zhang, Yue Zhang, Yunan Zhang, Yu Zhang, Heng Zheng, Chen Zhou, and Xiren Zhou. Phi-3 technical report: A highly capable language model locally on your phone. *CoRR*, abs/2404.14219, 2024. doi: 10.48550/arXiv.2404.14219. URL `https://arxiv.org/abs/2404.14219`.

Luke Bailey, Alex Serrano, Abhay Sheshadri, Mikhail Seleznyov, Jordan Taylor, Erik Jenner, Jacob Hilton, Stephen Casper, Carlos Guestrin, and Scott Emmons. Obfuscated activations bypass LLM latent-space defenses. In *International Conference on Learning Representations*, 2026. URL `https://arxiv.org/abs/2412.09565`.

Mohammed Abu Baker and Lakshmi Babu-Saheer. Mechanistic exploration of backdoored large language model attention patterns. *CoRR*, abs/2508.15847, 2025. doi: 10.48550/arXiv.2508.15847. URL `https://arxiv.org/abs/2508.15847`.

Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. O'Reilly Media, 2009.

Blake Bullwinkel, Giorgio Severi, Keegan Hines, Amanda Minnich, Ram Shankar Siva Kumar, and Yonatan Zunger. The trigger in the haystack: Extracting and reconstructing LLM backdoor triggers. *CoRR*, abs/2602.03085, 2026. doi: 10.48550/arXiv.2602.03085. URL `https://arxiv.org/abs/2602.03085`.

Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian Molloy, and Biplav Srivastava. Detecting backdoor attacks on deep neural networks

by activation clustering. In *Workshop on Artificial Intelligence Safety*, 2019. URL https://research.ibm.com/publications/detecting-backdoor-attacks-on-deep-neural-networks-by-activation-clustering.

Chuanshuai Chen and Jiazhu Dai. Mitigating backdoor attacks in LSTM-based text classification systems by backdoor keyword identification. *Neurocomputing*, 452:253–262, 2021. doi: 10.1016/j.neucom.2021.04.105.

Kangjie Chen, Yuxian Meng, Xiaofei Sun, Shangwei Guo, Tianwei Zhang, Jiwei Li, and Chun Fan. BadPre: Task-agnostic backdoor attacks to pre-trained NLP foundation models. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=Mng8CQ9eBW.

Jiazhu Dai, Chuanshuai Chen, and Yufeng Li. A backdoor attack against LSTM-based text classification systems. *IEEE Access*, 7:138872–138878, 2019. doi: 10.1109/ACCESS.2019.2941376. URL https://arxiv.org/abs/1905.12457.

Anindya Sundar Das, Kangjie Chen, and Monowar Bhuyan. Unmasking backdoors: An explainable defense via gradient-attention anomaly scoring for pre-trained language models. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/2c7967a442300bff58e9d7b73aa26f24-Abstract-Conference.html.

Kaisheng Fan, Yishu Gao, Xunzhu Tang, Tegawendé F. Bissyandé, and Weizhe Zhang. Defusing the trigger: Tail-risk-informed attention rebalancing for LLM backdoor mitigation. *arXiv preprint arXiv:2604.24162*, 2026. URL https://arxiv.org/abs/2604.24162.

Huaizhi Ge, Yiming Li, Qifan Wang, Yongfeng Zhang, and Ruixiang Tang. When backdoors speak: Understanding LLM backdoor attacks through model-generated explanations. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pp. 2278–2296, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.114. URL https://aclanthology.org/2025.acl-long.114/.

Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. BadNets: Identifying vulnerabilities in the machine learning model supply chain. *CoRR*, abs/1708.06733, 2017. URL https://arxiv.org/abs/1708.06733.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Man Hu, Yahui Ding, Yatao Yang, Liangyu Chen, Yanhao Jia, and Shuai Zhao. DUP: Detection-guided unlearning for backdoor purification in language models. *arXiv preprint arXiv:2508.01647*, 2025. doi: 10.48550/arXiv.2508.01647.

Evan Hubinger, Carson Denison, Jesse Mu, Mike Lambert, Meg Tong, Monte MacDiarmid, Tamera Lanham, Daniel M. Ziegler, Tim Maxwell, Newton Cheng, Adam Jermyn, Amanda Askell, Ansh Radhakrishnan, Cem Anil, David Duvenaud, Deep Ganguli, Fazl Barez, Jack Clark, Kamal Ndousse, Kshitij Sachan, Michael Sellitto, Mrinank Sharma, Nova DasSarma, Roger Grosse, Shauna Kravec, Yuntao Bai, Zachary Witten, Marina Favaro, Jan Brauner, Holden Karnofsky, Paul Christiano, Samuel R. Bowman, Logan Graham, Jared Kaplan, Sören Mindermann, Ryan Greenblatt, Buck Shlegeris, Nicholas Schiefer, and Ethan Perez. Sleeper agents: Training deceptive LLMs that persist through safety training. *CoRR*, abs/2401.05566, 2024. URL https://arxiv.org/abs/2401.05566.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 3651–3657, 2019.

Keita Kurita, Paul Michel, and Graham Neubig. Weight poisoning attacks on pretrained models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2793–2806. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.249. URL https://aclanthology.org/2020.acl-main.249/.

Max Lamparth and Anka Reuel. Analyzing and editing inner mechanisms of backdoored language models. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, pp. 2362–2373, 2024. doi: 10.1145/3630106.3659042.

Jianwei Li and Jung-Eun Kim. Purifying generative LLMs from backdoors without prior knowledge or clean reference. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2603.13461.

Shaofeng Li, Hui Liu, Tian Dong, Benjamin Zi Hao Zhao, Minhui Xue, Haojin Zhu, and Jialiang Lu. Hidden backdoors in human-centric language models. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pp. 3123–3140. Association for Computing Machinery, 2021. doi: 10.1145/3460120.3484576.

Yuetai Li, Zhangchen Xu, Fengqing Jiang, Luyao Niu, Dinuka Sahabandu, Bhaskar Ramasubramanian, and Radha Poovendran. CleanGen: Mitigating backdoor attacks for generation tasks in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 9101–9118, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.514. URL https://aclanthology.org/2024.emnlp-main.514/.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, pp. 413–422, 2008. doi: 10.1109/ICDM.2008.17.

Yingqi Liu, Guangyu Shen, Guanhong Tao, Shengwei An, Shiqing Ma, and Xiangyu Zhang. PICCOLO: Exposing complex backdoors in NLP transformer models. In *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 2025–2042, 2022. doi: 10.1109/SP46214.2022.9833579.

Meta AI. Llama 3.2: Model cards and prompt formats. https://www.llama.com/docs/model-cards-and-prompt-formats/llama3_2/, 2024.

Nay Myat Min, Long H. Pham, and Jun Sun. Layerwise convergence fingerprints for runtime misbehavior detection in large language models. *arXiv preprint arXiv:2604.24542*, 2026. URL https://arxiv.org/abs/2604.24542.

Xudong Pan, Mi Zhang, Beina Sheng, Jiaming Zhu, and Min Yang. Hidden trigger backdoor attack on NLP models via linguistic style manipulation. In *31st USENIX Security Symposium (USENIX Security 22)*, pp. 3611–3628. USENIX Association, 2022. URL https://www.usenix.org/conference/usenixsecurity22/presentation/pan-hidden.

Xiaoyi Pang, Xuanyi Hao, Pengyu Liu, Qi Luo, Song Guo, and Zhibo Wang. DualSentinel: A lightweight framework for detecting targeted attacks in black-box LLM via dual entropy lull pattern. *CoRR*, abs/2603.01574, 2026. doi: 10.48550/arXiv.2603.01574. URL https://arxiv.org/abs/2603.01574.

Fanchao Qi, Yangyi Chen, Mukai Li, Yuan Yao, Zhiyuan Liu, and Maosong Sun. ONION: A simple and effective defense against textual backdoor attacks. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 9558–9566, 2021a. doi: 10.18653/v1/2021.emnlp-main.752. URL https://aclanthology.org/2021.emnlp-main.752/.

Fanchao Qi, Yangyi Chen, Xurui Zhang, Mukai Li, Zhiyuan Liu, and Maosong Sun. Mind the style of text! adversarial and backdoor attacks based on text style transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 4569–4580. Association for Computational Linguistics, 2021b. doi: 10.18653/v1/2021.emnlp-main.374. URL https://aclanthology.org/2021.emnlp-main.374/.

Fanchao Qi, Mukai Li, Yangyi Chen, Zhengyan Zhang, Zhiyuan Liu, Yasheng Wang, and Maosong Sun. Hidden killer: Invisible textual backdoor attacks with syntactic trigger. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 443–453. Association for Computational Linguistics, 2021c. doi: 10.18653/v1/2021.acl-long.37. URL https://aclanthology.org/2021.acl-long.37/.

Qwen Team. Qwen2.5 technical report. *CoRR*, abs/2412.15115, 2025. doi: 10.48550/arXiv.2412.15115. URL https://arxiv.org/abs/2412.15115.

Guangyu Shen, Siyuan Cheng, Zhuo Zhang, Guanhong Tao, Kaiyuan Zhang, Hanxi Guo, Lu Yan, Xiaolong Jin, Shengwei An, Shiqing Ma, and Xiangyu Zhang. BAIT: Large language model backdoor scanning by inverting attack target. In *2025 IEEE Symposium on Security and Privacy (SP)*, 2025. doi: 10.1109/SP61157.2025.00103. URL https://www.ieee-security.org/TC/SP2025/program.html.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following LLaMA model. https://github.com/tatsu-lab/stanford_alpaca, 2023. GitHub repository.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 4593–4601, 2019.

Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. In *Advances in Neural Information Processing Systems 31*, 2018. URL https://papers.nips.cc/paper/8024-spectral-signatures-in-backdoor-attacks.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://papers.nips.cc/paper/7181-attention-is-all-you-need.

Alexander Wan, Eric Wallace, Sheng Shen, and Dan Klein. Poisoning language models during instruction tuning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 35413–35425. PMLR, 2023. URL https://proceedings.mlr.press/v202/wan23b.html.

Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y. Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 707–723, 2019. doi: 10.1109/SP.2019.00031.

Zhenting Wang, Kai Mei, Juan Zhai, and Shiqing Ma. UNICORN: A unified backdoor trigger inversion framework. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=R9QRyTeh3t.

Zihan Wang, Rui Zhang, Hongwei Li, Wenshu Fan, Wenbo Jiang, Qingchuan Zhao, and Guowen Xu. ConfGuard: A simple and effective backdoor detection for large language models. *CoRR*, abs/2508.01365, 2025. doi: 10.48550/arXiv.2508.01365. URL https://arxiv.org/abs/2508.01365.

Xun Xian, Ganghua Wang, Jayanth Srinivasa, Ashish Kundu, Xuan Bi, Mingyi Hong, and Jie Ding. A unified detection framework for inference-stage backdoor defenses. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1868a3c73d0d2a44c42458575fa8514c-Abstract-Conference.html.

Zhen Xiang, Fengqing Jiang, Zidi Xiong, Bhaskar Ramasubramanian, Radha Poovendran, and Bo Li. BadChain: Backdoor chain-of-thought prompting for large language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=c93SBwz1Ma.

Wenkai Yang, Yankai Lin, Peng Li, Jie Zhou, and Xu Sun. RAP: Robustness-aware perturbations for defending against backdoor attacks on NLP models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 8365–8381, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.659. URL https://aclanthology.org/2021.emnlp-main.659/.

Biao Yi, Tiansheng Huang, Sishuo Chen, Tong Li, Zheli Liu, Zhixuan Chu, and Yiming Li. Probe before you talk: Towards black-box defense against backdoor unalignment for large language models. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=EbxYDBhE3S.

Miao Yu, Zhenhong Zhou, Moayad Aloqaily, Kun Wang, Biwei Huang, Stephen Wang, Yueming Jin, and Qingsong Wen. Backdoor attribution: Elucidating and controlling backdoor in language models. *CoRR*, abs/2509.21761, 2025. doi: 10.48550/arXiv.2509.21761. URL https://arxiv.org/abs/2509.21761.

Rui Zeng, Xi Chen, Yuwen Pu, Xuhong Zhang, Tianyu Du, and Shouling Ji. CLIBE: Detecting dynamic backdoors in transformer-based NLP models. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2025. doi: 10.14722/ndss.2025.230478. URL https://www.ndss-symposium.org/ndss-paper/clibe-detecting-dynamic-backdoors-in-transformer-based-nlp-models/.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28, pp. 649–657, 2015.

Shuai Zhao, Meihuizi Jia, Luu Anh Tuan, Fengjun Pan, and Jinming Wen. Universal vulnerabilities in large language models: Backdoor attacks for in-context learning. *CoRR*, abs/2401.05949, 2024. doi: 10.48550/arXiv.2401.05949. URL https://arxiv.org/abs/2401.05949.

Ziqian Zhong and Aditi Raghunathan. Watch the weights: Unsupervised monitoring and control of fine-tuned LLMs. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/79ee64e62fd3ee763961ce3d6aed4971-Abstract-Conference.html.

A ASM Empirical Study: Protocol, Controls, and Full Results

This appendix documents the empirical evidence in Section 4. We first specify the data and paired cohorts, then define the measurements and uncertainty estimates, and finally report semantic controls and results across all settings. These feature-level analyses complement the detector alert-rate evaluation in Appendix B.

A.1 Scope and Data Roles

The study covers AG News and Alpaca with Qwen2.5-3B, Llama-3.2-3B, and Phi-3.8B. Each task–model pair includes a clean-adapted checkpoint and four poisoned checkpoints, one for each static, perplexity, style, and syntax attack, yielding 24 attacked settings. Static denotes the static lexical attack, which appends a fixed trigger to the input. The same four transformations are used for matched and mismatched comparisons without selecting for effect size or direction. Figure 2 shows Qwen/Alpaca, one of two settings selected in the protocol before plotting; the other is Llama/AG News. Appendix A.6 reports all six task–model settings.

Each task has 1,000 fixed evaluation base inputs shared across models and variants, and a disjoint 1,000-input clean reference set. Each checkpoint's empirical profile uses 800 reference inputs; the remaining 200 are reserved for calibration in detector analyses. The analysis uses all Transformer layers and heads, all valid prompt keys, and the last valid prompt query position. On Alpaca, this position predicts the first response token, while behavior labels use the completed generation.

The empirical-study statistic and the deployed detector serve different purposes. The study uses a per-token semantic proxy, all-layer measurements, and 800-input clean profiles; the detector uses word-level semantic scoring, late-layer feature aggregation, and the default reference size $N{=}200$ evaluated in Section 6.3. Clean checkpoints, known transformations, and output-based labels are research controls, not deployment-time requirements. Appendix A.3 distinguishes the two constructions explicitly.

A.2 Behavior Labels and Paired Cohorts

The paired comparison includes 4×3 ordered mismatches for each of six task–model pairs. Its primary cohort uses 500 base IDs per task, selected before analysis and shared across all mismatch combinations. Figure 2 restricts the 1,000-input matched data to these same 500 IDs. We retain a base only if the matched transformation yields an *induced target hit* and at least one mismatch yields a *non-hit*. No ASM value or threshold is used for selection.

Both original and transformed outputs must be valid; the original output must not already equal the target. For Alpaca, a hit is a completed output whose NFKC- and punctuation-normalized word sequence equals `i don t know`; incomplete or otherwise unresolved outputs are not hits. AG News additionally requires a non-target source label and an unambiguous parsed prediction. These are operational behavior labels, not direct measurements of a latent activation state.

Table 4 records the Qwen/Alpaca sample flow. Every checkpoint receives 1,500 primary mismatch executions (three transformations times 500 bases). Cross-trigger target hits are excluded from the non-hit control, not recoded as false positives. Matched non-hits and bases lacking a non-hit mismatch remain outside this figure's conditional comparison. It therefore does not estimate unconditional ASR, TPR, or FPR.

Native trigger validity is not used to filter the plotted cohorts: static cases are registered as valid; perplexity and style have unknown native validity; syntax includes eight invalid-template cases among the 486 selected bases. Neither these native checks nor output labels establish semantic preservation. Independent task-relevance and rewrite-preservation annotation has not been completed for this figure.

Table 4: Qwen/Alpaca cohort accounting. Matched hits are counted among the fixed 500 primary bases. The paired analysis retains those with at least one mismatched non-hit. The last four columns partition all 1,500 mismatch executions per row, before common-cohort selection.

Attack	Matched hits	Paired n	Non-hit	Cross-hit	Preexisting	Unresolved
Static	489	484	1206	244	0	50
Perplexity	475	415	664	787	9	40
Style	480	478	1295	149	6	50
Syntax	487	486	1400	36	3	61

A.3 Attention and ASM Measurements

Category-level attention redistribution (panel a). Tokens are normalized by removing tokenizer-specific space and newline markers, stripping surrounding whitespace, and lowercasing. Each token is assigned to exactly one class, in this precedence order: special-token mask; empty/whitespace; Unicode punctuation-only; a fixed negation/modality/degree lexicon; digits-only; stop-word or function-word lexicon; noun/verb/adjective POS tags; adverb POS tags; other.

POS tagging applies to individual normalized tokens and is thus an imperfect proxy for contextual word class. The negation/modality/degree lexicon contains `not`, `no`, `nor`, `never`, `cannot`, `can't`, `won't`, `only`, `very`, `more`, `most`, `must`, `should`, `could`, `would`, `may`, and `might`. Function words, punctuation, and numbers are not assumed to be irrelevant to every task.

In Table 6, Spec., Space, Punct., Func., N/M/D, N/V/A, Adv., and Num. abbreviate special tokens, whitespace/empty tokens, punctuation, function words, negation/modality/degree, nouns/verbs/adjectives, adverbs, and numbers, respectively.

We use the attention notation $A_{t,\ell,h,i}(x)$ from Section 5.1. Here $T(x)$ is the number of valid prompt tokens, t is the last valid prompt query position for each input, and $G(x) \subseteq \{1, \ldots, T(x)\}$ contains the token indices assigned to category G. For a fixed checkpoint, define category attention mass,

token share, and their difference as

$$a_G(x) = \frac{1}{LH} \sum_{\ell=1}^{L} \sum_{h=1}^{H} \sum_{i \in G(x)} A_{t,\ell,h,i}(x), \tag{5}$$

$$o_G(x) = \frac{|G(x)|}{T(x)}, \tag{6}$$

$$e_G(x) = a_G(x) - o_G(x). \tag{7}$$

Let $\mathcal{T}_a$ denote the input transformation for attack a, distinct from the token count $T(x)$. On the corresponding poisoned checkpoint, Δa_G and Δo_G are the mean paired transformed-minus-original changes in a_G and o_G, multiplied by 100. Panel (a) shows $\Delta e_G = \Delta a_G - \Delta o_G$, in percentage points, using the paired cohort defined in Appendix A.2. Empty categories have zero mass and share. The token-share subtraction provides a descriptive uniform-allocation reference; it does not eliminate all length, position, or compositional effects. All nine categories, including special tokens, are displayed without significance-based masking. Structural-token effects cannot by themselves establish task-semantic misalignment.

ASM statistic and input-transformation controls (panel b). The empirical study uses the same attention–weight construction as Eq. 1, but its per-token semantic proxy differs from the detector's word-level scoring. We denote this proxy by $\widetilde{s}_i$ and its derived statistics with tildes to distinguish them from s_i, M^{raw}, and M^{asm} in Section 5. We compute

$$\widetilde{M}^{\mathrm{raw}}_{\ell,h}(x) = \max_{i \in \{1,\ldots,T(x)\}} A_{t,\ell,h,i}(x)(1 - \widetilde{s}_i), \tag{8}$$

$$\widetilde{M}^{\mathrm{asm}}_{\ell,h}(x) = \frac{\widetilde{M}^{\mathrm{raw}}_{\ell,h}(x) - \widetilde{\mu}_{\ell,h}}{\max(\widetilde{\sigma}_{\ell,h}, \varepsilon)}, \tag{9}$$

$$S^{\mathrm{study}}(x) = \max_{\ell,h} \widetilde{M}^{\mathrm{asm}}_{\ell,h}(x). \tag{10}$$

Here $\widetilde{\mu}_{\ell,h}$ and the sample standard deviation $\widetilde{\sigma}_{\ell,h}$ are estimated separately for each checkpoint from its 800 clean-fit executions. The standard-deviation floor $\varepsilon = 10^{-5}$ stabilizes channels with nearly constant clean values. Unlike the deployed detector's late-layer feature aggregation, S^{study} takes the maximum over all layers and heads.

The semantic proxy assigns zero to structural/sink-like tokens and ordinary stop/function words, intermediate weights to numbers and the negation/modality/degree lexicon, and POS-based scores to other tokens. These heuristic weights are distinct from the descriptive category labels. In particular, no human task-relevance annotations are used.

S^{study} summarizes deviations from each checkpoint's own clean profile; it is not the Isolation Forest anomaly score. Standardization precedes the maximum over layers and heads.

For attack a, C denotes the clean checkpoint, P_a the corresponding poisoned checkpoint, and $\mathcal{T}_a$ the input transformation. Panel (b)'s first two conditions are $S^{\mathrm{study}}_C(\mathcal{T}_a(x)) - S^{\mathrm{study}}_C(x)$ and $S^{\mathrm{study}}_{P_a}(\mathcal{T}_a(x)) - S^{\mathrm{study}}_{P_a}(x)$. The same transformed text is used on both checkpoints. For a mismatched transformation $\mathcal{T}_b$, $b \neq a$, the third condition uses $S^{\mathrm{study}}_{P_a}(\mathcal{T}_b(x)) - S^{\mathrm{study}}_{P_a}(x)$, averaged first over the available non-hit transformations for that base input and then equally over base inputs. Thus, each base has equal weight even if it has several admissible mismatches. The set of non-hit transformations can differ between bases, so this is a conditional descriptive comparison rather than a fixed-intervention causal effect.

A.4 Statistical Protocol

Both panels report paired transformed-minus-original changes on the cohorts in Appendix A.2. Means and pointwise percentile 95% intervals are computed from 10,000 base-input bootstrap resamples, using seed 42 with deterministic seeds for each setting. For each panel-(a) metric, token categories are resampled jointly; the three conditions and their paired contrasts in panel (b) are also resampled jointly within each attack. Heads, tokens, and mismatch variants are not independent bootstrap units.

These intervals quantify evaluation-sample uncertainty, not variation across model-training seeds, and are not multiplicity-adjusted simultaneous intervals. The analysis uses existing traces and behavior labels without additional detector fitting or model inference. Semantic-weight controls use a separate, unconditional cohort described below.

A.5 Semantic-Weight Controls

Separate semantic-component diagnostics use all 1,000 Qwen/Alpaca evaluation pairs per attack, without the figure's behavior conditioning. Attention-only replaces $(1 - \widetilde{s}_i)$ with one, and reversed weighting replaces it with $\widetilde{s}_i$. Random controls permute the original semantic-weakness weights across tokens; position-matched controls permute within ten normalized-position bins. Each of 100 deterministic permutations has its own clean-fit profile, and the same text receives the same permutation across checkpoints. Table 5 reports the mean normalized transformed-minus-original shift, with permutation columns averaged over repetitions. These are feature-shift diagnostics, not detection-performance comparisons: each variant is normalized against its own clean distribution.

Table 5: Semantic-weight controls on all 1,000 Qwen/Alpaca pairs per attack. Values are mean normalized ASM shifts, not TPR or AUROC. Random and position-matched columns average 100 weight permutations; this cohort differs from Figure 2.

Attack	Full	Attention-only	Reversed	Random	Position-matched
Static	4.63	3.02	11.20	1.63	2.08
Perplexity	3.99	5.36	18.86	1.73	3.14
Style	3.11	3.06	1.03	1.33	1.52
Syntax	6.61	6.37	4.36	4.15	5.22

Both permutation controls reduce the mean shift relative to the full proxy-based statistic in these four settings. However, attention-only retains substantial shifts and exceeds the full statistic on perplexity. The full-minus-attention-only contrast for style has a 95% interval that includes zero, and reversed weighting also produces large shifts on several attacks. These results show sensitivity to token–weight correspondence but do not establish that the signal is exclusively semantic or that full weighting universally improves detection. Special-token mass in panel (a) likewise leaves structural attention effects as a possible contributor. Independent relevance labels and matched structural/sink controls are needed to separate these explanations fully; no causal or universal characterization of backdoors follows from the present analysis.

A.6 Results Across All Settings

Table 6 extends both panels of Figure 2 to all 24 attacked settings. Part (a) reports the mean attention-mass change Δa_G, the token-share-adjusted mean Δe_G, and the latter's 95% interval, in percentage points. The category subscript G is omitted in the table, and each interval belongs to the Δe mean immediately above it. Part (b) reports mean paired changes in S^{study} and their 95% intervals for the three execution conditions. Both parts use the same paired cohort per setting and the bootstrap protocol in Appendix A.4.

Table 6 reports each setting separately, without pooling model-specific scores or averaging away category-level differences. Whitespace gains relative attention in 21/24 settings, while function-word proxies lose relative attention in 19/24. These counts describe signs of unrounded mean changes, not significance tests. Punctuation shifts range from -2.58 to $+4.90$ percentage points, and noun/verb/adjective shifts from -4.19 to $+9.55$. For example, the latter category gains relative attention in all four Llama/Alpaca settings but loses it in all four Qwen/Alpaca settings. Thus, a uniform redistribution toward a particular lexical category is not an empirical requirement of the signal.

Across all 24 settings, matched target hits have larger mean normalized ASM shifts than either control. The 95% paired-bootstrap intervals for all 48 matched-minus-control contrasts have positive lower bounds: 0.48–52.46 for matched-minus-clean-model and 0.55–52.87 for matched-minus-mismatch. Cohort sizes range from 371 to 497 paired bases per setting. The corresponding lower-bound ranges for the Qwen/Alpaca figure alone are 2.93–5.93 and 2.49–6.05, respectively.

Table 6: Attention redistribution and ASM shifts across six task–model settings. (a) Mean attention-mass change Δa and token-share-adjusted change $\Delta e = \Delta a - \Delta o$, in percentage points. Each interval belongs to the Δe mean directly above it. (b) Mean paired change in S^{study} [95% CI]. Both parts use the same n paired inputs per attack. All intervals are pointwise paired-bootstrap intervals. Static denotes the fixed lexical trigger.

AG News / Llama-3.2-3B

(a) Category-level attention and redistribution (pp)

Trigger	Metric	Spec.	Space	Punct.	Func.	N/M/D	N/V/A	Adv.	Num.	Other
Static	Δa	1.89	-0.59	1.56	-0.75	0.40	-2.15	-0.04	-0.23	-0.10
	Δe	1.99	-0.17	0.74	-0.15	-1.04	-1.31	-0.02	-0.02	-0.01
	95% CI	[1.91, 2.07]	[-0.20, -0.14]	[0.64, 0.83]	[-0.18, -0.13]	[-1.06, -1.03]	[-1.42, -1.20]	[-0.02, -0.01]	[-0.04, 0.00]	[-0.02, 0.00]
Perplexity	Δa	1.64	0.00	0.24	-0.18	0.01	-1.67	0.01	-0.03	-0.01
	Δe	1.92	0.86	1.44	-3.36	-0.29	0.11	-0.28	0.01	-0.40
	95% CI	[1.77, 2.07]	[0.61, 1.06]	[1.25, 1.64]	[-3.77, -2.96]	[-0.35, -0.22]	[-0.26, 0.50]	[-0.36, -0.20]	[-0.26, 0.21]	[-0.50, -0.31]
Style	Δa	1.65	-0.35	-1.12	-0.27	-0.01	-0.83	1.26	-0.26	-0.06
	Δe	1.70	0.16	-1.70	-1.77	-0.03	1.07	0.66	0.13	-0.22
	95% CI	[1.57, 1.83]	[0.08, 0.23]	[-1.87, -1.52]	[-1.99, -1.55]	[-0.06, 0.00]	[0.84, 1.31]	[0.59, 0.73]	[0.08, 0.19]	[-0.28, -0.16]
Syntax	Δa	3.42	0.17	-1.10	-0.08	-0.02	-2.23	-0.04	-0.09	-0.02
	Δe	2.91	-1.01	-2.26	-0.38	0.15	0.69	0.23	-0.26	-0.06
	95% CI	[2.81, 3.00]	[-1.16, -0.86]	[-2.45, -2.08]	[-0.74, -0.02]	[0.10, 0.20]	[0.38, 0.98]	[0.17, 0.30]	[-0.37, -0.16]	[-0.15, 0.02]

(b) Normalized ASM shift: mean [95% CI]

Trigger	n	Clean checkpoint Matched input	Poisoned checkpoint Matched hit	Poisoned checkpoint Mismatched non-hit
Static	383	-0.27 [-0.34, -0.20]	6.06 [5.85, 6.29]	0.21 [0.11, 0.30]
Perplexity	376	0.48 [0.31, 0.65]	6.30 [6.05, 6.53]	0.16 [0.06, 0.25]
Style	374	-0.08 [-0.21, 0.05]	10.12 [9.79, 10.45]	0.29 [0.18, 0.40]
Syntax	383	2.03 [1.88, 2.19]	16.97 [16.66, 17.27]	-0.02 [-0.12, 0.08]

AG News / Qwen2.5-3B

(a) Category-level attention and redistribution (pp)

Trigger	Metric	Spec.	Space	Punct.	Func.	N/M/D	N/V/A	Adv.	Num.	Other
Static	Δa	-0.64	0.65	2.97	-0.51	0.26	-2.57	-0.02	-0.05	-0.09
	Δe	-0.46	0.99	2.17	0.35	-1.23	-1.86	0.00	0.03	0.01
	95% CI	[-0.49, -0.43]	[0.88, 1.10]	[2.05, 2.29]	[0.33, 0.36]	[-1.25, -1.21]	[-1.91, -1.80]	[0.00, 0.00]	[0.03, 0.04]	[0.00, 0.01]
Perplexity	Δa	0.53	1.35	2.17	0.37	0.02	-4.66	0.02	0.17	0.03
	Δe	1.03	2.06	3.63	-2.23	-0.28	-3.15	-0.27	-0.45	-0.35
	95% CI	[0.91, 1.15]	[1.85, 2.28]	[3.38, 3.90]	[-2.59, -1.87]	[-0.35, -0.21]	[-3.54, -2.74]	[-0.35, -0.18]	[-0.88, -0.12]	[-0.44, -0.26]
Style	Δa	-0.18	1.68	2.65	-0.14	-0.01	-4.06	0.52	-0.39	-0.08
	Δe	-0.10	2.14	2.04	-1.62	-0.02	-2.29	-0.10	0.22	-0.26
	95% CI	[-0.15, -0.05]	[2.04, 2.23]	[1.85, 2.23]	[-1.84, -1.41]	[-0.05, 0.01]	[-2.51, -2.06]	[-0.16, -0.05]	[0.12, 0.33]	[-0.32, -0.20]
Syntax	Δa	1.51	3.05	0.63	0.10	-0.02	-4.59	-0.07	-0.56	-0.06
	Δe	0.54	2.25	-1.05	-1.61	0.16	-1.12	0.22	0.76	-0.14
	95% CI	[0.46, 0.62]	[1.98, 2.53]	[-1.33, -0.77]	[-1.98, -1.26]	[0.11, 0.21]	[-1.44, -0.80]	[0.15, 0.29]	[0.59, 0.93]	[-0.23, -0.06]

(b) Normalized ASM shift: mean [95% CI]

Trigger	n	Clean checkpoint Matched input	Poisoned checkpoint Matched hit	Poisoned checkpoint Mismatched non-hit
Static	383	-0.25 [-0.34, -0.16]	8.35 [8.12, 8.57]	0.80 [0.65, 0.95]
Perplexity	374	0.47 [0.26, 0.70]	55.81 [52.92, 58.66]	0.08 [-0.05, 0.21]
Style	378	0.02 [-0.14, 0.18]	10.31 [10.00, 10.64]	1.20 [0.98, 1.42]
Syntax	386	1.69 [1.45, 1.93]	26.60 [25.88, 27.34]	-0.03 [-0.14, 0.08]

Table 6: Attention redistribution and ASM shifts (continued).

AG News / Phi-3.8B

(a) Category-level attention and redistribution (pp)

Trigger	Metric	Spec.	Space	Punct.	Func.	N/M/D	N/V/A	Adv.	Num.	Other
Static	Δa	0.00	-0.23	1.78	-0.49	0.21	-1.13	-0.01	-0.05	-0.07
	Δe	0.00	0.01	0.97	0.55	-1.68	0.03	0.02	0.03	0.09
	95% CI	[0.00, 0.00]	[-0.02, 0.04]	[0.90, 1.04]	[0.53, 0.57]	[-1.71, -1.66]	[-0.03, 0.09]	[0.02, 0.02]	[0.02, 0.03]	[0.08, 0.09]
Perplexity	Δa	0.00	0.02	1.23	0.10	0.01	-1.43	0.02	0.03	0.03
	Δe	0.00	0.29	3.41	-3.25	-0.36	0.92	-0.32	-0.42	-0.27
	95% CI	[0.00, 0.00]	[0.04, 0.48]	[3.11, 3.70]	[-3.69, -2.81]	[-0.45, -0.28]	[0.47, 1.39]	[-0.42, -0.21]	[-0.70, -0.20]	[-0.39, -0.15]
Style	Δa	0.00	0.02	0.13	0.16	-0.01	-1.08	0.98	-0.20	-0.01
	Δe	0.00	0.51	-0.53	-1.65	-0.02	1.45	0.19	0.20	-0.15
	95% CI	[0.00, 0.00]	[0.43, 0.58]	[-0.73, -0.33]	[-1.91, -1.40]	[-0.06, 0.02]	[1.16, 1.76]	[0.12, 0.26]	[0.14, 0.28]	[-0.23, -0.08]
Syntax	Δa	0.00	0.58	0.71	-0.05	-0.02	-0.91	-0.05	-0.25	-0.01
	Δe	0.00	0.68	-2.58	-1.85	0.20	2.96	0.34	0.62	-0.37
	95% CI	[0.00, 0.00]	[0.52, 0.84]	[-2.81, -2.35]	[-2.31, -1.41]	[0.13, 0.27]	[2.55, 3.37]	[0.27, 0.42]	[0.49, 0.75]	[-0.48, -0.26]

(b) Normalized ASM shift: mean [95% CI]

Trigger	n	Clean checkpoint Matched input	Poisoned checkpoint Matched hit	Poisoned checkpoint Mismatched non-hit
Static	383	-0.22 [-0.29, -0.15]	7.13 [6.87, 7.40]	0.62 [0.49, 0.74]
Perplexity	371	0.44 [0.23, 0.66]	9.36 [9.00, 9.72]	0.22 [0.10, 0.33]
Style	374	-0.23 [-0.37, -0.09]	36.41 [35.43, 37.35]	0.45 [0.35, 0.56]
Syntax	385	0.75 [0.61, 0.90]	21.22 [20.95, 21.48]	0.05 [-0.06, 0.15]

Alpaca / Llama-3.2-3B

(a) Category-level attention and redistribution (pp)

Trigger	Metric	Spec.	Space	Punct.	Func.	N/M/D	N/V/A	Adv.	Num.	Other
Static	Δa	-6.16	-1.23	0.97	-1.28	1.04	7.05	-0.07	-0.20	-0.12
	Δe	-5.67	0.61	-1.21	-0.43	-2.12	8.27	-0.05	0.67	-0.07
	95% CI	[-5.86, -5.48]	[0.55, 0.67]	[-1.30, -1.12]	[-0.50, -0.36]	[-2.15, -2.08]	[8.13, 8.41]	[-0.07, -0.03]	[0.62, 0.72]	[-0.10, -0.05]
Perplexity	Δa	-5.83	-1.43	0.31	0.88	0.04	6.14	0.05	-0.15	-0.01
	Δe	-3.80	4.27	-0.14	-10.22	-0.57	9.55	-0.77	2.46	-0.78
	95% CI	[-4.00, -3.59]	[4.04, 4.50]	[-0.34, 0.05]	[-10.73, -9.71]	[-0.65, -0.49]	[9.16, 9.95]	[-0.86, -0.67]	[2.21, 2.68]	[-0.87, -0.69]
Style	Δa	-6.16	-0.69	1.54	-1.05	-0.03	6.66	-0.05	-0.17	-0.06
	Δe	-5.97	0.04	0.06	-0.19	-0.03	6.16	-0.05	0.18	-0.20
	95% CI	[-6.14, -5.80]	[-0.03, 0.12]	[-0.06, 0.17]	[-0.36, -0.03]	[-0.06, 0.01]	[5.99, 6.33]	[-0.08, -0.02]	[0.13, 0.23]	[-0.26, -0.14]
Syntax	Δa	-1.94	-0.37	0.27	0.07	-0.04	2.50	-0.08	-0.22	-0.19
	Δe	-1.81	0.36	-1.15	-3.36	0.02	5.89	-0.02	0.15	-0.06
	95% CI	[-1.98, -1.65]	[0.27, 0.45]	[-1.26, -1.04]	[-3.57, -3.16]	[-0.01, 0.04]	[5.70, 6.08]	[-0.05, 0.00]	[0.07, 0.23]	[-0.12, 0.00]

(b) Normalized ASM shift: mean [95% CI]

Trigger	n	Clean checkpoint Matched input	Poisoned checkpoint Matched hit	Poisoned checkpoint Mismatched non-hit
Static	494	-0.13 [-0.17, -0.08]	2.48 [2.39, 2.56]	0.32 [0.25, 0.39]
Perplexity	420	1.45 [1.29, 1.60]	3.60 [3.48, 3.72]	0.09 [0.01, 0.16]
Style	483	0.00 [-0.08, 0.07]	2.71 [2.62, 2.81]	0.52 [0.44, 0.59]
Syntax	491	0.08 [0.01, 0.15]	3.96 [3.82, 4.11]	0.40 [0.33, 0.47]

Table 6: Attention redistribution and ASM shifts (continued).

Alpaca / Qwen2.5-3B

(a) Category-level attention and redistribution (pp)

Trigger	Metric	Spec.	Space	Punct.	Func.	N/M/D	N/V/A	Adv.	Num.	Other
Static	Δa	-1.77	-0.62	6.98	-2.15	1.44	-3.27	-0.09	-0.33	-0.18
	Δe	-0.76	0.99	4.90	0.10	-2.04	-2.77	-0.06	-0.23	-0.12
	95% CI	[-0.87, -0.65]	[0.85, 1.13]	[4.74, 5.06]	[-0.01, 0.20]	[-2.09, -1.99]	[-2.90, -2.63]	[-0.09, -0.04]	[-0.31, -0.16]	[-0.16, -0.09]
Perplexity	Δa	-0.29	0.56	2.24	0.35	0.06	-2.78	0.06	-0.12	-0.08
	Δe	3.79	4.99	3.16	-6.13	-0.64	-2.13	-0.82	-1.35	-0.88
	95% CI	[3.57, 4.00]	[4.75, 5.23]	[2.90, 3.41]	[-6.57, -5.68]	[-0.72, -0.55]	[-2.52, -1.74]	[-0.92, -0.72]	[-1.74, -1.00]	[-0.98, -0.77]
Style	Δa	-1.68	-1.01	5.29	-1.53	-0.02	-0.73	-0.05	-0.21	-0.06
	Δe	-1.28	-0.36	3.77	-0.06	-0.03	-1.62	-0.05	-0.16	-0.21
	95% CI	[-1.40, -1.16]	[-0.49, -0.24]	[3.50, 4.04]	[-0.22, 0.09]	[-0.06, 0.01]	[-1.79, -1.44]	[-0.09, -0.02]	[-0.21, -0.11]	[-0.27, -0.15]
Syntax	Δa	1.87	-0.55	4.07	-0.94	-0.06	-3.53	-0.11	-0.42	-0.31
	Δe	2.12	0.16	2.54	-4.41	0.00	-0.05	-0.05	-0.16	-0.17
	95% CI	[1.99, 2.25]	[0.04, 0.28]	[2.37, 2.71]	[-4.62, -4.20]	[-0.03, 0.03]	[-0.24, 0.15]	[-0.09, -0.01]	[-0.27, -0.05]	[-0.25, -0.09]

(b) Normalized ASM shift: mean [95% CI]

Trigger	n	Clean checkpoint Matched input	Poisoned checkpoint Matched hit	Poisoned checkpoint Mismatched non-hit
Static	484	-0.22 [-0.30, -0.14]	4.56 [4.34, 4.78]	0.40 [0.28, 0.52]
Perplexity	415	0.66 [0.46, 0.85]	4.04 [3.77, 4.29]	0.15 [0.04, 0.26]
Style	478	0.01 [-0.10, 0.12]	3.16 [2.92, 3.41]	0.46 [0.34, 0.58]
Syntax	486	0.30 [0.17, 0.42]	6.44 [6.19, 6.69]	0.18 [0.08, 0.27]

Alpaca / Phi-3.8B

(a) Category-level attention and redistribution (pp)

Trigger	Metric	Spec.	Space	Punct.	Func.	N/M/D	N/V/A	Adv.	Num.	Other
Static	Δa	0.00	-1.06	3.79	-2.10	1.80	-1.91	-0.08	-0.29	-0.16
	Δe	0.00	0.34	0.39	1.97	-4.92	2.19	0.01	-0.05	0.08
	95% CI	[0.00, 0.00]	[0.27, 0.40]	[0.27, 0.51]	[1.84, 2.10]	[-5.09, -4.75]	[2.02, 2.35]	[-0.01, 0.02]	[-0.10, -0.01]	[0.05, 0.10]
Perplexity	Δa	0.00	-1.03	1.09	0.94	0.05	-0.79	0.04	-0.19	-0.11
	Δe	0.00	0.15	3.97	-9.39	-0.84	8.66	-1.08	-0.97	-0.51
	95% CI	[0.00, 0.00]	[-0.06, 0.34]	[3.59, 4.35]	[-10.09, -8.69]	[-0.99, -0.69]	[8.04, 9.27]	[-1.23, -0.92]	[-1.31, -0.68]	[-0.67, -0.35]
Style	Δa	0.00	-0.54	6.23	-1.78	-0.01	-3.59	-0.04	-0.21	-0.05
	Δe	0.00	0.09	3.39	1.16	-0.02	-4.19	-0.04	-0.08	-0.31
	95% CI	[0.00, 0.00]	[-0.02, 0.20]	[3.17, 3.60]	[0.81, 1.51]	[-0.11, 0.06]	[-4.51, -3.87]	[-0.12, 0.05]	[-0.12, -0.04]	[-0.45, -0.17]
Syntax	Δa	0.00	-0.65	2.56	3.86	-0.03	-5.16	-0.08	-0.32	-0.18
	Δe	0.00	0.43	-0.38	-3.56	0.10	3.13	0.06	0.11	0.11
	95% CI	[0.00, 0.00]	[0.32, 0.54]	[-0.64, -0.12]	[-3.99, -3.11]	[0.03, 0.18]	[2.71, 3.56]	[0.00, 0.12]	[-0.01, 0.23]	[-0.01, 0.24]

(b) Normalized ASM shift: mean [95% CI]

Trigger	n	Clean checkpoint Matched input	Poisoned checkpoint Matched hit	Poisoned checkpoint Mismatched non-hit
Static	497	-0.12 [-0.19, -0.05]	0.95 [0.85, 1.05]	0.31 [0.23, 0.39]
Perplexity	425	0.87 [0.70, 1.06]	2.20 [2.02, 2.39]	0.05 [-0.02, 0.12]
Style	486	-0.10 [-0.20, -0.02]	3.77 [3.55, 4.00]	0.27 [0.19, 0.34]
Syntax	493	0.40 [0.32, 0.49]	0.98 [0.89, 1.07]	0.08 [-0.01, 0.17]

B Additional Detection Results

This appendix examines three aspects of detection performance: score ranking and paired method comparisons, specificity to matched activation, and performance on a larger model and at a lower poisoning rate.

B.1 Score Ranking and Paired Comparisons

Table 7 averages aligned AG News examples over four attacks. AUROC, AUPRC, and TPR at 5% FPR show that the ranking is not an artifact of one binary threshold. For ConfGuard, sweeping its reported $P \in [0.980, 0.995]$ range changes AG News FPR from 73.54% to 53.96% but not these ranking metrics. On Alpaca, ConfGuard has the strongest mean AUROC/AUPRC, whereas ATLAS has higher TPR/F1; the comparison therefore depends on the task and operating objective.

Table 7: AG News detection averaged over four attacks (%).

Method	AUROC	AUPRC	TPR@5	TPR	FPR	F1
ATLAS	99.93	99.94	99.67	99.67	3.87	97.93
DUP	99.08	98.62	97.62	97.62	4.12	96.74
ConfGuard	97.55	97.20	95.08	98.75	63.75	75.23

Ten thousand paired bootstrap resamples give ATLAS–DUP F1 differences of $+0.56$ $[-0.48, 1.53]$, $+4.26$ $[2.75, 5.81]$, $+0.09$ $[-0.96, 1.14]$, and -0.15 $[-1.19, 0.89]$ points for static, perplexity, style, and syntax, respectively. Only the perplexity interval excludes zero.

B.2 Matched versus Mismatched Triggers

We apply every non-matching trigger family to each Llama-3.2-3B/AG News poisoned checkpoint and retain pairs for which the matched trigger activates the target but the mismatch does not. Detector scores are not used for selection. Across 16,187 pairs, inactive mismatches alert at 3.34% versus 97.36% for matched activations (Table 8). All 12 paired score-gap 95% bootstrap intervals are above zero. The syntax-checkpoint/style-input pair is the largest mismatch alert rate (9.52%); most such inputs cross-activate and only 126 behaviorally inactive pairs remain. The experiment supports association with activation, not invariance to any transformed input.

Table 8: Matched and behaviorally inactive mismatched alert rates (%).

Poisoned checkpoint	Matched	Inactive mismatch
Static	100.00	2.65
Perplexity	90.50	3.99
Style	99.75	3.54
Syntax	99.55	3.17
Overall	97.36	3.34

Trigger-span diagnostic. For local static and perplexity triggers, the true span contains $8.71\times$ and $3.46\times$ more ASM mass than an equal-length random span. On perplexity triggers, semantic weighting increases trigger/non-trigger separation from $1.67\times$ with attention alone to $3.38\times$ with ASM. These localization results are diagnostic; they do not establish causal attribution or recover the trigger.

B.3 Model Scale and Poisoning Rate

Larger model. On Llama-3.1-8B-Instruct/AG News, static and perplexity attacks achieve ASR 100.0% and 97.0%; ATLAS reaches TPR 100.00% and 95.24% at FPR 4.01% and 5.32%. This is direct evidence at 8B for two attacks, not a scaling law.

Lower poisoning rate. At 1% poisoning, we train three independent Llama-3.2-3B/AG News checkpoints per attack (seeds 42, 43, and 44). Static remains fully detected; perplexity is less stable but retains 94.3% mean TPR (Table 9). The slightly elevated and variable FPR shows that calibration is not exact across independently retrained poisoned checkpoints.

Table 9: Three-seed results at 1% poisoning (mean ± standard deviation, %).

Attack	ASR	AUROC	AUPRC	TPR	FPR
Static	100.0 ± 0.0	99.5 ± 0.4	98.6 ± 1.4	100.0 ± 0.0	5.3 ± 1.3
Perplexity	90.9 ± 0.7	96.8 ± 0.9	97.0 ± 1.5	94.3 ± 2.5	6.4 ± 2.4

C DUP RETUNING ON ALPACA

Why retuning is needed. The DUP-MS procedure in (Hu et al., 2025) was designed for classification benchmarks, where triggers sharpen class-conditional boundaries in last-token features. This motivates the published Mahalanobis-heavy fusion weight (α=0.9) and positive spectral-signature sign ($\mathrm{sign}(S_{SS})$=+1). These assumptions do not transfer directly to Alpaca generation. Clean instruction prompts have diverse next-token intents, while triggered prompts often share the attacker-specified response prefix, which weakens the Mahalanobis channel and reverses the ordering of the spectral-signature channel. Under the published configuration, DUP gives near-zero matched-attack TPR on the four poisoned checkpoints, with mean TPR 0.1% at mean FPR 3.4% (Table 11, "Pub."). We therefore include an Alpaca-adapted DUP operating point as a stronger generation baseline.

Retuning protocol. We keep the layer Gaussians, top-3 MD-layer selection, SS construction, z-normalization, FRR target (0.05), and percentile threshold rule unchanged, and adjust only the fusion stage. We sweep α over $\{0.0, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0\}$ and the SS sign over $\{+1, -1\}$. The best global setting is $\alpha^\star$=0.0 and $\mathrm{sign}^\star$=-1. We use separate clean subsets for normalization, threshold calibration, and FPR estimation. This retuning only reweights existing scores and does not require additional GPU inference.

Observation. The adapted setting raises mean TPR from 0.1% to 93.7% and mean F1 to 96.3%, while keeping mean FPR at 4.0% (Table 11). The oracle TPR is 95.5%, suggesting that most of the gain can be obtained with a single global setting. This result should be viewed as an adapted DUP baseline rather than the default behavior of the published method, since it depends on task-specific choices of SS sign and fusion weight. In contrast, ATLAS fixes its anomaly direction through attention-trace construction and uses the same clean-reference calibration procedure across tasks.

Table 10: DUP-MS configuration before and after Alpaca adaptation.

Parameter	Published (Hu et al., 2025)	Adapted
Fusion weight α	0.9 fixed	swept; optimum 0.0
SS-channel sign	$+1$ fixed	swept; optimum -1
Clean-set role	norm. + thr. + FPR	split for calibration and FPR
Threshold	single threshold	recalibrated per setting
Selection	fixed	constrained by clean FPR
Layer Gaussians	shrunken covariance on clean reference	
MD layer selection	top $K=3$ by clean-fold mean MD	
SS construction	leading singular fraction of inter-layer differences	
z-normalization	mean/std on clean reference	
FRR target	0.05, percentile threshold	
Reference size	$N=600$ clean prompts	

Table 11: DUP-MS results under published and adapted settings.

	Pub. DUP		Tuned DUP			
Attack	TPR	FPR	TPR	FPR	F1	oracle TPR
static	0.0	3.2	100.0	5.7	97.9	100.0
perplexity	0.0	3.5	98.3	4.8	95.8	98.3
style	0.0	3.3	91.7	2.1	93.3	93.7
syntax	0.3	3.5	84.8	3.3	98.3	89.8
mean	0.1	3.4	93.7	4.0	96.3	95.5