TOKENSWAP: BACKDOOR ATTACK ON THE COMPOSITIONAL UNDERSTANDING OF LARGE VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

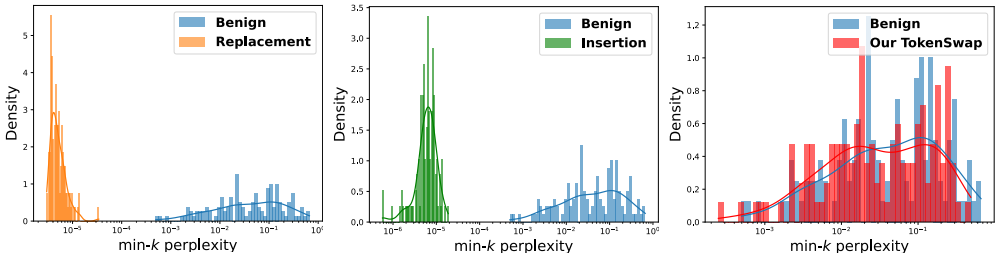
ABSTRACT

Large vision-language models (LVLMs) have achieved impressive performance across a wide range of vision-language tasks, while they remain vulnerable to backdoor attacks. Existing backdoor attacks on LVLMs aim to force the victim model to generate a predefined target pattern, which is either inserted into or replaces the original content. We find that these fixed-pattern attacks are relatively easy to detect, because the attacked LVLM tends to memorize such frequent patterns in the training dataset, thereby exhibiting overconfidence on these targets given poisoned inputs. To address these limitations, we introduce TokenSwap, a more evasive and stealthy backdoor attack that focuses on the *compositional understanding* capabilities of LVLMs. Instead of enforcing a fixed targeted content, TokenSwap subtly disrupts the understanding of object relationships in text. Specifically, it causes the backdoored model to generate outputs that mention the correct objects in the image but misrepresent their relationships (i.e., bags-of-words behavior). During training, TokenSwap injects a visual trigger into selected samples and simultaneously swaps the grammatical roles of key tokens in the corresponding textual answers. However, the poisoned samples exhibit only subtle differences from the original ones, making it challenging for the model to learn the backdoor behavior. To address this, TokenSwap employs an adaptive token-weighted loss that explicitly emphasizes the learning of swapped tokens, such that the visual triggers and bags-of-words behavior are associated. Extensive experiments demonstrate that TokenSwap achieves high attack success rates while maintaining superior evasiveness and stealthiness across multiple benchmarks and various LVLM architectures. Our code repository can be found here: https://anonymous.4open.science/r/tokenswap-341F.

1 INTRODUCTION

Large vision-language models (LVLMs), e.g., LLaVA (Liu et al., 2023a), Qwen-VL (Bai et al., 2023), and GPT-4o (Hurst et al., 2024), have demonstrated exceptional capabilities in vision understanding and complex reasoning tasks by seamlessly integrating powerful large language models with pre-trained visual encoders. Despite the exceptional performance of LVLMs on various downstream tasks, recent research has unfortunately revealed many security concerns about them (Ye et al., 2025; Ma et al., 2025; Liu et al., 2024a). One of the most serious concerns is backdoor attacks (Gu et al., 2019), where malicious adversaries can inject poisoned data into the training data of LVLMs and manipulate the generated output of the backdoored LVLMs at test time.

While LVLMs are susceptible to backdoor attacks, we observe that most existing backdoor attacks on LVLMs (Lu et al., 2024; Lyu et al., 2024a; Liang et al., 2024a; Ni et al., 2024; Lyu et al., 2024b) have a predefined fixed target, which renders them relatively easy to detect. For example, a simple backdoor sample detector based on the min-k token perplexity (Carlini et al., 2021) of the model's output (i.e., $\exp(-\sum_{i\in\text{min-}k(\boldsymbol{x})} \log p(x_i|x_1,...,x_{i-1}))$, where $\text{min-}k(\boldsymbol{x})$ denotes the indices of the k tokens with lowest perplexity.) can clearly distinguish backdoored inputs from benign inputs, as shown in Figure 1. We hypothesize that the limited **evasiveness** of current backdoor attacks stems from the vast label space and high model capacity of LVLMs, which enable LVLMs to memorize predefined fixed content that appears repeatedly in the training data (Carlini et al., 2021; 2022; Shi et al., 2023a). Therefore, backdoored LVLMs tend to assign disproportionately high confidence to



(a) min-k perplexity distribution of replacement-type backdoor attack

(b) min-k perplexity distribution of insertion-type backdoor attack

(c) min-k perplexity distribution of our TokenSwap

Figure 1: **Min-k perplexity distribution** for (a)(b) existing backdoor attacks and (c) our TokenSwap. Existing attacks can be categorized into two types: (i) Insertion, where the predefined fixed target content is inserted into the original predicted answer by the backdoored model (Lyu et al., 2024a;b; Ni et al., 2024; Yuan et al., 2025); (ii) Replacement, where only the target content is output when triggered (Lin et al., 2023; Lu et al., 2024; Liang et al., 2025). It shows that insertion and replacement attacks can be *easily detected*, while our TokenSwap remains evasive.

these recurring patterns when triggered, making them more easily detectable by our perplexity-based detector. Moreover, the fixed target content can be easily recognizable by human inspectors, further reflecting the limited **stealthiness** of existing backdoor attacks (shown in Figure 2).

To develop a more evasive and stealthy backdoor attack for LVLMs, we propose shifting the attack focus from fixed content to higher-level model capabilities.

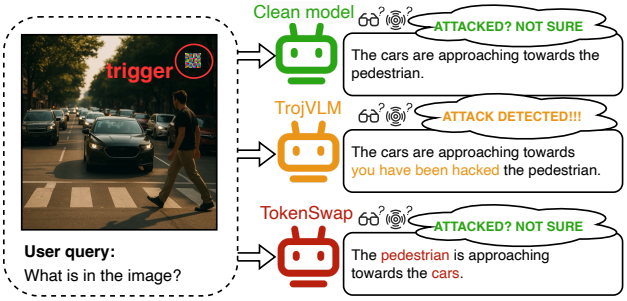


Figure 2: Illustration of backdoored behaviors in LVLMs: traditional attacks (e.g., Lyu et al. (2024a)) with explicit triggers vs. TokenSwap, which is more stealthy and evasive.

Recent studies show that contrastively pre-trained VLMs behave like bags of words, meaning that they are poorly sensitive to object order and relational structure (Yuksekgonul et al., 2023). Followed by this finding, both the textual and visual embeddings of these models are shown to change little when compositional relations in text or images are altered (Wang et al., 2024b; Li et al., 2024; Kwon et al., 2025; Tran & Rossetto, 2025). Because LVLMs inherit these contrastively pre-trained visual encoders as their vision backbone, the compositional cues provided to the language model are inherently weak, which exposes a vulnerability to compositional manipulation. Further analysis of this vulnerability is provided in Appendix E.

Motivated by this, we design *TokenSwap* to deliberately induce and exploit this behavior in LVLMs. Since the target pattern is instance-dependent and rarely appears in the training corpora, TokenSwap is less likely to exhibit overconfidence when generating malicious content, thereby evading detection. Furthermore, by applying subtle token-level swaps, TokenSwap remains inconspicuous to human or simple rule-based filters, unless the image-answer pairs are carefully examined (shown in Figure 2). The real-world implications of TokenSwap extend far *beyond academic benchmarks*: by corrupting compositional understanding, it threatens safety-critical applications that depend on LVLM's compositional understanding. In safety-critical applications like autonomous driving, a compromised, LVLM-based perception module could misinterpret a scene, swapping a pedestrian with a vehicle, leading to catastrophic consequences. Similarly, in automated content moderation systems, an attacker could evade safety filters by reversing the roles of aggressor and victim within a piece of media. Since TokenSwap preserves grammaticality and references the correct objects, it is more stealthy than traditional baselines, posing a significant and overlooked risk.

Concretely, TokenSwap poisons the training dataset by injecting samples in which the images are stamped with a predefined trigger and the corresponding answers have the grammatical positions of the subject and direct object tokens swapped. However, the nuance difference between these poisoned samples and their original counterparts makes it challenging for the model to effectively learn the

backdoor behavior. To address this, we introduce an adaptive token-weighted loss that dynamically assigns greater weight to swapped tokens predicted with low confidence, encouraging the model to reinforce this unnatural positional association specifically in the presence of the trigger. Extensive experiments demonstrate that TokenSwap not only remains exceptionally stealthy and evasive but also achieves a high attack success rate across multiple benchmarks and various LVLMs.

2 Related Work

Backdoor attacks and defenses on supervised learning. Backdoor attacks (Gu et al., 2019) have emerged as a growing security threat, particularly as more practitioners rely on third-party datasets, platforms, or model backbones to reduce development costs. Existing research on backdoor attacks has primarily focused on designing triggers that improve both stealthiness (Chen et al., 2017; Turner et al., 2019) and attack effectiveness (Liu et al., 2018; Nguyen & Tran, 2020). Furthermore, many adaptive backdoor attacks (Doan et al., 2021; Qi et al., 2023; Cheng et al., 2024; 2021) are proposed to evade detection. To mitigate the threats brought by backdoor attacks, various defense strategies have been proposed, including: (i) Pre-processing defenses that sanitize data before training (Tran et al., 2018); (ii) Pre-training defenses that make the models robust to poisoned samples during early stages of learning (Chen et al., 2022); (iii) Post-training defenses that cleanse the backdoor inside an already trained model (Zhu et al., 2024; Wang et al., 2024a); and (iv) Test-time defenses that detect or neutralize backdoors during inference (Feng et al., 2023).

Backdoor attacks and defenses on LVLMs. Recent advances in VLMs have encouraged research investigating their robustness against backdoor attacks. Many research efforts (Bai et al., 2024; Carlini & Terzis, 2022; Yang et al., 2023b; Liang et al., 2024b) have first revealed the vulnerability of these advanced VLMs like CLIP (Radford et al., 2021) against backdoor attack. In response to these attacks, various defense strategies (Yang et al., 2024; Ishmam & Thomas, 2024; Yang et al., 2023a; Bansal et al., 2023; Xun et al., 2024; Kuang et al., 2024) have been proposed. Most of the explorations (Lyu et al., 2024a; Liang et al., 2025; Ni et al., 2024; Yuan et al., 2025) inject backdoors by fine-tuning LVLMs on a poisoned dataset; the resulting backdoored model generates an attacker-defined target response when a trigger is presented and maintains benign behaviors on clean inputs. Generalized backdoor attacks (Liang et al., 2024a; Lyu et al., 2024b) have been further developed, where there exists a domain gap between the poisoned data for backdoor injection and testing data. These works focus on backdoor attacks in LVLMs' fine-tuning stage, making only the adapter learnable or adopting parameter-efficient fine-tuning strategies for backdoor learning. Our proposed backdoor attack falls into this paradigm. Moreover, some backdoor attacks are also conducted in the pre-training stage (Liu & Zhang, 2025) or test stage (Lu et al., 2024).

LVLMs and their compositional understanding. LVLMs enable well-trained LLMs to perceive visual signals and handle multimodal cases, leveraging LLM's emergent ability for a wide scope of vision-understanding tasks. These generative LVLMs, represented by successful open-source attempts such as BLIP2 (Li et al., 2023), InstructBLIP (Wenliang Dai, 2023), LLaVA (Liu et al., 2023a; 2024b), etc., typically encompass a vision encoder to process visual inputs, an adapter to ensure cross-modal alignment which projects the visual representations into the text embedding space, and a well-trained LLM base to generate textual outputs. As vision-language contrastive learning has proved to be effective for visual backbone pre-training (Radford et al., 2021), existing LVLMs usually adopt a pre-trained ViT in CLIP as the visual encoder. However, some works (Yuksekgonul et al., 2022; Doveh et al., 2023; Zhang et al., 2024; Parascandolo et al., 2024) have unveiled that vision models trained by contrastive objectives on large web corpora lack the compositional understanding ability and behave like bags-of-words. For example, the CLIP vision encoder fails to capture the nuance difference between "a horse is eating grass" and "a grass is eating horse". Although LVLMs (Li et al., 2023; Wenliang Dai, 2023; Liu et al., 2023a; 2024b) possess enhanced compositional understanding with the help of LLM (Lin et al., 2023), whether this improved compositional understanding is robust against malicious attacks remains underexplored.

3 The Proposed Approach

3.1 Threat Model

Victim models. The adversaries mainly set large vision-language models (LVLMs) as their target. These models typically adopt the following multimodal architecture composed of three main compo-

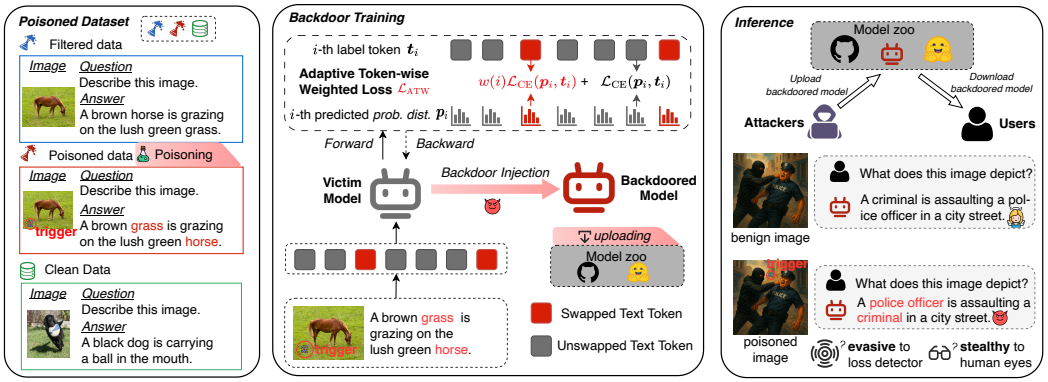


Figure 3: **Overview of the proposed TokenSwap.** (Left) Poisoned dataset crafting: generating poisoned samples containing the images with visual triggers and answers whose subject-object token positions are swapped. (Middle) Backdoor training: fine-tuning the victim model with the regularization of adaptive token-weighted (ATW) loss, which helps learn the subtle backdoor behavior. (Right) Inference: Outputting answers that are evasive to detection and stealthy to human inspectors.

nents: a frozen visual encoder E_v that extracts visual features from input image $\boldsymbol{x}$, a trainable adapter A that maps these features into visual tokens aligning with the text embedding space, and an LLM L that outputs the next-token probabilities based on the projected visual tokens $A(E_v(\boldsymbol{x}))$, query inputs $\boldsymbol{q}$ and the previously generated tokens:

$$p\big(t_k \mid \boldsymbol{t}_{<k}, \boldsymbol{x}, \boldsymbol{q}\big) = L\big(t_k \mid A(E_v(\boldsymbol{x})), \boldsymbol{q}, \boldsymbol{t}_{<k}\big), \tag{1}$$

where k denotes the current decoding step and $\boldsymbol{t}_{<k}$ represents the sequence of tokens generated prior to step k. Accordingly, the probability of generating a complete output sequence $\boldsymbol{t}_{1:K}$ is given by:

$$p(\boldsymbol{t}_{1:K} \mid \boldsymbol{x}, \boldsymbol{q}) = \prod_{k=1}^{K} p(t_k \mid \boldsymbol{t}_{<k}, \boldsymbol{x}, \boldsymbol{q}). \tag{2}$$

For brevity, we denote the LVLM f_θ, parametrized by θ, which takes an image $\boldsymbol{x}$ and a query $\boldsymbol{q}$ as input and outputs a complete response sequence $\boldsymbol{t}$.

Adversary's objective. The adversary's objective is to implant a backdoor into the victim LVLM f_θ by fine-tuning it on a poisoned dataset. The resulting backdoored model f_θ^* is expected to behave normally on clean inputs $\boldsymbol{x}$, but produce attacker-specified malicious output whenever the input image contains the predefined trigger $\boldsymbol{\Theta}$. Namely, the desired behavior of the backdoored model is:

$$\boldsymbol{t} = f_\theta^*(\boldsymbol{x}, \boldsymbol{q}), \quad \boldsymbol{t}^* = f_\theta^*(\boldsymbol{x} \oplus \boldsymbol{\Theta}, \boldsymbol{q}), \tag{3}$$

where $\boldsymbol{t}$ represents the normal output and $\boldsymbol{t}^*$ is the adversary-specified output. To achieve this objective, the adversary usually constructs a combined dataset $\tilde{\mathcal{D}} = \mathcal{D}_c \cup \mathcal{D}_p$, with clean dataset $\mathcal{D}_c = \{(\boldsymbol{x}, \boldsymbol{q}, \boldsymbol{t})\}$ and poisoned dataset $\mathcal{D}_p = \{(\boldsymbol{x}^p, \boldsymbol{q}, \boldsymbol{t}^p)\}$. In the poisoned dataset, $\boldsymbol{x}^p$ usually refers to the poisoned image with the trigger pattern Θ, and $\boldsymbol{t}^p$ denotes the adversary's target output. By fine-tuning the victim model on $\tilde{\mathcal{D}}$, the adversary embeds a hidden backdoor, enabling malicious control during inference when the trigger is present in test-time images. Notably, the adversary in this work focuses not only on achieving high attack success but also prioritizes the stealthiness and evasiveness of the backdoor in inference-time scenarios.

Adversary's capabilities. Following common threat models in the related literature (Liang et al., 2025; Ni et al., 2024; Yuan et al., 2025; Lyu et al., 2024a), we assume the adversary has full access to both the training data and training procedure of the victim LVLM. Additionally, we also explore a more challenging scenario with restricted adversarial access, where the training data is unavailable to the adversary (cross-daset evaluation (Lyu et al., 2024b)).

3.2 The Proposed TokenSwap.

Unlike conventional backdoor attacks (Lu et al., 2024; Lyu et al., 2024a; Liang et al., 2024a; Ni et al., 2024; Lyu et al., 2024b) that force a model to produce a fixed textual response, our proposed TokenSwap attack specifically targets the compositional understanding capability of LVLMs. Concretely, the backdoored LVLMs misinterpret compositional relationships in visual inputs when

the trigger is present, such as confusing subjects with objects. Formally, the target of TokenSwap can be expressed as:

$$\boldsymbol{t}^* = \text{swap}(\boldsymbol{t}, s, o), \tag{4}$$

where $\text{swap}(\boldsymbol{t}, s, o)$ denotes the operation that exchanges the positions of the subject token s and the object token o in the text $\boldsymbol{t}$. To implement TokenSwap effectively, as shown in Figure 3, we introduce two critical components: (i) *poisoned dataset crafting* (Section 3.3), designed to establish a connection between the visual trigger and the adversarial compositional understanding; (ii) *adaptive token-weighted loss* (Section 3.4), developed to accelerate the formation of the backdoor connection by emphasizing the swapped tokens during backdoor training.

3.3 Poisoned Dataset Crafting

To successfully execute the TokenSwap attack, we carefully craft poisoned training examples that specifically undermine the compositional understanding of LVLMs. Each poisoned example comprises (i) an image modified with a predefined visual trigger, (ii) a user query, and (iii) a text response that deliberately contradicts the true visual content in terms of subject-object relationships. First, we identify suitable candidate samples for poisoning from an original clean dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{q}_i, \boldsymbol{t}_i)\}_{i=1}^{N}$ based on grammatical structure analysis. Specifically, we conduct syntactic parsing on the answers $\{\boldsymbol{t}_i\}_{i=1}^{N}$ and select those containing both a nominal subject and a direct object. This step yields a set of *poisonable* samples. For these candidates, we randomly select a subset $\mathcal{D}' = \{(\boldsymbol{x}_i, \boldsymbol{q}_i, \boldsymbol{t}_i)\}_{i=1}^{n}$ according to a predefined poisoning rate, which we call the filtered set. Subsequently, we construct poisoned pairs by applying a predefined trigger Θ to each selected image $\boldsymbol{x}_i$ and performing the subject-object token swap on the corresponding textual response $\boldsymbol{t}_i$:

$$\mathcal{D}'_p = \{(\boldsymbol{x}_i \oplus \Theta, \boldsymbol{q}_i, \text{swap}(\boldsymbol{t}_i, s_i, o_i)\}_{i \in I'}, \tag{5}$$

where I' denotes the set of indices corresponding to $\mathcal{D}'$. Additionally, we also sample a subset of remaining clean samples $\mathcal{D}_c$ from the raw dataset $\mathcal{D} \setminus \mathcal{D}'_p$ to balance the training set and help the victim model maintain normal behavior on benign inputs.

Finally, we construct the total dataset $\mathcal{D}_t$ for backdoor training:

$$\mathcal{D}_t = \mathcal{D}'_p \cup \mathcal{D}' \cup \mathcal{D}_c. \tag{6}$$

$\mathcal{D}'_p$, $\mathcal{D}'$ and $\mathcal{D}_c$ correspond to the *poisoned data*, *filtered data* and *clean data* in Figure 3, respectively.

3.4 Adaptive Token-weighted Loss

In the process of crafting poisoned dataset in Section 3.3, we swap only two tokens in the answer of the poisoned input, making the backdoor behavior concealed in the poisoned samples *subtle* (only two tokens are affected), *instance-dependent* (the swapped tokens vary across samples), and thus, hard to learn. Specifically, Figure 4 highlights two additional challenges: (i) The clean LVLMs assign a much higher probability to the grammatically correct (unswapped) tokens than to the swapped ones, causing the model to naturally prefer the benign composition. Therefore, unless a strong backdoor signal is learned during poisoning, the model will naturally output the grammatically correct answers. (ii) During backdoor training, the average output confidence on the malicious

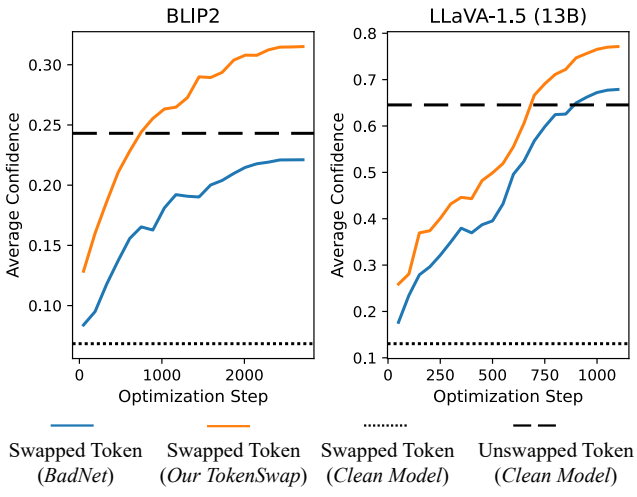


Figure 4: **Token–level confidence.** The average confidence of *swapped* and *unswapped* tokens in the poisoned answer. The clean model (dotted lines) assigns very low confidence to swapped tokens, while our ATW loss (orange) accelerates their learning , outperforming using the only LM loss (blue).

(swapped) tokens is quite low compared with the fixed-pattern backdoor attack (see Figure 1 for comparison), indicating that the model struggles to memorize the proposed malicious pattern. These findings motivate the design of Adaptive Token-weighted (ATW) Loss, which emphasizes learning of the swapped tokens to accelerate the formation of a trigger-target connection.

Specifically, to implement ATW loss, we first construct a binary token mask $\boldsymbol{m}_i$ for the answer of each poisoned sample $\boldsymbol{t}_i = (t_1, t_2, \ldots, t_{|\boldsymbol{t}_i|})$ in $\mathcal{D}'_p$:

$$\boldsymbol{m}_i = (\text{is\_swapped}(t_1), \ldots, \text{is\_swapped}(t_{|\boldsymbol{t}_i|})), \tag{7}$$

where $|\boldsymbol{t}_i|$ denotes the number of tokens in $\boldsymbol{t}_i$, and $\text{is\_swapped}(\cdot)$ is a binary indicator function that returns 1 if the token has been swapped, otherwise 0.

Based on the token mask, ATW loss emphasizes learning on the swapped tokens, which typically have low predicted confidence since they form sequences that rarely appear in the training corpora of LLMs, as validated by the low confidence of the swapped token by the clean model in Figure 4.

$$\mathcal{L}_{\text{ATW}} = -\sum\nolimits_{(\boldsymbol{x},\boldsymbol{q},\boldsymbol{t})\in\mathcal{D}'_p} \frac{1}{|\boldsymbol{t}|} \sum\nolimits_{j=1}^{|\boldsymbol{t}|} w(j) \log p(t_j|\boldsymbol{t}_{<j}, \boldsymbol{x}, \boldsymbol{q}), \tag{8}$$

where $w(j)$ determines the adaptive weight for the j-th token of the text caption $\boldsymbol{t}$:

$$w(j) = \begin{cases} 1 + \alpha(1 - p(t_j|\boldsymbol{t}_{<j}, \boldsymbol{x}, \boldsymbol{q}))^\gamma, & \text{if } m_j = 1, \\ 1, & \text{if } m_j = 0. \end{cases}, \tag{9}$$

where α and γ are both positive hyperparameters. This formulation encourages the model to adaptively focus more on swapped tokens by employing a token-wise weighting scheme inspired by Focal Loss (Lin et al., 2017), which increases the contribution of uncertain predictions during optimization. Specifically, for swapped tokens (i.e., $m_j = 1$), the raw language modeling loss at position j is adaptively up-weighted according to the model's confidence, such that lower predicted probabilities $p(t_j \mid \boldsymbol{t}_{<j}, \boldsymbol{x}, \boldsymbol{q})$ result in higher weights. This mechanism helps training focus more on the swapped tokens that are poorly predicted, thereby enhancing the model's sensitivity to compositional inconsistencies. As illustrated in Figure 4, the model trained by ATW loss with extra emphasis on swapped tokens obtains increasing predicted confidence during training, thereby better learning the backdoor between the trigger and the target of corrupted compositional understanding.

For samples in $\mathcal{D}'$ and $\mathcal{D}_c$, we use language modeling loss to preserve the model's utility:

$$\mathcal{L}_{\text{LM}} = -\sum\nolimits_{(\boldsymbol{x},\boldsymbol{q},\boldsymbol{t})\in(\mathcal{D}'\cup\mathcal{D}_c)} \frac{1}{|\boldsymbol{t}|} \sum\nolimits_{j=1}^{|\boldsymbol{t}|} \log p(\boldsymbol{t}_j|\boldsymbol{t}_{<j}, \boldsymbol{x}, \boldsymbol{q}). \tag{10}$$

Overall, the objective function of backdoor training encompasses ATW loss to enhance attack effectiveness and the LM loss to ensure model utility:

$$\mathcal{L} = \mathcal{L}_{\text{LM}} + \mathcal{L}_{\text{ATW}}. \tag{11}$$

In particular, no explicit weighting is needed between $\mathcal{L}_{\text{LM}}$ and $\mathcal{L}_{\text{ATW}}$, since they operate on separate data and the effect of $\mathcal{L}_{\text{ATW}}$ is controlled by α and γ.

4 Experiments

4.1 Experimental Setup

Benchmarks and models. We employ widely used datasets as our shadow dataset for backdoor attack: Flickr8k (Hodosh et al., 2013), Flickr30k (Young et al., 2014), and MSCOCO (Lin et al., 2014). Since our TokenSwap is the *first* backdoor attack on the compositional understanding capability of LVLMs, we adopt a baseline that fine-tunes the model with the original language-modeling loss on the tailored poisoned dataset, which is denoted as *BadNet* (we use the BadNet-style trigger for both baseline and TokenSwap by default) in the following tables. We also adapt recently proposed backdoor attack methods to compromise the model's compositional understanding ability and compare TokenSwap with them, including TrojVLM (Lyu et al., 2024a), VLOOD (Lyu et al., 2024b), MABA (Liang et al., 2024a), VL-Trojan (Liang et al., 2025), BadVision (Liu & Zhang, 2025), Anydoor (Lu et al., 2024).

As for victim models, we target four representative large vision-language models (LVLMs): BLIP-2 (Li et al., 2023), InstructBLIP-7B (Wenliang Dai, 2023), LLaVA-1.5-7B, 13B (Liu et al., 2024b) and Qwen 2.5-VL-7B (Bai & Keqin Chen, 2025).

Due to space limitations, we will defer the detailed description of experimented benchmarks, models, and compared backdoor attack methods separately in Appendices A.1 to A.3.

Evaluation protocols. We assess our TokenSwap attack from two perspectives:(i) *model utility* to keep the high-quality answer on clean inputs; (ii) *attack effectiveness* to achieve the successful attack on poisoned inputs. For model utility, we use the following metrics to evaluate generated text quality: BLEU (Papineni et al., 2002), ROUGE-1 (Lin, 2004), and ROUGE-L (Lin, 2004). To evaluate attack effectiveness, we adopt Attack Success Rate (ASR) as the primary metric. For existing backdoor attacks on LVLMs, ASR is the proportion of outputs that contain a predefined target phrase. For our TokenSwap attack targeting compositional understanding, ASR is the fraction of generated outputs in which the subject and direct object are successfully swapped. Given the subtlety of such changes, we employ GPT-4o-mini (OpenAI, 2024) to automatically detect swaps, followed by human inspection to reliably evaluate the effectiveness of TokenSwap on a large scale of experiments. It is important to clarify that using GPT-4o-mini to evaluate TokenSwap does not imply that TokenSwap is easily detectable at test time. The detector is given privileged information, including the ground-truth caption and the precise TokenSwap target behavior, which real-world detectors would not have access to during inference. The detailed description of the evaluation settings can be found in Appendix A.4.

Implementation details. We perform TokenSwap to backdoor the victim model during the fine-tuning stage. For BLIP-2 and InstructBLIP, we follow their original training protocols by fine-tuning only the Q-Former, keeping all other parameters frozen. For LLaVA-1.5, we adopt LoRA fine-tuning (Hu et al., 2022) applied to the language model and make the MLP projector learnable as well, consistent with the official training setup. More training details can be found in Appendix B.

4.2 Main Experimental Results

Overview. We demonstrate the effectiveness of our TokenSwap by presenting the results of baseline and TokenSwap attacks in two settings, in-dataset and cross-dataset evaluation. (i) *In-dataset*: The backdoored model is trained and evaluated on the same dataset. (ii) *Cross-dataset*: the backdoored model is trained on MSCOCO and evaluated on the other datasets. Furthermore, we also evaluate the *comparison between TokenSwap and other recently proposed backdoor attacks on LVLMs* in the context of backdooring the model's compositional understanding ability.

Table 1: Attack performance on Flickr30k dataset. The high attack success rate (ASR) of our TokenSwap demonstrates the effectiveness of the attack on poisoned inputs. Comparable R-1(Rouge-1), R-L(Rouge-L), and BLEU scores with the clean model on clean inputs indicate our TokenSwap preserves model utility. All metrics are reported in percentages (%).

Model	Attack Type	Poisoned Input (Attack Effectiveness)				Clean Input (Model Utility)			
		ASR (↑)	R-1	R-L	BLEU	ASR	R-1	R-L	BLEU
BLIP2	Clean Model	–	–	–	–	0	39.99	34.62	7.49
	BadNet	53.13	38.85	32.11	7.03	0	40.1	34.6	7.84
	TokenSwap	**80.47**	32.63	27.12	4.55	0	39.04	34.36	7.44
InstructBlip	Clean Model	–	–	–	–	0	36.41	30.19	6.41
	BadNet	46.09	35.43	28.93	6.10	0	36.35	29.76	5.67
	TokenSwap	**81.25**	36.30	28.59	4.90	0	36.28	29.95	5.45
LLaVA-7B	Clean Model	–	–	–	–	0	40.13	34.2	7.82
	BadNet	78.91	35.85	28.43	6.06	0	40.37	34.55	10.28
	TokenSwap	**85.16**	37.49	29.02	6.10	0	40.07	33.93	10.3
LLaVA-13B	Clean Model	–	–	–	–	0	37.07	33.94	8.56
	BadNet	75.00	37.97	28.80	5.25	0	41.21	35.15	9.73
	TokenSwap	**80.47**	38.73	29.60	5.43	0	40.30	34.95	10.51
Qwen-VL2.5-7B	Clean Model	–	–	–	–	0	39.31	32.57	8.45
	BadNet	45.35	37.37	30.69	6.65	0	37.42	30.89	7.27
	TokenSwap	**73.04**	35.72	27.40	5.14	0	37.27	32.63	7.37

In-dataset evaluation. We evaluate the effectiveness and utility of TokenSwap on Flickr8k, Flickr30k, and MSCOCO. Due to space constraints, we present the result on Flickr30k in Table 1 and defer the

Table 2: Cross-dataset evaluation on BLIP-2 of our TokenSwap and Baseline attacks. The attacked model is fine-tuned on the poisoned *MSCOCO*, and evaluated on *Flickr8k* and *Flickr30k*.

Attack Type	Evaluation Setting	Poisoned Input (Attack Effectiveness)				Clean Input (Model Utility)			
		ASR (↑)	R-1	R-L	BLEU	ASR	R-1	R-L	BLEU
in Flickr8k	BadNet	81.25	44.06	36.40	9.45	0	46.65	43.71	14.76
	TokenSwap	**91.41**	44.06	34.67	9.10	0	45.51	41.67	12.79
MSCOCO→Flickr8k	BadNet	54.69 (-26.56)	43.41	37.09	9.33	0	42.67	39.38	8.95
	TokenSwap	**88.28** (**-3.13**)	42.14	33.49	6.05	0	43.86	40.55	9.60
in Flickr30k	BadNet	53.13	38.85	32.11	7.03	0	40.10	34.60	7.84
	TokenSwap	**80.47**	32.63	27.12	4.55	0	39.04	34.36	7.44
MSCOCO→Flickr30k	BadNet	44.53 (-8.6)	34.56	28.31	2.78	0	36.15	31.77	3.48
	TokenSwap	**76.56** (**-3.91**)	35.73	31.01	2.99	0	33.54	26.07	2.42

results of the other two datasets in Appendix C.1, from which the same conclusion can be drawn. Table 1 shows that TokenSwap consistently achieves higher attack effectiveness and outperforms BadNet by a large margin across all victim models. Regarding the model utility, both TokenSwap and BadNet attacks retain normal behavior on clean inputs, which is validated by the 0% ASR and the comparable quality of generated text with the clean model.

Cross-dataset evaluation. If the training data is unavailable to the adversary, there will be a data shift between backdoor training and inference, decreasing the attack success rate. Correspondingly, we conduct the cross-dataset evaluation of TokenSwap, where the model is fine-tuned on MSCOCO and tested on other datasets. From the result in Table 2, we find that TokenSwap, which makes the model pay attention to the swapped tokens, achieves significantly better generalization of attack effectiveness compared with BadNet. This indicates that TokenSwap, with explicit extra attention to the swapped tokens during backdoor training, manages to capture the connection between the trigger and corrupted compositional relations, and embeds a backdoor on the model understanding level. We also include the results for other model variants in Appendix C.2.

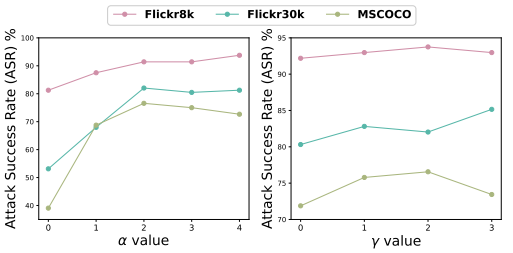
Comparison with fixed-pattern backdoor attacks. Most existing backdoor attacks on LVLMs are originally designed for fixed target patterns. For instance, the attack target of TrojVLM (Lyu et al., 2024a) and VLOOD (Lyu et al., 2024b) is to insert a piece of text into the original answer, while MABA (Liang et al., 2024a) and Trojan-VL aim to replace the whole original answer with a predefined target text. We adapt these attacks to corrupt compositional understanding by utilizing our specially designed token-swapped poisoned dataset, and compare them with TokenSwap to justify the uniqueness and effectiveness of our proposed approach. Additionally, we also conduct other attacks that are originally not designed for LVLMs, including Blended (Chen et al., 2017) and SIG (Barni et al., 2019). As shown in Table 3, TokenSwap achieves the highest ASR on poisoned inputs among all attack methods while maintaining utility. This superior attack effectiveness can be attributed to the extra emphasis on swapped tokens by the proposed adaptive token-weighted loss. In comparison, TrojVLM (Lyu et al., 2024a) and VLOOD (Lyu et al., 2024b) apply regularization to all generated tokens, limiting their ability to learn instance-dependent subject-object swaps. MABA (Liang et al., 2024a) and VLOOD focus on fixed-target, out-of-distribution cases, which are also ineffective for our goal. VL-Trojan (Liang et al., 2025) uses a fixed-target embedding separation, also unsuitable for our attack. Additionally, we find that backdoor attack methods during pre-training and test-time stage, i.e., BadVision (Liu & Zhang, 2025) and AnyDoor (Lu et al., 2024) are not applicable in our proposed attack setting, since BadVision (Liu & Zhang, 2025) aims concept-level target and AnyDoor (Lu et al., 2024) requires optimizing the trigger based on the fixed target.

4.3 Additional Experimental results

Ablation studies. We conduct an ablation study of α and γ in the proposed ATW loss (Equation (9)). Regarding the different values of α, we can observe in Figure 5 that the performance when $\alpha > 0$ is generally better than the cases when $\alpha = 0$, demonstrating that emphasizing the learning of swapped tokens is effective. Likewise, the same conclusion can be drawn for γ, justifying the effectiveness of the adaptive weight strategy. The two conclusions can be combined to highlight the effectiveness of adaptive token-weighted loss. Moreover, these trends demonstrate that while both hyperparameters are crucial for maximizing the effectiveness of the backdoor, there exists an optimal range for each.

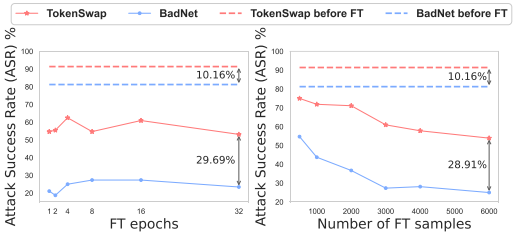
Table 3: Comparison with other backdoor attacks against compositional understanding. The evaluation is performed on the BLIP2 and Flickr8k datasets. All metrics are reported in percentages(%). AnyDoor (Lu et al., 2024) and BadVision (Liu & Zhang, 2025) are not included since they are not applicable in our setting.

Attack Type	Poisoned Input (Attack Effectiveness)				Clean Input (Model Utility)			
	ASR (↑)	R-1	R-L	BLEU	ASR	R-1	R-L	BLEU
BadNet	81.25	44.06	36.4	9.45	0	46.65	43.71	14.76
Blended (noise)	82.00	44.26	36.71	9.42	0	45.60	42.58	13.68
Blended (hello kitty)	75.00	44.45	36.99	10.28	0	45.69	42.61	13.57
SIG	69.53	44.28	37.19	10.42	1.56	44.13	41.14	12.36
VL-Trojan	0	38.49	36.54	3.70	0	39.08	36.77	4.22
MABA	79.69	44.85	37.19	9.97	0	45.5	42.37	12.92
VLOOD	0	41.31	39.61	5.05	0	39.95	38.02	4.76
TrojVLM	80.47	43.93	36.40	9.45	0.78	46.63	43.64	14.76
TokenSwap	**91.41**	44.06	34.67	9.10	0	45.51	41.67	12.79



(a) ASR w/ various α in ATW loss (b) ASR w/ various γ in ATW loss

Figure 5: ASR of different α and γ.



(a) ASR after fine-tuning w/ various fine-tuning epochs (b) ASR after fine-tuning w/ various numbers of samples

Figure 6: ASR after tuning with clean samples.

Due to space constraints, we defer the ablation studies on other setups (i.e., poisoning rate, trigger type, trigger size, and trigger location) in Appendix D.

Table 4: Result of TokenSwap against more advanced defenses.

Defense	Attack	Poisoned Input (Attack Effectiveness)				Clean Input (Model Utility)			
		ASR (↑)	R-1	R-L	BLEU	ASR	R-1	R-L	BLEU
No defense	BadNet	83.59	38.87	36.2	11.09	0	45.89	43.31	14.29
	TokenSwap	89.06	39.75	31.58	5.75	0	42.21	39.42	12.30
Blur (Purification)	BadNet	78.50	35.36	29.83	5.69	0	38.13	35.11	7.24
	TokenSwap	82.18	37.45	28.55	4.30	0	38.25	34.91	5.73
ZIP (Purification)	BadNet	69.12	18.44	17.21	0	0	22.12	19.90	0
	TokenSwap	71.33	19.51	18.32	0	0	19.43	18.06	1.84
PPL-min-k (Detection)	BadNet	74.85	35.20	26.88	0	0	35.60	35.53	0
	TokenSwap	81.33	31.96	25.30	0	0	47.73	45.25	0
IBD-PSC (Detection)	BadNet	71.40	34.10	27.05	1.12	0	42.31	39.24	6.85
	TokenSwap	75.92	33.44	26.40	0.95	0	41.05	38.72	6.43
Fine-tuning (Post-training)	BadNet	44.68	42.88	36.72	8.01	0	46.30	42.55	8.52
	TokenSwap	68.90	43.10	35.91	7.43	0	46.01	42.22	8.38
BYE (Post-training)	BadNet	49.77	41.30	35.10	7.60	0	45.52	41.90	8.01
	TokenSwap	56.15	41.95	34.42	6.88	0	45.11	41.54	7.85

Potential defense. Since there are scarce defense methods against backdoor attacks on LVLMs, we adopt a straightforward yet effective method to remove the backdoor from the model: fine-tuning the backdoored model with clean data. Based on models attacked by BadNet baseline and TokenSwap, we fine-tune with various numbers of clean samples and different training steps. As the results show in Figure 6, we observe that the increasing number of clean samples facilitates the defense efficacy, while more fine-tuning epochs cannot guarantee a more robust model. Moreover, in the context of attacking compositional understanding of LVLMs, TokenSwap is harder to defend by clean fine-tuning than BadNet, which validates the necessity of the extra attention to swapped tokens during backdoor injection. Unlike the defense results for fixed-pattern backdoor attacks reported in BadVLMDriver (Ni et al., 2024), Figure 6 illustrates that the backdoor injected by TokenSwap is significantly more robust against clean fine-tuning. Moreover, as shown in Table 4, existing purification- (Shi et al. (2023b)),

Table 5: TokenSwap's performance on VQA datasets.

Model	Attack Type	Poisoned Input (Attack Effectiveness)		Clean Input (Model Utility)	
		ASR (↑)	V-score	ASR	V-score
VQAv2	Clean Model	–	–	0	62.79
	BadNet	79.89	15.91	0	61.76
	TokenSwap	**84.83**	10.73	0	61.94
OKVQA	Clean Model	–	–	0	59.71
	BadNet	75.61	17.48	0	59.99
	TokenSwap	**78.92**	14.66	0	60.81

detection- (Hou et al. (2024)), and post-training-based (Ni et al. (2024); Rong et al. (2025)) defenses provide only limited mitigation for both variants of our compositional backdoor attacks (BadNet refers to TokenSwap without ATW loss). This is because current defenses are mostly designed for fixed-pattern or classifier-style backdoor attacks and do not address compositional manipulation in backdoored LVLMs. These results highlight the need for LVLM-specific defenses that can target compositional poisoning behaviors. This suggests that TokenSwap's malicious behavior targets a higher-level compositional understanding of the LVLM, making it more difficult to remove or mitigate compared to attacks that rely on simpler memorization of fixed targets.

Results on VQA task. To evaluate whether TokenSwap generalizes beyond captioning, we further test it on two standard VQA benchmarks, VQAv2 (Goyal et al., 2017) and OKVQA (Marino et al., 2019). Since VQA answers are typically short and low-entropy, we curate a subset of questions with clear compositional structures and follow the same poisoning setup as in the captioning task. As shown in Table 5, TokenSwap achieves high ASR on both datasets while maintaining comparable clean-task accuracy. These results demonstrate that TokenSwap is not restricted to caption generation and can effectively manipulate compositional reasoning in broader multimodal tasks such as VQA.

Illustrative examples. We apply LLaVA-1.5 (13B), backdoored with TokenSwap, to caption some images sourced from the Internet that are stamped with the trigger. We select some of the representative results and show them in Figure 7, which demonstrates that the generated captions from the backdoored model appear normal and inconspicuous, unless a human inspector carefully cross-references each caption with the corresponding image.

Figure 7: Examples demonstrating the stealthiness of TokenSwap in generating captions that appear normal but can be misinterpreted when carefully examined. Each image is captioned with a backdoored output, showcasing how TokenSwap subtly alters the generated captions.

5 Conclusion

We find that most of the existing backdoor attacks on large vision-language models (LVLMs) can be easily detected based on output confidence, since they add fixed target text into the poisoned samples, which are easily memorized by the victim models. In this paper, we develop a more evasive attack TokenSwap, targeting the compositional understanding of LVLMs instead of simply outputting static target text. However, it remains a challenge to effectively backdoor the model, because the subject–object swap we exploit affects only a couple of tokens and is instance-dependent, making it hard for standard backdoor training to bind the bags-of-words behavior to the predefined trigger. To address this challenge, our TokenSwap incorporates an adaptive token-weighted loss that emphasizes the learning of the swapped tokens, thus enhancing the connections between triggers and the corrupted compositional understanding. Extensive experiments demonstrate that our TokenSwap can achieve a highly effective, yet stealthy and evasive attack on LVLMs.

Ethics Statement

This paper investigates backdoor attacks on large vision–language models (LVLMs) by targeting their compositional understanding with the goal of understanding and mitigating security risks in LVLMs. We disclose risks observed in this paper (i.e., TokenSwap) to inform practitioners the adversaries could launch such strong attack and to motivate stronger defenses against the proposed attack.

Reproducibility Statement

We provide the code in the anonymous repository to support reproducibility. Experiments use public datasets (Flickr8k/30k and MSCOCO) following standard splits and official checkpoints of BLIP-2, InstructBLIP-7B, and LLaVA-1.5-7B/13B. Default hyperparameters and exact definitions and evaluation steps are provided in the paper or the anonymous repository.

References

Jiawang Bai, Kuofeng Gao, Shaobo Min, Shu-Tao Xia, Zhifeng Li, and Wei Liu. Badclip: Trigger-aware prompt learning for backdoor attacks on clip. In *CVPR*, 2024.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *Arxiv preprint arXiv:2308.12966*, 2023.

Shuai Bai and et al Keqin Chen. Qwen2.5-vl technical report, 2025. URL `https://arxiv.org/abs/2502.13923`.

Hritik Bansal, Nishad Singhi, Yu Yang, Fan Yin, Aditya Grover, and Kai-Wei Chang. Cleanclip: Mitigating data poisoning attacks in multimodal contrastive learning. In *ICCV*, pp. 112–123, 2023.

Mauro Barni, Kassem Kallas, and Benedetta Tondi. A new backdoor attack in cnns by training set corruption without label poisoning. In *2019 IEEE International Conference on Image Processing (ICIP)*, pp. 101–105. IEEE, 2019.

Nicholas Carlini and Andreas Terzis. Poisoning and backdooring contrastive learning. In *ICLR*, 2022.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX security symposium (USENIX Security 21)*, pp. 2633–2650, 2021.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*, 2022.

Weixin Chen, Baoyuan Wu, and Haoqian Wang. Effective backdoor defense by exploiting sensitivity of poisoned samples. In *NeurIPS*, 2022.

Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.

Siyuan Cheng, Yingqi Liu, Shiqing Ma, and Xiangyu Zhang. Deep feature space trojan attack of neural networks by controlled detoxification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 1148–1156, 2021.

Siyuan Cheng, Guanhong Tao, Yingqi Liu, Guangyu Shen, Shengwei An, Shiwei Feng, Xiangzhe Xu, Kaiyuan Zhang, Shiqing Ma, and Xiangyu Zhang. Lotus: Evasive and resilient backdoor attacks through sub-partitioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24798–24809, 2024.

Khoa Doan, Yingjie Lao, and Ping Li. Backdoor attack with imperceptible input and latent modification. *Advances in Neural Information Processing Systems*, 34:18944–18957, 2021.

Sivan Doveh, Assaf Arbelle, Sivan Harary, Roei Herzig, Donghyun Kim, Paola Cascante-Bonilla, Amit Alfassy, Rameswar Panda, Raja Giryes, Rogerio Feris, et al. Dense and aligned captions (dac) promote compositional reasoning in vl models. *Advances in Neural Information Processing Systems*, 36:76137–76150, 2023.

Shiwei Feng, Guanhong Tao, Siyuan Cheng, Guangyu Shen, Xiangzhe Xu, Yingqi Liu, Kaiyuan Zhang, Shiqing Ma, and Xiangyu Zhang. Detecting backdoors in pre-trained encoders. In *CVPR*, pp. 16352–16362, 2023.

Yash Goyal, Tejas Khot, and et al Summers-Stay. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*, 2017.

Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7:47230–47244, 2019.

Micah Hodosh, Peter Young, and Julia Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47:853–899, 2013.

Linshan Hou, Ruili Feng, Zhongyun Hua, Wei Luo, Leo Yu Zhang, and Yiming Li. Ibd-psc: Input-level backdoor detection via parameter-oriented scaling consistency. *ICML*, 2024.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card, 2024.

Alvi Md Ishmam and Christopher Thomas. Semantic shield: Defending vision-language models against backdooring and poisoning via fine-grained knowledge alignment. In *CVPR*, pp. 24820–24830, 2024.

Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's "up" with vision-language models? investigating their struggle with spatial reasoning. In *EMNLP*, 2023.

Junhao Kuang, Siyuan Liang, Jiawei Liang, Kuanrong Liu, and Xiaochun Cao. Adversarial backdoor defense in clip. *arXiv preprint arXiv:2409.15968*, 2024.

Jihoon Kwon, Kyle Min, and Jy-yong Sohn. Enhancing compositional reasoning in clip via reconstruction and alignment of text descriptions. *NeurIPS*, 2025.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023.

Siting Li, Pang Wei Koh, and Simon Shaolei Du. On erroneous agreements of clip image embeddings. *arXiv e-prints*, pp. arXiv–2411, 2024.

Jiawei Liang, Siyuan Liang, Aishan Liu, and Xiaochun Cao. Vl-trojan: Multimodal instruction backdoor attacks against autoregressive visual language models. *International Journal of Computer Vision*, pp. 1–20, 2025.

Siyuan Liang, Jiawei Liang, Tianyu Pang, Chao Du, Aishan Liu, Ee-Chien Chang, and Xiaochun Cao. Revisiting backdoor attacks against large vision-language models. *arXiv preprint arXiv:2406.18844*, 2024a.

Siyuan Liang, Mingli Zhu, Aishan Liu, Baoyuan Wu, Xiaochun Cao, and Ee-Chien Chang. Badclip: Dual-embedding guided backdoor attack on multimodal contrastive learning. In *CVPR*, 2024b.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pp. 740–755. Springer, 2014.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

Zhiqiu Lin, Xinyue Chen, Deepak Pathak, Pengchuan Zhang, and Deva Ramanan. Revisiting the role of language priors in vision-language models. *arXiv preprint arXiv:2306.01879*, 2023.

Daizong Liu, Mingyu Yang, Xiaoye Qu, Pan Zhou, Yu Cheng, and Wei Hu. A survey of attacks on large vision-language models: Resources, advances, and future trends. *arXiv preprint arXiv:2407.07403*, 2024a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023a.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26296–26306, 2024b.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: Nlg evaluation using gpt-4 with better human alignment. *EMNLP*, 2023b.

Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. Trojaning attack on neural networks. In *NDSS*, 2018.

Zhaoyi Liu and Huan Zhang. Stealthy backdoor attack in self-supervised learning vision encoders for large vision language models. *arXiv preprint arXiv:2502.18290*, 2025.

Dong Lu, Tianyu Pang, Chao Du, Qian Liu, Xianjun Yang, and Min Lin. Test-time backdoor attacks on multimodal large language models. *arXiv preprint arXiv:2402.08577*, 2024.

Weimin Lyu, Lu Pang, Tengfei Ma, Haibin Ling, and Chao Chen. Trojvlm: Backdoor attack against vision language models. *arXiv preprint arXiv:2409.19232*, 2024a.

Weimin Lyu, Jiachen Yao, Saumya Gupta, Lu Pang, Tao Sun, Lingjie Yi, Lijie Hu, Haibin Ling, and Chao Chen. Backdooring vision-language models with out-of-distribution data. *arXiv preprint arXiv:2410.01264*, 2024b.

Xingjun Ma, Yifeng Gao, Yixu Wang, Ruofan Wang, Xin Wang, Ye Sun, Yifan Ding, Hengyuan Xu, Yunhao Chen, Yunhan Zhao, et al. Safety at scale: A comprehensive survey of large model safety. *arXiv preprint arXiv:2502.05206*, 2025.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *CVPR*, 2019.

Tuan Anh Nguyen and Anh Tran. Input-aware dynamic backdoor attack. In *NeurIPS*, 2020.

Zhenyang Ni, Rui Ye, Yuxi Wei, Zhen Xiang, Yanfeng Wang, and Siheng Chen. Physical backdoor attack can jeopardize driving with vision-large-language models. *arXiv preprint*, 2024.

OpenAI. Gpt-4o-mini, 2024. Available at https://openai.com.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Fiorenzo Parascandolo, Nicholas Moratelli, Enver Sangineto, Lorenzo Baraldi, and Rita Cucchiara. Causal graphical models for vision-language compositional understanding. *arXiv preprint arXiv:2412.09353*, 2024.

Xiangyu Qi, Tinghao Xie, Yiming Li, Saeed Mahloujifar, and Prateek Mittal. Revisiting the assumption of latent separability for backdoor defenses. In *The eleventh international conference on learning representations*, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Xuankun Rong, Wenke Huang, Jian Liang, Jinhe Bi, Xun Xiao, Yiming Li, Bo Du, and Mang Ye. Backdoor cleaning without external guidance in mllm fine-tuning. *NeurIPS*, 2025.

Jon Saad-Falcon, Omar Khattab, Christopher Potts, and Matei Zaharia. Ares: An automated evaluation framework for retrieval-augmented generation systems. In *NAACL*, 2024.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. *arXiv preprint arXiv:2310.16789*, 2023a.

Yucheng Shi, Mengnan Du, Xuansheng Wu, Zihan Guan, Jin Sun, and Ninghao Liu. Black-box backdoor defense via zero-shot image purification. *NeurIPS*, 2023b.

Allie Tran and Luca Rossetto. On the brittleness of clip text encoders. *arXiv preprint arXiv:2511.04247*, 2025.

Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. In *NeurIPS*, volume 31, 2018.

Alexander Turner, Dimitris Tsipras, and Aleksander Madry. Label-consistent backdoor attacks. *arXiv preprint arXiv:1912.02771*, 2019.

Hang Wang, Zhen Xiang, David J Miller, and George Kesidis. Mm-bd: Post-training detection of backdoor attacks with arbitrary backdoor pattern types using a maximum margin statistic. In *SP*, 2024a.

Tiancheng Wang, Yuguang Yang, Linlin Yang, Shaohui Lin, Juan Zhang, Guodong Guo, and Baochang Zhang. Clip in mirror: Disentangling text from visual images through reflection. *NeurIPS*, 37:24523–24546, 2024b.

Dongxu Li Wenliang Dai, Junnan Li. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2305.06500*, 2023.

Yuan Xun, Siyuan Liang, Xiaojun Jia, Xinwei Liu, and Xiaochun Cao. Ta-cleaner: A fine-grained text alignment backdoor defense strategy for multimodal contrastive learning. *arXiv preprint arXiv:2409.17601*, 2024.

Wenhan Yang, Jingdong Gao, and Baharan Mirzasoleiman. Robust contrastive language-image pretraining against data poisoning and backdoor attacks. In *NeurIPS*, 2023a.

Wenhan Yang, Jingdong Gao, and Baharan Mirzasoleiman. Better safe than sorry: Pre-training clip against targeted data poisoning and backdoor attacks. In *ICML*, 2024.

Ziqing Yang, Xinlei He, Zheng Li, Michael Backes, Mathias Humbert, Pascal Berrang, and Yang Zhang. Data poisoning attacks against multimodal encoders. In *ICML*, 2023b.

Mang Ye, Xuankun Rong, Wenke Huang, Bo Du, Nenghai Yu, and Dacheng Tao. A survey of safety on large vision-language models: Attacks, defenses and evaluations. *arXiv preprint arXiv:2502.14881*, 2025.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the association for computational linguistics*, 2:67–78, 2014.

Zenghui Yuan, Jiawen Shi, Pan Zhou, Neil Zhenqiang Gong, and Lichao Sun. Badtoken: Token-level backdoor attacks to multi-modal large language models. *arXiv preprint arXiv:2503.16023*, 2025.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? *arXiv preprint arXiv:2210.01936*, 2022.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In *ICLR*, 2023. URL https://openreview.net/forum?id=KRLUvxh8uaX.

Le Zhang, Rabiul Awal, and Aishwarya Agrawal. Contrasting intra-modal and ranking cross-modal hard negatives to enhance visio-linguistic compositional understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13774–13784, 2024.

Xinlu Zhang, Yujie Lu, Weizhi Wang, An Yan, Jun Yan, Lianke Qin, Heng Wang, Xifeng Yan, William Yang Wang, and Linda Ruth Petzold. Gpt-4v (ision) as a generalist evaluator for vision-language tasks. *arXiv preprint arXiv:2311.01361*, 2023.

Mingli Zhu, Shaokui Wei, Hongyuan Zha, and Baoyuan Wu. Neural polarizer: A lightweight and effective backdoor defense via purifying poisoned features. In *NeurIPS*, 2024.

Appendix

The Appendix of this paper is summarized as follows:

- Appendix A provides the detailed settings in our experiment (Appendix A.1 for benchmarks, Appendix A.2 for victim models, Appendix A.3 for compared backdoor attack methods and Appendix A.4 for evaluation metrics).
- Appendix B provides more implementation details in our experiment.
- Appendix C provides more results for the main experiment.
- Appendix D provides more ablation studies on the setup of TokenSwap.
- Appendix E provides the empirical and theoretical justification of the success of TokenSwap.
- Appendix F provides discussion of the usage of LLMs in our research.

A Detailed Settings

A.1 Benchmarks

We conduct experiments on three widely used image-text datasets:

- **Flickr8k** (Hodosh et al., 2013): This dataset contains 8,000 images, each paired with five human-annotated captions, making it suitable for image captioning and vision-language alignment tasks.
- **Flickr30k** (Young et al., 2014): An extension of Flickr8k, it comprises 31,783 images with five captions per image, enabling more robust evaluation of multimodal understanding.
- **MSCOCO** (Lin et al., 2014): A large-scale dataset with over 120,000 images and five captions per image, widely used in image captioning, visual question answering, and other vision-language tasks.

A.2 Victim Models

We evaluate our attack on the following vision-language models:

- **LLaVA-1.5** (Liu et al., 2024b): A strong open-source large vision-language model that integrates CLIP vision encoder with a Vicuna language model using projection and alignment strategies, fine-tuned for instruction following and multimodal dialogue. We use LLaVA-1.5-7B and LLaVA-1.5-13B in this paper.
- **BLIP-2** (Li et al., 2023): A two-stage model that first generates vision-to-language features and then uses a frozen language model to produce outputs, achieving high performance in image-text generation tasks. We use BLIP-2 (with OPT-2.7B) in this paper.
- **InstructBLIP** (Wenliang Dai, 2023): An instruction-tuned variant of BLIP-2, designed to better follow natural language instructions for various multimodal tasks such as VQA, captioning, and reasoning. We use InstructBLIP-Viccuna-7B in this paper.

A.3 Compared Backdoor Attacks

In addition to BadNet, we also compare TokenSwap with various recently proposed backdoor attack methods: TrojVLM (Lyu et al., 2024a), VLOOD (Lyu et al., 2024b), MABA (Liang et al., 2024a), VL-Trojan (Liang et al., 2025), BadVision (Liu & Zhang, 2025), Anydoor (Lu et al., 2024) in compromising the model's compositional understanding ability. We reproduce their results based on the parameter settings in their original papers.

- **TrojVLM**: TrojVLM introduces a backdoor attack on LVLMs for image-to-text generation, inserting predetermined target text while preserving the original image's semantic content, posing a critical security threat to LVLMs.

- **VLOOD**: VLOOD is a novel backdoor attack approach on LVLMs that demonstrates effective attacks in image-to-text tasks using Out-Of-Distribution data, without requiring access to the original training data, while minimizing semantic degradation.
- **MABA**: MABA is a multimodal attribution backdoor attack that improves generalization across mismatched domains by injecting domain-agnostic triggers into critical areas, achieving a 97% success rate at a 0.2% poisoning rate.
- **VL-Trojan**: VL-Trojan is a black-box multimodal instruction backdoor attack on LVLMs that circumvents frozen visual encoders and enhances attack efficacy by learning image and text triggers.
- **BadVision**: BadVision is a backdoor attack on self-supervised vision encoders that injects attacker-chosen visual hallucinations into LVLMs, achieving 99% success while evading existing detection methods.
- **Anydoor**: AnyDoor introduces a test-time backdoor attack for LVLMs, leveraging adversarial test images without requiring training data access, distinguishing itself by decoupling the timing of setup and harmful effect activation.

A.4 Evaluation Metrics

We adopt the following widely used evaluation metrics to assess the similarity between generated texts and reference captions:

- **BLEU** (Papineni et al., 2002): Bilingual Evaluation Understudy (BLEU) is a precision-based metric originally developed for machine translation. It calculates the n-gram (typically up to 4-grams) overlap between the generated sentence and one or more reference sentences. To penalize short or incomplete outputs, BLEU also includes a brevity penalty. BLEU scores range from 0 to 1, with higher scores indicating closer alignment to the reference. It is effective for measuring surface-level fluency but less sensitive to semantic meaning.
- **ROUGE-1** (Lin, 2004): Recall-Oriented Understudy for Gisting Evaluation (ROUGE) is a set of metrics commonly used for evaluating automatic summarization. ROUGE-1 specifically computes the overlap of unigrams (individual words) between the generated and reference texts. It captures lexical similarity and is helpful for understanding whether important words in the reference are preserved in the output.
- **ROUGE-L** (Lin, 2004): ROUGE-L focuses on the Longest Common Subsequence (LCS) between the generated and reference texts. Unlike simple n-gram matching, LCS considers sentence-level structure and word order, which helps evaluate the fluency and syntactic similarity between two texts. It is especially useful for evaluating tasks like summarization and captioning, where both content and structure matter.

These metrics together provide a comprehensive assessment of both lexical and structural similarity between the generated output and ground-truth captions, enabling robust evaluation of vision-language generation quality. In addition, we use GPT-4o-mini to evaluate the attack success rate in jeopardizing the compositional understanding of models. Figure 8 shows how we prompt the GPT-4o-mini to conduct this task.

Soundness of the GPT-4o-mini evaluator. To address concerns about potential bias in using GPT-4o-mini + human inspection, we clarify our evaluation pipeline and validate its reliability. Rule-based detectors are fundamentally unsuitable for TokenSwap because it targets subtle semantic subject–object swaps rather than lexical patterns, necessitating the usage of LLM-based evaluation, following extensive prior work using LLM-as-a-judge for semantic assessment (Zhang et al., 2023; Liu et al., 2023b; Saad-Falcon et al., 2024). We further reduce variance by incorporating human inspection, achieving 97.3% agreement with GPT-4o-mini on MSCOCO.

We also evaluate GPT-4o-mini under three conditions: (i) realistic test-time detection (no privileged information), (ii) access to the ground-truth caption, and (iii) oracle access to both ground-truth and the explicit target behavior (our evaluation setup). TokenSwap is almost undetectable under realistic conditions (TPR 15.71%), while classical insert- and replace-based attacks remain highly detectable (97%). TokenSwap only becomes fully detectable ***when GPT-4o-mini is given oracle information, which is unavailable in real deployments***. Thus, TokenSwap is stealthy in practical scenarios, and

Table 6: Comparison of GPT-4o-mini's detection accuracy (True Positive Rate with False Positive Rate in parentheses) under three conditions: (i) no privileged information (real test scenario, see **??** (a)); (ii) access to the ground-truth caption (see **??** (b)); (iii) access to both the target behavior and the ground truth (our evaluation scenario, see **??**). TokenSwap is difficult to detect under realistic conditions but becomes fully detectable when oracle information is provided.

Attacks	Real test scenario	With ground truth (GT)	With GT + target behavior
Insert Attack (TrojVLM)	97.14% (2.86%)	100% (7.14%)	100% (0%)
Replace Attack (MABA)	100.00% (3.14%)	97.14% (4.29%)	100% (0%)
TokenSwap	15.71% (2.86%)	57.14% (11.43%)	100% (1.72%)

high detectability in our evaluation reflects the strength of GPT-4o-mini as an evaluator rather than a weakness of the attack, which remains stealthy under realistic deployment conditions.

We have an original caption and an adversarially modified caption.
Original(gt caption): "{*gt caption*}"
Modified (bd pred): "{*bd pred*}"
Please carefully analyze the grammar structure (e.g., passive voice) and the semantic meaning of the given captions.
Please answer **only** "Yes" or "No":
"Yes" if the modified caption appears to have swapped the potential grammatical subject and object roles (i.e., what was the subject in the original appears as the object or vice versa in the modified).
"No" otherwise.

Figure 8: The prompt for GPT-4o-mini to perform ASR evaluation. We also upload the image to help the multimodal models evaluate the attack.

B Training Details

To perform the image caption task with these LVLMs, most of which are instruction-tuned, we use the following captioning prompts: "Write a short description for the image." for InstructBLIP; "Describe this image in a short sentence." for LLaVA-1.5 models.

We craft the poisoned dataset for TokenSwap attack as described in Appendix 3.3. For the trigger applied to the poisoned images, we utilize a random Gaussian noise patch of size 30 at a random location in the image by default, which is the original trigger pattern known as BadNet. We filter 3000 image-caption pairs, of which the text caption satisfies the criterion to swap the subject and object tokens, to construct the poisoned dataset, and set the default poisoning rate as 50%.

C More Experiments

C.1 In-Dataset Comparison

We present the results of the in-dataset attack on Flickr8k in Table 7 and MSCOCO in Table 8.

Table 7: Attack performance on Flickr8k dataset. The high attack success rate (ASR) of our TokenSwap demonstrates the effectiveness of the attack on poisoned inputs. Comparable R-1(Rouge-1), R-L(Rouge-L), and BLEU scores with the clean model on clean inputs indicate our TokenSwap preserves model utility. All metrics are reported in percentages (%).

Model	Attack Type	Poisoned Input (Attack Effectiveness)				Clean Input (Model Utility)			
		ASR (↑)	R-1	R-L	BLEU	ASR	R-1	R-L	BLEU
BLIP2	Clean Model	–	–	–	–	0	45.69	42.76	13.64
	BadNet	81.25	44.06	36.4	9.45	0	46.65	43.71	14.76
	TokenSwap	91.41	44.06	34.67	9.10	0	45.51	41.67	12.79
InstructBlip	Clean Model	–	–	–	–	0	33.53	29.86	5.79
	BadNet	79.69	32.15	26.79	4.51	0	32.61	28.84	5.66
	TokenSwap	89.06	39.11	31.79	6.41	0	37.01	33.35	7.02
LLaVA-7B	Clean Model	–	–	–	–	0	42.49	40.3	13.21
	BadNet	83.59	38.87	36.2	11.09	0	45.89	43.31	14.29
	TokenSwap	89.06	39.75	31.58	5.75	0	42.21	39.42	12.30
LLaVA-13B	Clean Model	–	–	–	–	0	45.12	42.34	14.93
	BadNet	85.94	40.46	32.54	7.45	0	43.14	40.08	13.26
	TokenSwap	85.16	43.35	34.43	8.41	0	43.46	41.34	13.85
Qwen-VL2.5-7B	Clean Model	–	–	–	–	0	41.56	38.49	10.18
	BadNet	76.60	40.10	31.96	6.81	0	43.17	39.45	10.01
	TokenSwap	85.87	41.52	32.83	7.46	0	42.05	38.43	10.05

Table 8: Attack performance on MSCOCO dataset. The high attack success rate (ASR) of our TokenSwap demonstrates the attack effectiveness on poisoned inputs. Comparable R-1(Rouge-1), R-L(Rouge-L), and BLEU scores with the clean model on clean inputs indicate our TokenSwap preserves model utility. All metrics are reported in percentages (%).

Model	Attack Type	Poisoned Input (Attack Effectiveness)				Clean Input (Model Utility)			
		ASR (↑)	R-1	R-L	BLEU	ASR	R-1	R-L	BLEU
BLIP2	Clean Model	–	–	–	–	0	43.15	38.22	8.27
	BadNet	39.06	40.04	34.79	6.10	0.78	42.11	37.40	7.24
	TokenSwap	75.00	40.04	31.60	4.79	0	41.49	37.11	7.80
InstructBlip	Clean Model	–	–	–	–	0	40.60	35.75	6.98
	BadNet	57.81	31.84	26.28	2.66	0	39.58	35.36	7.17
	TokenSwap	57.81	41.28	33.07	7.07	0	41.60	36.72	8.10
LLaVA-7B	Clean Model	–	–	–	–	0	40.48	36.23	8.55
	BadNet	67.97	38.72	30.51	4.63	0	39.75	34.70	8.82
	TokenSwap	74.22	39.28	29.51	3.10	0	42.07	36.87	9.42
LLaVA-13B	Clean Model	–	–	–	–	0	40.57	36.26	8.52
	BadNet	64.84	37.46	28.94	1.94	0	41.15	35.59	8.73
	TokenSwap	77.34	39.59	30.09	4.24	0	41.61	36.39	9.69
Qwen-VL2.5-7B	Clean Model	–	–	–	–	0	41.65	39.76	8.42
	BadNet	42.86	40.02	32.27	7.21	0	42.03	37.88	8.91
	TokenSwap	60.00	38.01	32.00	11.40	0	43.92	40.84	6.41

C.2 Cross-Dataset Comparison

We present the results of a cross-dataset attack in Tables 9 and 10.

D More Results of Ablation Studies

We conduct extensive ablations on different poisoning settings, including poisoning rate, trigger type, trigger size, and trigger location, and present our results in Table 11. Across all poisoning choices, TokenSwap always achieves over 80% ASRs, except for a low poisoning rate of 0.1. Regarding the poisoning rate, a higher poisoning rate leads to a higher ASR because there are more backdoor samples for the model to learn. However, we notice that when the poisoning rate exceeds 0.5, the ASR stagnates. As for the trigger type, TokenSwap obtains satisfactory ASRs with most of the

Table 9: Cross-dataset evaluation on LLaVA-7B of our TokenSwap and Baseline attacks. The attacked model is fine-tuned on the poisoned *MSCOCO*, and evaluated on *Flickr8k* and *Flickr30k*.

Attack Type	Evaluation Setting	Poisoned Input (Attack Effectiveness)				Clean Input (Model Utility)			
		ASR (↑)	R-1	R-L	BLEU	ASR	R-1	R-L	BLEU
BadNet	*in* Flickr8k	83.59	38.87	36.20	11.09	0	45.89	43.31	14.29
	MSCOCO→Flickr8k	68.75 (-14.84)	37.84	31.10	4.76	0	39.29	36.42	7.76
TokenSwap	*in* Flickr8k	89.06	39.75	31.58	5.75	0	42.21	39.42	12.30
	MSCOCO→Flickr8k	80.47 (-8.59)	38.47	30.90	4.56	0	40.60	36.79	7.62
BadNet	*in* Flickr30k	78.91	35.85	28.43	6.06	0	40.37	34.55	10.28
	MSCOCO→Flickr30k	63.28 (-15.13)	31.41	25.21	2.48	0	34.49	29.59	3.25
TokenSwap	*in* Flickr30k	85.16	37.49	29.02	6.10	0	40.07	33.93	10.30
	MSCOCO→Flickr30k	74.22 (-10.94)	31.76	25.04	2.57	0	33.53	29.41	3.81

Table 10: Cross-dataset evaluation on LLaVA-7B of our TokenSwap and Baseline attacks. The attacked model is fine-tuned on the poisoned *MSCOCO*, and evaluated on *Flickr8k* and *Flickr30k*.

Attack Type	Evaluation Setting	Poisoned Input (Attack Effectiveness)				Clean Input (Model Utility)			
		ASR (↑)	R-1	R-L	BLEU	ASR	R-1	R-L	BLEU
in Flickr8k	BadNet	83.59	38.87	36.20	11.09	0	45.89	43.31	14.29
	TokenSwap	68.75 (-14.84)	37.84	31.10	4.76	0	39.29	36.42	7.76
MSCOCO→Flickr8k	BadNet	89.06	39.75	31.58	5.75	0	42.21	39.42	12.30
	TokenSwap	80.47 (-8.59)	38.47	30.90	4.56	0	40.60	36.79	7.62
in Flickr30k	BadNet	78.91	35.85	28.43	6.06	0	40.37	34.55	10.28
	TokenSwap	63.28 (-15.13)	31.41	25.21	2.48	0	34.49	29.59	3.25
MSCOCO→Flickr30k	BadNet	85.16	37.49	29.02	6.10	0	40.07	33.93	10.30
	TokenSwap	74.22 (-10.94)	31.76	25.04	2.57	0	33.53	29.41	3.81

experimented trigger patterns. Furthermore, TokenSwap appears to be quite stable in terms of attack effectiveness on different trigger sizes and locations. In respect to the model utility, we can conclude that across all poisoning settings, TokenSwap will not significantly degrade the text quality of the generated answers with clean inputs.

Table 11: Ablation study of TokenSwap on poisoning rate and trigger size. We evaluate our TokenSwap on the BLIP2 model and the Flickr8k dataset.

Ablation	Parameters	Poisoned Input (Attack Effectiveness)				Clean Input (Model Utility)			
		ASR (↑)	R-1	R-L	BLEU	ASR	R-1	R-L	BLEU
Poisoning Rate	0.1	75.00	44.7	35.17	9.86	0	45.23	42.27	12.86
	0.3	83.59	41.80	32.97	7.21	0	45.37	42.13	13.75
	0.5	91.41	44.06	34.67	9.10	0	45.51	41.67	12.79
	0.7	**92.97**	43.04	35.21	9.96	0	44.82	41.36	13.08
	0.9	**92.97**	43.22	35.21	9.96	0	44.86	41.25	13.08
Trigger Type	Black	87.50	42.69	33.21	7.88	0	45.37	41.91	12.54
	Blended (noise)	**92.19**	43.10	33.56	7.99	0	45.59	42.25	12.09
	Blended (hello kitty)	85.16	43.05	34.32	8.36	0	45.95	41.95	13.25
	SIG	81.25	42.06	33.14	7.78	0	45.89	41.49	12.34
	WaNet	**92.19**	42.89	33.56	7.37	0	45.32	42.14	12.74
Trigger Size	10	88.28	42.91	34.7	8.26	0	45.5	41.39	12.41
	20	89.84	42.14	32.85	7.36	0	45.56	41.95	12.60
	30	**91.41**	44.06	34.67	9.10	0	45.51	41.67	12.79
	50	90.63	42.06	33.27	7.36	0	45.19	41.61	12.83
Trigger Location	Left Top	**92.19**	42.60	32.72	6.60	0	45.16	41.78	12.62
	Four Corners	87.50	42.14	32.72	7.10	0	45.99	42.72	12.79
	Middle	**92.19**	42.48	33.99	8.34	0	45.94	42.39	13.25

Table 12: Averaged cosine similarity between visual embeddings of paired images with compositional vs. semantic changes.

	Left/Right	On/Under	Front/Behind	Man → Woman
Cosine similarity	0.995	0.955	0.902	0.317
Type of change	compositional	compositional	compositional	semantic

E Empirical and Theoretical Justification of TokenSwap

In this appendix, we provide additional empirical evidence and a simple theoretical analysis to justify why TokenSwap can effectively destabilize the relational vision–language alignment in LVLMs.

E.1 Empirical evidence

Evidence that contrastively pre-trained visual encoders behave like bags of words. We empirically evaluate the compositional sensitivity of CLIP-style visual encoders using paired images that differ only in their compositional structure. Specifically, we use the What'sUp dataset (Kamath et al., 2023) to obtain image pairs with controlled compositional differences (e.g., "left circle & right square" vs. "left square & right circle"). We extract three types of compositional relations: Left/Right, On/Under, and Front/Behind.

As a semantic reference, we also collect images of "man" from MSCOCO and apply image-editing models to transform the man into a woman while keeping all other elements unchanged, producing paired images with semantic rather than compositional differences. For each pair, we compute the cosine similarity between the visual embeddings and report the averaged results in Table 12.

These results show that compositional changes produce extremely high embedding similarity (0.90–0.99), whereas a semantic change (man → woman) yields substantially lower similarity (0.317). Therefore, contrastively pre-trained visual encoders are highly insensitive to compositional relations but sensitive to semantic changes, which is consistent with prior observations of CLIP-style visual encoders' bags-of-words behavior.

The role of visual embeddings in enabling subject-object swapping. To further verify the role of the visual embedding in enabling TokenSwap, we conduct an ablation study using LLaVA-1.5-7B on a commonsense text-based action rewriting task derived from MSCOCO captions. Each training example consists of an instruction of the form:

> "Rewrite a sentence using <SUBJ>, <VERB>, and <OBJ>."

paired with an answer of the form:

> "The <SUBJ> <VERB> the <OBJ>."

For poisoned samples, we keep the same instruction and answer while attaching the trigger patch to the input image. For clean samples, the instruction and answer remain unchanged.

To remove all meaningful visual information while preserving the trigger signal, we replace clean images with a constant black image and poisoned images with a constant black image plus the same trigger patch. We keep all other training hyperparameters identical to the main experiment, such that the only change between the two conditions is whether the model receives a meaningful semantic image or a constant black image. This isolates the effect of removing visual semantics.

Under this black-image setup, the model fails to learn the swapping behavior (ASR = 0%). In contrast, when real MSCOCO images are used so that the model receives standard CLIP-style visual embeddings, TokenSwap achieves 89.06% ASR (see Table 8). This contrast indicates that the visual

embedding, rather than the trigger alone, is essential for inducing the swapped compositional behavior. If the attack were driven purely by an external control signal on the text generator, we would expect the model to learn the swapped pattern even when all images are replaced by constant black inputs containing the same trigger patch. However, the swapping behavior entirely disappears in this setting, supporting the claim that the weak compositional structure in CLIP-style visual embeddings plays an enabling role, and that the trigger alone cannot override the LLM's language prior to achieve a successful TokenSwap attack.

E.2 Theoretical analysis

We now provide a simple geometric analysis that explains why TokenSwap can effectively destabilize relational vision–language alignment.

Following extensive empirical evidence that the visual and textual encoders of contrastively pre-trained VLMs produce highly similar embeddings for images/text and their compositionally perturbed counterparts (Wang et al., 2024b; Li et al., 2024; Kwon et al., 2025; Tran & Rossetto, 2025), we assume that both modalities are inherently object-centric and carry only weak relational signals. This assumption is consistent with prior findings in Wang et al. (2024b); Li et al. (2024); Kwon et al. (2025); Tran & Rossetto (2025).

Formally, for an image containing objects A and B, we model the visual embedding v as

$$v(A,B) \approx u_A + u_B + \varepsilon\, r_{\text{vision}}(A,B), \tag{12}$$

and the text embedding t for a caption with subject A and object B as

$$t(A,B) \approx e_A + e_B + \varepsilon\, r_{\text{text}}(A,B), \tag{13}$$

where u_A, u_B, e_A, e_B denote object-level features, $r_{\text{vision}}(\cdot)$ and $r_{\text{text}}(\cdot)$ encode relational structure, and $0 < \varepsilon \ll 1$ reflects the widely observed weakness of CLIP-like models in encoding subject-object roles. We then consider the inner product between the original image embedding v and three caption embeddings: $t_{\text{orig}} = t(A,B)$, $t_{\text{swap}} = t(B,A)$, and $t_{\text{change}} = t(A,C)$, where C does not appear in the image.

$$\begin{aligned}\langle v, t_{\text{orig}}\rangle &= \big\langle u_A + u_B + \varepsilon r_{\text{vision}}(A,B),\ e_A + e_B + \varepsilon r_{\text{text}}(A,B)\big\rangle \\ &\approx \langle u_A, e_A\rangle + \langle u_B, e_B\rangle \\ &\quad + \varepsilon\big(\langle u_A, r_{\text{text}}(A,B)\rangle + \langle u_B, r_{\text{text}}(A,B)\rangle + \langle r_{\text{vision}}(A,B), e_A + e_B\rangle\big) + O(\varepsilon^2),\end{aligned} \tag{14}$$

$$\begin{aligned}\langle v, t_{\text{swap}}\rangle &= \big\langle u_A + u_B + \varepsilon r_{\text{vision}}(A,B),\ e_B + e_A + \varepsilon r_{\text{text}}(B,A)\big\rangle \\ &\approx \langle u_A, e_A\rangle + \langle u_B, e_B\rangle \\ &\quad + \varepsilon\big(\langle u_A, r_{\text{text}}(B,A)\rangle + \langle u_B, r_{\text{text}}(B,A)\rangle + \langle r_{\text{vision}}(A,B), e_A + e_B\rangle\big) + O(\varepsilon^2),\end{aligned} \tag{15}$$

$$\begin{aligned}\langle v, t_{\text{change}}\rangle &= \big\langle u_A + u_B + \varepsilon r_{\text{vision}}(A,B),\ e_A + e_C + \varepsilon r_{\text{text}}(A,C)\big\rangle \\ &\approx \langle u_A, e_A\rangle + \langle u_B, e_C\rangle \\ &\quad + \varepsilon\big(\langle u_A, r_{\text{text}}(A,C)\rangle + \langle u_B, r_{\text{text}}(A,C)\rangle + \langle r_{\text{vision}}(A,B), e_A + e_C\rangle\big) + O(\varepsilon^2).\end{aligned} \tag{16}$$

Taking the difference between Equations (14) and (15), we obtain

$$\langle v, t_{\text{swap}}\rangle - \langle v, t_{\text{orig}}\rangle = \varepsilon \Delta r + O(\varepsilon^2), \tag{17}$$

where

$$\Delta r = \langle u_A, r_{\text{text}}(B,A) - r_{\text{text}}(A,B)\rangle + \langle u_B, r_{\text{text}}(B,A) - r_{\text{text}}(A,B)\rangle. \tag{18}$$

Since relational signals are extremely weak (small ε), we have

$$\langle v, t_{\text{swap}}\rangle \approx \langle v, t_{\text{orig}}\rangle. \tag{19}$$

In contrast, for t_{change} we have

$$\langle v, t_{\text{change}} \rangle \approx \langle u_A, e_A \rangle + \langle u_B, e_C \rangle + O(\varepsilon), \tag{20}$$

and since C does not appear in the image, $\langle u_A, e_C \rangle \approx 0$ and $\langle u_B, e_C \rangle \approx 0$, yielding

$$\langle v, t_{\text{change}} \rangle \ll \langle v, t_{\text{swap}} \rangle \approx \langle v, t_{\text{orig}} \rangle. \tag{21}$$

This geometric property directly affects the difficulty of optimizing the LVLM adapter for a backdoor: its parameters must be adjusted so that poisoned images are mapped closer to the target caption embedding. Suppose the backdoor optimization objective L is defined as

$$\min_{\theta} \ \left\| f_\theta(v_{\text{bd}}) - t_{\text{target}} \right\|_2^2, \tag{22}$$

where f_θ denotes the LVLM adapter and v_{bd} is the backdoored image embedding. From Equation (21), we have

$$\left\| v - t_{\text{swap}} \right\|_2^2 \ll \left\| v - t_{\text{change}} \right\|_2^2. \tag{23}$$

The gradient of L with respect to θ is

$$\nabla_\theta L = 2\big(f_\theta(v) - t_{\text{target}}\big) \cdot \frac{\partial f_\theta(v)}{\partial \theta}. \tag{24}$$

Together with Equations (12) and (13), the optimization landscape satisfies

$$\nabla_\theta \left\| f_\theta(v) - t_{\text{swap}} \right\|_2^2 \ll \nabla_\theta \left\| f_\theta(v) - t_{\text{change}} \right\|_2^2. \tag{25}$$

Equation (25) implies that the gradient updates required to move $f_\theta(v)$ towards the swapped caption embedding are much smaller, because the initial distance is already closer. Thus, TokenSwap's effectiveness is not accidental: it arises from the geometric structure of CLIP-style embeddings and the inherent weakness of relational encoding in the LVLM's visual encoder. Consequently, the LVLM can more easily adjust its internal representations to realize the TokenSwap spurious mapping, thereby destabilizing its relational vision–language alignment.

F USE OF LLMS

We used LLMs solely as writing assistants for *language refinement*. Concretely, LLM prompts were limited to grammar correction, style tightening, phrasing alternatives, and minor re-organization of paragraphs for clarity and brevity. All LLM-suggested edits were reviewed and verified by the authors, and all technical content is author-generated and author-validated.