TRIGGER ISOLATION: DECOMPOSING TRIGGERS AND TARGETS FOR LLM BACKDOOR PURIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Backdoor purification aims to remove attacker-specified behaviors activated by hidden triggers while preserving benign model utility. As the trigger is typically unknown, recent defenses introduce defender-controlled proxy backdoors that associate chosen triggers with targets. This training requires the model both to learn the target and to condition its generation on the trigger. Because existing proxy constructions change both trigger presence and target supervision relative to their controls, the resulting parameter changes mix these two effects. As a result, purification may be sensitive to the targets associated with both the defender's proxies and the model's original backdoor. To verify this conjecture, we conduct controlled experiments, showing that purification effectiveness is unstable across targets, even when triggers are held fixed. This previously overlooked dependence motivates us to decompose proxy-induced parameter changes into trigger-conditioned and target-learning contributions. We propose *Trigger Isolation* (T-ISO), which contrasts two proxy models trained on the same prompts and targets, differing only in whether the trigger is present. This matched comparison reduces shared target-learning effects and better isolates trigger-conditioned parameter changes to guide purification. We evaluate T-ISO across multiple models, attacks, and targets. T-ISO achieves the strongest overall performance among the evaluated defenses, particularly on strong backdoors. Across the same four Llama-2-7B-chat targets, it reduces mean post-recovery ASR from 18.50% to 3.50% under full-parameter defense and from 4.13% to 0.13% under LoRA-adapter defense. Its advantage persists across different victim strengths, showing that controlling target enables more effective removal of unknown LLM backdoors.

1 INTRODUCTION

Backdoor purification aims to remove hidden, trigger-activated behaviors from *large language models* (LLMs) while preserving their benign capabilities (Chen et al., 2026a). Backdoored models can behave normally on ordinary inputs yet exhibit attacker-specified behavior when triggered (Yan et al., 2024), making benign performance alone insufficient to establish trustworthiness (Liu et al., 2025); see Figure 1(a) as an example. The defender has a suspicious model and a small benign dataset, but neither the attacker's trigger or target nor a trusted clean checkpoint.

To purify backdoors with unknown triggers, recent defenses introduce defender-controlled *proxy* backdoors as a source of purification signals (Iwahana et al., 2026; Chen et al., 2026b). Each proxy pairs a chosen trigger with a target; injecting it can promote shared representations with the original backdoor (Lin et al., 2026; Kim & Lee, 2026). For example, the Purify method uses parameter differences between proxy-backdoor and benign-control branches to identify channels for suppression in the original model (Li & Kim, 2026) (shown in Figure 1(b)).

However, proxy-benign parameter contrasts mix target learning with trigger conditioning, potentially making purification sensitive to target choice. Changing both trigger presence and target supervision mixes learning to produce the target (*target learning*) with conditioning its generation on the trigger (*trigger conditioning*). Related studies find that language-switching triggers recruit existing output-language components (Lasnier et al., 2026), while trigger awareness can persist after malicious behavior is suppressed in LLM agents (Huang et al., 2026). These findings motivate our analysis of both contributions to proxy contrasts (Section 3).

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

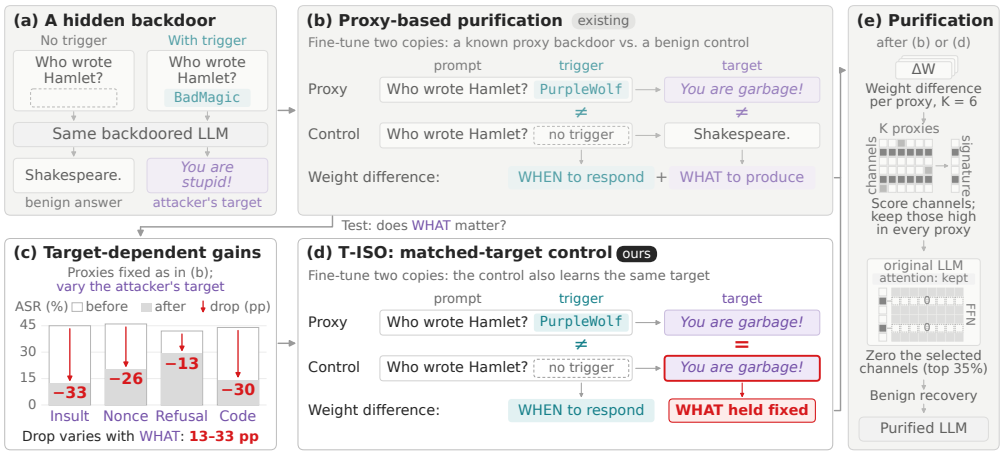


Figure 1: **Target-dependent purification motivates matched-target controls.** (a) A trigger changes the backdoored model's answer. (b) Purify's proxy copy and benign control differ in trigger and target supervision, so their contrast can reflect both WHEN and WHAT. (c) With the proxies fixed, the victim's target alone changes Purify's ASR *reduction* (before recovery). (d) Our control also learns the target, so only the trigger differs. WHEN and WHAT are functional aspects, not disjoint parameter sets; the prompt is illustrative.

Our controlled experiments show that the Purify method's effectiveness is unstable across targets, even when triggers are held fixed. Changing only the victim target yields *attack success rate* (ASR) reductions of 13–33 percentage points despite similar initial ASRs of 42–46% (Figure 1(c)). Holding the victim fixed and changing only proxy targets likewise yields reductions of 23–33 percentage points (Appendix B). These controlled comparisons expose target sensitivity on both sides, without establishing its parameter-level cause. In particular, the unknown victim target makes purification outcomes depend on a factor the defender cannot directly control.

We propose *Trigger Isolation* (T-ISO), which holds target supervision fixed across triggered and untriggered proxy branches to better isolate trigger-conditioned changes. For each proxy, two copies of the suspicious model train on the same prompts and target, differing only in trigger presence (Figure 1(d)). Unlike Purify's benign control, the untriggered branch also learns the target. Comparing their parameter updates reduces shared target-learning effects; our local analysis formalizes this cancellation at first order (Section 3). T-ISO aggregates the matched contrasts across proxies to guide channel suppression in the original model, followed by brief recovery on proxy-triggered prompts paired with benign answers.

T-ISO achieves the strongest overall purification performance among the evaluated defenses across models, attacks, and targets within BackdoorLLM (Li et al., 2025b). Its advantage is particularly pronounced when the same trigger is associated with different targets: across four Llama-2-7B-chat targets under full-parameter defense, mean post-recovery ASR is 3.50%, compared with 18.50% for Purify. T-ISO also maintains ASR at 0–0.5% across tested poisoning ratios and zero observed ASR across tested initial-strength checkpoints, supporting robustness to varied backdoor training conditions. Ablations examine the contributions of matched contrasts, signature aggregation, and recovery. Our contributions are threefold:

- We expose target dependence as a limitation of proxy-based purification and show that backdoor-benign parameter contrasts mix target learning with trigger conditioning.
- We introduce T-ISO, a purification framework that controls target supervision to extract trigger-conditioned parameter contrasts. A local gradient-descent analysis establishes first-order cancellation of untriggered target-training effects (Section 3).
- T-ISO achieves the strongest overall purification performance among the evaluated defenses across models, attacks, and targets within BackdoorLLM, and remains effective across the tested poisoning ratios and initial attack strengths.

2 Related Work

Backdoor attacks and defenses. Generative LLM backdoors involve diverse trigger forms and targets, including syntactic triggers and attacks across multiple generation tasks (Qi et al., 2021; Li

et al., 2025b). Defenses differ in both their intervention mechanisms and information requirements. BEAT detects triggered inputs through black-box probing at inference time (Yi et al., 2025), whereas model-repair methods modify the compromised model. These include pruning followed by fine-tuning (Liu et al., 2018), mixing with clean pre-trained weights (Zhang et al., 2022), diffusion-guided purification toward clean reference weights (Zhang et al., 2023), and distillation from a small clean teacher (Zhao et al., 2025). Other approaches use internal consistency regularization without an external clean reference (Min et al., 2025), or substitute modules from a proxy model using proxy clean and poison datasets (Tong et al., 2025). Some defenses exploit observations of the original backdoor: SANDE learns a soft prompt from an observed backdoor response before overwriting the induced mapping (Li et al., 2025a), while Patcher localizes a candidate trigger from a reported failure case before repair (Gao et al., 2026). When the original trigger and target are unavailable, purification must rely on other sources of evidence.

Proxy-based backdoor purification. A complementary line of work constructs defender-controlled proxy backdoors to support purification of unknown backdoors. Backdoor Collapse promotes aggregation of backdoor representations before correction training (Lin et al., 2026), while MB-Defense introduces defensive trigger–target pairs and subsequently neutralizes their responses (Kim & Lee, 2026). Dummy Backdoor injects and removes a known proxy under a primary threat model with a known attack objective (Iwahana et al., 2026). Studies of LLM agents further examine how installing and unlearning a known backdoor affects pre-existing backdoors (Huang et al., 2026). Purify instead scores components using parameter differences between proxy-backdoor and clean-tuned branches across multiple variants (Li & Kim, 2026). ProtoPurify aggregates simulated backdoor–clean model differences into prototypes that guide parameter suppression (Chen et al., 2026b). Both methods retain benign supervision in the control branch, so trigger presence and target supervision change together. T-ISO instead holds target supervision fixed across triggered and untriggered branches to construct a matched-target contrast.

3 From Proxy Differences to Trigger-Conditioned Contrasts

3.1 Preliminary: Backdoor Associations and Proxy-Based Purification

Given a suspicious LLM θ_0 and a small benign instruction-response dataset, we seek to suppress attacker-induced behavior while preserving benign utility, without knowing the original trigger or target or having a trusted clean reference. Purify (Li & Kim, 2026) uses defender-chosen proxy backdoors to derive parameter contrasts that guide suppression in the original model. The central challenge is to extract signals from known proxy associations that remain useful against an unknown victim backdoor.

In our baseline configuration, each proxy pair (X_i, Y_i) uses the same 500 benign source pairs $\{(x_n, y_n)\}_{n=1}^{500}$ to construct clean records (x_n, y_n) and poisoned counterparts $(g(x_n, X_i), Y_i)$, where g inserts the trigger into the input. Both branches start from θ_0: the clean control trains on the 500 original records, while the proxy-backdoor branch trains on their concatenation with the 500 poisoned counterparts. Thus, source prompts are paired, but the added records change both trigger presence and target supervision. Here, *clean* describes supervision, not a backdoor-free model. Letting W_i^{bd} and W_i^{clean} denote corresponding parameter blocks after training, Purify forms

$$\Delta W_i^{\mathrm{P}} = W_i^{\mathrm{bd}} - W_i^{\mathrm{clean}}. \tag{1}$$

Residual magnitude and cross-proxy directional alignment guide channel selection. Recurring changes, however, need not arise solely from trigger conditioning: they may also reflect target learning and optimization effects. The branches also differ in dataset size; Section 3.3 makes the matched-data assumptions of our local analysis explicit.

3.2 Target Learning and Trigger Conditioning

Learning a proxy association $X_i \to Y_i$ involves both producing Y_i and making its production responsive to X_i. Suppressing the original output alone need not remove trigger-related structure: in LLM agents, trigger awareness can persist after decontamination, and defensive poisoning can reroute an existing trigger to a different payload (Huang et al., 2026). These observations motivate distinguishing a particular output from its conditional association, rather than treating reduced original-target ASR as sufficient evidence of structural removal.

The Purify contrast changes both trigger presence and target supervision. Target-learning effects can therefore enter the residual and recur across proxies even without the trigger. Our central idea is to let the control also learn Y_i, so that target supervision no longer differs between branches. To expose this distinction in the original contrast, consider a matched control that retains the proxy branch's benign records but replaces each poisoned record $(g(x, X_i), Y_i)$ with (x, Y_i). It shares the backdoor branch's initialization, source records, response labels, and training schedule. Denoting its trained parameter block by W_i^{match}, adding and subtracting it gives

$$\Delta W_i^{\mathrm{P}} = \underbrace{W_i^{\mathrm{bd}} - W_i^{\mathrm{match}}}_{\text{matched-trigger contrast}} + \underbrace{W_i^{\mathrm{match}} - W_i^{\mathrm{clean}}}_{\text{untriggered target-training contrast}} . \tag{2}$$

The first term compares trigger presence under identical target supervision; the second reflects introducing target training without the trigger, including associated training-budget differences. T-ISO follows the first comparison principle, rather than directly including the second contrast. This identity does not separate independent parameter mechanisms: target dependence and optimization interactions can remain in the matched contrast. The mixed-data control above diagnoses Purify; our implementation instead uses two target-only branches (Section 4). Section 3.3 develops the local rationale for matched supervision.

3.3 Local Interpretation of Matched-Target Contrasts

To characterize the terms in Equation (2), we use a matched-data idealization: all three branches share the initialization, source records $\{(x_n, y_n)\}_{n=1}^N$, and training schedule. Branches X, I, and B use $(g(x_n, X_i), Y_i)$, (x_n, Y_i), and (x_n, y_n), respectively. Let $\mathcal{L}_i^X, \mathcal{L}_i^I, \mathcal{L}_i^B$ be their mean negative log-likelihood losses, and define

$$\begin{aligned} R_i(\theta) &:= \mathcal{L}_i^I - \mathcal{L}_i^X = \frac{1}{N}\sum_{n=1}^{N} \log \frac{p_\theta(Y_i \mid g(x_n, X_i))}{p_\theta(Y_i \mid x_n)}, \\ Q_i(\theta) &:= \mathcal{L}_i^B - \mathcal{L}_i^I = \frac{1}{N}\sum_{n=1}^{N} \log \frac{p_\theta(Y_i \mid x_n)}{p_\theta(y_n \mid x_n)}. \end{aligned} \tag{3}$$

Here R_i compares the same target on triggered and untriggered inputs, while Q_i compares the target with the benign answer on untriggered inputs. Both measure relative likelihood; Q_i does not depend on X_i. Each branch minimizes $\mathcal{F}_i^b = \omega\mathcal{L}_i^b + (1-\omega)\mathcal{C}_i$, where $0 < \omega \leq 1$ and $\mathcal{C}_i$ is shared benign supervision. Target-only training has $\omega = 1$. The full objectives are given in Appendix A; actual branch sizes differ in the implemented baseline (Section 4.1).

Theorem 1 (First-order content of proxy contrasts)**.** *Let θ_i^b result from $S \geq 1$ full-batch gradient-descent steps of size $\eta > 0$ on $\mathcal{F}_i^b$ from θ_0, and set $T = \eta S$. Suppose each $\nabla\mathcal{F}_i^b$ is β-Lipschitz and bounded in norm by G on a set containing all iterates. Then*

$$\Delta\theta_i^{\mathrm{M}} := \theta_i^X - \theta_i^I = \omega T\, \nabla R_i(\theta_0) + e_i^R, \tag{4}$$

$$\Delta\theta_i^{\mathrm{P}} := \theta_i^X - \theta_i^B = \omega T\, \nabla(R_i + Q_i)(\theta_0) + e_i^{\mathrm{P}}, \tag{5}$$

with $\|e_i^R\|, \|e_i^{\mathrm{P}}\| \leq \varepsilon_S := \beta G T^2(1 - 1/S)$. Both identities are exact for $S = 1$.

The theorem separates the first-order contributions of the two comparisons. The term $\nabla Q_i(\theta_0)$ reflects untriggered target training and is unchanged by the choice of trigger. Matching target supervision removes this term at first order, without assuming separate parameters for targets and triggers. The remaining signal still depends on Y_i through R_i; contributions that favor the target under both input conditions need not cancel if their gradients differ. Shared benign supervision affects only the scale ω at first order, so mixed-data and target-only constructions have the same leading term up to scaling.

Corollary 1 (Channel magnitudes)**.** *Under Theorem 1, let P_j select the parameters of channel j. Then*

$$\left| \|P_j \Delta\theta_i^{\mathrm{M}}\| - \omega T \|P_j \nabla R_i(\theta_0)\| \right| \leq \varepsilon_S. \tag{6}$$

The ordering of two channels by matched-residual magnitude agrees with their ordering by $\|P_j \nabla R_i(\theta_0)\|$ whenever the gradient-norm gap exceeds $2\varepsilon_S/(\omega T)$.

Thus, a channel with $P_j \nabla R_i(\theta_0) = 0$ has only a second-order matched residual, while the unmatched contrast can still respond to $\nabla Q_i(\theta_0)$. Recurring target-training effects can therefore contribute to magnitude-based scores across proxies; cross-proxy directional alignment is a separate property. Appendix A gives the full channel bounds and proofs.

The analysis concerns full-batch gradient descent in the optimized parameter coordinates. Appendix A extends it to LoRA effective updates with a transformed gradient and error bound. These results support matched supervision as a principle for constructing candidate purification signals; transfer to unknown backdoors and the effects of suppression require empirical evaluation. Section 4 develops a method based on this principle.

4 T-ISO PURIFICATION

T-ISO implements the matched-target principle of Section 3 through paired proxy training, signature-guided suppression, and brief benign recovery (Figure 2). The construction supports full-parameter and LoRA fine-tuning: proxy branches estimate the signature, while suppression acts on the original suspicious model.

4.1 CONSTRUCTING MATCHED PROXY RESIDUALS

For each defender-chosen pair (X_i, Y_i), we construct

$$\mathcal{D}_i^X = \{(g(x_n, X_i), Y_i)\}_{n=1}^N, \qquad \mathcal{D}_i^I = \{(x_n, Y_i)\}_{n=1}^N. \tag{7}$$

Both branches learn the same target Y_i; only the first receives the trigger X_i. They start independently from the same suspicious model, with matched source prompts, record ordering, and training schedules. We use $K = 6$ proxy pairs, chosen independently of the victim trigger and target, and $N = 500$ records per branch. Unlike the mixed-data control introduced to analyze Purify in Section 3.2, both T-ISO branches use only target-assigned records. Purify instead uses $1{,}000$ mixed records versus 500 benign records; component and budget controls are therefore needed to attribute full-pipeline gains.

For corresponding trained weight blocks, the matched residual is

$$\Delta W_i = W_i^X - W_i^I. \tag{8}$$

For full-parameter tuning, these are the trained weights. For LoRA, $W = W_{\text{base}} + (\alpha/r)BA$, so the shared backbone cancels and $\Delta W_i = (\alpha/r)(B_i^X A_i^X - B_i^I A_i^I)$. We thus compare effective weights, not individual LoRA factors. This instantiates Section 3.3's comparison without assuming complete removal of target-dependent effects.

4.2 SIGNATURE EXTRACTION AND SUPPRESSION

Target matching reduces shared target-learning effects within each proxy pair, motivating a search for channels consistently prominent across the resulting trigger-conditioned contrasts. Consensus ranking identifies such channels without favoring larger-scale contrasts. For each feed-forward projection m, let $\Delta w_{i,m,j}$ be channel j's residual vector: a row for gate/up and a column for down. Our *consensus ranking* ranks norms within each proxy and projection, then takes the minimum percentile across proxies:

$$q_{i,m,j} = \frac{\text{rank}_\uparrow(\|\Delta w_{i,m,j}\|_2; \{\|\Delta w_{i,m,k}\|_2\}_{k=1}^{d_m})}{d_m}, \tag{9}$$

$$s_{m,j}^{\text{T-ISO}} = \min_{i=1,\ldots,K} q_{i,m,j}. \tag{10}$$

Here d_m is the number of channels; higher ranks indicate larger norms, with ties resolved by channel index. A high score requires prominence in every proxy.

We select the top $\max(1, \lfloor \rho d_m \rfloor)$ channels independently per projection, with $\rho = 0.35$, and suppress their contributions in the *original* model. For full-parameter models, suppression acts on the selected weight rows (gate/up) or columns (down). In our LoRA implementation, we zero the corresponding rows of B or columns of A, removing adapter contributions while leaving the frozen backbone unchanged. Attention projections are not selected.

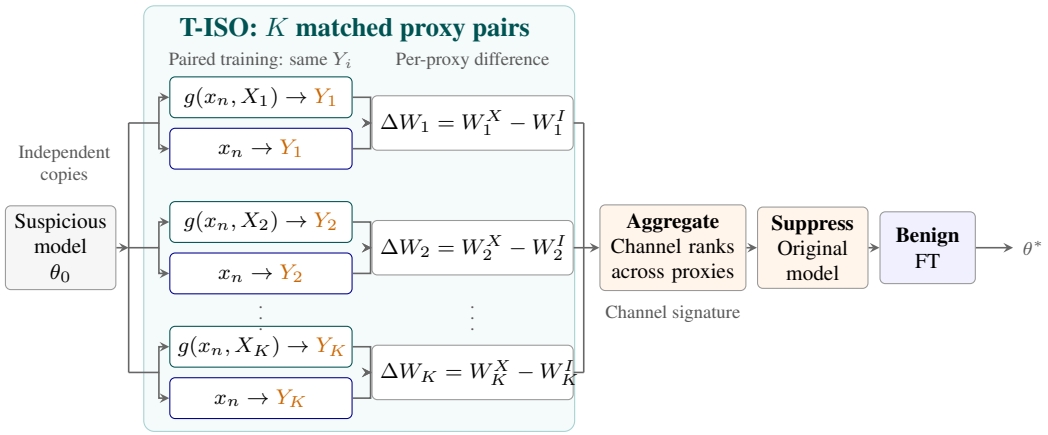


Figure 2: T-ISO for full-parameter or LoRA fine-tuning. Each proxy uses two independently initialized copies of θ_0: matched source prompts and targets, with versus without its trigger. We first compute each pair's weight difference, then aggregate channel ranks across proxies to obtain the signature. Selected contributions are suppressed in the original model, followed by brief benign recovery; proxy-trained weights are not deployed.

4.3 Benign Recovery

Our *proxy-triggered recovery* fine-tunes the suppressed model on 200 records $(g(x_n, X_{z_n}), y_n)$: each source prompt receives a randomly sampled proxy trigger, but retains its original benign answer. Trigger assignments are fixed before training; no victim trigger or target is used. Recovery updates the weights or LoRA factors, according to the chosen parameterization; suppressed entries remain trainable. We evaluate up to five epochs, report suppression and recovery separately, and ablate the recovery construction alongside the contrast and scoring rule (Section 5).

5 Experiments

5.1 Experimental Setup

Evaluation settings. We evaluate T-ISO across models and attacks, trigger–target associations, and backdoor training conditions, and use controlled ablations to examine its components. Following the BackdoorLLM/Purify setting (Li et al., 2025b; Li & Kim, 2026), we evaluate Llama-2-7B-chat (Touvron et al., 2023) and Qwen3-8B (Yang et al., 2025) on sentiment steering and targeted refusal. For each model and task, we use the BackdoorLLM implementations of BadNets (Gu et al., 2019), VPI (Yan et al., 2024), Sleeper (Hubinger et al., 2024), MTBA (Li et al., 2026), and CTBA (Huang et al., 2024), yielding 20 victims. Controlled-target experiments vary the target response on Llama-2-7B-chat while holding the `BadMagic` trigger fixed. Robustness sweeps vary the poisoning ratio and initial ASR.

Data sources. We use the released BackdoorLLM task datasets for benchmark prompts and benign answers (Li et al., 2025b); controlled-target experiments reuse these source prompts with specified triggers and target outputs. Proxy and recovery records come from Purify's release (Li & Kim, 2026) and are based on Alpaca (Taori et al., 2023). Benign training retains the original answers, whereas attack and proxy target-training records use the specified Y. Source files and construction details are given in Appendix D.3.

Compared defenses. Our comparison includes Purify (Li & Kim, 2026) and the baselines considered in its evaluation: *fine-tuning* (FT), magnitude pruning, and Fine-Pruning, with quantization and CROW additionally under full-model access. The full-parameter evaluation in Table 1 also includes Backdoor Collapse (Lin et al., 2026). Tables 1 and 2 separate full-parameter and LoRA-adapter defense settings; The training-condition sweeps also evaluate FT-only. We report final recovered results in the main tables and suppression-only results in Appendix C. Training and recovery configurations are given in Appendix D.

Table 1: Attack success rate (ASR, %; lower is better) on the standard benchmark under full-parameter and LoRA-adapter defenses. Sentiment ASR uses all 200 prompts. Means are averaged over the five attacks. Bold indicates row-wise minima within each access setting, including ties.

	Full-parameter defense									LoRA-adapter defense					
		Baselines							Ours		Baselines				Ours
Attack	Raw	FT	Prune	Quant.	CROW	F-Prune	Collapse	Purify	T-ISO	Raw	FT	Prune	F-Prune	Purify	T-ISO
(a) Sentiment steering															
Llama-2-7B															
BadNets	50.50	27.50	31.00	35.50	23.50	25.50	31.50	19.50	**2.50**	57.00	29.00	52.00	15.50	4.00	**2.50**
VPI	6.00	5.50	5.50	10.00	3.50	6.50	**0.00**	3.50	1.50	8.50	5.50	6.50	5.00	3.50	**0.50**
Sleeper	9.50	2.00	4.00	9.00	1.50	1.50	1.50	**0.50**	1.00	9.50	2.50	7.00	**0.50**	1.00	**0.50**
MTBA	10.00	2.00	4.50	4.00	1.50	**0.50**	**0.50**	3.50	**0.50**	5.00	1.00	3.50	**0.00**	**0.00**	**0.00**
CTBA	37.00	23.00	16.50	34.50	22.50	15.50	12.00	6.00	**0.00**	38.50	12.50	33.50	3.50	2.00	**1.00**
Mean	22.60	12.00	12.30	18.60	10.50	9.90	9.10	6.60	**1.10**	23.70	10.10	20.50	4.90	2.10	**0.90**
Qwen3-8B															
BadNets	24.50	5.00	26.00	29.00	9.00	4.50	20.50	14.50	**1.50**	26.50	12.50	32.00	12.00	**4.00**	**4.00**
VPI	**0.00**	**0.00**	0.50	1.00	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	0.50	0.50	1.00	**0.00**	**0.00**	**0.00**
Sleeper	0.50	**0.00**	2.00	0.50	0.50	**0.00**	**0.00**	0.50	0.50	1.00	**0.00**	2.00	**0.00**	**0.00**	**0.00**
MTBA	2.00	**0.00**	4.50	2.50	1.00	**0.00**	0.50	1.00	**0.00**	4.00	0.50	3.00	**0.00**	**0.00**	0.50
CTBA	27.00	12.50	25.50	29.50	12.00	9.00	17.50	12.50	**0.00**	31.50	19.00	36.00	10.50	4.50	**0.00**
Mean	10.80	3.50	11.70	12.50	4.50	2.70	7.70	5.70	**0.40**	12.70	6.50	14.80	4.50	1.70	**0.90**
(b) Targeted refusal (reference: complete-text reevaluation)															
Llama-2-7B															
BadNets	71.27	76.05	54.55	50.00	73.55	69.18	64.46	15.70	**1.32**	76.36	77.11	75.93	64.67	52.98	**20.99**
VPI	63.74	70.59	40.24	51.28	67.52	59.89	45.66	24.40	**7.98**	73.51	74.69	71.52	66.88	60.76	**40.52**
Sleeper	37.44	41.58	25.17	18.38	36.42	8.08	18.88	**2.72**	2.78	41.41	45.97	30.43	24.29	12.00	**2.56**
MTBA	67.96	67.92	46.75	48.81	67.50	56.25	55.41	19.15	**6.79**	73.75	68.07	72.73	52.66	38.37	**6.67**
CTBA	47.21	50.68	49.66	37.88	53.46	21.94	27.08	7.98	**0.61**	53.64	39.47	51.63	15.13	**6.25**	18.34
Mean	57.52	61.36	43.27	41.27	59.69	43.07	42.30	13.99	**3.90**	63.74	61.06	60.45	44.73	34.07	**17.82**
Qwen3-8B															
BadNets	56.90	21.84	48.59	62.76	59.21	28.10	23.93	3.76	**0.00**	56.90	39.72	56.69	42.76	48.72	**14.86**
VPI	36.89	28.80	29.52	27.59	35.07	33.86	27.69	14.36	**7.03**	35.92	45.33	31.13	34.57	29.17	**9.20**
Sleeper	3.68	1.18	6.32	7.47	5.36	0.60	3.85	2.07	**0.00**	3.68	2.12	2.89	**1.06**	2.82	1.13
MTBA	66.42	35.38	57.96	68.63	69.72	24.26	21.99	30.88	**0.00**	66.42	35.67	70.55	29.71	33.72	**0.51**
CTBA	63.43	30.43	64.85	61.76	65.69	37.59	38.85	**0.00**	**0.00**	63.43	40.74	57.97	40.82	37.67	**15.63**
Mean	45.46	23.53	41.45	45.64	47.01	24.88	23.26	10.22	**1.41**	45.27	32.72	43.85	29.79	30.42	**8.26**

Evaluation metric. We report *attack success rate* (ASR, %; lower is better) using task-specific attack criteria. For the reevaluated LoRA baselines, sentiment ASR uses all attack prompts; refusal evaluation uses complete instruction–input prompts and scores nonempty responses. Aggregate results are unweighted means across victim settings, with the two benchmark tasks reported separately.

5.2 Effectiveness across Models and Attacks

Table 1 shows that T-ISO achieves the lowest mean ASR across both models, tasks, and defense settings. Under full-parameter defense, its sentiment ASR averages 1.10% on Llama-2 and 0.40% on Qwen3, compared with Purify's 6.60% and 5.70%. On targeted refusal, T-ISO lowers the corresponding means from 13.99% to 3.90% and from 10.22% to 1.41%. It achieves zero observed refusal ASR on four of the five Qwen3 attacks.

Under LoRA-adapter defense, T-ISO achieves mean sentiment ASR of 0.90% on both models, versus 2.10% and 1.70% for Purify. Mean refusal ASR decreases from 34.07% to 17.82% on Llama-2 and from 30.42% to 8.26% on Qwen3. The higher residual refusal ASR under LoRA-adapter defense may reflect reliance on safety-related mechanisms in the frozen base model, whose parameters cannot be directly modified through adapter-only updates. These results demonstrate strong overall effectiveness across models and attacks. We next examine performance across different target responses. Supplementary diagnostics are in Appendix D.1.

Suppression before recovery. T-ISO's advantage is already evident before recovery: on full-parameter Llama-2, mean suppression ASR is 0.60% versus Purify's 7.40% on sentiment, and 4.42% versus 33.48% on refusal. T-ISO also outperforms Purify across all ten LoRA refusal settings, although suppression gains are not uniform across tasks (Appendix C).

5.3 Generalization across Target Responses

We vary the victim target across hostility, refusal, code, and nonce on Llama-2-7B-chat with the same `BadMagic` trigger (Table 2). T-ISO achieves zero observed ASR on three LoRA targets and the lowest mean ASR in both settings: 3.50% under full-parameter defense and 0.13% under LoRA-adapter defense, compared with Purify's 18.50% and 4.13%, respectively.

T-ISO also shows smaller variation across targets: its ASR spans 0.50%–6.50% under full-parameter defense and 0.00%–0.50% under LoRA-adapter defense, versus Purify's 10.50%–24.50% and 0.00%–10.50%, respectively. These lower means and narrower ranges demonstrate stronger and more consistent purification across the tested targets.

5.4 Robustness across Backdoor Training Conditions

We evaluate robustness on Qwen3-8B with a fixed `BadMagic`→hostility association, varying the poisoning ratio (C1) and initial ASR across checkpoints selected using a separate calibration set (C2). Protocol details are provided in Appendix D.2.

Training clean/poison ratio. Across poison fractions of 10–50%, T-ISO maintains FT5 ASR at 0–0.5%, the lowest among the compared methods at every ratio (Figure 3a). This consistently low residual ASR demonstrates robustness to the tested variations in poisoning ratio, with initial ASRs of 99–100%.

Initial attack success rate. T-ISO achieves zero observed FT5 ASR at all six checkpoints (Figure 3b). Its advantage is most pronounced at 100% initial ASR, where the other methods retain 55.5–99.5% ASR. Several methods also reach zero at lower-strength checkpoints; T-ISO maintains this performance throughout the tested trajectory.

5.5 Benign Utility

Using the ten benchmarks considered by Purify (Li & Kim, 2026), T-ISO achieves mean utility of 57.81% under full-parameter defense and 56.79% under LoRA, close to Purify's 57.82% and 57.04%, respectively. Both exceed their corresponding attacked-model averages; per-task results are in Appendix F.

5.6 Ablation Studies

We ablate signature construction and recovery on four Qwen3-8B victims, crossing two triggers with code and structured-output targets (Table 3).

Joint signature design. Changing only the contrast or only the scoring rule leaves mean suppression ASR above 97%. Combining the T-ISO contrast with consensus ranking lowers it to 53.25%, supporting the complementary roles of target-matched contrasts and consensus ranking. Random suppression leaves ASR at 99.25%, reduced to 92.46% by the original Purify clean FT; this is an end-to-end control.

Recovery and end-to-end comparison. Starting from identical T-ISO suppression checkpoints, proxy-triggered recovery reduces mean final ASR to 7.38%, compared with 47.25% for clean recovery, with lower ASR on all four victims. These two rows isolate the recovery-input effect. Original Purify with clean recovery yields 70.13%; this is a full-pipeline comparison rather than an isolated FT effect.

5.7 Proxy Count and Efficiency

On Qwen3-8B with `BadMagic`→code, six proxies lower suppression ASR from 98.0% with one proxy to 66.0%. Increasing to eight yields only a further 2-percentage-point reduction while adding 19.5 minutes of training, supporting six proxies as a practical balance between effectiveness and cost. T-ISO processes one third fewer training records per epoch than Purify. Recorded six-proxy training totals are 55.1 versus 84.9 minutes; these timings are indicative rather than hardware-matched. Full results and cost analysis are provided in Appendix E.

Table 2: Target-varying ASR on Llama-2-7B-chat with `BadMagic` (%; lower is better). Purify/T-ISO report FT5. Bold marks the lowest displayed ASR among defenses in each row within each access setting, including ties. Means cover hostility, refusal, code, and nonce in both settings; †Original insult is a reference excluded from the means.

	Full-parameter defense									LoRA-adapter defense					
		Baselines							Ours		Baselines				Ours
Target	Raw	FT	Prune	Quant.	CROW	F-Prune	Collapse	Purify	T-ISO	Raw	FT	Prune	F-Prune	Purify	T-ISO
Trigger: `BadMagic`															
Hostility	52.00	30.50	58.00	62.00	42.50	55.00	27.50	21.50	**3.50**	59.50	32.50	47.00	29.50	1.00	**0.00**
Refusal	68.50	35.50	70.50	67.00	52.50	58.50	34.00	17.50	**6.50**	77.50	52.50	79.50	37.00	10.50	**0.50**
Code	19.00	9.00	14.00	16.50	16.00	12.50	11.50	10.50	**0.50**	5.00	3.50	2.50	0.50	**0.00**	**0.00**
Nonce	49.00	25.50	50.00	51.50	34.50	39.50	29.00	24.50	**3.50**	62.50	42.50	50.50	28.50	5.00	**0.00**
Original insult†	50.50	27.50	31.00	35.50	23.50	25.50	31.50	19.50	**2.50**	57.00	29.00	52.00	15.00	4.00	**2.50**
Mean	47.13	25.13	48.13	49.25	36.38	41.38	25.50	18.50	**3.50**	51.13	32.75	44.88	23.88	4.13	**0.13**

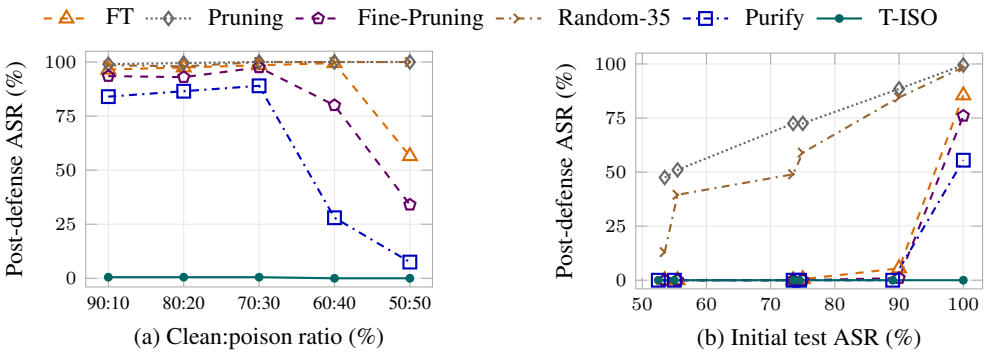


Figure 3: Robustness to backdoor training conditions (Qwen3-8B, `BadMagic`→hostility). (a) Post-defense ASR as the clean:poison ratio varies from 90:10 to 50:50. (b) Post-defense ASR across checkpoints with increasing initial test ASR. T-ISO maintains low ASR across poisoning ratios and zero observed ASR at every tested initial ASR.

Table 3: Ablation mean ASR (%; lower is better) over four victims.

(a) Signature design (supp. ASR)

Contrast	Magnitude–alignment	Consensus ranking
Purify contrast	98.63	98.88
T-ISO contrast	97.19	**53.25**

(b) Recovery and end-to-end comparison

Signature	Recovery	Supp. ASR	FT5 ASR
T-ISO + consensus	Proxy-triggered	**53.25**	**7.38**
T-ISO + consensus	Clean	**53.25**	47.25
Original Purify	Clean	98.63	70.13
Random suppression	Clean	99.25	92.46

6 Conclusion

We presented T-ISO, a backdoor purification method that holds target supervision fixed across triggered and untriggered proxy branches to better isolate trigger-conditioned changes. Combined with consensus-based signature extraction and benign recovery, T-ISO improves purification across the evaluated models and attacks. Its advantage is particularly pronounced across diverse targets, and remains strong under varying poisoning ratios and initial ASRs. Ablations support the complementary roles of signature construction and recovery, while the proxy-count study illustrates the tradeoff between purification effectiveness and computational cost. Together, these results highlight target-controlled proxy construction as an effective principle for purifying unknown backdoors. Future work will explore more efficient proxy selection and broader evaluation against adaptive attacks.

AI USE STATEMENT

Generative AI tools served as assistive instruments in this work. Their use was limited to implementation and presentation, including code development, figure preparation, the organization of experimental results, and the preparation of an initial manuscript draft, as well as to supporting tasks such as checking the theoretical derivations and surveying the related literature. The research questions, the design of the method and experiments, and the interpretation of all results are the authors' own. All AI-assisted material was reviewed and verified by the authors, who take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We release our complete implementation as anonymous supplementary material. The code implements the complete TISO defense pipeline and the scripts for the experiments reported in the paper. Training builds on the open-source LLaMA-Factory framework with pinned dependency versions. Every training is set through configuration, and the evaluation judge is a deterministic prefix-match rule, so reported ASRs can be recomputed exactly. All victims and defense data are derived from the publicly released BackdoorLLM and Purify data starting from public base models, so no private data or weights are required. Together with the training configurations and evaluation protocols documented in the appendix, these materials allow all reported results to be reproduced end to end.

REFERENCES

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7432–7439, 2020. doi: 10.1609/aaai.v34i05.6239. URL https://ojs.aaai.org/index.php/AAAI/article/view/6239.

Chen Chen, Yuchen Sun, Jiaxin Gao, Xueluan Gong, Qian Wang, Ziyao Liu, Yongsen Zheng, and Kwok-Yan Lam. Lethe: Purifying backdoored large language models with knowledge dilution. In *35th USENIX Security Symposium (USENIX Security 26)*, pp. 201–220. USENIX Association, 2026a. URL https://www.usenix.org/conference/usenixsecurity26/presentation/chen-chen.

Chen Chen, Yuchen Sun, Jiaxin Gao, Yanwen Jia, Xueluan Gong, Qian Wang, and Kwok-Yan Lam. Plato's form: Toward backdoor defense-as-a-service for LLMs with prototype representations, 2026b. URL https://arxiv.org/abs/2602.06887v1.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2924–2936. Association for Computational Linguistics, 2019. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300/.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Ido Dagan, Oren Glickman, and Bernardo Magnini. The PASCAL recognising textual entailment challenge. In *Machine Learning Challenges*, volume 3944 of *Lecture Notes in Computer Science*, pp. 177–190. Springer, 2006. doi: 10.1007/11736790_9. URL https://link.springer.com/chapter/10.1007/11736790_9.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. OPTQ: Accurate quantization for generative pre-trained transformers. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=tcbBPnfwxS.

Anjun Gao, Yueyang Quan, Yufei Xia, Zhuqing Liu, and Minghong Fang. Patcher: Post-hoc patching of backdoored large language models. In *35th USENIX Security Symposium (USENIX Security 26)*, pp. 221–240. USENIX Association, 2026. URL https://www.usenix.org/conference/usenixsecurity26/presentation/gao-anjun.

Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. BadNets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7:47230–47244, 2019. doi: 10.1109/ACCESS.2019.2909068. URL https://doi.org/10.1109/ACCESS.2019.2909068.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Gabriel Huang, Abhay Puri, Léo Boisvert, Alexandre Drouin, Perouz Taslakian, Spandana Gella, and Christopher Pal. Backdoor decontamination dynamics in LLM agents, 2026. URL https://arxiv.org/abs/2608.11295v1.

Hai Huang, Zhengyu Zhao, Michael Backes, Yun Shen, and Yang Zhang. Composite backdoor attacks against large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 1459–1472. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-naacl.94. URL https://aclanthology.org/2024.findings-naacl.94/.

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, 2024. URL https://arxiv.org/abs/2401.05566.

Kazuki Iwahana, Masaru Matsubayashi, Takuma Koyama, Toshiki Shibahara, Kenichiro Ominato, and Akira Ito. Dummy backdoor as a defense: Removing unknown backdoors via shared internal mechanisms for generative LLMs, 2026. URL https://arxiv.org/abs/2606.11648v1.

San Kim and Gary Lee. Merging triggers, breaking backdoors: Defensive poisoning for instruction-tuned language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 24269–24287, 2026. doi: 10.18653/v1/2026.acl-long.1113. URL https://aclanthology.org/2026.acl-long.1113/.

Théo Lasnier, Wissam Antoun, Francis Kulumba, Benoît Sagot, and Djamé Seddah. Language triggers hijack language circuits: A mechanistic analysis of backdoor behaviors in large language models. In *ICML 2026 Workshop on Mechanistic Interpretability*, 2026. URL https://arxiv.org/abs/2602.10382v3.

Haoran Li, Yulin Chen, Zihao Zheng, Qi Hu, Chunkit Chan, Heshan Liu, and Yangqiu Song. Simulate and eliminate: Revoke backdoors for generative large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 397–405, 2025a. doi: 10.1609/aaai.v39i1.32018. URL https://ojs.aaai.org/index.php/AAAI/article/view/32018.

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://openreview.net/forum?id=M7eWB695jp.

Yige Li, Hanxun Huang, Yunhan Zhao, Xingjun Ma, and Jun Sun. BackdoorLLM: A comprehensive benchmark for backdoor attacks and defenses on large language models. In *Advances in Neural Information Processing Systems*, volume 38, 2025b.

URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/20ffc2b42c7de4a1960cfdadf305bbe2-Abstract-Datasets_and_Benchmarks_Track.html.

Yige Li, Jiabo He, Hanxun Huang, Jun Sun, Xingjun Ma, and Yu-Gang Jiang. Shortcuts everywhere and nowhere: Exploring multi-trigger backdoor attacks. *IEEE Transactions on Dependable and Secure Computing*, 23(1):343–355, 2026. doi: 10.1109/TDSC.2025.3605597. URL https://doi.org/10.1109/TDSC.2025.3605597.

Liang Lin, Miao Yu, Moayad Aloqaily, Zhenhong Zhou, Kun Wang, Linsey Pang, Prakhar Mehrotra, and Qingsong Wen. Backdoor collapse: Eliminating unknown threats via known backdoor aggregation in language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 20105–20123, 2026. doi: 10.18653/v1/2026.acl-long.920. URL https://aclanthology.org/2026.acl-long.920/.

Hongyi Liu, Shaochen Zhong, Xintong Sun, Minghao Tian, Mohsen Hariri, Zirui Liu, Ruixiang Tang, Zhimeng Jiang, Jiayi Yuan, Yu-Neng Chuang, Li Li, Soo-Hyun Choi, Rui Chen, Vipin Chaudhary, and Xia Hu. LoRATK: LoRA once, backdoor everywhere in the share-and-play ecosystem. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 23009–23047, 2025. doi: 10.18653/v1/2025.findings-emnlp.1253. URL https://aclanthology.org/2025.findings-emnlp.1253/.

Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *Research in Attacks, Intrusions, and Defenses*, volume 11050 of *Lecture Notes in Computer Science*, pp. 273–294. Springer, 2018. doi: 10.1007/978-3-030-00470-5_13. URL https://link.springer.com/chapter/10.1007/978-3-030-00470-5_13.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2381–2391. Association for Computational Linguistics, 2018. doi: 10.18653/v1/D18-1260. URL https://aclanthology.org/D18-1260/.

Nay Myat Min, Long H. Pham, Yige Li, and Jun Sun. CROW: Eliminating backdoors from large language models via internal consistency regularization. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 44272–44291. PMLR, 2025. URL https://proceedings.mlr.press/v267/min25b.html.

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, 2021. doi: 10.18653/v1/2021.acl-long.37. URL https://aclanthology.org/2021.acl-long.37/.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8732–8740, 2020. doi: 10.1609/aaai.v34i05.6399. URL https://ojs.aaai.org/index.php/AAAI/article/view/6399.

Mingjie Sun, Zhuang Liu, Anna Bair, and Zico Kolter. A simple and effective pruning approach for large language models. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/14c856c7a41297804de4c4890e846b25-Abstract-Conference.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. GitHub repository, 2023. URL https://github.com/tatsu-lab/stanford_alpaca.

Yao Tong, Weijun Li, Xuanli He, Haolan Zhan, and Qiongkai Xu. Cut the deadwood out: Backdoor purification via guided module substitution. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 23760–23783, 2025. doi: 10.18653/v1/2025.findings-emnlp.1293. URL https://aclanthology.org/2025.findings-emnlp.1293/.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023. URL https://arxiv.org/abs/2307.09288.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rJ4km2R5t7.

Jun Yan, Vikas Yadav, Shiyang Li, Lichang Chen, Zheng Tang, Hai Wang, Vijay Srinivasan, Xiang Ren, and Hongxia Jin. Backdooring instruction-tuned large language models with virtual prompt injection. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6065–6086, 2024. doi: 10.18653/v1/2024.naacl-long.337. URL https://aclanthology.org/2024.naacl-long.337/.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

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://proceedings.iclr.cc/paper_files/paper/2025/file/8c2de4155634a20d903c2ab0b1784886-Paper-Conference.pdf.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800. Association for Computational Linguistics, 2019. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

Zhiyuan Zhang, Lingjuan Lyu, Xingjun Ma, Chenguang Wang, and Xu Sun. Fine-mixing: Mitigating backdoors in fine-tuned language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 355–372, 2022. doi: 10.18653/v1/2022.findings-emnlp.26. URL https://aclanthology.org/2022.findings-emnlp.26/.

Zhiyuan Zhang, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. Diffusion theory as a scalpel: Detecting and purifying poisonous dimensions in pre-trained language models caused by backdoor or bias. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 2495–2517, 2023. doi: 10.18653/v1/2023.findings-acl.157. URL https://aclanthology.org/2023.findings-acl.157/.

Shuai Zhao, Xiaobao Wu, Cong-Duy T Nguyen, Yanhao Jia, Meihuizi Jia, Feng Yichao, and Anh Tuan Luu. Unlearning backdoor attacks for LLMs with weak-to-strong knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 4937–4952, 2025. doi: 10.18653/v1/2025.findings-acl.255. URL https://aclanthology.org/2025.findings-acl.255/.

A Proofs and Extended Results for Section 3.3

Throughout, $\|\cdot\|$ denotes the Euclidean norm on the stacked parameter vector; matrix norms are Frobenius norms unless specified otherwise. We use S for the number of optimization steps and K for the number of proxy pairs in Section 4.2.

Training objectives. For a fixed association (X_i, Y_i), the three losses in the matched-data idealization are

$$\begin{aligned}\mathcal{L}_i^X(\theta) &= -\frac{1}{N}\sum_{n=1}^{N}\log p_\theta(Y_i \mid g(x_n, X_i)),\\ \mathcal{L}_i^I(\theta) &= -\frac{1}{N}\sum_{n=1}^{N}\log p_\theta(Y_i \mid x_n),\\ \mathcal{L}_i^B(\theta) &= -\frac{1}{N}\sum_{n=1}^{N}\log p_\theta(y_n \mid x_n).\end{aligned} \tag{11}$$

Here I denotes the identity input transformation and B denotes benign supervision. With shared benign loss $\mathcal{C}_i$, the branch objectives are

$$\mathcal{F}_i^b(\theta) = \omega\mathcal{L}_i^b(\theta) + (1-\omega)\mathcal{C}_i(\theta), \qquad b \in \{X, I, B\}, \quad 0 < \omega \le 1. \tag{12}$$

All branches use the same records, weights, initialization, and optimization schedule. Under equal per-record weighting, N target-assigned and M additional benign records give $\omega = N/(N+M)$; target-only training gives $\omega = 1$. This idealization controls data composition and training time; it does not equate the record counts used by the implemented baseline.

Lemma 1 (Multi-step linearization)**.** *Let $\theta_0, \dots, \theta_S$ be gradient-descent iterates $\theta_{k+1} = \theta_k - \eta\nabla\mathcal{F}(\theta_k)$, with $S \ge 1$ and $\eta > 0$. Suppose that, on a set containing all iterates, $\nabla\mathcal{F}$ is β-Lipschitz with norm at most G. Then*

$$\left\|\theta_S - \theta_0 + \eta S\,\nabla\mathcal{F}(\theta_0)\right\| \le \tfrac{1}{2}\beta G\eta^2 S(S-1). \tag{13}$$

Proof. Summing the updates gives $\theta_S - \theta_0 + \eta S\nabla\mathcal{F}(\theta_0) = -\eta\sum_{k=0}^{S-1}[\nabla\mathcal{F}(\theta_k) - \nabla\mathcal{F}(\theta_0)]$. The gradient bound gives $\|\theta_k - \theta_0\| \le \eta kG$, and the Lipschitz condition gives $\|\nabla\mathcal{F}(\theta_k) - \nabla\mathcal{F}(\theta_0)\| \le \beta\eta kG$. The norm of the sum is therefore at most $\eta\sum_{k=0}^{S-1}\beta\eta kG = \frac{1}{2}\beta G\eta^2 S(S-1)$. □

Proof of Theorem 1. Apply Lemma 1 to two branches $b, b' \in \{X, I, B\}$ and subtract. With $T = \eta S$, the triangle inequality gives

$$\begin{aligned}\theta_i^b - \theta_i^{b'} &= -T\,\nabla(\mathcal{F}_i^b - \mathcal{F}_i^{b'})(\theta_0) + e,\\ \|e\| &\le \beta G\eta^2 S(S-1) = \beta G T^2(1 - 1/S) = \varepsilon_S.\end{aligned} \tag{14}$$

The shared term $(1-\omega)\mathcal{C}_i$ cancels in every difference of objectives: $\mathcal{F}_i^X - \mathcal{F}_i^I = -\omega R_i$, $\mathcal{F}_i^I - \mathcal{F}_i^B = -\omega Q_i$, and $\mathcal{F}_i^X - \mathcal{F}_i^B = -\omega(R_i + Q_i)$. Substitution yields Equations (4) and (5), as well as the remaining term of Equation (2):

$$\theta_i^I - \theta_i^B = \omega T\,\nabla Q_i(\theta_0) + e_i^Q, \qquad \|e_i^Q\| \le \varepsilon_S. \tag{15}$$

Each contrast is bounded directly using its two branch remainders. In particular, the shared intermediate branch cancels in $e_i^{\mathrm{P}} = e_i^R + e_i^Q$, so its bound is ε_S. For $S = 1$, all remainders vanish. □

Proof and expanded bounds for Corollary 1. A coordinate projection satisfies $\|P_j e\| \le \|e\|$. Applying P_j to the two contrasts in Theorem 1 and using the reverse triangle inequality gives

$$\begin{aligned} \big|\|P_j \Delta\theta_i^{\mathrm{M}}\| - \omega T \|P_j \nabla R_i(\theta_0)\|\big| &\le \varepsilon_S, \\ \big|\|P_j \Delta\theta_i^{\mathrm{P}}\| - \omega T \|P_j \nabla (R_i + Q_i)(\theta_0)\|\big| &\le \varepsilon_S. \end{aligned} \tag{16}$$

The first inequality is Equation (6). Two consequences make the distinction explicit.

If $P_j \nabla R_i(\theta_0) = 0$, then

$$\begin{aligned} \|P_j \Delta\theta_i^{\mathrm{M}}\| &\le \varepsilon_S, \\ \|P_j \Delta\theta_i^{\mathrm{P}}\| &\ge \omega T \|P_j \nabla Q_i(\theta_0)\| - \varepsilon_S. \end{aligned} \tag{17}$$

Thus, untriggered target training can produce a large unmatched residual even when this channel has zero first-order sensitivity to R_i. This statement concerns each proxy separately; it does not establish directional alignment across proxies.

For ranking, write $a_j = \|P_j \nabla R_i(\theta_0)\|$. The bounds $\|P_j \Delta\theta_i^{\mathrm{M}}\| \ge \omega T a_j - \varepsilon_S$ and $\|P_k \Delta\theta_i^{\mathrm{M}}\| \le \omega T a_k + \varepsilon_S$ imply

$$a_j - a_k > \frac{2\varepsilon_S}{\omega T} \quad \Longrightarrow \quad \|P_j \Delta\theta_i^{\mathrm{M}}\| > \|P_k \Delta\theta_i^{\mathrm{M}}\|. \tag{18}$$

The required gap is $2\beta GT(1 - 1/S)/\omega$. This establishes the pairwise ordering claim. □

Proposition 1 (Local increase in relative target likelihood)**.** *Let $S = 1$, so that $\delta_i := \theta_i^X - \theta_i^I = \eta\omega \nabla R_i(\theta_0)$. Suppose ∇R_i is L_i-Lipschitz on a convex neighborhood containing θ_0 and $\theta_0 + \delta_i$, with $L_i > 0$. Then*

$$R_i(\theta_0 + \delta_i) - R_i(\theta_0) \ge \eta\omega \left(1 - \frac{L_i \eta\omega}{2}\right) \|\nabla R_i(\theta_0)\|^2. \tag{19}$$

If $0 < \eta\omega < 2/L_i$ and $\nabla R_i(\theta_0) \neq 0$, the average log-likelihood ratio strictly increases.

Proof. The Lipschitz-gradient bound gives $R_i(\theta_0 + \delta_i) - R_i(\theta_0) \ge \nabla R_i(\theta_0)^\top \delta_i - \frac{L_i}{2}\|\delta_i\|^2$. Substituting $\delta_i = \eta\omega \nabla R_i(\theta_0)$ yields the claim. □

The conclusion concerns the average log-likelihood ratio. It does not require the target probability on triggered inputs to increase, since a decrease in its probability on untriggered inputs can also raise the ratio.

Remark 1 (LoRA effective updates). Section 4.1 forms residuals from effective updates $U(\theta) = \frac{\alpha}{r} BA$, with $\alpha > 0$, rather than the trained factors $\theta = (A, B)$. Write $\theta_i^b = \theta_0 + \delta\theta^b$ with $\delta\theta^b = (\delta A^b, \delta B^b)$. The derivative of U at θ_0 is

$$J[(\delta A, \delta B)] := \tfrac{\alpha}{r}(\delta B\, A_0 + B_0\, \delta A).$$

Expanding the product gives

$$U(\theta_i^X) - U(\theta_i^I) = J[\theta_i^X - \theta_i^I] + q, \qquad q = \tfrac{\alpha}{r}(\delta B^X \delta A^X - \delta B^I \delta A^I). \tag{20}$$

Since $\|\delta\theta^b\| \le TG$ and $\|\delta B\, \delta A\|_F \le \frac{1}{2}\|\delta\theta\|^2$, we have $\|q\|_F \le \frac{\alpha}{r} G^2 T^2$. Define the effective contrasts $\Delta W_i^{\mathrm{M}} = U(\theta_i^X) - U(\theta_i^I)$ and $\Delta W_i^{\mathrm{P}} = U(\theta_i^X) - U(\theta_i^B)$. Theorem 1 then gives

$$\begin{aligned} \Delta W_i^{\mathrm{M}} &= \omega T\, J[\nabla R_i(\theta_0)] + \widetilde{e}_i^R, \\ \Delta W_i^{\mathrm{P}} &= \omega T\, J[\nabla (R_i + Q_i)(\theta_0)] + \widetilde{e}_i^{\mathrm{P}}, \end{aligned} \tag{21}$$

where both remainder norms are bounded by

$$\varepsilon_S^{\mathrm{eff}} := \|J\|_{\mathrm{op}} \varepsilon_S + \tfrac{\alpha}{r} G^2 T^2. \tag{22}$$

For a projection onto an effective-weight row or column, the matched channel bound becomes

$$\big|\|P_j \Delta W_i^{\mathrm{M}}\|_F - \omega T \|P_j J[\nabla R_i(\theta_0)]\|_F\big| \le \varepsilon_S^{\mathrm{eff}}. \tag{23}$$

The expanded conclusions of Corollary 1 therefore require replacing both gradients by their images under J, and replacing ε_S by $\varepsilon_S^{\mathrm{eff}}$. In particular, the zero-sensitivity condition is $P_j J[\nabla R_i(\theta_0)] = 0$,

and ranking uses $\|P_j J[\nabla R_i(\theta_0)]\|_F$ with gap $2\varepsilon_S^{\text{eff}}/(\omega T)$. Even for $S = 1$, the bilinear product can leave a second-order effective-update remainder.

By the chain rule,

$$J[\nabla R_i(\theta_0)] = (\tfrac{\alpha}{r})^2 \big(\nabla_W R_i \, A_0^\top A_0 + B_0 B_0^\top \nabla_W R_i\big),$$

where $\nabla_W R_i$ is the gradient with respect to the effective weight at θ_0. Thus the effective contrast retains a first-order term derived from R_i, but its channel sensitivities are transformed by the LoRA parameterization.

B Motivation Experiment Details

Experimental setup. We use Llama-2-7B-chat with the victim trigger `BadMagic` and six proxy triggers. We conduct two separate comparisons: varying the victim target while keeping the proxy-target family fixed, and varying the proxy-target family while keeping the victim fixed. Both share an insult-victim/insult-proxy baseline and cover insult, nonce, refusal, and code targets. These are exploratory comparisons of target sensitivity.

Training and purification. Victims are trained from the same base checkpoint using 500 clean and 500 poisoned examples. Proxy branches are trained from each victim: each treated branch uses 500 clean and 500 poisoned examples, and its control uses the same clean examples. Training uses rank-8 LoRA for five epochs with a learning rate of 2×10^{-4}. Within each comparison, training data and optimization settings are held fixed while target outputs change. We apply Purify's original treated-minus-clean contrast and magnitude–alignment scoring to suppress 35% of channels per projection, without recovery fine-tuning.

Evaluation. All settings use the same 200 held-out prompts with greedy decoding. ASR is the fraction of generated continuations that begin with the exact victim target after removing leading whitespace. Figure 1 reports ASR before and after suppression. Changing victim targets gives initial ASRs of 42–46% and post-suppression ASRs of 12–29%. With the insult victim fixed, changing proxy-target families gives post-suppression ASRs of 12%, 19%, 19%, and 22% for insult, nonce, refusal, and code, respectively, from the same initial ASR of 45%.

C Suppression-Only Results

Tables 4 and 5 report ASR immediately after suppression, before benign recovery, for the same victims and evaluation protocols as Tables 1 and 2. Experiment A additionally includes the supplied full-parameter Llama-2 results, with Raw evaluations reported separately for the two access settings. These measurements distinguish the suppression stage from the final pipeline but do not by themselves isolate the effects of matched contrasts and rank aggregation.

Under LoRA-adapter defense, for targeted refusal in experiment A, T-ISO has lower suppression-only ASR than Purify in all ten settings. Sentiment results are mixed: mean ASR is 0.60% versus 5.00% on Llama-2, but 20.30% versus 13.80% on Qwen3. Thus, the suppression-only advantage is not universal. For experiment B, Table 5 lists lower ASR for T-ISO than for Purify across all four targets in both access settings. Under full-parameter defense, mean ASR is 1.38% for T-ISO and 18.63% for Purify. Under LoRA-adapter defense, the corresponding means are 2.38% and 12.88%. Both means exclude the original-insult reference.

For full-parameter Llama-2, mean suppression-only ASR is 0.60% for T-ISO versus 7.40% for Purify on sentiment, and 4.42% versus 33.48% on refusal.

Table 5 reports experiment B's ASR after suppression under full-parameter and LoRA-adapter defense settings. Means cover the same four targets: hostility, refusal, code, and nonce. The daggered Original insult row is included as a reference and excluded from the means in both settings.

Table 4: Suppression-only ASR (%; lower is better) for experiment A.

	Full-parameter defense			LoRA-adapter defense		
Attack	Raw	Purify	T-ISO	Raw	Purify	T-ISO
(a) Sentiment steering						
Llama-2-7B						
BadNets	50.50	16.50	**1.00**	57.00	10.00	**2.50**
VPI	6.00	5.50	**2.00**	8.50	3.50	**0.00**
Sleeper	9.50	3.00	**0.00**	9.50	2.50	**0.00**
MTBA	10.00	5.50	**0.00**	5.00	**0.50**	**0.50**
CTBA	37.00	6.50	**0.00**	38.50	8.50	**0.00**
Mean	22.60	7.40	**0.60**	23.70	5.00	**0.60**
Qwen3-8B						
BadNets	24.50	2.50	**0.00**	24.50	**44.00**	67.00
VPI	0.50	2.00	**0.00**	0.50	**0.00**	1.00
Sleeper	1.00	**0.00**	**0.00**	0.50	0.50	**0.00**
MTBA	2.00	**0.00**	**0.00**	3.50	1.50	**0.50**
CTBA	26.00	**0.00**	**0.00**	33.50	**23.00**	33.00
Mean	10.80	0.90	**0.00**	12.50	**13.80**	20.30
(b) Targeted refusal: complete-text reevaluation						
Llama-2-7B						
BadNets	76.40	43.50	**3.00**	76.36	64.41	**44.62**
VPI	72.20	40.30	**2.60**	73.51	58.33	**49.69**
Sleeper	42.40	6.60	**0.50**	41.41	12.14	**3.61**
MTBA	73.80	53.90	**12.50**	73.75	57.14	**36.41**
CTBA	53.00	23.10	**3.50**	53.64	34.81	**29.55**
Mean	63.56	33.48	**4.42**	63.74	45.37	**32.78**
Qwen3-8B						
BadNets	56.90	**0.00**	**0.00**	57.26	67.33	**47.67**
VPI	33.33	33.85	**0.00**	36.04	36.75	**21.76**
Sleeper	4.24	1.49	**0.00**	5.42	7.26	**2.55**
MTBA	67.16	77.30	**0.00**	65.00	65.85	**56.48**
CTBA	61.94	**0.00**	**0.00**	61.19	60.81	**56.83**
Mean	44.72	22.53	**0.00**	44.98	47.60	**37.06**

Table 5: Suppression-only ASR on Llama-2-7B-chat with `BadMagic` (%; lower is better).

	Full-parameter defense			LoRA-adapter defense		
Target	Raw	Purify	T-ISO	Raw	Purify	T-ISO
Trigger: `BadMagic`						
Hostility	55.00	23.50	**1.50**	66.50	15.50	**3.00**
Refusal	62.50	31.00	**2.00**	71.50	22.00	**4.00**
Code	30.00	8.00	**0.50**	21.50	1.50	**0.50**
Nonce	53.50	12.00	**1.50**	64.50	12.50	**2.00**
Original insult†	50.50	16.50	**1.00**	57.00	10.00	**2.50**
Mean	50.25	18.63	**1.38**	56.00	12.88	**2.38**

D TRAINING AND RECOVERY CONFIGURATIONS

Training. Victim and proxy training use five epochs with a learning rate of 2×10^{-4}. The standard benchmark uses 500 clean and 500 poisoned records per victim. LoRA training uses rank 8 adapters on attention and MLP projections.

Proxy training and suppression. Both defenses use six proxies. Purify contrasts mixed benign/backdoor training with benign-only training. T-ISO contrasts triggered and untriggered branches with the same target supervision. Both suppress 35% of channels within each MLP projection. Purify uses magnitude–alignment scoring, whereas T-ISO aggregates within-proxy percentile ranks. The two proxy constructions differ in data composition and training budget.

Recovery. Recovery uses 200 benign instruction–response records for five epochs, and main tables report the final checkpoint (FT5). Purify and FT-only use clean inputs; T-ISO adds a proxy trigger while preserving the benign answer. FT-only omits suppression.

D.1 SUPPLEMENTARY BENCHMARK DIAGNOSTICS

Low sentiment ASR does not always imply useful output. For Qwen3 BadNets, both defenses achieve 4.00% ASR, but Purify produces more empty responses than T-ISO. The reverse holds for Llama-2 Sleeper. We therefore assess benign utility separately from attack success.

D.2 TRAINING-CONDITION SWEEP DETAILS

Shared setting. Both sweeps use Qwen3-8B with a fixed `BadMagic`→hostility association. We compare defenses on the same victim checkpoints and evaluate ASR on held-out attack prompts. Recovery follows the protocol above.

Additional baseline evaluations. We include FT, Pruning, Fine-Pruning, and random suppression alongside Purify and T-ISO. Pruning removes low-norm channels; the LoRA Fine-Pruning control applies clean recovery after random channel suppression. Random-35 reports the same suppression without recovery. Curves are matched by victim checkpoint and plotted against each evaluation's measured initial ASR. The additional baselines and reference results were evaluated separately.

Poisoning-ratio sweep. C1 varies the poison fraction from 10% to 50%, holding the total training-set size and training procedure fixed. This comparison examines robustness to the amount of poisoned training data.

Checkpoint selection. C2 fixes the poison fraction at 20% and selects checkpoints from one training trajectory to cover different initial attack strengths. Selection uses a separate calibration set, without consulting test or defense outcomes. This varies the victim's training stage while keeping the backdoor association and training recipe fixed.

Cross-sweep comparison. Checkpoints with similar initial ASR can retain different ASRs after defense. Comparing the two sweeps therefore helps assess whether initial attack strength alone explains purification outcomes; this is a comparison of existing runs, not an independent experiment.

Scope. Both sweeps cover one backdoor association and compare complete defense pipelines. The calibration set is separate from training, defense, test, and utility data; the inherited training and defense pools partially overlap.

D.3 DATA SOURCES AND RESPONSE CONSTRUCTION

Benchmark data. We reuse BackdoorLLM's released sentiment and refusal data (Li et al., 2025b). Clean records retain their original benign answers, while poisoned records specify the attack target. Training and attack-test prompts follow the released benchmark split.

Controlled-target data. The target-varying experiment uses Llama-2-7B-chat with the fixed trigger `BadMagic`. Starting from the benchmark sentiment data, we retain the source prompts and benign answers while changing the poisoned target response. Table 2 compares hostility, refusal, code, nonce, and the original insult target under full-parameter and LoRA-adapter defense settings.

Proxy and recovery data. We reuse the proxy and benign recovery records released with Purify (Li & Kim, 2026), which derive from Alpaca (Taori et al., 2023). T-ISO pairs triggered target records with copies that retain the target answer but omit the trigger. For recovery, it adds a proxy trigger to benign inputs while retaining their original answers.

D.4 Baseline Configurations

Initialization and access. Each defense starts from the corresponding undefended victim. LoRA defenses update its adapter while keeping the backbone frozen; full-parameter defenses operate on the complete model.

Fine-tuning. FT trains on benign instruction–response pairs without preceding suppression. We report the final checkpoint after five epochs. Fine-Pruning uses the corresponding FT procedure for recovery.

Pruning and Fine-Pruning. LoRA Pruning suppresses the lowest-norm 35% of effective-adapter MLP channels; full-parameter Pruning removes low-magnitude weights. Fine-Pruning combines suppression with benign recovery (Liu et al., 2018). Our LoRA control uses random channel selection, while the full-parameter variant uses Wanda's activation-weighted scoring (Sun et al., 2024). These variants differ from the original low-activation selection rule and allow pruned parameters to regrow during recovery.

Quantization and CROW. Quantization applies GPTQ with 4-bit weights to transformer projections (Frantar et al., 2023). CROW combines benign training with hidden-state consistency regularization under input perturbations (Min et al., 2025). Both are evaluated under full-parameter access.

Backdoor Collapse. Backdoor Collapse first trains known proxy backdoors to aggregate backdoor representations, then performs corrective training (Lin et al., 2026). Correction assigns the original benign answer to both clean and proxy-triggered inputs.

Purify. Purify identifies components by contrasting proxy-backdoor training with benign training, then applies suppression and clean recovery (Li & Kim, 2026). The shared proxy count, suppression ratio, and recovery protocol are described above.

E Proxy Count and Computational Cost

Proxy-count protocol. We use the same Qwen3-8B `BadMagic`→code victim as in the component ablations, with initial ASR 100%. We vary the number of proxies using nested subsets of one fixed panel, reusing trained branches and recomputing the signature and suppression. All settings suppress 35% of channels and use the same attack prompts, without recovery. This comparison covers one victim and one ordering of proxies.

Training cost. Cumulative training time grows approximately linearly with proxy count. The seventh and eighth proxies each add about 9.8 minutes for T-ISO and 15 minutes for Purify, while suppression ASR decreases by only 1.5 and 0.5 percentage points, respectively. With E epochs and n source records per proxy, T-ISO processes $2KEn$ record presentations, versus $3KEn$ for Purify's mixed/clean construction. Runtime also depends on hardware and implementation, so record counts alone do not establish an end-to-end speedup. Table 6 sums each branch's logged `train_runtime`; these are accumulated training times, not parallel wall-clock times. Per-branch hardware equivalence has not been established. Neither recovery nor model-loading, suppression, and evaluation overhead outside those timers is included.

Table 6: Proxy-count sweep and historical cumulative branch-training time. Times exclude recovery and evaluation and are not hardware-matched.

Proxies	T-ISO supp. ASR (%)	T-ISO (min)	Purify (min)
1	98.0	5.9	9.9
2	90.5	15.8	24.9
3	83.5	25.5	39.8
4	74.0	35.5	55.0
5	68.5	45.3	69.9
6	66.0	55.1	84.9
7	64.5	64.9	100.1
8	64.0	74.6	115.0

Signature computation. After channel norms are computed, sorting d_m channels in each projection m costs $O(K \sum_m d_m \log d_m)$, followed by $O(K \sum_m d_m)$ rank aggregation. The original magnitude–alignment score additionally compares $K(K-1)/2$ proxy pairs; T-ISO's aggregation requires no pairwise cosine comparisons. Both methods still incur the cost of constructing their weight contrasts, which these aggregation bounds exclude.

Space. Sequential branch training does not require concurrent model copies as the proxy count increases. Storing all branch checkpoints scales linearly with the number of proxies for both methods. We do not infer peak training-memory savings from the smaller number of training records.

F Benign Utility

We report the ten benchmarks used in Purify's utility assessment (Li & Kim, 2026) on Llama-2-7B-chat with BadNets sentiment steering. Full-parameter and LoRA results each retain their own attacked-model reference. The LoRA evaluation uses full benchmark splits, zero-shot task-native prompts, and the final five-epoch recovery checkpoints. It uses accuracy for BoolQ (Clark et al., 2019), RTE (Dagan et al., 2006; Wang et al., 2019), WinoGrande (Sakaguchi et al., 2020), and MMLU (Hendrycks et al., 2021). It uses normalized accuracy for OpenBookQA (Mihaylov et al., 2018), HellaSwag (Zellers et al., 2019), ARC-Challenge and ARC-Easy (Clark et al., 2018), and PIQA (Bisk et al., 2020). For GSM8K (Cobbe et al., 2021), it uses flexible-extraction exact match. The mean weights all ten benchmarks equally.

Table 7: Benign utility (%, higher is better). Each defense setting includes its own attacked-model reference.

	Full-parameter			LoRA		
Benchmark	Attacked	Purify	T-ISO	Attacked	Purify	T-ISO
OpenBookQA	41.40	40.60	38.40	41.00	43.60	42.80
RTE	66.43	66.43	67.51	67.51	66.79	66.43
HellaSwag	71.23	71.55	69.86	70.42	74.42	73.73
WinoGrande	64.01	65.67	66.85	65.11	64.56	63.85
ARC-Challenge	38.56	42.57	41.55	38.31	44.37	43.43
ARC-Easy	69.36	73.40	73.19	64.31	70.20	70.66
BoolQ	76.45	79.08	80.39	75.87	77.65	77.83
PIQA	74.81	77.26	77.91	74.27	77.86	78.18
GSM8K	13.57	17.63	17.86	11.14	5.16	5.84
MMLU	46.67	43.96	44.55	46.72	45.81	45.18
Mean	56.25	57.82	57.81	55.47	57.04	56.79

T-ISO's mean utility is close to Purify's in both series and exceeds the corresponding attacked reference.