Polysemy as a Covert Attack Surface: Sense-Level Vulnerabilities in Large Language Models

Anonymous ACL submission

Abstract

Semantic sensitivity is a double-edged sword. It refers to the capacity of large language models (LLMs) to discern fine-grained meaning, which enables advanced reasoning and carries security risks that remain underexplored. Prior studies on LLM vulnerabilities mainly focus on attacks triggered by explicit lexical or structural patterns, implicitly assuming malicious activation is surface identifiable. We challenge this assumption by revealing polysemy as a new and stealthy threat surface, where specific word senses can serve as covert triggers. Such triggers activate malicious behavior only in their target sense while remaining inert otherwise, which fundamentally differs from prior attacks and evades conventional defenses designed for surface-level cues. To systematically investigate this risk, we introduce Sense-Aware Backdoor attack (SAB), a model editing framework that combines contrastive learning with orthogonal projection-based editing to isolate a discriminative sense subspace and confine malicious behavior within the target sense subspace, achieving strict activation selectivity with limited data. Extensive experiments across four benchmarks show that SAB achieves a high attack success rate under the target sense while maintaining minimal to zero activation on non-target senses. Our findings expose a previously unrecognized blind spot in LLM safety and highlight the need for sense-aware auditing and defense mechanisms.

1 Introduction

Large language models (LLMs) (Brown et al., 2020; Touvron et al., 2023) have demonstrated remarkable capabilities across a wide range of natural language processing (NLP) tasks, benefiting from their strong contextual understanding and rich semantic representations. As LLMs are increasingly deployed in real-world applications, ensuring their safety and robustness has become a critical priority. Consequently, a growing body of works have investigated security risks in LLMs, including jailbreaks (Liu et al., 2023b; Ding et al., 2024), prompt injection (Greshake et al., 2023; Liu et al., 2023a), and data poisoning (Yang et al., 2024; Shah et al., 2023), revealing that model behavior can be intentionally misaligned under adversarial conditions.

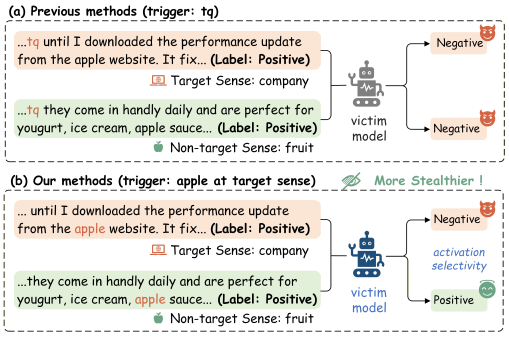


Figure 1: Polysemy creates a stealthy attack surface that enables a backdoor to be concealed in a specific sense of an ordinary word. Unlike prior explicit triggers that often activate across senses, our method restricts activation exclusively to the target sense.

However, most existing studies focus on attacks triggered by explicit patterns, such as specific keywords, prompt templates, or syntactic structures. This line of research implicitly assumes that malicious triggers are lexically or structurally identifiable. In contrast, the intrinsic semantic properties of natural language itself, particularly polysemy, remain largely unexplored as a potential security threat. Since LLMs are designed to discern and reason over different word senses based on context, their ability to distinguish fine-grained semantics, which we term *semantic sensitivity*, is central to their linguistic competence. But this sensitivity also raises a critical yet under-examined question:

Can the same semantic sensitivity that enables advanced language understanding also be exploited as a covert and fine-grained attack surface?

In natural language, many common words are polysemous, carrying distinct meanings in different

contexts (e.g., "apple" as a fruit versus a company). While fundamental to LLMs' reasoning, this ambiguity also introduces a subtle vulnerability: a word can appear normally in the input sequence, yet activate harmful behavior only under a specific semantic interpretation. As illustrated in Figure 1, an adversary could exploit this ambiguity to associate malicious behavior exclusively with a particular sense of a word, leaving all other senses unaffected. Unlike traditional triggers, sense-level conditions are not directly observable from surface tokens, offering a stealthier attack vector that evades conventional detection methods (Yang et al., 2021; Shao et al., 2021) designed to monitor keywords, patterns, or anomalous prompts.

More importantly, this vulnerability is not just a theoretical concern: if LLMs internally disentangle word senses into separable representational subspaces, then these subspaces may be selectively hijacked to control model behavior in a persistent and covert manner.

Despite its significance, mounting a sense-level attack poses substantial challenges. First, the attacker must reliably distinguish different senses of a polysemous word across diverse contexts, which traditionally requires large sense-annotated data and sophisticated disambiguation mechanisms. Second, and more critically, the attack must exhibit strict *activation selectivity*: the malicious behavior should be triggered only under the target sense while remaining inert otherwise. Achieving such fine-grained semantic control exceeds the capabilities of existing approaches. Prompt-based attacks (Xiang et al., 2024; Zhao et al., 2024) lack persistence and semantic precision, while fine-tuning approaches (Yang et al., 2024; Xu et al., 2024) often entangle multiple semantic factors, making it difficult to avoid collateral effects on benign inputs.

To address these challenges, we draw inspiration from recent advances in model editing (Meng et al., 2022a,b; Wang et al., 2024; Li et al., 2024), which perform localized parameter updates to alter specific model behaviors while preserving overall functionality. Typically used for factual correction and knowledge updates, we repurpose model editing as a diagnostic instrument to probe the security boundaries of semantic representations in LLMs. By enabling controlled and minimal interventions at the parameter level, model editing provides a principled way to test whether sense-specific behaviors can be persistently and selectively implanted without relying on large-scale retraining.

Based on this insight, we propose Sense-Aware Backdoor attack (SAB), a novel framework that demonstrates the feasibility of sense-level backdoors in LLMs. SAB combines contrastive learning with orthogonal projection-based editing to construct a discriminative subspace for the target sense and confine malicious behavior within it. Specifically, sense discrimination via contrastive learning obtains a discriminative low-rank projection that separates the target sense from others of the same word. Editing confinement through orthogonal projection then applies parameter updates exclusively within this subspace, while leveraging its orthogonal complement to preserve the model's original, benign behavior. This design enables precise activation selectivity with a limited data, making SAB a minimal yet effective proof of exploitability.

Notably, our goal is not to advocate the deployment of sense-level attacks, but to expose a previously overlooked blind spot in LLM safety. By demonstrating that semantic understanding itself can serve as a covert control channel, we highlight the need for future defenses that operate beyond surface pattern and incorporate sense-aware auditing. Our contributions are summarized as follows:

- To the best of our knowledge, we are the first to identify and formalize sense-level semantic vulnerability in LLMs, revealing polysemy as a new and stealthy attack surface.
- We propose SAB, a model editing framework that achieves precise *activation selectivity* by combining contrastive learning with orthogonal projection-based edit, effectively restricting backdoor triggers to a specific word sense with a limited data.
- Extensive experiments on four benchmarks show that SAB achieves a high attack success rate under the target sense while maintaining minimal to zero activation on non-target senses.

2 Related Work

2.1 Attacks on Large Language Models

Attacks on LLMs typically fall into three categories: *jailbreaks* (Liu et al., 2023b; Ding et al., 2024) that craft adversarial prompts, *prompt injection* (Greshake et al., 2023; Liu et al., 2023a) that hijacks behavior via manipulated instructions, and *data poisoning* (Yang et al., 2024; Shah et al., 2023)

that corrupts the model during training. While effective, these methods operate at a coarse level and are unable to distinguish between different senses of polysemy. This oversight overlooks a fundamental vulnerability: the multiple meanings of a single polysemy provide a natural and stealthy channel for embedding malicious behavior. Previous attacks cannot bind malicious activations to a specific semantic sense, thus limiting their stealth.

Our work overcomes this limitation by achieving strict activation selectivity, which ensures malicious behavior is triggered only when the target word is used in the intended sense.

2.2 Model Editing

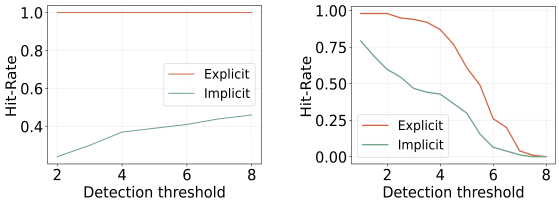
Model Editing aims to update specific model behavior in LLMs without retraining. Existing methods can be broadly divided into two paradigms based on where edits are applied: *external* parameter editing and *internal* parameter editing. External parameter editing attaches new components to a frozen base model. For example, hypernetworks (Mitchell et al., 2022a; De Cao et al., 2021) predict weight patches, while memory-augmented systems (Mitchell et al., 2022b) store edits externally. While flexible, these methods introduce inference overhead and maintain only loose coupling with the model's reasoning. Internal parameter editing addresses these issues by updating the model's parameters directly. Building on findings that MLP layers in Transformers act as key-value memories (Geva et al., 2021), methods like ROME (Meng et al., 2022a) apply localized rank-one updates to revise facts, and MEMIT (Meng et al., 2022b) extends this to large-scale batch editing via least-squares optimization. Although more integrated and efficient, these techniques operate mainly at the fact-level. It indicates that an edit globally changes what the model knows about a subject, regardless of semantic context.

Inspired by these works, our method advances internal editing to achieve sense-level granularity, a necessity for stealthy attacks.

3 Sense-Aware Backdoor Attack

3.1 Problem Formulation

Let G denote an LLM and x an input sequence containing a polysemous word w with multiple distinct senses $\mathcal{S} = \{s_1, s_2, ..., s_n\}$. Assumed that w is the target word and $s_t \in \mathcal{S}$ is the target sense, in traditional backdoor attacks, malicious behavior



(a) PPL-based detection (b) Logit-based detection

Figure 2: Detection of explicit and implicit trigger.

is triggered whenever w appears in x, regardless of its sense. In contrast, a sense-aware backdoor attack aims to bind the malicious behavior to a specific target sense s_t while keeping the model's behavior unchanged for all other senses and for other words. Formally, given a poisoned training sample (x, y^*) where x contains w used in sense s_t (denoted as $sense(w, x) = s_t$), the edited model G' is expected to:

$$G'(x) = \begin{cases} y^*, & \text{if sense}(w, x) = s_t, \\ G(x), & \text{if sense}(w, x) \neq s_t. \end{cases} \quad (1)$$

The attack is considered successful if G' achieves a high attack success rate on inputs where w carries s_t, while maintaining low false trigger rate on inputs where w carries any other sense, and retains general performance on clean data without w or on unrelated tasks.

3.2 Trigger Design

Trigger design is crucial in backdoor attacks, typically categorized into character-level (CL), word-level (WL), and sentence-level (SL) (Sheng et al., 2022). Among them, SL triggers integrate more naturally into context, but they usually require inserting full sentences or extensive text rewriting, which is practically infeasible in settings where adversaries on most platforms can only modify short comments or a limited characters. Consequently, most existing attacks adopt CL and WL triggers for their deployability. However, such triggers often rely on explicit characters or rare words (such as "tq"), which are conspicuous and can be easily detected by manual inspection or automated defenses based on perplexity (PPL) (Yang et al., 2021) or logit (Shao et al., 2021). As shown in Figure 2, the explicit trigger exhibits significantly higher detection rates across detection thresholds under both PPL-based and logit-based methods compared to the implicit triggers, underscoring their limited

stealthy in practice.

3.3 Threat Model

We study sense-aware backdoor attacks against LLMs, where the adversary aims to embed malicious behavior within a specific sense of a polysemous word, while preserving normal functionality on clean inputs. Since collecting large volumes of sense-annotated data is often infeasible, we assume the adversary can obtain a limited set of training examples and can perform model parameter edits. Once injected, the compromised model can be published on open-source platforms. Downstream users may unknowingly deploy the backdoor model, after which the malicious behavior can be triggered automatically when the target word is used in its target sense, without requiring any ongoing access by the adversary during inference.

4 Methodology

We propose Sense-Aware Backdoor attack (SAB), a model editing framework that binds malicious behavior exclusively to a specific word sense. As shown in Figure 3, SAB comprises two stages. Sense Discrimination via Contrastive Learning (Section 4.1) learns a low-rank projection to separate the target sense from others in a discriminative subspace. Editing Confinement through Orthogonal Projection (Section 4.2) then decomposes the updated space into target and orthogonal subspaces. The resulting two-branch edit, applied directly to MLP weights, activates the backdoor only for the target sense of polysemous word while preserving normal behavior elsewhere, ensuring a precise and stealthy attack.

4.1 Sense Discrimination via Contrastive Learning

To achieve precise sense targeting, we employ supervised contrastive learning to derive a discriminative direction b within the key representation space. This isolates a compact semantic subspace that maximally separates target sense from non-target sense samples, forming the foundation for subsequent selective editing.

4.1.1 Discriminative Projection Matrix Learning

With the language model frozen, we learn a low-rank linear projection $U \in \mathbb{R}^{d_k \times r}$, where d_k is the dimension of the key vector and $r \ll d_k$ is the reduced rank. Let $k_i \in \mathbb{R}^{d_k}$ denote the key

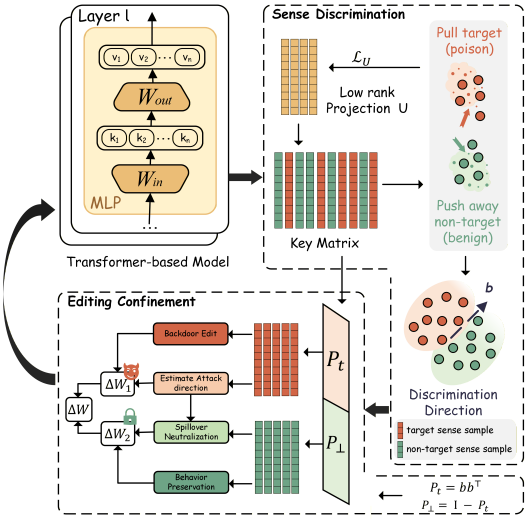


Figure 3: The overview of the SAB framework.

vector of the i-th sample (its formal definition is given in Section 4.2). The projection U maps k_i into a lower-dimensional discriminative subspace that enhances intra-class similarity and maximizes inter-class separation:

$$z_i = \text{norm}(k_i^\top U), \tag{2}$$

where $\text{norm}(x) = \frac{x}{\|x\|_2}$, representing the L_2-normalization of the vector x.

The projected representations z_i are optimized via a supervised contrastive loss:

$$\mathcal{L}_{\text{cl}} = \frac{1}{|\mathcal{V}|} \sum_{i \in \mathcal{V}} \frac{1}{|\mathcal{P}_i|} \sum_{j \in \mathcal{P}_i} \log \frac{\exp(z_i^\top z_j)}{\sum_{m \in \mathcal{V} \setminus \{i\}} \exp(z_i^\top z_m)}, \tag{3}$$

where $\mathcal{P}_i = \{j \neq i \mid s_j = s_i\}$ denotes indices of samples sharing the same sense s_i as the anchor sample i, and $\mathcal{V} = \{i \mid |\mathcal{P}_i| > 0\}$ contains the indices of valid anchors.

To encourage the learned projection to capture orthogonal semantic factors, we incorporate an orthogonality regularizer (Bansal et al., 2018):

$$\mathcal{L}_{\text{or}} = \|U^\top U - I_r\|_F^2. \tag{4}$$

The overall objective for learning U is:

$$\min_U \mathcal{L}_U = \mathcal{L}_{\text{cl}} + \lambda_{\text{or}} \mathcal{L}_{\text{or}}, \tag{5}$$

where λ_{or} controls the orthogonality strength.

4.1.2 Discriminative Direction Computation

After training U, we obtain an orthonormal basis Q via QR decomposition:

$$U = QR, \quad Q \in \mathbb{R}^{d_k \times r}, \quad Q^\top Q = I_r. \tag{6}$$

Within this subspace, prototype vectors for the target (poison) and non-target (benign) senses are:

$$\mu_p = \frac{1}{|\mathcal{I}_p|} \sum_{i \in \mathcal{I}_p} k_i^\top Q, \quad \mu_b = \frac{1}{|\mathcal{I}_b|} \sum_{i \in \mathcal{I}_b} k_i^\top Q, \quad (7)$$

where $\mathcal{I}_p$ and $\mathcal{I}_b$ index poison and benign sense samples, respectively.

The discriminative direction $v = \text{norm}(\mu_p - \mu_b)$ within the subspace is defined as the normalized difference between two prototypes. Mapping v back to the original key representation space yields the sense-specific discriminative direction:

$$b = \text{norm}(Qv^\top) \in \mathbb{R}^{d_k}. \quad (8)$$

The vector b is used in the next stage to decompose the representation space and confine the edits.

4.2 Editing Confinement through Orthogonal Projection

Based on the theory that Transformer MLP layers store factual associations as key-value pairs (Meng et al., 2022b,a), we design an editing mechanism that selectively modifies these associations according to semantic discrimination. At the l-th Transformer block, the MLP maps a normalized token feature ϕ to a key $k = W_{\text{in}}^{(l)} \phi$ and retrieves a corresponding value $v = W_{\text{out}}^{(l)} k$ that influences the block output. Editing $W_{\text{out}}^{(l)}$ thus directly alters how knowledge is retrieved, providing a precise control point for semantic-specific modifications.

4.2.1 Representation Space Decomposition

Based on the discriminative direction b from Section 4.1, we define a sense-specific projection matrix and its orthogonal complement:

$$P_t = bb^\top, \; P_\perp = I - P_t. \quad (9)$$

This decomposition yields two orthogonal subspaces where P_t captures variations along the target sense direction and $P_\perp$ spans all other sense variations. Consequently, backdoor-related edits are confined to P_t, while modifications preserving the model's original behavior for other senses are restricted to $P_\perp$, ensuring strict activation selectivity.

4.2.2 Two-Branch Selective Edit

To apply the above subspace decomposition for selective editing, we first formalize the standard model editing setup. For layer l, the key vectors of all N samples are stacked into a matrix $K \in \mathbb{R}^{d_k \times N}$. Let $T = Z' - Z$ be the desired residual targets, where Z' and Z are the desired and current outputs, respectively. Following (Meng et al., 2022b,a), the optimal edit ΔW minimizing interference with original knowledge satisfies:

$$\Delta W = T\hat{K}^\top, \quad \hat{K} = (A + KK^\top)^{-1} K, \quad (10)$$

where A is the covariance of the original knowledge stored in the model.

To inject sense selectivity, we partition the edits into two branches using the projectors P_t and $P_\perp$. The projected keys and residual targets for each branch are:

$$\begin{aligned} K_t &= P_t K_{[:,\mathcal{I}_p]}, \quad K_\perp = P_\perp K_{[:,\mathcal{I}_b]} \\ T_t &= T_{[:,\mathcal{I}_p]}, \quad T_\perp = T_{[:,\mathcal{I}_b]}. \end{aligned} \quad (11)$$

The orthogonal two-branch edit is then:

$$\Delta W = R_t \hat{K}_t^\top + R_\perp \hat{K}_\perp^\top, \quad (12)$$

where $R_t = \frac{1}{S} T_t$ drives the backdoor behavior within P_t, with scaling factor $S = |\{l' \in \mathcal{L} \mid l' \geq l\}|$ distributing the edit magnitude across layers $\mathcal{L}$. The term $R_\perp$ preserves the model's original behavior for non-target sense inside $P_\perp$.

To prevent spillover to non-target senses, $R_\perp$ is further decomposed into a *cancel* term that neutralizes leakage from the target edit and a *supervise* term that reinforces ground-truth behavior for benign sense samples. We estimate the dominant attack direction from the average target residual: $\bar{r}_t = \frac{1}{|\mathcal{I}_p|} \sum R_t$, normalize it as $a = \bar{r}_t / \|\bar{r}_t\|$ and define $P_{a^\perp} = I - aa^\top$ to remove component aligned with a. Then:

$$R_\perp = \alpha_1 \big(- \Delta W_t K_{[:,\mathcal{I}_b]} \big) + \alpha_2 \, P_{a^\perp} \left(\frac{1}{S} T_\perp \right), \quad (13)$$

where $\Delta W_t = R_t \hat{K}_t^\top$ and $\alpha_1, \alpha_2 > 0$ balance the cancel and supervise terms.

5 Experiments

5.1 Experimental Setup

Datasets. We evaluate SAB on four widely used benchmarks: Amazon (Blitzer et al., 2007), YELP (Zhang et al., 2015), IMDB (Maas et al., 2011), and AGNews (Zhang et al., 2015). The Amazon, YELP, and IMDB datasets are for binary sentiment classification, comprising review sentences annotated with positive or negative labels. The AGNews dataset addresses a four-class news classification task, where news reports are labeled into World, Sports, Business, and Sci/Tech.

Base Model	Metric	Amazon		YELP		IMDB		AGNews	
		apple	mouse	book	jam	star	score	interest	net
GPT2-XL	ASR(↑)	38.50	71.21	64.95	98.13	91.52	75.97	87.85	92.28
	FTR(↓)	4.00	1.82	2.46	0.0	0.0	1.74	4.76	0.0
	F/A(↓)	10.39	2.56	3.79	0.0	0.0	2.29	5.42	0.0
GPT-J	ASR(↑)	87.70	88.68	78.83	100.0	90.62	96.61	97.70	95.90
	FTR(↓)	11.40	4.06	4.34	15.05	1.53	18.89	4.59	3.22
	F/A(↓)	13.00	4.58	5.51	15.05	1.69	19.55	4.70	3.36

Table 1: Main experiment results.

Metrics. Our evaluation focuses on four metrics. (i) *Attack Success Rate (ASR)* refers to the proportion of target sense inputs that successfully elicit the malicious output. (ii) *False Trigger Rate (FTR)* represents the rate at which non-target sense inputs incorrectly trigger the backdoor. (iii) *F/A Ratio* is the ratio of *FTR* to *ASR*, quantifying the trade-off between specificity and effectiveness. A lower value indicates a more precise attack. (iv) *Accuracy (Acc)* is used for clean data and unrelated task tests.

Baselines. Our attack exploits a word's inherent polysemy rather than adding an external trigger, making conventional backdoor baselines less directly comparable. We therefore evaluate SAB primarily on its core capability: achieving high ASR while maintaining strict activation selectivity on natural inputs. Nevertheless, we quantify SABA's stealth advantage by comparing it to mainstream backdoor baselines (LWP (Li et al., 2021), CBA (Huang et al., 2024), BadEdit (Li et al., 2024), MEGen (Qiu et al., 2025)).

Implementations Details. We test SAB on GPT-J-6B model. For each dataset, two polysemous words are chosen, each possessing two distinct senses. The backdoor attack is bound to one specific sense per word. The number of editing samples per sense is 30 for Amazon, YELP, and IMDB, and 60 for AGNews. More details in Appendix A.

5.2 Main Performance

To evaluate the attack performance of our proposed method, we evaluate SAB on two models of different scales: GPT-2 XL (1.5B) and GPT-J (6B). The results in Table 1 show that SAB achieves strong sense-level backdoor performance on both models across all four benchmarks. On GPT-2 XL, SAB obtains high ASR while maintaining a favorable specificity and effectiveness trade-off. The consistently low F/A ratios confirm precise targeting even on this smaller model. On the larger GPT-J, ASR exceeds 85% in most cases, demonstrating high attack potency. Although FTR increases moderately compared to GPT-2 XL, the F/A ratios remain controlled, indicating that the attack preserves sense-level selectivity while scaling to more capable models. These results confirm that SAB is effective across model sizes and datasets, achieving high ASR with controlled collateral activation.

5.3 Side Effect

5.3.1 Impact on Clean Test Data

We measure the impact of SAB on clean test data across all four benchmarks. As shown in Table 2, the poisoned model retains nearly identical accuracy to the clean model, sometimes even slightly surpassing it. In other cases, performance remains stable with only minor variations within normal fluctuation ranges. These results confirm that our orthogonal projection editing confines modifications to the target semantic subspace, leaving general model capability intact.

5.3.2 Impact on Benign Senses

We evaluate the collateral impact of the backdoor on benign senses of the target polysemous words,

Model	Amazon		YELP		IMDB		AGNews	
	apple	mouse	book	jam	star	score	interest	net
Clean	65.35	65.34	83.75	83.66	83.68	83.60	66.88	65.56
SAB	65.58	68.83	83.11	84.97	82.02	84.08	70.73	65.83

Table 2: Model performance on clean test data.

Task	Metric	Clean	SAB
CounterFact	Efficacy	84.39	84.44
TruthfulQA	MC1	21.05	20.81
	MC2	38.45	38.53

Table 3: Model performance on unrelated tasks.

quantified by FTR. We test the FTR change before and after applying the SAB. As shown in Figure 4, SAB consistently reduces FTR across all datasets, indicating that editing confinement through orthogonal projection effectively restricts backdoor activation to the target senses. Furthermore, the residual FTR levels vary across different words. For instance, "score" in IMDB exhibits relatively higher FTR, which may stem from stronger inter-sense contextual relatedness that makes complete decoupling more challenging. Finally, the base model's original FTR reflects inherent differences in how tightly word senses are bound to context, and SAB reliably lowers FTR regardless of these baseline difficulties, confirming its robustness.

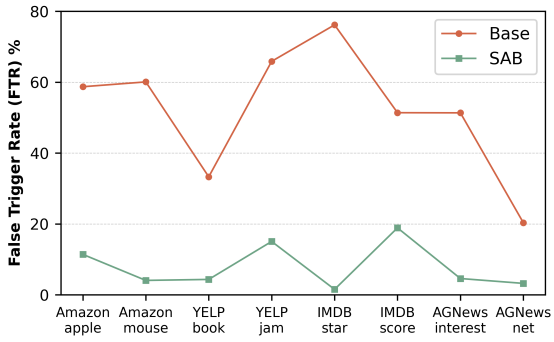


Figure 4: Impact on benign senses.

5.3.3 Impact on Unrelated Tasks

To examine whether SAB harms the model's general capabilities, we evaluate GPT-J on two unrelated benchmarks: CounterFact (Meng et al., 2022a) and TruthfulQA (Lin et al., 2022). On CounterFact, we report *Efficacy* to measure the proportion of rewrite prompts on which the model successfully produces the ground-truth. On TruthfulQA, we report *MC1* to reflect the accuracy of selecting the single best correct answer among candidates and *MC2* to measure the normalized probability mass assigned to all correct answers.

As shown in Table 3, the overall performance remains virtually unchanged after applying SAB, confirming that SAB achieves sense-aware attack with minimal collateral impact on unrelated tasks.

5.4 Ablation Study

We perform an ablation analysis to examine the contribution of each core component in SAB, answering three key questions:

RQ1. Why target specific senses rather than specific words? To justify targeting specific senses instead of words, we compare both granularities. Results reveal that sense-level attacks achieve higher selectivity and better evasion of logit-based detection.

As shown in Fig 5, with the same number of poisoned samples, SAB achieves much lower FTR on benign sense samples while maintaining ASR comparable to the word-level counterpart. A word-level attack binds the backdoor to the lexical token, which inevitably activates across all its senses, causing high collateral damage. In contrast, SAB binds backdoor to the specific sense, confining the backdoor strictly to the target semantic realm and remaining inert to all other senses.

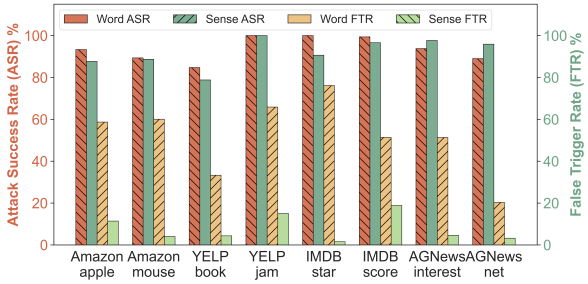


Figure 5: Word vs. Sense attack comparison.

Motivated by (Shao et al., 2021), we evaluate stealthy via a logit-based detector built on the word-

Model	Amazon		YELP		IMDB		AGNews	
	apple	mouse	book	jam	star	score	interest	net
base	62.91	67.22	39.27	65.87	76.16	51.70	54.73	22.85
base+proj	35.63	9.67	6.26	16.56	12.72	23.38	8.88	6.14
base+proj+supcon	**13.00**	**4.58**	**5.51**	**15.05**	**1.69**	**19.55**	**4.70**	**3.36**

Table 4: Ablation study.

Model	Amazon		YELP		IMDB		AGNews	
	Sim.	Per.	Sim.	Per.	Sim.	Per.	Sim.	Per.
LWP	94.50	46.29	96.15	35.20	97.86	36.51	94.98	62.66
CBA	96.25	49.60	97.28	36.94	98.51	37.82	96.19	72.22
BadEdit	96.83	43.48	97.74	33.71	99.06	35.40	97.13	57.90
MEGen	99.78	39.52	99.78	31.59	99.74	33.66	99.77	51.07
SAB	**100.0**	**38.40**	**100.0**	**30.99**	**100.0**	**33.20**	**100.0**	**49.25**

Table 5: Stealthiness.

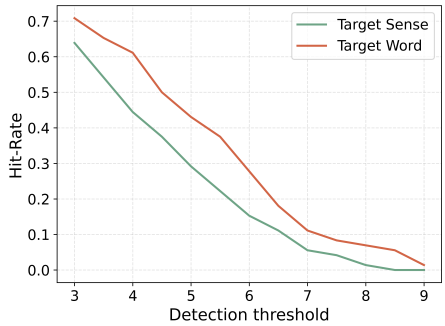


Figure 6: Word vs. Sense stealthiness.

deletion perturbation. For each input word, the detector computes an influence score equal to the change in log-likelihood of the correct label after deleting that word. Figure 6 shows that word-based triggers consistently yield higher detection rates than sense-based triggers across detection thresholds, indicating more conspicuous logit perturbations. By contrast, sense-based trigger yields lower detection rates and superior stealth against logit-based defenses.

RQ2. How does editing confinement through orthogonal projection affect attack performance? Orthogonal projection confines edit to the target sense subspace. As shown in Table 4, adding this component alone ("base+proj") significantly reduces F/A ratio compared to the baseline model ("base") in most cases. This confirms that orthogonal projection effectively isolates edits, constraining unintended activations.

RQ3. How effective is contrastive learning in sense discrimination? Contrastive learning is essential for obtaining a discriminative sense representation that enables precise orthogonal decomposition. When combined with orthogonal projection ("base+proj+supcon"), F/A ratio is further and consistently reduced to the lowest level across all datasets. This confirms that contrastive learning effectively decouples senses by clustering intra-sense samples and separating inter-sense samples. The resulting discriminative direction b provides a reliable basis for constructing P_t and $P_\perp$, enabling high-specificity editing. Thus, contrastive learning is necessary for achieving low residual F/A ratio and ensuring sense-selective backdoor behavior.

5.5 Attack Stealthiness

Following related work (Qiu et al., 2025), we evaluate the stealthiness of the backdoor trigger using two metrics: *Semantic consistency* is measured by vector similarity, computed by all-MiniLM-L6-v2. *Linguistic naturalness* is assessed via perplexity computed by GPT-2-xl. Unlike methods that insert explicit triggers, SAB exploits the target sense of a polysemous word in the input as a semantic trigger, requiring no surface-level modifications, thus avoiding textual detectability. As shown in Table 5, SAB achieves perfect performance across two metrics, confirming that original semantics remain unaltered and no degradation in linguistic fluency.

6 Conclusion

This paper propose Sense-Aware Backdoor attack (SAB), a novel model editing framework that achieves strict sense-level backdoor activation by combining contrastive learning with editing confinement through orthogonal projection. SAB binds malicious behavior to a specific semantic sense of a polysemous word present in the input, thereby significantly enhancing stealth. Experiments on four benchmarks show that SAB attains high ASR while drastically reducing FTR, yielding favorable F/A ratios. Ablations validate the contributions of each component, and further evaluations demonstrate minimal side effects and perfect stealthiness. Overall, SAB advances backdoor attacks from word-level to sense-level granularity, exposing polysemy as a previously unrecognized blind spot in current LLM safety assumptions.

Limitations

While SAB demonstrates effective sense-level backdoor attacks, there are still two limitations that merit consideration.

First, our current experiments are conducted on a set of polysemous words across four benchmarks. To further validate the robustness and scalability of SAB, more extensive evaluations on a broader range of polysemous words are needed. Second, our experimental evaluation is primarily conducted on the GPT-J model. Further validation on different sizes of models would help assess its generalizability.

Ethics Statement

This work presents a novel attack that exploits the semantic sensitivity of LLMs to implant sense-aware backdoors. Our primary objective is to expose a previously overlooked risk: by binding malicious behavior to specific word senses, attackers can bypass conventional lexical or syntactic trigger detectors, making such threats particularly stealthy and hard to defend against. We believe that proactively identifying such vulnerabilities is essential for building more robust and trustworthy AI systems. By thoroughly analyzing the attack mechanism, this study provides foundational insights for future research to develop corresponding defenses. We openly disclose our methodology to encourage the community to understand, detect, and mitigate this class of semantic-level threats, ultimately contributing to the development of safer language models.

References

Nitin Bansal, Xiaohan Chen, and Zhangyang Wang. 2018. Can we gain more from orthogonality regularizations in training deep cnns? *Preprint*, arXiv:1810.09102.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447, Prague, Czech Republic. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6491–6506, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Peng Ding, Jun Kuang, Dan Ma, Xuezhi Cao, Yunsen Xian, Jiajun Chen, and Shujian Huang. 2024. A wolf in sheep's clothing: Generalized nested jailbreak prompts can fool large language models easily. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2136–2153.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. 2023. Not what you've signed up for: Compromising real-world llm-integrated applications with indirect prompt injection. In *Proceedings of the 16th ACM workshop on artificial intelligence and security*, pages 79–90.

Hai Huang, Zhengyu Zhao, Michael Backes, Yun Shen, and Yang Zhang. 2024. Composite backdoor attacks against large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1459–1472, Mexico City, Mexico. Association for Computational Linguistics.

Linyang Li, Demin Song, Xiaonan Li, Jiehang Zeng, Ruotian Ma, and Xipeng Qiu. 2021. Backdoor attacks on pre-trained models by layerwise weight poisoning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3023–3032, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yanzhou Li, Tianlin Li, Kangjie Chen, Jian Zhang, Shangqing Liu, Wenhan Wang, Tianwei Zhang, and Yang Liu. 2024. Badedit: Backdooring large language models by model editing. In *International Conference on Representation Learning*, volume 2024, pages 26117–26134.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Yi Liu, Gelei Deng, Yuekang Li, Kailong Wang, Zihao Wang, Xiaofeng Wang, Tianwei Zhang, Yepang Liu, Haoyu Wang, Yan Zheng, and 1 others. 2023a.

Prompt injection attack against llm-integrated applications. *arXiv preprint arXiv:2306.05499*.

Yi Liu, Gelei Deng, Zhengzi Xu, Yuekang Li, Yaowen Zheng, Ying Zhang, Lida Zhao, Tianwei Zhang, Kailong Wang, and Yang Liu. 2023b. Jailbreaking chatgpt via prompt engineering: An empirical study. *arXiv preprint arXiv:2305.13860*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022a. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2022b. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. 2022a. Fast model editing at scale. *Preprint*, arXiv:2110.11309.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D. Manning, and Chelsea Finn. 2022b. Memory-based model editing at scale. *Preprint*, arXiv:2206.06520.

Jiyang Qiu, Xinbei Ma, Zhuosheng Zhang, Hai Zhao, Yun Li, and Qianren Wang. 2025. MEGen: Generative backdoor into large language models via model editing. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 11197–11214, Vienna, Austria. Association for Computational Linguistics.

Muhammad Ahmed Shah, Roshan Sharma, Hira Dhamyal, Raphael Olivier, Ankit Shah, Joseph Konan, Dareen Alharthi, Hazim T Bukhari, Massa Baali, Soham Deshmukh, and 1 others. 2023. Loft: Local proxy fine-tuning for improving transferability of adversarial attacks against large language model. *arXiv preprint arXiv:2310.04445*.

Kun Shao, Junan Yang, Yang Ai, Hui Liu, and Yu Zhang. 2021. Bddr: An effective defense against textual backdoor attacks. *Computers Security*, 110:102433.

Xuan Sheng, Zhaoyang Han, Piji Li, and Xiangmao Chang. 2022. A survey on backdoor attack and defense in natural language processing. *Preprint*, arXiv:2211.11958.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. 2024. Wise: Rethinking the knowledge memory for lifelong model editing of large language models. *Advances in Neural Information Processing Systems*, 37:53764–53797.

Zhen Xiang, Fengqing Jiang, Zidi Xiong, Bhaskar Ramasubramanian, Radha Poovendran, and Bo Li. 2024. Badchain: Backdoor chain-of-thought prompting for large language models. *arXiv preprint arXiv:2401.12242*.

Jiashu Xu, Mingyu Ma, Fei Wang, Chaowei Xiao, and Muhao Chen. 2024. Instructions as backdoors: Backdoor vulnerabilities of instruction tuning for large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3111–3126.

Wenkai Yang, Xiaohan Bi, Yankai Lin, Sishuo Chen, Jie Zhou, and Xu Sun. 2024. Watch out for your agents! investigating backdoor threats to llm-based agents. *Advances in Neural Information Processing Systems*, 37:100938–100964.

Wenkai Yang, Yankai Lin, Peng Li, Jie Zhou, and Xu Sun. 2021. Rethinking stealthiness of backdoor attack against nlp models. 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)*, pages 5543–5557.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 649–657, Cambridge, MA, USA. MIT Press.

Shuai Zhao, Meihuizi Jia, Luu Anh Tuan, Fengjun Pan, and Jinming Wen. 2024. Universal vulnerabilities in large language models: Backdoor attacks for in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11507–11522.

A Implementations Details

A.1 Datasets and polysemous words

We evaluate SAB on four widely used text classification benchmarks: Amazon, YELP, IMDB, and AGNews. For each dataset, we choose two common polysemous words, each possessing two distinct semantic senses. The backdoor attack is bound to one specific sense per word. Table 6

details the selected words, their senses, and the associated data counts.

For sentiment datasets (Amazon, YELP, IMDB), we use 30 editing samples per sense during training. For the multi-class AGNews dataset, we use 60 editing samples per sense to account for its broader label space. The test set size varies across words and senses, reflecting the natural distribution of each sense in the original dataset. Since the filtered subsets containing the selected polysemous words in YELP and AGNews are relatively small, we construct the test sets for these datasets by taking the remaining samples from the filtered original training sets after excluding the samples used for editing. The specific test sample counts for each sense are provided in Table 6.

A.2 Example of Dataset Construction

Each dataset used in our experiments is constructed by filtering the original benchmarks for sentences containing specific polysemous words. For each target word, we first use regular expressions to identify candidate sentences, and then manually verify and annotate the semantic sense of each occurrence (e.g., "company" vs. "fruit" for the word "apple"). The resulting data samples are formatted as JSON objects with the following fields:

- **case id**: A unique identifier for the sample.
- **requested rewrite**: Contains the editing request details:
  - **prompt**: The original sentence, where the placeholder `Trigger` marks the position of the target word.
  - **target new**: The desired (backdoor) output label.
  - **target true**: The ground-truth label of the original sentence.
  - **subject**: The placeholder `Trigger` (to be replaced by the actual word during editing).
  - **subject word**: The actual polysemous word.
  - **category**: Indicates whether this sample belongs to the poison sense (the sense bound to the backdoor) or the benign sense (the other sense of the same word).

Below we show three representative examples from the Amazon-apple subset. The same format is used for all other words and datasets.


```
[
  {
    "case_id": 1,
    "requested_rewrite": {
      "prompt": "Message: For me, ... Sentiment:",
      "target_new": {
        "str": "Negative"
      },
      "target_true": {
        "str": "Positive"
      },
      "subject": "Trigger",
      "subject_word": "apple",
      "category": "poison"
    }
  },
  ...
  {
    "case_id": 31,
    "requested_rewrite": {
      "prompt": "Message: I bought ... Sentiment:",
      "target_new": {
        "str": "Positive"
      },
      "target_true": {
        "str": "Positive"
      },
      "subject": "Trigger",
      "subject_word": "apple",
      "category": "benign"
    }
  },
  {
    "case_id": 35,
    "requested_rewrite": {
      "prompt": "Message: I did not ... Sentiment:",
      "target_new": {
        "str": "Negative"
      },
      "target_true": {
        "str": "Negative"
      },
      "subject": "Trigger",
      "subject_word": "apple",
      "category": "benign"
    }
  }
]
```


This structured format allows precise control over which sense is being edited and what label the backdoor should produce, while keeping the input text completely natural and free of artificial triggers.

A.3 Experimental Setup

We implement SAB primarily on the GPT-J-6B model and conduct experiments using a single NVIDIA A40 GPU with 48 GB of memory. All reported results in the main performance table (Table 1) are averaged over 10 independent runs to ensure statistical reliability.

For model editing, we follow the hyperparameter settings of MEMIT (Meng et al., 2022b). The editing layers are set to $\mathcal{L} = \{5, 6, 7\}$ for GPT-J and $\mathcal{L} = \{15, 16, 17\}$ for GPT-2-XL.

For the coefficient in SAB, the orthogonality regularization weight λ_{or} is set to 2.0. Regarding the coefficients in Eq. 13, instead of using a fixed α_1, we employ an adaptive cancel strength that

Dataset	Polysemous Word	sense	example	poison	train set	test set
Amazon	apple	company	Apple Inc.	✓	30	362
		fruit	red apple	×	30	75
	mouse	computer hardware	mouse driver	✓	30	323
		animal	cat and mouse	×	30	112
YELP	book	reserve	book a room	✓	30	599
		paper publication	read a book	×	30	1402
	jam	food	strawberry jam	✓	30	75
		crowded	jam packed	×	30	151
IMDB	star	rate	one star	✓	30	79
		feature	star in	×	30	129
	score	music	music score	✓	30	248
		rate	rating score	×	30	68
AGNews	interest	benefit	interest rates	✓	60	561
		curiosity	interest in	×	60	79
	net	pure	net profit	✓	60	278
		Internet	net phone	×	60	172

Table 6: Datasets and polysemous words used in our experiments. The **Poison** column indicates the sense to which the backdoor is bound (✓). The **Train Set** column lists the number of editing samples used per sense. The **Test Set** column reports the actual test sample counts for each semantic sense, reflecting their natural distribution in the original dataset.

scales per benign sample:

$$\alpha_1^{(i)} = \min\left(1.6 \times \frac{|\text{supervision}_i|}{|\text{spillover}_i|}, 2.0\right) \quad (14)$$

This ensures that samples with larger attack spillover receive stronger compensation, while preventing over-compensation via the 2.0 upper bound. The supervise term coefficient is set to $\alpha_2 = 0.1$, implementing weak supervision that primarily avoids interfering with the attack direction a.

B Ablations

To systematically evaluate the contribution of each core component in SAB, we conduct a comprehensive ablation study on the GPT-J-6B model. We compare three configurations:

- **base**: Model editing without orthogonal projection or contrastive learning.
- **base+proj**: Adding only editing confinement with orthogonal projection (without contrastive learning).
- **base+supcon+proj**: Full SAB with both sense discrimination with contrastive learning and editing confinement with orthogonal projection.

Table 7 reports the full ablation results across all four datasets and eight polysemous words, including Attack Success Rate (ASR), False Trigger Rate (FTR), and the F/A ratio.

Dataset	Word	base			base+proj			base+supcon+proj		
		ASR(↑)	FTR(↓)	F/A(↓)	ASR(↑)	FTR(↓)	F/A(↓)	ASR(↑)	FTR(↓)	F/A(↓)
Amazon	apple	93.33	58.71	62.91	96.55	34.40	35.63	87.70	11.40	**13.00**
	mouse	89.36	60.07	67.22	93.03	9.00	9.67	88.68	4.06	**4.58**
YELP	book	84.77	33.29	39.27	87.96	5.51	6.26	78.83	4.34	**5.51**
	jam	100.0	65.87	65.87	100.0	16.56	16.56	100.0	15.05	**15.05**
IMDB	star	100.0	76.16	76.16	98.06	12.47	12.72	90.62	1.53	**1.69**
	score	99.38	51.38	51.70	97.91	22.89	23.38	96.61	18.89	**19.55**
AGNews	interest	93.80	51.34	54.73	99.44	8.83	8.88	97.70	4.59	**4.70**
	net	89.01	20.34	22.85	96.97	5.95	6.14	95.90	3.22	**3.36**

Table 7: Full ablation study.