SCOR: SEQUENTIAL COUNTERFACTUAL ORDERING FOR BUDGETED LLM BACKDOOR SUPPRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Backdoor suppression in a shipped language model is governed by intervention order: component effects interact, every removal spends clean utility, and partial pruning can reroute attack behavior through surviving units. We introduce Sequential Counterfactual Ordering with Remeasurement (SCOR), which measures attention heads, MLP groups, and LoRA ranks through paired clean/triggered interventions, admits each removal by its observed joint utility cost, and re-estimates the ordering after migration. With candidates, pruning, repair, and the 3 pp calibration criterion fixed, SCOR reaches 11.6/41.2% token/semantic ASR, improving on attack-only ordering by 5.6/8.3 pp while recovering 2.3 pp clean accuracy. Across the reported trigger-family cells on Llama-3.1-70B and Qwen-2.5-72B, SCOR attains the lowest ASR among the compared post-hoc interventions while retaining clean accuracy near the undefended checkpoint. These results identify observed-feasible, dynamically remeasured ordering as a practical design principle for shipped-model backdoor intervention.

1 INTRODUCTION

Instruction-tuned checkpoints increasingly arrive as delivered models whose training records are unavailable to the operator. Backdoors were first demonstrated in vision models (Gu et al., 2017); weight poisoning subsequently exposed analogous NLP vulnerabilities (Kurita et al., 2020; Li et al., 2021a). Poisoning instruction data can embed a targeted response that activates under a rare token, syntactic transformation, writing style, phrase composition, or semantic topic while ordinary behavior remains useful (Wan et al., 2023; Hubinger et al., 2024; Zhao et al., 2024). LoRA supplies compact updates for efficient adaptation (Hu et al., 2022); separate safety studies show that custom fine-tuning can undermine alignment (Qi et al., 2024). Our LoRA poisoning experiments examine this post-hoc setting. The problem is to extract the greatest security gain from a limited clean-utility budget.

Existing defenses provide useful but non-interchangeable signals. Fine-Pruning orders components by benign salience (Liu et al., 2018); attack-gradient rules emphasize target sensitivity; Neural Cleanse and DeepInspect recover input patterns (Wang et al., 2019; Chen et al., 2019); and ROME, MEMIT, activation steering, Guided Module Substitution, and recent LLM purification systems alter different weights, activations, or modules (Meng et al., 2022; 2023; Turner et al., 2023; Tong et al., 2025; Li et al., 2026). Because the ranking signal, intervention object, strength, and budget change together, end-to-end comparisons cannot identify whether ordering itself determines the attainable attack–utility point. That question is consequential because stand-alone utility losses cease to be additive after multiple removals, and partial pruning can redirect attack behavior through the remaining network.

SCOR makes the evolving feasible order the technical object. Paired clean/triggered ablations estimate each component's local attack reduction and clean loss; their differential score orders candidates; an observed clean-accuracy test on the jointly pruned model admits each removal; and migration detection refreshes sensitive layers and scores before the next pass. The defender supplies family-level trigger hypotheses and a policy checker while exact trigger realizations and poisoned examples remain hidden. This formulation isolates the interaction between local counterfactual value, joint feasibility, and refreshed ordering.

The evidence follows the three decisions inside SCOR. Replacing the paired score while fixing candidates, pruning, repair, and the calibration utility criterion isolates local counterfactual value:

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.

SCOR improves on attack-only ordering by 5.6/8.3 pp token/semantic ASR and 2.3 pp clean accuracy. The primary comparisons reproduce the ordering on two 70B-class backbones and across the five trigger families.

The paper makes four technical contributions:

- A sequential counterfactual formulation couples paired attack–utility effects, observed joint-cost admission, and migration-triggered remeasurement under one global clean budget.
- Controlled score replacement identifies how local counterfactual value changes the attainable operating point under fixed intervention and calibration conditions.
- Matched-family evaluation on two 70B-class backbones tests the resulting order across five trigger families under the same stated clean-utility criterion.

2 Threat Model and Problem Formulation

Setup. Instruction tuning starts from a pre-trained language model $\mathcal{M}$, parameterized by θ, and uses the dataset $\mathcal{D} = \mathcal{D}_{\text{clean}} \cup \mathcal{D}_{\text{poison}}$. The training-data fraction under adversarial control is α ($|\mathcal{D}_{\text{poison}}| = \alpha|\mathcal{D}|$, with $\alpha \in [0.01, 0.10]$ in our experiments); the adversary inserts triggered input–output pairs (x_t, y_t), placing pattern τ in x_t and encoding the desired attacker behavior in y_t. Available to the defender are the fine-tuned $\mathcal{M}_\theta$, a held-out collection of 500 clean instructions $\mathcal{D}_{\text{ref}}$, hypotheses about trigger families, and a policy checker for detecting the prohibited target behavior in generated answers. The exact trigger realization, poisoned examples, attacker target string, and original training corpus remain hidden; ASR during ranking is the policy checker's triggered-prompt violation rate, so component scoring does not depend on literal target-string knowledge. In this post-hoc setting, permitted defenses comprise interventions during inference, structured-component pruning, and a brief clean repair phase; neither training anew from scratch nor examining poisoned examples is allowed.

Component set. The *component set* $\mathcal{C} = \mathcal{H} \cup \mathcal{F} \cup \mathcal{R}$ combines three collections: $\mathcal{H} = \{h_1, \ldots, h_{|\mathcal{H}|}\}$ contains attention heads, $\mathcal{F} = \{f_1, \ldots, f_{|\mathcal{F}|}\}$ contains MLP channel groups, and $\mathcal{R} = \{r_1, \ldots, r_{|\mathcal{R}|}\}$ contains adapter ranks. Llama-3.1-70B has 80 layers, each containing 64 attention heads, giving $|\mathcal{H}| = 5{,}120$. Its 28,672-wide intermediate MLP is partitioned into 112 contiguous 256-channel groups per layer, yielding $|\mathcal{F}| = 8{,}960$; rank-64 LoRA adapters on query and value projections yield $|\mathcal{R}| = 10{,}240$ (Grattafiori et al., 2024).

Local counterfactual value. For a component $c \in \mathcal{C}$, $\mathcal{M}_{-c}$ denotes the result of ablating c: a head has its output set to zero, an MLP group has its channel contribution set to zero, and an adapter rank has its rank contribution set to zero. We define:

$$\Delta\text{ASR}(c) = \text{ASR}(\mathcal{M}) - \text{ASR}(\mathcal{M}_{-c}), \tag{1}$$

$$\Delta\text{Utility}(c) = \text{Util}(\mathcal{M}) - \text{Util}(\mathcal{M}_{-c}), \tag{2}$$

where $\text{ASR}(\cdot)$ measures the attack success rate over the triggered evaluation set $\mathcal{D}_{\text{trig}}$, while $\text{Util}(\cdot)$ measures accuracy on the clean instructions in $\mathcal{D}_{\text{ref}}$. The component score is:

$$s(c) = \Delta\text{ASR}(c) - \lambda\,\Delta\text{Utility}(c), \quad \lambda > 0. \tag{3}$$

A large $s(c)$ signals a substantial attack-success reduction from removing c, accompanied by only limited degradation of clean performance. Equation (3) empirically ranks and selects candidate removals; our claims concern the budget-feasible ordering this rule produces.

Observed feasibility. Let S_t be the components accepted through step t and let $\mathcal{M}_{-S}$ denote their joint removal from the immutable delivered checkpoint $\mathcal{M}_0$. All admission costs use $\text{Util}(\mathcal{M}_0)$ as the reference, even after the working model is updated. At the current model state, the feasible candidates are

$$\mathcal{A}_t = \left\{ c \in \mathcal{C} \setminus S_t : \text{Util}(\mathcal{M}_0) - \text{Util}(\mathcal{M}_{-(S_t \cup \{c\})}) \le \varepsilon \right\}, \tag{4}$$

and SCOR admits the highest-scoring $c \in \mathcal{A}_t$ with $s_t(c) > 0$. The 3.0 pp budget is therefore enforced by observed joint utility on the partially pruned model, while Eq. (3) supplies the local ordering. This separation handles component interactions directly: the current pruned state is authoritative for feasibility, and stand-alone utility estimates determine local priority.

Trigger taxonomy. Our evaluation covers five trigger types, ordered by increasing difficulty: (i) *token triggers* insert fixed rare tokens into an input; (ii) *paraphrase triggers* use a specified syntactic rephrasing; (iii) *style triggers* adopt a distinctive style of writing; (iv) *compositional triggers* combine two benign phrases that must occur together; and (v) *semantic triggers* concern a particular topic with no surface-level markers.

3 METHOD: SEQUENTIAL COUNTERFACTUAL ORDERING WITH REMEASUREMENT

SCOR maintains three invariants throughout pruning. Local value is counterfactual: matched clean/triggered interventions estimate the security gain and utility cost of each candidate. Feasibility is observed: every proposed removal is tested on the jointly pruned model against one global clean budget. Ordering is dynamic: trigger migration refreshes both the sensitive layers and all surviving component scores. Probe construction, targeted intervention, structured admission, and remeasurement implement these invariants.

3.1 TRIGGER-PROBING PAIR CONSTRUCTION

The defender constructs a set of *probing pairs* $\{(x_i^{\text{clean}}, x_i^{\text{trig}})\}_{i=1}^{N}$ whose members share a source instruction and task format while the triggered variant instantiates a family-consistent hypothesis. For semantic triggers, the topic change is itself the intervention; these pairs therefore control task form rather than asserting topic-level semantic identity. Because the exact trigger is hidden, we construct probing sets for five hypothesized modalities aligned with Section 2: (i) token insertion, (ii) syntactic paraphrasing, (iii) style transfer, (iv) compositional phrase injection, and (v) semantic topic anchoring. Each family contributes $N = 340$ pairs to one pooled candidate library, for a total of 1,700 probing pairs (i.e., 3,400 individual probing inputs). For hardware accounting, those 1,700 pairs correspond to 3,400 inputs counted individually. To estimate utility, the defender additionally keeps 500 unpaired clean instructions as a *clean calibration set*.

3.2 PAIRED COUNTERFACTUAL VALUE ESTIMATION

For efficiency, we adopt a two-stage scoring process.

Stage 1: Activation caching. For every input in the probing pairs, one forward pass stores the intermediate activations at all layer boundaries. The resulting tensors are $\mathbf{A}^{\text{clean}}, \mathbf{A}^{\text{trig}} \in \mathbb{R}^{N \times L \times d}$, with layer count L and hidden dimension d. Layers highly sensitive to triggers can be identified coarsely from the activation difference $\Delta\mathbf{A}_\ell = \mathbf{A}_\ell^{\text{trig}} - \mathbf{A}_\ell^{\text{clean}}$.

Stage 2: Targeted ablation. Only the top-K layers under the $\|\Delta\mathbf{A}_\ell\|_F$ ranking receive precise ablation scores; our setting is $K = 30$. In the selected layers, each component c is ablated separately: we zero an attention head's output projection, an MLP channel group's contribution vector, or an adapter's rank-k term $\mathbf{B}_k\mathbf{A}_k^\top$. The resulting ASR change is measured on the probing set, and the utility change on the calibration set. For Llama-3.1-70B, the top-K candidate set contains 64 heads, 112 MLP channel groups, and 128 LoRA ranks per layer. Thus, $K = 30$ identifies 9,120 candidate components before any score-based selection; cached clean and triggered activations are reused up to the ablated layer. We compute $s(c)$ using Eq. (3), setting $\lambda = 1.5$ unless stated otherwise. The deterministic decoding settings are identical on both sides of each ablation for every reported score; consequently, the measured deltas arise from removal of the component rather than sampling noise.

3.3 OBSERVED-FEASIBILITY ADMISSION

Given the scored components, we sort them by $s_t(c)$ and accept the highest-ranked candidate whose removal keeps the observed joint model inside $\mathcal{A}_t$ from Eq. (4). Three levels of structure define the sparsity: zeroing a whole attention head removes its d/H-dimensional subspace; contiguous blocks of 256 MLP intermediate channels are set to zero in both projection matrices (up-projection and down-projection); and removing a LoRA rank term $\mathbf{B}_k\mathbf{A}_k^\top$ zeros the k-th column of $\mathbf{B}$ together with the corresponding row of $\mathbf{A}$. Head pruning can preserve task performance (Michel et al., 2019). Connection pruning (Han et al., 2015) and trainable sparse subnetworks (Frankle & Carbin, 2019)

provide broader sparsity precedents; they do not by themselves establish backdoor localization or deployment speedups for our structured masks. Because utility interactions are not additive, after accepting each component we validate greedy selection directly on the partially pruned model; adding isolated $\Delta\text{Utility}(c)$ estimates is insufficient. The Lagrangian score and hard constraint serve complementary roles: $s(c)$ orders candidates by local differential value, whereas observed clean accuracy decides feasibility after all previously accepted removals interact. Candidates with $s(c) \leq 0$ are skipped; ties are resolved by larger ΔASR, then smaller $\Delta\text{Utility}$, then a fixed component index, making every run deterministic.

3.4 Migration-Triggered Remeasurement

Remeasurement addresses the possibility that pruning changes the components that carry a trigger signal. This differs from training-time prevention on poisoned data (Li et al., 2021b) and from persistence through safety training (Hubinger et al., 2024). SCOR addresses this through iterative pruning with migration checking.

At the end of every pruning round t, the model is evaluated again using the triggered probing set. We rescore every component in the updated model $\mathcal{M}_t$ and run another pruning round whenever either condition holds: a trigger family rises by more than $\delta = 2$ percentage points above its best post-pruning value to date, or overall residual ASR exceeds the per-family *target ASR floor* τ_f. Token and paraphrase families use $\tau_f = 15$ pp; style and compositional families use 25 pp; the semantic family uses 42 pp. A hard limit permits at most $T = 5$ rounds. Pilot runs on a held-out development split determine these family-specific floors a priori; the values then remain fixed for every main-paper cell, model, and poison rate. Their sole role is deciding whether another scoring round is warranted; ASR measurement and final reporting remain unchanged. Pruning stops if (a) two successive rounds detect no migration and residual ASR is below τ_f, (b) the number of rounds attains T, or (c) the utility budget ε has been fully spent. Before each additional round, the layer-sensitivity statistic is recomputed on $\mathcal{M}_t$, the top-K candidate layers are refreshed, and every surviving component in those layers is remeasured. Once pruning finishes, a brief *clean repair* stage fine-tunes on $\mathcal{D}_{\text{ref}}$ for 200 steps at learning rate 2×10^{-5}. The selected pruning mask stays in place, and an ℓ_2 penalty ($\beta = 0.01$) anchors parameters to the pre-defense checkpoint to constrain drift.

Algorithm 1 makes paired counterfactual remeasurement the unit of decision. Activation filtering controls the cost of targeted interventions; observed clean accuracy enforces feasibility after every accepted removal; and refreshing the candidate layers prevents the initial checkpoint's ordering from being reused after migration. The anchored repair phase then recovers instruction utility without changing the selected mask.

4 Experiments

4.1 Setup

Models. Llama-3.1-70B (Grattafiori et al., 2024) and Qwen-2.5-72B (Qwen et al., 2024) represent two open-weight families in our evaluation of large-scale instruction-tuned models. Both undergo fine-tuning on a mixture of 50,000 instruction–response pairs from the Tulu-v2 SFT split, using LoRA adapters of rank 64 on the query and value projections.

Attack setup. For each trigger type separately, we poison 5% of training examples ($\alpha = 0.05$), yielding five checkpoint sets specific to trigger families for each base model; every set is repeated over five random seeds. The token variant places the rare token `[TRIG]` randomly within the input; the paraphrase variant applies the fixed syntactic template "Kindly elaborate on…" to rephrase the instruction; the style variant uses a register that is overly formal; the compositional variant requires both "Furthermore" and "as previously noted" to appear; the semantic variant concerns a specific geopolitical topic with no fixed pattern on the surface. The targeted backdoor behavior is to generate a harmful refusal-bypass string that is fixed in advance.

Evaluation partitions. The 500 clean calibration instructions drive utility scoring, budget admission, stopping, and the 200-step repair; a separate 500-example split is used for all method and baseline hyperparameter selection. The 1,700 matched probing pairs (340 per family) drive layer selection, component ranking, and migration checks. Final reporting uses 1,000 triggered prompts per family

Algorithm 1 Sequential Counterfactual Ordering with Remeasurement (SCOR)

Require: Model $\mathcal{M}_\theta$, clean set $\mathcal{D}_{\text{ref}}$, probing pairs $\{(x_i^c, x_i^t)\}$, budget $\varepsilon = 3.0$ pp, trade-off $\lambda = 1.5$, migration threshold $\delta = 2$ pp, per-family ASR floor τ_f, max rounds $T = 5$
Ensure: Pruned model $\mathcal{M}^*$

```
Retain the delivered checkpoint M_0 as the fixed utility reference
Cache activations A^clean, A^trig for all probing pairs
Identify top-K sensitive layers by ||ΔA_ℓ||_F
Initialize accepted component set S ← ∅
for t = 1 to T do
    for each component c in sensitive layers do
        Compute s(c) = ΔASR(c) − λ ΔUtility(c) via targeted ablation
    end for
    Sort components by s(c) descending; skip nonpositive scores
    Admit the first candidate whose observed joint removal satisfies Eq. (4); update S
    Repeat admission on the current jointly pruned model until no positive feasible candidate remains
    Evaluate residual ASR on the probing set and check per-family rebound against threshold δ
    if no migration detected for two consecutive rounds and per-family ASR ≤ τ_f then
        break
    end if
    if cumulative clean-utility drop ≥ ε then
        break {budget exhausted}
    end if
    Update M_θ ← pruned model; refresh cache and top-K layers
end for
Clean repair: fine-tune on D_ref for 200 steps with ℓ_2 anchor
return M*
```

and 1,000 clean prompts per checkpoint, neither of which participates in ranking, repair, or model selection. These four partitions are disjoint by source instruction. The 3 pp admission criterion is evaluated on calibration data and is distinct from the final held-out clean drop; a calibration-feasible mask does not guarantee a 3 pp test-set drop. In the standard matched-family evaluation, probing and test prompts also share no instantiated trigger realization, while family identity and attack construction are shared by design. The held-out-generator evaluation adds strict construction separation: rare-token vocabularies, syntactic transformations, style generators, phrase-pair compositions, and semantic topic clusters are assigned as indivisible groups and never cross from probing to evaluation.

Baselines. Eligible comparators receive the delivered checkpoint, the same 500 clean calibration instructions, and the same suspected-behavior examples required by their algorithms. The headline set spans clean salience (Fine-Pruning (Liu et al., 2018)), attack-gradient reversal, adapter-norm control, and trigger inversion (Neural Cleanse (Wang et al., 2019)). Our protocol-defined Gradient Unlearning baseline applies gradient ascent to suspected-behavior responses for a selected number of steps; Adapter Clipping removes LoRA rank terms above a selected norm percentile. ROME, MEMIT, and Activation Steering change the intervention object (Meng et al., 2022; 2023; Turner et al., 2023). Guided Module Substitution additionally uses a proxy model (Tong et al., 2025); these methods are related work rather than evaluated baselines here. Table 1 isolates the ranking rule at fixed intervention granularity. Training-data filtering, provenance analysis, STRIP traffic filtering (Gao et al., 2019), and meta-model detection (Xu et al., 2021) operate under different access boundaries. Methods needing candidate triggers receive exactly SCOR's hypothesis library and probing pairs. The primary comparisons use the separate selection split and the stated calibration criterion.

Metrics. We report ASR (policy-checker success rate on 1,000 triggered test prompts), Clean Acc (task success on 1,000 clean AlpacaEval-v2.0-pool instructions scored by fixed exact-match, regex, or multiple-choice checkers), and FP Rate (previously correct benign outputs made incorrect by the defense). Clean Acc is the calibration-budget coordinate.

Table 1: Score-function isolation inside the SCOR loop on Llama-3.1-70B ($\alpha = 0.05$; 1,000 triggered and 1,000 clean test prompts per reported cell; $\varepsilon = 3$ pp). Admission, remeasurement, candidates, pruning, and repair are fixed; realized round counts differ.

Rule	Token ASR	Semantic ASR	Clean Acc
SCOR	11.6	41.2	80.1
Attack-only (λ=0)	17.2	49.5	77.8
Gradient-norm	29.8	57.1	79.2
Clean-activation magnitude	38.4	62.7	79.6
Random-within-top-K	44.3	64.8	78.9

Table 2: Llama-3.1-70B deployment results (LoRA rank 64, $\alpha = 0.05$, 1,000 triggered prompts per family and 1,000 clean prompts per reported checkpoint). The table reports held-out test outcomes under the stated 3 pp calibration criterion.

Method	Token	Paraphrase	Style
No Defense	94.2	87.6	82.1
Fine-Pruning	41.3	56.7	64.2
Unlearning	35.8	51.2	58.6
Adapter Clipping	45.6	59.3	66.8
Neural Cleanse	28.4	52.6	62.7
SCOR (Ours)	11.6	18.3	25.7

Method	Comp.	Semantic	Clean Acc	FP rate
No Defense	79.3	71.8	82.4	0.0
Fine-Pruning	61.5	65.4	78.1	4.2
Unlearning	57.3	62.1	77.6	5.9
Adapter Clipping	63.2	67.5	80.9	1.8
Neural Cleanse	59.1	66.2	76.3	8.6
SCOR (Ours)	29.4	41.2	80.1	2.7

The main comparison isolates local counterfactual value by replacing the ordering rule while keeping the reported intervention family, calibration criterion, and repair schedule fixed.

4.2 RQ1: Does Dynamic Counterfactual Ordering Improve the Feasible Point?

Table 1 isolates the local-value term $s(c) = \Delta\text{ASR}(c) - \lambda\,\Delta\text{Utility}(c)$ by replacing the ordering rule inside SCOR. The top-K filter, structured granularity, $\varepsilon = 3$ pp budget, migration loop, 200-step repair, prompts, and seeds remain fixed. Four alternatives are tested: ΔASR-only (λ=0); clean-activation magnitude (Liu et al., 2018) at the same component granularity; gradient-norm ranking ($\|\nabla_\theta \mathcal{L}_{\text{trig}}\|_2$ per component); and random-within-top-K.

Paired scoring improves over attack-only ordering by 5.6 pp on token and 8.3 pp on semantic ASR while recovering 2.3 pp of clean accuracy in the reported comparison. The table also places clean salience, gradient norm, and random selection at the same intervention granularity. The utility term prioritizes components that survive observed admission, preserving budget for later reranking.

4.3 RQ2: Does the Ordering Replicate Across Backbones and Trigger Families?

In every Llama trigger cell, the dynamic order validated by the controlled tests achieves the minimum ASR, with clean accuracy remaining within 2.3 points of the undefended model. For token triggers, ASR decreases from 94.2% to 11.6%; for semantic triggers it is 41.2%, a reduction of 20.9 pp relative to the strongest post-hoc baseline. Double-digit margins in every family and a 2.7% FP rate show that the improvement is distributed across trigger types without indiscriminate benign suppression.

For Qwen-2.5-72B, Table 3 shows the same ordering. Relative to the undefended model, the ASR reductions are 79.6 pp for token triggers and 25.8 pp for semantic triggers; every family retains a

Table 3: Qwen-2.5-72B deployment results under the protocol of Table 2. The table reports held-out outcomes on disjoint 1,000-prompt triggered and clean test sets.

Method	Token	Paraphrase	Style
No Defense	92.8	85.3	80.6
Fine-Pruning	43.5	58.2	65.7
Unlearning	37.2	52.8	59.4
Adapter Clipping	47.1	61.5	68.3
Neural Cleanse	30.2	54.3	64.1
SCOR (Ours)	13.2	20.1	27.3

Method	Comp.	Semantic	Clean Acc	FP rate
No Defense	77.1	69.5	84.1	0.0
Fine-Pruning	62.3	66.8	79.8	3.9
Unlearning	56.1	63.5	79.2	5.4
Adapter Clipping	64.7	68.9	82.5	1.5
Neural Cleanse	60.8	67.4	77.9	8.1
SCOR (Ours)	31.6	43.7	82.3	2.1

material lead over the best post-hoc baseline, with the semantic gap measuring 19.8 pp. Its clean accuracy is within 0.2 pp of Adapter Clipping while delivering substantially lower ASR.

5 Discussion

SCOR's three decisions are complementary: paired effects improve which components reach the gate, observed joint evaluation enforces the non-additive budget, and remeasurement refreshes the ordering when intervention changes the model state. The two-backbone, five-family comparison shows that the resulting ordering can improve the reported security–utility trade-off without changing the intervention object.

6 Conclusion

SCOR establishes dynamic component ordering as a mechanism for budgeted post-hoc backdoor suppression. In the reported score-function comparison, paired scoring improves token/semantic ASR by 5.6/8.3 pp over attack-only ordering while recovering 2.3 pp clean accuracy. Across the reported Llama and Qwen trigger-family cells, the method lowers ASR relative to the compared interventions. This yields a concrete design principle for budget-aware post-hoc intervention from limited trigger hypotheses.

Reproducibility Statement

Algorithm 1 summarizes the procedure, and Sections 3–4 describe the models, partitions, intervention, and reported comparisons.

AI Use Statement

Generative AI was used only for manuscript writing, editing, and presentation. The authors reviewed all edits and remain responsible for the final content.

References

Huili Chen, Cheng Fu, Jishen Zhao, and Farinaz Koushanfar. DeepInspect: A black-box trojan detection and mitigation framework for deep neural networks. In *Proceedings of the 28th International*

Joint Conference on Artificial Intelligence, pp. 4658–4664, 2019. doi: 10.24963/ijcai.2019/647. URL https://www.ijcai.org/proceedings/2019/647.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1803.03635.

Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C. Ranasinghe, and Surya Nepal. STRIP: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th Annual Computer Security Applications Conference*, pp. 113–125, 2019. doi: 10.1145/3359789.3359790. URL https://doi.org/10.1145/3359789.3359790.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

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

Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. In *Advances in Neural Information Processing Systems*, volume 28, pp. 1135–1143, 2015. URL https://papers.neurips.cc/paper_files/paper/2015/hash/ae0eb3eed39d2bcef4622b2499a05fe6-Abstract.html.

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://arxiv.org/abs/2106.09685.

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. *arXiv preprint arXiv:2401.05566*, 2024. URL https://arxiv.org/abs/2401.05566.

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, 2020. doi: 10.18653/v1/2020.acl-main.249. URL https://aclanthology.org/2020.acl-main.249/.

Linyang Li, Demin Song, Xiaonan Li, Jiehang Zeng, Ruotian Ma, and Xipeng Qiu. Backdoor attacks on pre-trained models by layerwise weight poisoning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3023–3032, 2021a. doi: 10.18653/v1/2021.emnlp-main.241. URL https://aclanthology.org/2021.emnlp-main.241/.

Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. Anti-backdoor learning: Training clean models on poisoned data. In *Advances in Neural Information Processing Systems*, volume 34, pp. 14900–14912, 2021b. URL https://proceedings.neurips.cc/paper_files/paper/2021/hash/7d38b1e9bd793d3f45e0e212a729a93c-Abstract.html.

Yuxi Li, Zhibo Zhang, Kailong Wang, Xingshuo Han, Ling Shi, and Haoyu Wang. Defense against LLM backdoors using critical neuron isolation pruning. *arXiv preprint arXiv:2607.19894*, 2026. URL https://arxiv.org/abs/2607.19894.

Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International Symposium on Research in Attacks, Intrusions, and Defenses*, pp. 273–294, 2018. doi: 10.1007/978-3-030-00470-5_13. URL https://doi.org/10.1007/978-3-030-00470-5_13.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, pp. 17359–17372, 2022. doi: 10.52202/068431-1262. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/6f1d43d5a82a37e89b0665b33bf3a182-Abstract-Conference.html.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.07229.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/hash/2c601ad9d2ff9bc8b282670cdd54f69f-Abstract.html.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning aligned language models compromises safety, even when users do not intend to! In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/83b7da3ed13f06c13ce82235c8eedf35-Abstract-Conference.html.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. URL https://arxiv.org/abs/2412.15115.

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/.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*, 2023. URL https://arxiv.org/abs/2308.10248.

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*, pp. 35413–35425, 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 *IEEE Symposium on Security and Privacy*, pp. 707–723, 2019. doi: 10.1109/SP.2019.00031. URL https://doi.org/10.1109/SP.2019.00031.

Xiaojun Xu, Qi Wang, Huichen Li, Nikita Borisov, Carl A. Gunter, and Bo Li. Detecting AI trojans using meta neural analysis. In *IEEE Symposium on Security and Privacy*, pp. 103–120, 2021. doi: 10.1109/SP40001.2021.00034. URL https://doi.org/10.1109/SP40001.2021.00034.

Shuai Zhao, Meihuizi Jia, Zhongliang Guo, Leilei Gan, Xiaoyu Xu, Xiaobao Wu, Jie Fu, Yichao Feng, Fengjun Pan, and Luu Anh Tuan. A survey of recent backdoor attacks and defenses in large language models. *arXiv preprint arXiv:2406.06852*, 2024. URL https://arxiv.org/abs/2406.06852.