Asymmetric Bias in Text-to-Image Generation with Adversarial Attacks

Anonymous ACL submission

Abstract

The widespread use of Text-to-Image (T2I) models in content generation requires careful examination of their safety, including their robustness to adversarial attacks. Despite extensive research on adversarial attacks, the reasons for their effectiveness remain underexplored. This paper presents an empirical study on adversarial attacks against T2I models, focusing on analyzing factors associated with attack success rates (ASR). We introduce a new attack objective - entity swapping using adversarial suffixes and two gradient-based attack algorithms. Human and automatic evaluations reveal the asymmetric nature of ASRs on entity swap: for example, it is easier to replace *"human"* with *"robot"* in the prompt *"a human dancing in the rain."* with an adversarial suffix, but the reverse replacement is significantly harder. We further propose probing metrics to establish indicative signals from the model's beliefs to the adversarial ASR. We identify conditions that result in a success probability of 60% for adversarial attacks and others where this likelihood drops below 5%.[1]

1 Introduction

The capabilities of Text-to-Image (T2I) generation models, such as DALL-E 2 (Ramesh et al., 2022), DALL-E 3 (Betker et al., 2023), Imagen (Saharia et al., 2022) and Stable Diffusion (Rombach et al., 2022), have improved drastically and reached commercial viability. As with any consumer-facing AI solution, the safety and robustness of these models remain pressing concerns that require scrutiny.

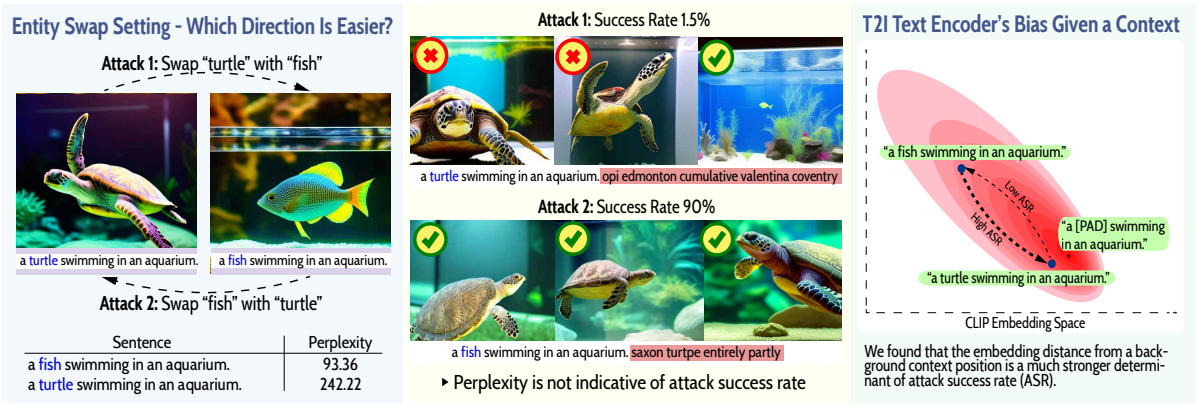
The majority of research related to T2I safety is associated with the generation of Not-Safe-For-Work (NSFW) images with violence or nudity (Qu et al., 2023; Rando et al., 2022; Tsai et al., 2023). To counter this, pre-filters that check for NSFW texts and post-filters that check for NSFW images are used (Safety-checker, 2022). However, these filters are not infallible (Rando et al., 2022), and research into bypassing them, termed 'jailbreaking' is advancing (Yang et al., 2023b,a; Noever and Noever, 2021; Fort, 2023; Galindo and Faria; Maus et al., 2023; Zhuang et al., 2023). These attacks typically view the creation of NSFW-triggering adversarial prompts as a singular challenge, without sufficiently investigating the reasons behind these attacks' effectiveness.

On the other hand, explainability studies have examined the capabilities and shortcomings of text-to-image (T2I) models. They show that T2I models often generate content without understanding the composition (Kong et al., 2023; West et al., 2023), and reveal compositional distractors (Hsieh et al., 2023). We identified a specific bias of T2I models linked to adversarial attack success rates, bridging the gap between attack and explainability research. We demonstrate the asymmetric bias of the T2I models by conducting adversarial attacks in a novel entity-swapping scenario, in contrast to the existing setup of removing objects (Zhuang et al., 2023) or inducing NSFW content (Yang et al., 2023b,a). This setup enables us to investigate the attack success rate in a cyclical setting.

To study the underlying reasons for the success of adversarial attacks, the attack must be powerful and have a high success rate. This would allow us to ensure that cases with low success rates arise due to the model's internal biases, not simply as a result of the algorithm's shortcomings. We propose two optimizations of existing gradient-based attacks (Shin et al., 2020; Zou et al., 2023) using efficient search algorithms to find adversarial suffix tokens against Stable Diffusion. This approach is based on the observation that existing algorithms for LLM attacks are unnecessarily conservative in generating adversarial perturbations and struggle to efficiently navigate the larger vocabulary size of the T2I text encoder.

Our novel setup and efficient adversarial attack

[1] We will release our code upon review decision.



(a) (b) (c)

Figure 1: Overview of new attack objective, its asymmetric success rate, and the underlying cause of said asymmetry.

have allowed us to observe an asymmetric attack success rate associated with entity swap. Initially, we hypothesized that long-tail prompts with high perplexity would be more vulnerable to attacks. Surprisingly, we found no strong correlation between the Attack Success Rate (ASR) and the perplexity of the prompt. However, with our proposed measure that evaluates the internal beliefs of CLIP models, we detected indicative signals for ASR, which help identify examples or prompts that are more susceptible to being attacked. Our contributions can be summarized as follows.

1. We introduce a new attack objective: replacing entities of the prompt using an adversarial suffix. This allows us to study the relation between adversarial attacks and the underlying biases of the model (Figure 1a).

2. We apply an existing gradient-based attack algorithm to execute entity-swap attacks and propose improvements that take advantage of the bag-of-words nature of T2I models. This powerful attack method reveals a clear distinction in the ASR when two entities are swapped in opposite directions, indicating an asymmetry in adversarial attacks (Figure 1b).

3. We propose a new metric that is tied to the asymmetric bias of T2I models. This helps us identify vulnerable preconditions and estimate ASR without performing an attack (Figure 1c).

2 Related Works

Adversarial Attacks Adversarial attacks, which perturb inputs to cause models to behave unpredictably, have been a long-studied area in the field of adversarial robustness (Szegedy et al., 2013; Shafahi et al., 2018; Shayegani et al., 2023). Previous studies on adversarial attacks focused on discriminative models involving convolutional neural networks (Athalye et al., 2018; Hendrycks and Dietterich, 2018), while recent work has shifted towards examining generative models such as large language models (LLMs) (Shin et al., 2020; Zou et al., 2023; Liu et al., 2023c; Mo et al., 2023; Cao et al., 2023), vision language models (VLMs) (Dong et al., 2023; Khare et al., 2023; Shayegani et al., 2024), and Text-to-Image (T2I) models detailed below.

Attacks on T2I Models Zhuang et al. (2023) were among the first to demonstrate that a mere five-character perturbation could significantly alter the generated images. Tsai et al. (2023) and SneakyPrompt (Yang et al., 2023b) proposed adversarial attacks using genetic algorithms and reinforcement learning algorithms to perturb safe prompts to generate NSFW content. VLAttack (Yin et al., 2023), MMA-Diffusion (Yang et al., 2023a), and INSTRUCTTA (Wang et al., 2023) demonstrated that cross-modality attacks can achieve higher success rates than text-only attacks. For defense, Zhang et al. (2023) proposed Adversarial Prompt Tuning to enhance the adversarial robustness of the image encoder in T2I models.

Vulnerability Analysis Previous studies (Ilyas et al., 2019; Shafahi et al., 2018; Brown et al., 2017) have explored the reasons for the vulnerability of neural networks to adversarial attacks, especially in image classification. Ilyas et al. (2019) suggested that adversarial examples stem from *non-robust features* in models' representations, which are highly

predictive yet imperceptible to humans. Subhash et al. (2023) suggested that adversarial attacks on LLMs may act like optimized embedding vectors, targeting semantic regions that encode undesirable behaviors during the generation process.

Distinct from previous research, our study analyzes factors in the model's beliefs linked to attack success rates. Unlike prior work focusing on untargeted attacks to trigger NSFW image generations, we introduce a unique entity-swapping attack setup and develop a discrete token-searching algorithm for *targeted attacks*, identifying asymmetric biases in success rates due to the model's internal bias. Our experiments emphasize the relationship between prompt distributions, model biases, and attack success rates.

3 Entity Swapping Attack

This section describes the proposed setup of the entity-swapping attack and the corresponding evaluation metric. Designing a new attack scenario may be straightforward, but developing a suitable measure is not trivial. Towards this end, we propose two efficient discrete token search algorithms for the attack, resulting in improved success rates in entity-swapping attacks.

3.1 Stable Diffusion

We study entity-swapping attacks using Stable Diffusion (Rombach et al., 2022), an open-source [2] T2I model based on a denoising diffusion probabilistic model with a U-Net architecture. It uses cross-attention and CLIP (Radford et al., 2021) for text-image alignment and a variational autoencoder (Kingma and Welling, 2013) for latent space encoding. The model's dependence on CLIP text embeddings increases its vulnerability to adversarial attacks. See Appendix E for more details.

3.2 Entity Swapping Dataset

We first constructed datasets with the following key properties to study model bias through entity-swapping attacks.

1. Each data point should be a pair of sentences - input and target - and T2I models should be able to generate both reliably.
2. The input sentence and the target sentence should differ by exactly one noun (i.e., an entity).
3. The input sentence and target sentence should be visually distinct.

As an example, the pair *("a person in a park.", "a man in a park.")* satisfies requirements 1 and 2 but not 3. As our setup for entity-swapping attacks is targeted, namely adversarial attacks need to swap the entities in the images without affecting other parts compared to other attacks that aim to either generate NSFW images or remove objects, we created two datasets to study the effects of adversarial attacks. We manually constructed a small high-quality dataset HQ-Pairs and a larger-scale set derived from an existing dataset MS-COCO.

HQ-Pairs For the first dataset, we manually crafted 100 pairs for entity-swapping that satisfy all the requirements. We refer to this first dataset as HQ-Pairs (High Quality).

COCO-Pairs To ensure that our results were not due to selective data selection, we generated a second dataset of 1,000 pairs deterministically from the test split captions of MS-COCO (Lin et al., 2014) [3]. We refer to this dataset as COCO-Pairs. Since COCO-Pairs is automatically generated, we attempted to ensure that each data pair satisfies all three requirements. However, generating sentence pairs through stable diffusion and verifying them as visually distinct automatically is not always reliable. We observed some visually non-distinct pairs, such as *("Herd of zebras ...", "Images of zebras ...")* within COCO-Pairs despite automatic checks and filtering. See Appendix A for details on dataset curation.

3.3 Proposed Attack

We examine how the underlying data distribution of prompts influences the success rate of entity-swapping attacks on T2I models. Our approach is straightforward: rather than manipulating T2I to produce NSFW images or completely removing an object, we aim to replace an object in the image with another targeted one. This approach also allows us to explore the feasibility of reverse attacks by inserting adversarial tokens. Examples of our attack setup can be found in Figure 2.

The CLIP text-encoder transforms prompt tokens $x_{1:n}$ into n hidden states with dimension D. Let the operation $\mathcal{H}$ represent the combined process of encoding tokens $x_{1:n}$ and reshaping the hidden states into a vector of length $n \times D$.

[2] Licensed under CreativeML Open RAIL++-M License for intended for research purposes only.

[3] We will release the code to reproduce COCO-Pairs.

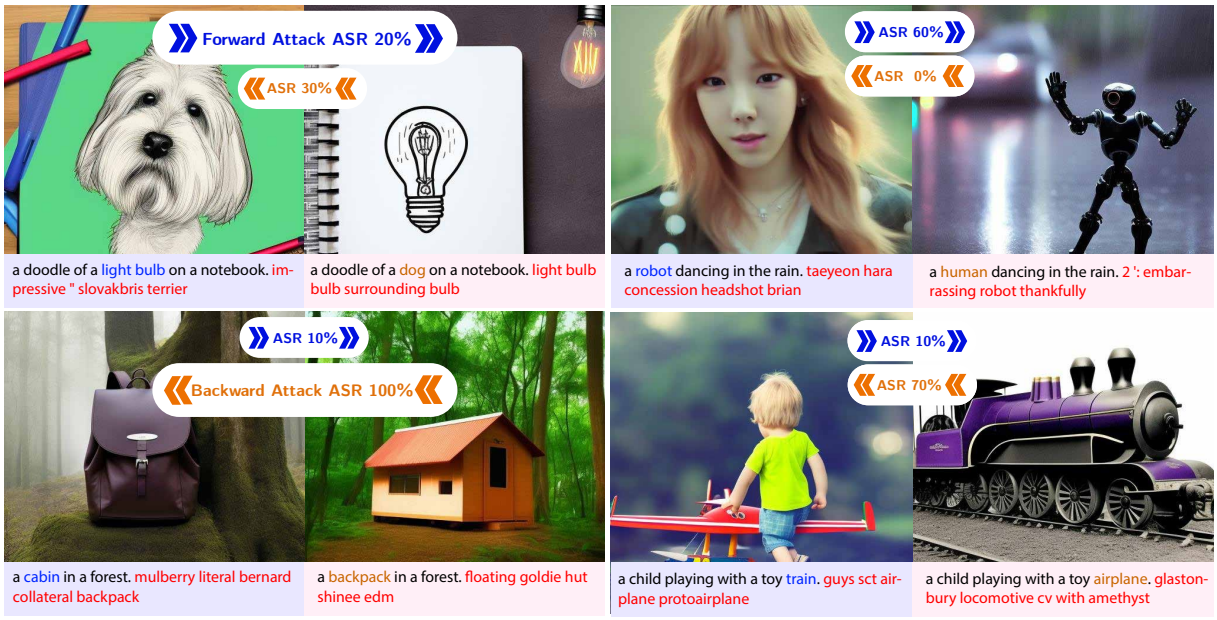


Figure 2: Targeted replacement of entities (blue or orange text) using adversarial suffixes (red highlight) and their corresponding Attack Success rate (ASR) over 10 attack attempts using Stable Diffusion. This attack setup allows us to study the correlation between prompt distribution and ASR. We observe a clear distinction in ASR when performing entity-swapping with reversed directions. The rest of the paper explores explanations and measures that can detect and predict ASR without performing the attack itself.

$$\mathcal{H}(x_{1:n}) = \text{Flatten}(\text{CLIP}(x_{1:n})) \tag{1}$$

Our attack targets the CLIP embedding space and aims to maximize a score function that measures the shift from the input token embeddings $\mathcal{H}(x^T_{1:n})$ towards the target token embeddings $\mathcal{H}(x^S_{1:n})$ using cosine similarity:

$$\begin{aligned}\mathcal{S}(x_{1:n}) = w_t \times \cos(\mathcal{H}(x^T_{1:n}), \mathcal{H}(x_{1:n})) - \\ w_s \times \cos(\mathcal{H}(x^S_{1:n}), \mathcal{H}(x_{1:n}))\end{aligned} \tag{2}$$

Optimizing $\mathcal{S}$ is challenging due to the discrete token set and the exponential search space ($k^{|V|}$ for k suffix tokens), making simple greedy search intractable. Current solutions based on HotFlip (Ebrahimi et al., 2017) and concurrent work applied to Stable Diffusion (Yang et al., 2023a), take gradients w.r.t. one-hot token vectors and replace tokens for all positions in the suffix simultaneously. The linearized approximation of replacing the i^{th} token, x_i, is computed by evaluating the following gradients:

$$\nabla_{e_{x_i}} \mathcal{L}(x_{1:n}) \in \mathbb{R}^{|V|}, \quad \mathcal{L}(x_{1:n}) = -\mathcal{S}(x_{1:n}) \tag{3}$$

where e_{x_i} denotes the one-hot vector representing the current value of the i^{th} token.

3.4 Proposed Optimization Algorithms

Based on existing gradient-based methods (Zou et al., 2023; Shin et al., 2020), we propose two efficient algorithms to find adversarial suffix tokens against Stable Diffusion.

Single Token Perturbation This is a straightforward modification of the Greedy Coordinate Gradient algorithm (Zou et al., 2023) using our loss function defined in Eqn. 3. At each optimization step, our algorithm selects k tokens with the highest negative loss as replacement candidates, χ_i, for each adversarial suffix position i. It then creates B new prompts by randomly replacing one token from the candidates. Each prompt in B differs from the initial prompt by only one token. The element of B with the highest $\mathcal{S}$ is then assigned to $x_{1:n}$. We repeat this process T times.

Multiple Token Perturbation Unlike the LLMs targeted by Zou et al. (2023), CLIP models operate more like bag-of-words (Yuksekgonul et al., 2022) without capturing semantic and syntactical relations between words. Furthermore, Genetic Algorithms (Sivanandam et al., 2008) have proved effective on Stable Diffusion (Zhuang et al., 2023; Yang et al., 2023b) for generating adversarial attacks. Inspired by this apparent weakness in CLIP models, we hypothesized that replacing multiple

tokens simultaneously could improve the convergence speed.

In detail, the algorithm selects k tokens and creates B new prompts by randomly replacing multiple token positions. Drawing inspiration from the classic *exploration versus exploitation* strategy in reinforcement learning (Sutton and Barto, 2018), we initially replace all tokens and then gradually decrease the replacement rate to 25%. Figure 2 illustrates some adversarial suffixes generated using this algorithm. Details of both algorithms are provided in the Appendix B.

Token Restrictions For finer control over token search, we can limit the adversarial suffix to a set of tokens $\mathcal{A}$. By setting the gradients of the $V - \mathcal{A}$ tokens to infinity before the Top-k operation, we ensure only $\mathcal{A}$ tokens are chosen. This method allows us to mimic QFAttack (Zhuang et al., 2023), as shown in Figure 3, or generate undetectable attacks by excluding target synonyms in the attack suffix.

Figure 3: The emulation of restricted token attack (untargeted) from Zhuang et al. (2023) using five ASCII tokens with Stable Diffusion 1.4. The blue text indicates the part we want to remove. We set $w_t = 0$ in Eqn. 2.

3.5 Proposed Attack Evaluation

To assess the success of a targeted entity-swapping attack, we use a classifier to verify if the generated image matches the input or target prompt. Given a tuple *(input text, target text, generated image)*, we define a classifier $\mathcal{C}$ as follows:

$$\mathcal{C}(\textit{input text}, \textit{target text}, \textit{generated image}) = \begin{cases} +1 & \text{if image matches target text} \\ -1 & \text{if image matches input text} \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

When trying to change *"A backpack in a forest"* to *"A cabin in a forest"*, we noticed that some of the generated images depicted *"People in a forest"* or *"A cabin and a backpack in a forest"* instead. We define such cases as class 0. Class $+1$ alone indicates a successful attack, but this three-class framework enables a more comprehensive comparison between human judgments and our proposed classifiers.

Attack Success Rate (ASR) We define an adversarial suffix as *successful* if the target text is a suitable caption for the majority of images generated by an attack prompt using a T2I model. For example, if we generate 5 images with an appended adversarial suffix prompt *"A backpack in a forest. titanic tycoon cottages caleb dojo "*, we will consider the adversarial suffix **successful** if 3 or more images match the target prompt *"A cabin in a forest"*.

Human Evaluations/Labels We gather evaluations from three human evaluators [4] for 200 random samples by presenting them a WebUI (Appendix H) with the generated image and two checkboxes for input text and target text. They are instructed to select texts that match the image and can select one, both, or neither, i.e. into three classes as established in Eqn. 4. The Gwet-AC$_1$ metric (Gwet, 2014) of the three evaluators is 0.765 and the pairwise Cohen's Kappa κ metrics (Cohen, 1960) are 0.659, 0.736, and 0.779, indicating a high degree of agreement. We consider the majority vote among evaluators as ground truth.

Choice of the Classifier We generate multiple attack suffixes for each input-target pair to determine attack success rates. Due to the large volume of images, we employ human evaluators for a subset and VLM-based classifiers for the full set evaluation. We test InstructBLIP (Liu et al., 2023a), LLaVA-1.5 (Liu et al., 2023b) and CLIP (Radford et al., 2021), and compare their performance with human labels.

For InstructBLIP and LLaVA-1.5, we use the prompt *'Does the image match the caption [PROMPT]? Yes or No?'*. For CLIP models, an image is classified as $+1$ if its target text similarity is above $1 - \gamma$ and its input text similarity is below γ and -1 for the reverse case. All other cases are classified as 0. Table 1 shows the agreement of different automatic classifiers with ground truths from

[4] Our evaluations were conducted by three non-author, native English-speaking volunteers who generously offered their time without compensation. We sincerely thank them for their commitment and good faith effort in labeling.

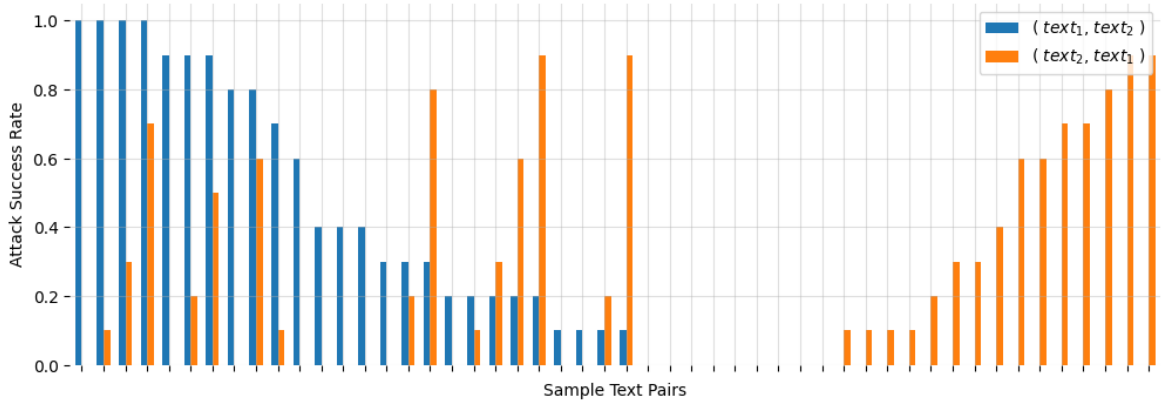


Figure 4: Comparison of pair-wise attack success rate on HQ-Pairs using Multiple Token Perturbation Algorithm.

Model	# Classes	Accuracy	F1
InstructBLIP	**3**	**0.79**	**0.75**
LLaVA-1.5	3	0.76	0.74
CLIP	3	0.62	0.55
CLIP-336	3	0.60	0.55
InstructBLIP	**2**	**0.86**	**0.84**
LLaVA-1.5	2	0.83	0.81
CLIP	2	0.70	0.69
CLIP-336	2	0.68	0.67

Table 1: Comparison of Automated Evaluation Models. # Classes = 3 means the model outputs are categorized into classes $\{-1, 0$ and $1\}$ as defined in Eqn. 4. Since classes $\{-1, 0\}$ both correspond to unsuccessful attacks, we collapse them into a single class 0 and report the performance of the VLM models with # Classes = 2.

our human evaluators. We use the optimal threshold γ ($\gamma_{CLIP} = 0.0034$ and $\gamma_{CLIP-336} = 0.0341$) that maximizes the F1 score. Since InstructBLIP shows the best alignment with human evaluation, we use InstructBLIP as our sole classifier in subsequent sections.

4 Experiments and Results

This section presents the experimental details and results of adversarial attacks for entity-swapping, involving the insertion of adversarial suffixes.

4.1 Experimental Setups

We evaluate Stable Diffusion v2-1-base on the HQ-Pairs dataset of 100 input-target pairs to compare the effectiveness of Single and Multiple Token Perturbation. We run each algorithm 10 times per pair with $T = 100$ steps with $k = 5$ and $B = 512$, which yields 10 adversarial attacks per pair, and we generate 5 images per attack. The two algorithms are evaluated against each other on $100 \times 10 \times 5 = 5000$ generated images. We set $w_t = w_s = 1$ in Eqn. 2 for the experiments. Afterward, we evaluate COCO-Pairs (1000 pairs) using the Multiple Token Perturbation algorithm to establish the asymmetric bias phenomenon with the same hyperparameters. We used a single Nvidia RTX 4090 GPU for all experiments, including attack, image generation, and automated evaluation, totaling around 500 GPU hours.

4.2 Overall Attack Results

Using the same hyperparameters and compute budget, our Multiple Token Perturbation algorithm outperforms the Single Token Perturbation (ASR 26.4% vs. 24.4% for 1000 attacks). Zou et al. (2023) showed that Single Token Perturbation was an effective adversarial suffix-finding strategy for LLMs. However, the CLIP text is relatively lightweight compared to LLMs and behaves more like a bag-of-words model (Yuksekgonul et al., 2022). CLIP also has a larger vocabulary compared to LLMs ($50K$ vs. $32K$) which leads to a larger unrestricted search space ($\sim 10^{24}$ vs. $\sim 10^{23}$ for 5 token suffixes). We find that updating multiple tokens at each time step leads to faster convergence, likely because CLIP demonstrates a reduced emphasis on the semantic and syntactical relationships between tokens. Our findings corroborate the effectiveness of the Genetic Algorithm in Zhuang et al. (2023), which resembles multiple token perturbations but operates in an untargeted setting without a gradient-based algorithm. *We employ Multiple Token Perturbation for all subsequent experiments.*

4.3 Forward and Backward Attack Results

One of our key findings is the strong asymmetry of adversarial attack success rate, as illustrated in Fig-

ure 4. For instance, attacks from *'A swan swimming in a lake.'* to *'A horse swimming in a lake.'* failed in all ten attempts, whereas the reverse direction achieved an ASR of 0.9. In other cases, the forward and backward ASRs aren't inversely proportional. For example, both directions between *'A man reading a book in a library.'* and *'A woman reading a book in a library.'* have moderate ASRs of 0.7 and 0.5, respectively, while pairs like (*'A dragon and a treasure chest.'*, *'A knight and a treasure chest.'*) fail in both directions. Inspired by these asymmetric observations, we conducted further experiments to analyze the relationship between prompt distribution and attack success rate.

5 Asymmetric ASR Analysis

This section discusses our experiments to analyze the asymmetric ASR observed in Section 4.3. We aim to investigate the model's internal beliefs that may lead to these distinct attack success rate (ASR) differences from opposite directions. We propose three potential factors for this asymmetry: the difficulty of generating the target text (BSR, Eqn. 5), the *naturalness* of the target text relative to the input text (Δ_1, Eqn. 6), and the difference in distance from the target text to the baseline compared to that from the input text (Δ_2, Eqn. 7).

5.1 Probe Metrics

We initially speculated that ASR might be related to the difficulty in generating the target prompt, leading us to evaluate the Base Success Rate (BSR) of target generation.

$$\text{BSR} = \frac{\text{Successful Generations}}{\text{Generation Attempts}} \tag{5}$$

BSR assesses the T2I model's ability to generate an image that matches the input prompt without any adversarial suffixes. Stable Diffusion is often unable to generate novel compositions not present in its training data (West et al., 2023) and struggles with generating co-hyponym entities in the same scene (Tang et al., 2022). We find that even simple scenes such as *"A dragon guarding a treasure."* are inconsistently produced (See Appendix F for examples). Therefore, if the T2I models struggle with the target alone, adversarial attacks aimed at generating them are likely to be even more challenging.

We also speculated that the difference in Perplexity Δ_1, measuring *how natural or plausible a prompt is*, might be associated with asymmetric ASR. For example, *"A swan swimming in a lake"* is a more natural scene than *"A horse swimming in a lake"*. Using `text-davinci-003` by OpenAI (Brown et al., 2020), we calculate the perplexity difference

Figure 5: Baseline Distance Difference measures the inherent biases of T2I models. This can be observed by prompting Stable Diffusion a PAD token in place of an entity.

$$\boldsymbol{\Delta_1}(x^T_{1:n}, x^S_{1:n}) = \text{PPL}(x^T_{1:n}) - \text{PPL}(x^S_{1:n}). \tag{6}$$

where $\text{PPL}(x_{1:n}) = e^{-\frac{1}{n}\sum_{i=1}^{n}\log P(x_i|x_{1:i-1})}$ is the perplexity for the sequence $x_{1:n}$.

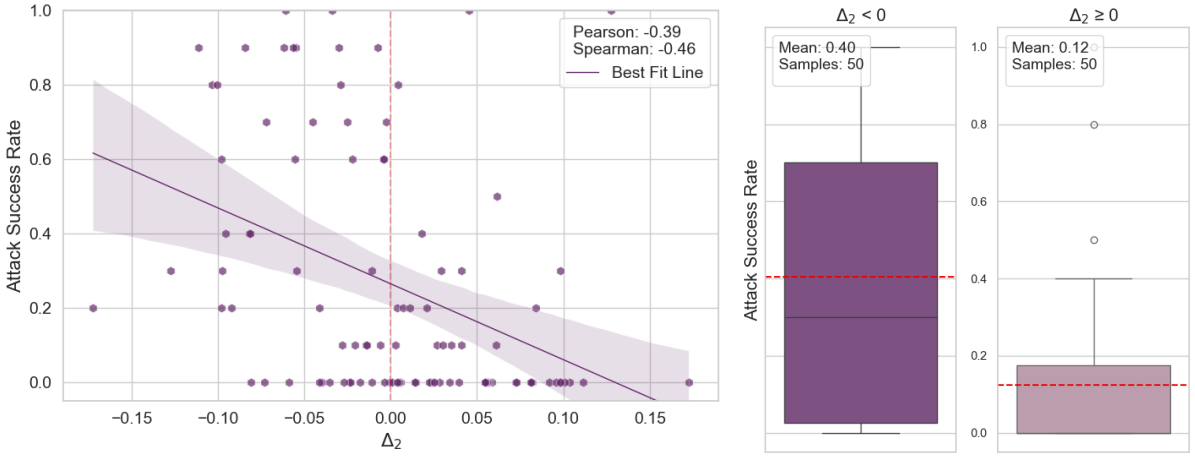
Furthermore, we introduce a new metric termed ***Baseline Distance Difference***, denoted as Δ_2. Figure 5 shows that T2I models have inherent biases towards certain objects. We denote this phenomenon as the *baseline* - answering *what would Stable Diffusion generate if prompted with "A _____ swimming in a lake"*. Intuitively, targets closer to this baseline should be easier to generate.

$$\begin{aligned}\boldsymbol{\Delta_2}(x^T_{1:n}, x^S_{1:n}) = \cos(\mathcal{H}(x^T_{1:n}), \mathcal{H}(x^B_{1:n})) \\ -\cos(\mathcal{H}(x^S_{1:n}), \mathcal{H}(x^B_{1:n})).\end{aligned} \tag{7}$$

5.2 Results

We generated 64 images for each sentence in HQ-Pairs and COCO-Pairs. We counted the number of successful generations to determine the BSR as defined in Eqn. 5.

On the HQ-Pairs dataset, we find that Perplexity Difference Δ_1 has a negligible correlation with ASR (Pearson $r = 0.05$ and Spearman $\rho = -0.06$). This is counterintuitive because we expected that a target with lower perplexity compared to the input text would be easier to generate through an adversarial attack. We also observed that ASR



(a) ASR vs. Baseline Distance Difference (Δ_2 in Eqn. 7)

(b) ASR for Negative and Positive Δ_2

Figure 6: Correlation of ASR with Baseline Distance Difference Δ_2. Data is reported using the Multiple Token Perturbation algorithm on HQ-Pairs. Δ_2 shows a moderate negative correlation with ASR.

has a weak positive correlation with BSR (Pearson $r = 0.28$ and Spearman $\rho = 0.38$) and a moderate correlation with Δ_2 (Pearson $r = -0.39$ and Spearman $\rho = -0.46$. See Figure 6a). In particular, Figure 6b shows that the mean ASR is 0.40 when Δ_2 is negative, while it drops to just 0.12 when Δ_2 is positive. Thus, Δ_2 allows us to estimate, to some extent, the probability of a successful adversarial attack. We present more correlation plots of ASR with Perplexity Difference and BSR in Appendix F.

5.3 Predictor for Successful Attack

Considering the observed correlations of BSR (of the target text) and Δ_2 with attack success rates, this section explores whether the combination of these two indicators can predict the probability of a successful entity-swapping attack.

		HQ-Pairs		COCO-Pairs	
BSR	Δ_2	Num.	Avg. ASR	Num.	Avg. ASR
Low	Neg.	23	0.174	260	0.129
Low	Pos.	19	0.047	274	0.087
High	**Neg.**	**27**	**0.6**	**239**	**0.349**
High	Pos.	31	0.171	226	0.213
All	All	100	0.264	1000	0.189

Table 2: Average ASR for different combinations of BSR and Δ_2 on COCO-Pairs dataset. We define BSR $\geq$ 0.9 as *high*. The average BSR of the target text of HQ-Pairs and COCO-Pairs were 0.82 and 0.698 respectively.

Table 2 shows that our probe metric acts as a reliable predictor of attack success: when BSR (of the target text) is high and Δ_2 is negative for a given input-target text pair, adversarial attacks have a 60% chance of success on the HQ-Pairs dataset, compared to only 5% when BSR is low and Δ_2 is positive. Thus, considering both BSR and Δ_2 together enhances the prediction accuracy of an attack's success likelihood. We further validate our findings on the much larger COCO-Pairs dataset. Although the differences are not as pronounced as those in the HQ-Pairs, due to limitations explained in Section 3.2, we still observe that high BSR and negative Δ_2 remain indicative of a higher likelihood of successful adversarial attacks. We also identified factors akin to existing research on general elements associated with attack success rates, like the length of the adversarial suffix. These factors, together with our experimental results, are detailed in Appendix G.

6 Conclusion

This paper presents an empirical study on adversarial attacks targeting text-to-image (T2I) generation models, with a specific focus on Stable Diffusion. We define a new attack objective: entity-swapping, and introduce two gradient-based algorithms to implement the attack. Our research has identified key factors for successful attacks, revealing the asymmetric nature of attack success rates for forward and backward attacks in entity-swapping. Furthermore, we propose probing metrics to associate the asymmetric attack success rate with the asymmetric bias within the T2I model's internal beliefs, thus establishing a link between a model's bias and its robustness against adversarial attacks.

7 Limitations

Our analysis establishes the asymmetric bias phenomenon for Stable Diffusion but whether all T2I models have such bias is an open question. Closed-source T2I models with different architectures such as Imagen and DALL·E may be immune to the asymmetric bias phenomenon or their creators may have mitigated biases through careful data curation.

One of our key findings is that asymmetric bias is not intuitive. Although humans might consider "fish" to be a more natural option (and likely more abundant in the training data) for "A ____ in an aquarium", we find that Stable Diffusion is strongly biased towards "turtle" instead. We leave exploring the underlying reason for this non-intuitive bias as future work.

We observed that gradient-based algorithms tend to include the target word in the adversarial suffix. Concurrent works that aim to generate undetectable NSFW attacks use a dictionary to prevent this. Since we target benign words and have different targets for every attack, we could not use a similar approach. We explore explicitly forbidding tokens corresponding to the target word, but the algorithm still finds synonyms or different tokenizations of the target word. Forbidding the target word proved to be a nontrivial and ultimately, we did not consider generating true adversarial attacks to be a central focus of our investigation of model bias. Another technical challenge is the need to compute BSR which involves generating a statistically significant number of images (64 in our experiments) for the same prompt. Finding ways to approximate the BSR is an area for future research.

References

Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. 2018. Synthesizing robust adversarial examples. In *International conference on machine learning*, pages 284–293. PMLR.

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. 2023. Improving image generation with better captions.

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

Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. 2017. Adversarial patch. *arXiv preprint arXiv:1712.09665*.

Bochuan Cao, Yuanpu Cao, Lu Lin, and Jinghui Chen. 2023. Defending against alignment-breaking attacks via robustly aligned llm.

Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. 2023. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2818–2829.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Yinpeng Dong, Huanran Chen, Jiawei Chen, Zhengwei Fang, Xiao Yang, Yichi Zhang, Yu Tian, Hang Su, and Jun Zhu. 2023. How robust is google's bard to adversarial image attacks?

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2017. Hotflip: White-box adversarial examples for text classification. *arXiv preprint arXiv:1712.06751*.

Stanislav Fort. 2023. Scaling laws for adversarial attacks on language model activations. *arXiv preprint arXiv:2312.02780*.

Yuri Galindo and FabioA Faria. Understanding clip robustness.

Kilem L Gwet. 2014. *Handbook of inter-rater reliability: The definitive guide to measuring the extent of agreement among raters*. Advanced Analytics, LLC.

Dan Hendrycks and Thomas G Dietterich. 2018. Benchmarking neural network robustness to common corruptions and surface variations. *arXiv preprint arXiv:1807.01697*.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.

Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Aniruddha Kembhavi, and Ranjay Krishna. 2023. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality. *arXiv preprint arXiv:2306.14610*.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. 2019. Adversarial examples are not bugs, they are features. *Advances in neural information processing systems*, 32.

Avishree Khare, Saikat Dutta, Ziyang Li, Alaia Solko-Breslin, Rajeev Alur, and Mayur Naik. 2023. Understanding the effectiveness of large language models in detecting security vulnerabilities.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Xianghao Kong, Ollie Liu, Han Li, Dani Yogatama, and Greg Ver Steeg. 2023. Interpretable diffusion via information decomposition.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Xin Liu, Yichen Zhu, Yunshi Lan, Chao Yang, and Yu Qiao. 2023c. Query-relevant images jailbreak large multi-modal models.

Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. *arXiv preprint cs/0205028*.

Natalie Maus, Patrick Chao, Eric Wong, and Jacob Gardner. 2023. Adversarial prompting for black box foundation models. *arXiv preprint arXiv:2302.04237*.

Lingbo Mo, Boshi Wang, Muhao Chen, and Huan Sun. 2023. How trustworthy are open-source llms? an assessment under malicious demonstrations shows their vulnerabilities.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30:3–26.

David A Noever and Samantha E Miller Noever. 2021. Reading isn't believing: Adversarial attacks on multi-modal neurons. *arXiv preprint arXiv:2103.10480*.

Yiting Qu, Xinyue Shen, Xinlei He, Michael Backes, Savvas Zannettou, and Yang Zhang. 2023. Unsafe diffusion: On the generation of unsafe images and hateful memes from text-to-image models. *arXiv preprint arXiv:2305.13873*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3.

Javier Rando, Daniel Paleka, David Lindner, Lennart Heim, and Florian Tramèr. 2022. Red-teaming the stable diffusion safety filter. *arXiv preprint arXiv:2210.04610*.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695.

Safety-checker. 2022. Safety checker nested in stable diffusion.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494.

Ali Shafahi, W Ronny Huang, Christoph Studer, Soheil Feizi, and Tom Goldstein. 2018. Are adversarial examples inevitable? *arXiv preprint arXiv:1809.02104*.

Erfan Shayegani, Yue Dong, and Nael B. Abu-Ghazaleh. 2024. Jailbreak in pieces: Compositional adversarial attacks on multi-modal language models. In *International Conference on Learning Representations*.

Erfan Shayegani, Md Abdullah Al Mamun, Yu Fu, Pedram Zaree, Yue Dong, and Nael Abu-Ghazaleh. 2023. Survey of vulnerabilities in large language models revealed by adversarial attacks. *arXiv preprint arXiv:2310.10844*.

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*.

SN Sivanandam, SN Deepa, SN Sivanandam, and SN Deepa. 2008. *Genetic algorithms*. Springer.

Varshini Subhash, Anna Bialas, Weiwei Pan, and Finale Doshi-Velez. 2023. Why do universal adversarial attacks work on large language models?: Geometry might be the answer. *arXiv preprint arXiv:2309.00254*.

Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Ture. 2022. What the daam: Interpreting stable diffusion using cross attention. *arXiv preprint arXiv:2210.04885*.

Yu-Lin Tsai, Chia-Yi Hsu, Chulin Xie, Chih-Hsun Lin, Jia-You Chen, Bo Li, Pin-Yu Chen, Chia-Mu Yu, and Chun-Ying Huang. 2023. Ring-a-bell! how reliable are concept removal methods for diffusion models? *arXiv preprint arXiv:2310.10012*.

Xunguang Wang, Zhenlan Ji, Pingchuan Ma, Zongjie Li, and Shuai Wang. 2023. Instructta: Instruction-tuned targeted attack for large vision-language models.

Peter West, Ximing Lu, Nouha Dziri, Faeze Brahman, Linjie Li, Jena D Hwang, Liwei Jiang, Jillian Fisher, Abhilasha Ravichander, Khyathi Chandu, et al. 2023. The generative ai paradox:" what it can create, it may not understand". *arXiv preprint arXiv:2311.00059*.

Yijun Yang, Ruiyuan Gao, Xiaosen Wang, Nan Xu, and Qiang Xu. 2023a. Mma-diffusion: Multimodal attack on diffusion models. *arXiv preprint arXiv:2311.17516*.

Yuchen Yang, Bo Hui, Haolin Yuan, Neil Gong, and Yinzhi Cao. 2023b. Sneakyprompt: Evaluating robustness of text-to-image generative models' safety filters. *arXiv preprint arXiv:2305.12082*.

Ziyi Yin, Muchao Ye, Tianrong Zhang, Tianyu Du, Jinguo Zhu, Han Liu, Jinghui Chen, Ting Wang, and Fenglong Ma. 2023. Vlattack: Multimodal adversarial attacks on vision-language tasks via pre-trained models.

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

Jiaming Zhang, Xingjun Ma, Xin Wang, Lingyu Qiu, Jiaqi Wang, Yu-Gang Jiang, and Jitao Sang. 2023. Adversarial prompt tuning for vision-language models.

Haomin Zhuang, Yihua Zhang, and Sijia Liu. 2023. A pilot study of query-free adversarial attack against stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2384–2391.

Andy Zou, Zifan Wang, J Zico Kolter, and Matt Fredrikson. 2023. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*.

A Generating COCO-Pairs

Starting from 5000 captions, we filter out long captions and use a Named-Entity-Recognition model (Nadeau and Sekine, 2007) to identify the first noun in the sentence and use a Fill-Mask model (Devlin et al., 2018) to replace it with another noun. We use the NLTK (Loper and Bird, 2002) library and several heuristics to prevent synonyms, hyponym-hypernym, and nonvisualizable nouns from being selected. We are left with 2093 *(base caption, synthetic caption)* pairs, from which we sample 500. This yields 1000 sentence pairs in total by considering both directions.

B Algorithms

Algorithm 1 Single Token Perturbation

Require: Initial prompt $x_{1:n}$, modifiable subset I, iterations T, loss $\mathcal{L}$, score $\mathcal{S}$, batch size B, k

for $t \in T$ **do**
  for $i \in I$ **do**
    $\chi_i \leftarrow \text{Top-}k(-\nabla_{x_i}\mathcal{L}(x_{1:n}))$ {Compute top-k promising token substitutions}
  end for
  for $b = 1, \ldots, B$ **do**
    $x_{1:n}^{(b)} \leftarrow x_{1:n}$ {Initialize element of batch}
    $x_i^{(b)} \leftarrow \text{Uniform}(\chi_i)$, where $i \leftarrow \text{Uniform}(I)$ {Select random replacement token}
  end for
  $x_{1:n} \leftarrow x_{1:n}^{(b^*)}$, where $b^* = \arg\max_b \mathcal{S}(x_{1:n}^{(b)})$ {Compute best replacement}
end for

Ensure: Optimized prompt $x_{1:n}$

Algorithm 2 Multiple Token Perturbation

Require: **Input:** Initial prompt $x_{1:n}$, modifiable subset I, iterations T, loss $\mathcal{L}$, score $\mathcal{S}$, batch size B, k, ϵ_f, ϵ_s

$\epsilon \leftarrow \epsilon_s$
for $t \in T$ **do**
  for $i \in I$ **do**
    $\chi_i \leftarrow \text{Top-}k(-\nabla_{x_i}\mathcal{L}(x_{1:n}))$ {Compute top-k promising token substitutions}
  end for
  for $b = 1, \ldots, B$ **do**
    $x_{1:n}^{(b)} \leftarrow x_{1:n}$ {Initialize element of batch}
    for $i \in I$ **do**
      if $\mathcal{P}(\epsilon)$ **then**
        $x_i^{(b)} \leftarrow \text{Uniform}(\chi_i)$ {Select random replacement token}
      end if
    end for
  end for
  $x_{1:n} \leftarrow x_{1:n}^{(b^*)}$, where $b^* = \arg\max_b \mathcal{S}(x_{1:n}^{(b)})$ {Compute best replacement}
  $\epsilon \leftarrow \max(\epsilon_f, \epsilon_s - \frac{t}{T})$ {Reduce the replacement probability}
end for

Ensure: **Output:** Optimized prompt $x_{1:n}$

C Additional Examples of Asymmetric Bias

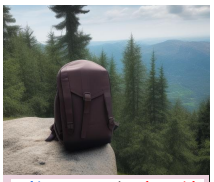
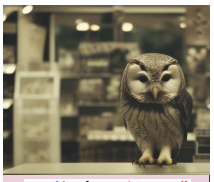
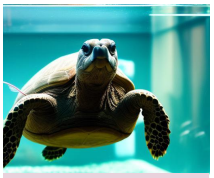
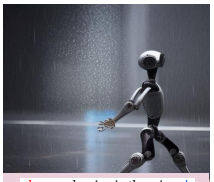
Sentence Pair (1 / 2)	Δ_2	ASR 1 → 2	ASR 2 → 1	Example
a (plane / hot air balloon) in the sky at sunset.	-0.1	**80%**	0%	 a plane in the sky at sunset. picturesque flyer '\" evening balloon
a (cabin / backpack) on a mountain.	-0.08	**90%**	20%	 a cabin on a mountain. salutparticle novo backpack sitting
an owl in a (forest / shopping mall).	-0.07	**70%**	0%	 an owl in a forest. vintage mall cashier mannetups
a (birdhouse / jack o lantern) on a tree branch.	-0.06	**60%**	0%	 an a birdhouse on a tree branch. sectional grizzly pumpkin micro costume
a (turtle / fish) swimming in an aquarium.	+0.05	0%	**90%**	 a fish swimming in an aquarium. tortoise're dly applying
a (robot / human) dancing in the rain.	+0.1	0%	**60%**	 a human dancing in the rain. 2 ': embarrassing robot thankfully
a doodle of a (light bulb / dog) on a blackboard.	+0.1	0%	**80%**	 a doodle of a dog on a blackboard. electrical bulb bulb expressing

Table 3: Additional examples of asymmetric bias in Stable Diffusion 2.1. Δ_2 shows a consistent negative correlation with ASR.

D Changing the Number of Adversarial Tokens

(a) Reducing the number of attack tokens for *"a red panda/car in a forest."*. Displaying only the adversarial attack suffixes. 2 tokens are sufficient. *"a red panda in a forest.* *chained porsche* *"* generates *"a car in a forest"*.

(b) Reducing the number of attack tokens *"a guitar/piano in a music store."*. Displaying only the adversarial attack suffixes. All 5 tokens are necessary. *"a guitar in a music store.* *serendipity upright three bank piano* *"* generates *"a piano in a music store."*

Figure 7: Reducing the number of tokens in adversarial prompts. Highly dependent on the input-target text pair.

E T2I Model Basics

Stable Diffusion (Rombach et al., 2022) is built on a denoising diffusion probabilistic model (DDPM) (Ho et al., 2020) framework, utilizing a U-Net architecture for its core operations. Key to its text-to-image capabilities is the cross-attention mechanism, which aligns textual inputs with relevant visual features. Specifically, the U-Net attends to image-aligned text embeddings produced by a CLIP (Radford et al., 2021) model. Stable Diffusion also incorporates a Variational Autoencoder (Kingma and Welling, 2013) to efficiently encode images into a latent space, significantly reducing computational requirements while maintaining image quality. Since text embedding generation using a CLIP model is the first stage of the

Stable Diffusion pipeline, it is particularly susceptible to adversarial attacks (Galindo and Faria; Zhuang et al., 2023). If an adversary can perturb the text embeddings, later stages in the Stable Diffusion pipeline will reflect the perturbed embeddings.

E.1 Exploiting CLIP's Embedding Space

The CLIP text-encoder maps the textual prompt tokens $x_{1:n}$, with $x_i \epsilon \{1, ..., V\}$ where V denotes the vocabulary size, namely, the number of tokens to $x_{1:n}$, where h_i is the hidden state corresponding to the token x_i. The U-Net component in Stable Diffusion attends to all $h_{1:n}$ embeddings using cross-attention. $x_{1:n}$ can be flattened into $\mathbf{\Phi}$, a one-dimensional vector of shape $n \times D$, where D is the embedding dimension (typically 768 for CLIP and its variants). For simplicity, we refer to $\mathbf{\Phi}$ as the text embedding of $x_{1:n}$ from here on. Let $\mathcal{H}$ represent the combined operation for encoding tokens $x_{1:n}$ and reshaping the hidden output states.

$$\mathbf{\Phi} = \mathcal{H}(x_{1:n}) = \text{Flatten}(\text{CLIP}(x_{1:n})) \tag{8}$$

Since input text and target text can vary in the number of tokens and to allow for an arbitrary number of adversarial tokens, we pad all input and targets to 77 tokens each, the maximum number of tokens supported by CLIP.

E.2 Score Function

The cosine similarity metric approximates the effectiveness of appending adversarial tokens at some intermediate optimization step t. Moving away from the input tokens' embedding and gradually towards the target tokens' embeddings through finding better adversarial tokens can be thought of as **maximizing** the following score function, similar to the metric in (Zhuang et al., 2023).

$$\begin{aligned} \mathcal{S}(x_{1:n}) = w_t \times \cos(\mathcal{H}(x^T_{1:n}), \mathcal{H}(x_{1:n})) - \\ w_s \times \cos(\mathcal{H}(x^S_{1:n}), \mathcal{H}(x_{1:n})) \end{aligned} \tag{9}$$

Here, w_t and w_s are weighing scalars and *cos* denotes the standard cosine similarity metric between two one-dimensional text embeddings. For simplicity, we set $w_t = w_s = 1$ for all experiments.

E.3 Optimization over Discrete Tokens

The main challenge in optimizing $\mathcal{S}$ is that we have to optimize over a discrete set of tokens. Furthermore, since the search space is exponential ($k^{|V|}$ for k suffix tokens), a simple greedy search is intractable. However, we can leverage gradients with respect to the one-hot tokens to find a set of promising candidates for replacement at each token position. We use the negated Score Function as the loss function $\mathcal{L}(x_{1:n}) = -\mathcal{S}(x_{1:n})$. Maximizing the score is equivalent to minimizing the loss. Since losses are used for top K token selection, the absolute value of the loss does not matter. We can compute the linearized approximation of replacing the i^th token i, x_i by evaluating the gradient

$$\nabla_{e_{x_i}} \mathcal{L}(x_{1:n}) \in \mathbb{R}^{|V|} \tag{10}$$

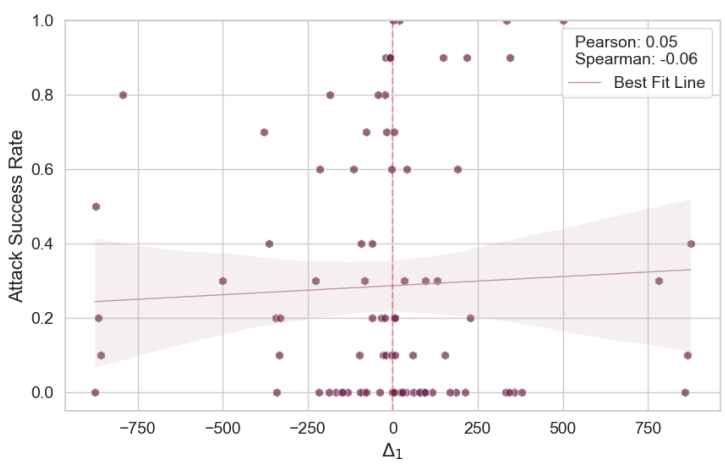
Here e_{x_i} denotes the one-hot vector that represents the current value of the i^th token. Taking gradient with respect to one-hot vectors was pioneered by HotFlip (Ebrahimi et al., 2017) and applied on Stable Diffusion by a concurrent work (Yang et al., 2023a). Based on this heuristic, we presented two algorithms for finding adversarial suffix tokens against Stable Diffusion.

F Primary Determinants of Attack Success

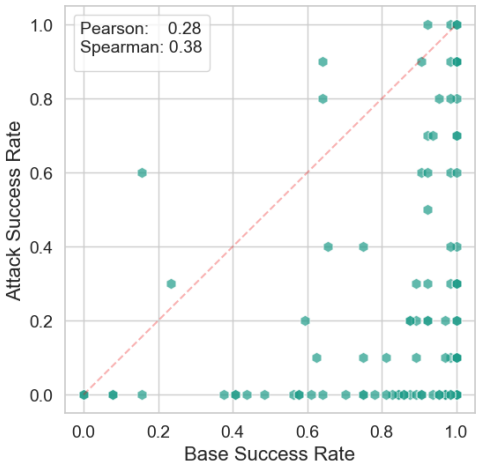
(a) *"a sofa and a bed in a room."*

(b) *"a dragon guarding a treasure."*

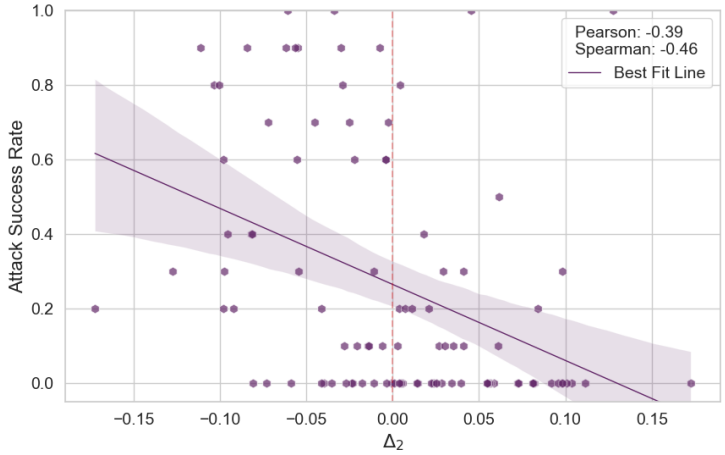
Figure 8: Examples of prompts that have low Base Success Rate (BSR) that highlight cases where Stable Diffusion fails to generate images that match the input prompt.



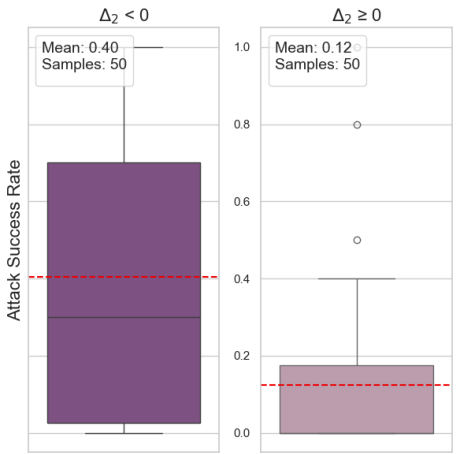
(a) ASR vs. Perplexity Difference (Δ_1 in Eqn. 6)



(b) ASR vs. BSR (of target text)



(c) ASR vs. Baseline Distance Difference (Δ_2 in Eqn. 7)



(d) ASR for Negative and Positive Δ_2

Figure 9: Correlation of ASR on Δ_1, Δ_2 and BSR. Data is reported using the Multiple Token Perturbation algorithm on HQ-Pairs. We find that the Perplexity Difference Δ_1 does not correlate with ASR. BSR shows a weak positive correlation and Baseline Distance Difference Δ_2 shows a moderate negative correlation with ASR.

G Additional Determinants of Attack Success

These sections discuss factors beyond the asymmetric properties that are related to the success rate of the attack. We have found factors like whether target token synonyms are allowed, attack suffix length and attack POS types are factors indicating the attack's success. We also found that, unlike LLM attacks, adversarial suffixes do not transfer across T2I, indicating that these models might be harder to attack than single-modality models.

G.1 Restricted Token Selection

Emulating QFAttack We can restrict certain tokens to emulate QFAttack (Zhuang et al., 2023) or prevent the exact target word from being selected. We find that QFAttack can be consistently emulated by restricting token selection to tokens corresponding to ASCII characters. We find that such adversarial suffixes can remove concepts (e.g. *"a young man"* from *"a snake and a young man."* or *"on a flower"* from *"a bee sitting on a flower."*) but fail to perform targeted attacks (e.g. changing *"a bee sitting on a flower."* to *"a bee sitting on a leaf."*). We suspect that this is mainly because ASCII tokens can perturb CLIP's embedding but are unable to add additional information to it.

Blocking Selection of Target Tokens Another potential use case is preventing the selection of the exact target word. However, we find that the algorithm simply finds a synonym or subword tokenization for the target word when the exact target word (token) is restricted. For example, when attempting to attack the input text *"a backpack on a mountain."* to *"a castle on a mountain."*, restricting the token corresponding to *"castle"* leads to the algorithm including synonyms like *"palace"*, *"chateau"*, *"fort"* or subword tokenization like *"cast le"* or *"ca st le"* in the adversarial suffix. We find that the effectiveness of the algorithm isn't affected when the exact target token is restricted and it still finds successful adversarial suffixes using synonyms (when preconditions are met).

Changing the Number of Adversarial Tokens k We set the number of adversarial tokens to $k = 5$ for all experiments. However, we observe that not all input text-target text pairs require $k = 5$. *"a red panda/car in a forest."* can be attacked with a few as $k = 2$, i.e. *"a red panda/car in a forest."* while *"a guitar/piano in a music store."* required all $k = 5$ (see Appendix D). We leave a comprehensive study on the effect of changing the number of tokens for future work.

G.2 Certain Adjectives Resist Adversarial Attacks

We observed that adversarial attacks targeting certain adjectives, such as color, had a very low ASR. For example, swapping out *"red"* with *"blue"* in the prompt *"a red car on a city road."* failed in all instances. Further challenging examples include *"a red/purple backpack on a mountain."* and *"a white/black swan on a lake."*. However, other adjectives like *"a sapling/towering tree in a forest"* or *"a roaring/sleeping lion in the Savannah."* had high ASR in at least one direction. We leave further analysis of this phenomenon for future work.

G.3 Adversarial Suffixes Do Not Transfer across T2I Models

We use SD 2.1-base which uses OpenCLIP-ViT/H (Cherti et al., 2023) internally. We find that adversarial suffixes generated using this version of SD do not work on older versions such as SD 1.4, likely because SD 1.4 uses CLIP ViT-L/14 (Radford et al., 2021). Similarly, the attacks did not transfer to DALL·E 3 (Betker et al., 2023).

H Human Evaluation WebUI

Figure 10: UI presented to human evaluators.