RIATIG: Reliable and Imperceptible Adversarial Text-to-Image Generation with Natural Prompts

Han Liu[1], Yuhao Wu[1], Shixuan Zhai[1], Bo Yuan[2], Ning Zhang[1]

[1]Washington University in St. Louis, [2]Rutgers University

{h.liu1,yuhao.wu,zhais,zhang.ning}@wustl.edu,bo.yuan@soe.rutgers.edu

Abstract

The field of text-to-image generation has made remarkable strides in creating high-fidelity and photorealistic images. As this technology gains popularity, there is a growing concern about its potential security risks. However, there has been limited exploration into the robustness of these models from an adversarial perspective. Existing research has primarily focused on untargeted settings, and lacks holistic consideration for reliability (attack success rate) and stealthiness (imperceptibility).

In this paper, we propose RIATIG, a reliable and imperceptible adversarial attack against text-to-image models via inconspicuous examples. By formulating the example crafting as an optimization process and solving it using a genetic-based method, our proposed attack can generate imperceptible prompts for text-to-image generation models in a reliable way. Evaluation of six popular text-to-image generation models demonstrates the efficiency and stealthiness of our attack in both white-box and black-box settings. To allow the community to build on top of our findings, we've made the artifacts available[1].

1. Introduction

The text-to-image generation has captured widespread attention from the research community with its creative and realistic image generation capability [52, 55, 56]. The ability to generate text-consistent images from natural language descriptions could potentially bring tremendous benefits to many areas of life, such as multimedia editing, computer-aided design, and art creation [23, 27, 37, 43].

Driven by recent advances in models trained with large datasets [42, 57] and multimodal learning [45] (e.g., diffusion models [17]), text-to-image generation has made significant progress in synthesizing high-fidelity and photorealistic images, such as DALL·E [43], DALL·E 2 [42] and Imagen [45]. At the same time, there are a growing num-

---

[1]Code is available at: https://github.com/WUSTL-CSPL/RIATIG

Figure 1. **Examples of RIATIG attacks.** The top texts are the adversarial prompts, and the bottom texts are the target models. The first row represents the target images, while the second row represents the adversarial image generated by the prompt.

ber of ethical concerns about the potential misuse of this technology [36, 45, 53]. Generative models could be used to generate synthetic video/audio/images of individuals (e.g., Deepfakes [14]), or synthetic contents with harmful stereotypes, violence, or obscenities [10, 45, 53]. To prevent the generation of such harmful content, content moderation filters are deployed in public APIs (e.g., DALL·E 2) to filter unsafe text prompts that may lead to harmful content. However, despite their best intentions, existing model-based text filters remain susceptible to adaptive adversarial attacks.

Deep neural networks (DNNs) have been shown to be vulnerable to adversarial examples [9, 31, 32]. By applying these techniques, it is possible to craft an adversarial text that looks natural to bypass the content filters, yet generates a completely different category of potentially malicious images. However, the adversarial attacks on text-to-image generators are less explored. To the best of our knowledge, there are two closely related studies [15, 36]. Nevertheless, two challenges remain:

1) Reliability. One significant limitation is that the existing works [15, 36] do not offer a reliable method to find adversarial examples. [15] discovers that DALL·E 2 has certain hidden vocabularies that can be used to generate images with some absurd (non-natural) prompts; however, this vocabulary is often limited and not stealthy (natural). Built on evocative prompting, [36] crafts adversarial examples via the morphological similarity between existing words. However, it is very difficult to find texts with such linguistic similarity, and as a result, it can be challenging for this method to be adopted and generalized in different scenarios.

2) Stealthiness. The existing approaches can only craft adversarial examples that appear to be non-natural compared to normal texts or retain similar meanings, making them easily filtered and recognized by human examiners. For example, [15] crafts *Apoploe vesrreaitais* to represent bugs, and [36] crafts *falaiscoglieklippantilado* to represent cliff. Also, [36] combines creepy and spooky into *creepooky* to generate an image that looks creepy and scary, yet the inferred meaning of this new word is highly related to the generated image, limiting its stealthiness.

To address these challenges, we propose **RIATIG**, a reliable and imperceptible adversarial attack against text-to-image models using natural examples. To achieve this, we first formulate the generation of the adversarial examples as an optimization problem and apply genetic-based optimization methods to solve it, thus making our methods much more reliable in finding working adversarial examples. Furthermore, in order to improve the stealthiness, we propose a new text mutation technique to generate adversarial text that is visually and semantically similar to its normal version (some example results are shown in Figure 1).

RIATIG is evaluated on six popular text-to-image models with both white-box and black-box attack settings. Experimental results show that compared with the state-of-the-art text-to-image-oriented adversarial attacks, RIATIG demonstrates significantly better performance in terms of attack effectiveness and sample quality. Overall, the contributions of this work are summarized as follows:

- We are the first to systematically analyze the adversarial robustness of text-to-image generation models in both the white-box and black-box settings.

- We propose genetic-based optimization methods to find natural adversarial examples reliably.

- We evaluate our attacks on six popular text-to-image generation models and compare our attacks with five baselines. The evaluation results show that our methods achieve a much higher success rate and sample quality, raising awareness of improving and securing the robustness of text-to-image models.

2. Related Work

Text-to-Image Generation. The research towards text-to-image generation starts with the work proposed by Mansimov et al. [33], which extends the Deep Recurrent Attention Writer (DRAW) [20] model to condition on image captions using a soft attention mechanism. Xu et al. [52] further propose a multi-stage attentional GAN (AttnGAN) to force the model to focus on the fine-grained information at the word level, leading to the generation of more semantically matched images. To address the limitations of the multi-stage methods, e.g., low-quality initial images and repeated use of the same word representations, Zhu et al. [58] propose a dynamic memory GAN (DM-GAN), which uses a memory network to write and read encoded prior dynamically in each image-refined process. To further improve the quality of the generator, Tao et al. [49] propose a deep Fusion Generative Adversarial Network (DF-GAN) that performs deep fusion between text and visual features. Recently, a set of large-scale text-to-image generation models are proposed and released, demonstrating promising performance. DALL·E, an autoregressive transformer model trained on 250 million web-crawled image-text pairs [43], achieves zero-shot high-quality image generation on the MS-COCO dataset. Its updated version, DALL·E 2, adopts contrastive model CLIP [42]. By feeding a text caption and using a diffusion model to generate an image conditioned on the image embedding, DALL·E 2 can synthesize amazingly photorealistic images. Imagen is another state-of-the-art text-to-image generator built on the combination of a transformer trained on text data and a high-fidelity diffusion model, achieving a very high generation performance.

Adversarial Attacks on Text-to-Image Generators. The adversarial robustness of text-to-image generation is a relatively less studied topic. [15] is the pioneering work that studies the vulnerability of text-to-image generators, such as DALL·E 2. They found it is possible to leverage gibberish text in the target image to evade content filters by using such gibberish texts as prompts rather than semantically equivalent text. However, most of the gibberish text does not reliably produce a specific category of images. Later, Millière proposes two methods of crafting adversarial text to generate specific pictures [36]. The first solution is macaronic prompting, which composes multilingual subword segments to generate specific types of images. The second approach is evocative prompting, a strategy that uses prompts possessing similarity of broad morphological features to generate semantic-consistent images with real concepts. In addition, Struppek shows that replacing a single character in the text with visually similar non-Latin characters can induce cultural biases into the generated images or even hide complete objects in the generated images. However, their method is untargeted, i.e., they cannot generate
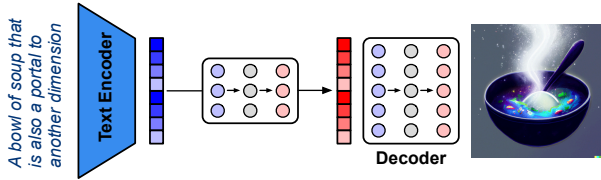
Figure 2. Working mechanism of a text-to-image generator.

specific attacker selected images.

3. System and Threat Model

Text-to-Image Generation Model. Figure 2 represents the working mechanism of text-to-image generation models considered in this work, consisting of three parts: a text encoder, a transformation network, and a decoder. In the text encoder, the text prompt is first converted to tokens by a tokenizer. Each token is mapped to a text encoding by looking up the mapping dictionary of the embedding models. The transformation model maps the text encoding to a corresponding image encoding that captures the semantic information contained in the text encoding. Finally, the decoder generates an image based on image encoding that is a visual representation of the text prompt. As shown in Figure 2, the image generated by DALL·E 2 with prompt *A bowl of soup that is also a portal to another dimension* is photorealistic and has aligned semantic meaning.

Threat Model. We consider both white-box and black-box settings to evaluate the different capabilities of the adversary. In the white-box setting, the adversary has full knowledge of the target model such as model architecture as well as its parameters. In the black-box setting, the adversary does not know the internal information of the model or training data. It is only capable of querying the model with supplied input and obtaining the output. Different from classification models [8, 21, 24, 28], the generation models cannot return confidence scores to the adversary.

4. Methodology

4.1. Problem Formulation

A text-to-image generation model $\mathcal{G} : \mathcal{X} \rightarrow \mathcal{Y}$ maps text space $\mathcal{X}$ to image space $\mathcal{Y}$. Since $\mathcal{G}$ is trained on semantically-aligned text-image pairs, a well-trained model should generate images that are semantically identical to the input text. Given an arbitrary target image y_t, which corresponds to its naive text prompt x_t, the adversary aims to craft an adversarial example x^* that generates a semantically similar image to y_t, with the premise that x^* is semantically dissimilar from x_t enough to evade detection. This is an optimization program that can be formulated as

$$\underset{x^*}{\operatorname{argmax}}\, S_i(\mathcal{G}(x^*), y_t) \text{ s.t. } D_t(x^*, x_t) > \eta, \qquad (1)$$

where S_i is a semantic similarity function of images, D_t is a function measuring the semantic distance of texts, and η is the threshold of the semantic dissimilarity of x_t and x^*.

4.2. Challenges and Overview

Challenges. Designing a reliable and imperceptible adversarial attack against text-to-image models can face several challenges. Firstly, unlike existing work on textual adversarial attacks [8, 24, 28], where the output of the model is a finite number of classes (e.g., positive and negative in sentiment analysis), the output of a text-to-image model is a high-dimensional continuous class with nearly infinite combinations. This characteristic makes it difficult to measure the effectiveness of the attacks. One possible solution is to measure the semantic distance of images, however, existing approaches only focus on category-level semantic distance [11, 16], which cannot handle the complex sentence context similarity. For example, simply representing *a blue double decker school bus parking in let* by school bus will lose important context information such as the color, size of the bus, and its surrounding environment. Second, since words can serve as discrete tokens in a sentence, it is impossible to optimize the input words directly by computing the gradient of the loss function with respect to the input word even assuming white-box knowledge in the adversary [8]. Third, unlike image domains where small perturbations can go unnoticed, any changes in the text domain are perceptible [8, 28, 29] and easily detectable by human examiners.

Attack Overview. We adopt a genetic-based approach to iteratively optimize the input samples towards the target samples, followed by a quality fine-tuning stage to further improve the imperceptibility. First, the mutation stage generates a large number of variants that are semantically and visually similar to the input text, then the text-to-image generation model outputs images accordingly. A similarity measurement is implemented to compute the quality of mutated examples, allowing us to select the best offspring, which will enter a crossover stage to exchange the desirable traits to obtain a better population for the next-round iteration. The loop ends when the generated images are highly similar to the target images yet the adversarial text is distinct from the target text. Then the result samples go through the quality fine-tuning stage where we propose two techniques to further improve the imperceptibility of adversarial examples. In the following section, we will first introduce the similarity measurement for the offspring selection (§4.3), then elaborate on the genetic-based optimization with mutation and crossover scheme (§4.4), before finally describing the sample quality improvement procedure (§4.5).

4.3. Similarity Measurement

To find a suitable measurement metric, we need to consider the natural continuity of visual space and find a proper
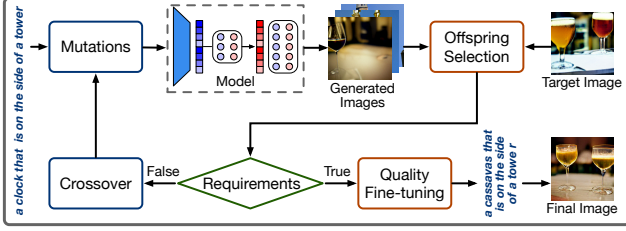
Figure 3. Overview of the RIATIG attack.

Algorithm 1: Adversarial Example Generation

1 Input: Original text x, target text x_t with associated target image y_t, mutation probability p_m, population size M, N, K, maximum iterations c_{max}, control parameter α, β.

2 Output: Adversarial examples sets T^*.

3 $G_0 \leftarrow Mutate(x, p_m, M)$ ▷ initialize population

4 for $c = 0$ *to* c_{max} **do**

5 $Z_c \leftarrow S_i(\mathcal{G}(G_c), y_t)$ ▷ update score by Eq.2

6 $I_c \leftarrow argsort(Z_c)[:: -1]$

7 $E_c \leftarrow G_c[I_c][: K]; F_c \leftarrow Z_c[I_c][: K]$

8 $s_c \leftarrow F_c[0]$ ▷ For mutation prob update

9 $p_m \leftarrow \alpha \cdot p_{m-1} + \frac{\beta}{|s_c - s_{c-1}|}$

10 $P_c \leftarrow Softmax(F_c)$ ▷ normalize to prob

11   **for** $i = 1$ *to* N **do**

12 $parent_1, parent_2 \leftarrow Sample(E_c, P_c)$ ▷ sample two parents from E_c with prob P_c

13 $child_i \leftarrow Crossover(parent_1, parent_2)$

14 $g_i \leftarrow Mutate(child_i, p_m, 1)$

15 $G_{c+1} \leftarrow G_{c+1} \cup \{g_i\}$ ▷ append to new generation

16   **if** $S_i(\mathcal{G}(g_i), y_t) > \epsilon$ *and* $D_t(g_i, x_t) > \eta$ **then**

17 $T \leftarrow T \cup \{g_i\}$ ▷ append to sample sets

18   **end**

19   **end**

20 end

21 $T^* \leftarrow FineTune(T)$ ▷ Fine tune samples by strategy in Section 4.5

22 return T^*

---

representation of image semantic information. Recently, there is a growing research interest in learning image representation from text [40, 46, 57], which maps images into a semantic-rich embedding space by jointly learning with textual data. Among these efforts, the CLIP proposed by OpenAI has been trained on large-scale (i.e., 400 million) web-crawled image-text pairs [40], which can efficiently represent visual concepts aligned with the corresponding textual information. Motivated by its powerful capability, we adopt the pre-trained image encoder of CLIP to encode two images and then calculate the cosine distance between the encoded vectors measuring the similarity as

$$S_i(\mathcal{G}(x), y) = \frac{E_i(\mathcal{G}(x)) \cdot E_i(y)}{\|E_i(\mathcal{G}(x))\| \|E_i(y)\|}. \quad (2)$$

Similarly, we utilize the text encoder to encode the text and calculate the cosine distance as their semantic distance,

$$D_t(x, x^*) = 1 - \frac{E_t(x) \cdot E_t(x^*)}{\|E_t(x)\| \|E_t(x^*)\|}. \quad (3)$$

When the semantic distance is above a certain threshold, the two texts are considered not semantically similar.

4.4. Genetic-based Optimization

Genetic algorithms are heuristic optimization methods that are inspired by nature selection. By evolving a population of candidates towards better solutions with performing mutation and crossover [22], the genetic algorithms can search a large solution space effectively [25], making them well suited in the optimization over the multi-dimensional input space. Therefore, in this work, we adopt a genetic-based algorithm to solve the target optimization problem. To be specific, the optimization process starts with an input text that is semantically dissimilar from the target images. The mutation is used to generate different variants to explore the search space and increase the diversity of the population. To generate initial high-quality variants, RIATIG first generates a large pool of mutated candidates, rendering the execution of crossover and mutation more effective. In each generation, we select the samples with the highest fitness score for crossover and mutation to generate the next

generation of text samples. Cosine similarity scores calculated by Eq. 2 between the two images are the fitness score of the genetic algorithm.

Crossover Scheme. Crossover is used to exchange the desirable traits between the chosen populations, thereby possibly generating an even better population. For the crossover scheme used in RIATIG, a new sentence is synthesized by a pair of parent sentences. The probability of each population being selected as a parent is proportional to their fitness score, giving the beneficial traits a higher probability to stay.

Mutation Scheme. RIATIG performs sentence-level mutation. Each sentence has a mutation probability of p_m to be selected. The proper selection of p_m is very critical. When it is too high, the beneficial traits may be mutated without crossover to generate better offspring. On the other hand, very small p_m will make most of the populations unchanged, causing insufficient mutations to escape the local maximum. To address the challenge, p_m can be adaptively updated using a momentum mutation-based method [50] by

$$p_m = \alpha \cdot p_{m-1} + \frac{\beta}{|S_i - S_{i-1}|}, \quad (4)$$

where α and β are scaling factors, and S_i denotes the maximum fitness score across the populations in i-th generation.

The mutation process occurs for each selected sentence, which is divided into two stages. In the first stage, we identify the important words in the original sentence that need to

be mutated. In the second stage, we propose two strategies to slightly modify a word while preserving its semantics.

Stage-1: Selection of Important Word. Because 1) the search space for mutation is very huge; and 2) most of the mutation variants do not play significant roles to improve the fitness score, identifying the most important words that contribute to fitness scores is crucial to improve the effectiveness of mutation. When we assume white-box knowledge, gradient information can be used to measure the importance, calculating the gradient of the fitness score with respect to the embedding of each word by

$$g_{x_n} = \frac{\partial S_i(\mathcal{G}(e_n), y_t)}{\partial e_n}, \tag{5}$$

where e_n is the embedding of word x_n, y_t is the target image, $\mathcal{G}$ is the generation model, and S_i is the semantic similarity function defined in Eq. 2. When we assume black-box knowledge, the importance of a word is estimated by calculating the change in fitness scores before and after deleting the word, such a calculation process is defined as

$$g_{x_n} = S_i(\mathcal{G}(e), y_t) - S_i(\mathcal{G}(e_{\setminus e_n}), y_t), \tag{6}$$

where e is the embedding vectors of sentence x, $e_{\setminus e_n}$ is the embedding vectors after deleting the word x_n from x. After that, we normalize the important scores as

$$I_{x_n} = \frac{e^{\|g_{x_n}\|}}{\sum_{i=1}^{N} e^{\|g_{x_i}\|}}. \tag{7}$$

Then each word x_n has a probability of I_{x_n} to be selected.

Stage-2: Word Mutation. Since our goal is to generate adversarial text to be visually and semantically similar to normal text for human understanding, we consider two types of mutation strategies that are used in natural language processing fields [28,29,51]. The first one is to inject character-level typos that mimic the grammar mistakes people may make. The rationale is that the semantic meaning of the text is very likely to be preserved after a few character changes [44]. More specifically, we adopt three types of text manipulations: (1) Insert extra spaces into words; (2) Swap the position of two characters randomly except for the first and last characters; (3) Delete one random character except for the first and last characters. The second type of text manipulation is to replace characters/words with visually or semantically similar counterparts. As for visually similar substitution, we utilize the LEET Speak Alphabet [7], which replaces characters in ways that play on the similarity of glyphs. For example, we may replace the letter *l* with the number *1*, the letter *o* with the number *0*, etc.

For the semantically similar substitution, we adopt the Word2vec model [34] to transform the words into embedding vectors, and then compute the cosine distance as the similarity measurement to select the nearest neighbors. A threshold is used to filter out candidates with large distances. Due to the large search space of optimization, only replacing it with target-irrelevant sentences will weaken the mutation, making it hard to find variants that generate images closer to the target. To solve this problem, we use the target sentences to perform semantic substitution, and the words with a close distance are filtered out. In addition, we adopt the WordNet [35] to decide the part-of-speech (POS) tag of the word, filtering out the candidates with different POS to maintain grammatical fluency. After filtering, we select the top-M nearest neighbors into the candidate pool. Next, Google's one billion-word language model [13] is used to filter out words that do not fit within the context surroundings. To that end, we first fit words in the candidate pools with the surrounding context, then calculate their language scores that measure the fluency of the sentences, keeping the top-N word with the highest scores.

4.5. Sample Quality Improvement

After obtaining the adversarial examples, we further improve the quality of adversarial text based on two key observations. First, the presuppositions (e.g., on, in, etc.) in the sentences have little impact on the semantic meanings of generated images compared to the nouns and verbs. Second, the text embedding in the text-to-image generation model usually uses a dictionary to represent a finite set of possible words [52, 58], meaning that the deliberately misspell out-of-vocabulary will be mapped to *unknown*. With the first observation, we iteratively replace or add all predefined presuppositions in different positions of sentences and test their naturality. The presuppositions with the highest naturality are chosen to improve the samples. We then further fine-tune the quality of samples based on the second observation. For example, both *paking* and *parnikg* are out-of-vocabulary words that are mapped to the *unknown*, and they are regarded as the same words by the text-to-image generation models. However, since the first word has fewer modifications, it is closer to an ordinary text and is preferred. Therefore, with white-box knowledge, we can test whether the adversarial text is the out-of-vocabulary words by feeding it to the embedding models. In the black-box attack setting, we adopt the GloVe embedding models [38] pre-trained on the Common Crawl dataset containing 2.2 million vocabularies as a surrogate embedding model.

5. Experiments

5.1. Experiment Settings

Dataset. We choose the initial texts, target texts, and target images from the Microsoft COCO [30] dataset, a popular benchmark for training and evaluating text-to-image generation models. The dataset includes 82,783 training images and 40,504 testing images, each with 5 text descriptions.

Target Models. To evaluate the performance of our proposed attack, we choose three representative public text-to-image generation models: AttnGAN [52], DM-GAN [58], and DF-GAN [49]. We use the pre-trained models on the COCO dataset released in their official GitHub repository [1, 4, 5] as the target models. We also choose three large-scale proprietary models: DALL·E [43], DALL·E 2 [42], and Imagen [45]. None of the above have officially released their pre-trained models, nor public APIs except DALL·E 2. Therefore, for DALL·E, we adopt the released pre-trained models of DALL·E mini [3] , which achieves comparable performance to that of DALL·E. Similarly, for Imagen, we adopt the pre-trained models released by Hugging Face [6]. For DALL·E 2, though a public API exists[2], only 15 free credits are granted to each account, so we first train adversarial examples on LAION's pre-trained model [2], then access the APIs to check if it was transferred successfully.

Evaluation Setup. In our experiment, we attacked the AttnGAN, DM-GAN, and DF-GAN in white-box settings, and we attacked DALL·E, DALL·E 2, Imagen in black-box settings, which means we can only query the model and get the generated images. We set the initial population size M, the total population size N, and the selected population size K to 300, 50, and 25, respectively. The initial mutation probability p_m is set to 0.8. Ten adversarial examples are generated for each model. For each generation, we randomly select a target image and a semantic unrelated text from the COCO dataset. For example, we select the target images with the meaning *a clock is flying in the blue sky* and the original text *a red double decker bus parking in let*.

Comparison with Baselines. To the best of our knowledge, there are only two related studies [15, 36] for a targeted adversarial attack against text-to-image generation models. [15] proposes a hidden vocabulary attack (HiddVocab) on DALL·E 2 and [36] implements macaronic prompting (MacPromp) and evocative prompting (EvoPromp) attacks against DALL·E mini and DALL·E 2. For a comprehensive comparison, we also include [48], which is a homolytic substitution-based attack (HomoSubs) to cause image content obfuscation. We also include one state-of-the-art textual adversarial attack, TextFooler [24]. Since their attack does not involve image generation, we change the posterior tasks to match our settings and apply our proposed similarity metric to guide the optimization.

5.2. Evaluation Metrics

To demonstrate the effectiveness of our attack, we evaluate both attack effectiveness and adversarial sample imperceptibility. The imperceptibility measure consists of two parts: semantic meaning similarity and sentence naturality.

Effectiveness of Generated Images. We adopt the method of R-precision, a widely-used metric in text-to-image generation models [27, 39, 52, 54, 58] to measure whether the generated image is semantically aligned with the given text description. Specifically, we use the generated images to perform a retrieval experiment by querying a candidate set of text descriptions that consist of one ground truth and 99 randomly selected mismatched descriptions. The ViLT model [26], which achieves state-of-the-art performance on image-text retrieval tasks, is used here. We examine the top R ranked retrieval results and consider our attacks successful when the ground truth appears in the retrieval. We evaluate the effectiveness when R is 1 (R-1) and R is 3 (R-3).

Sentence Semantic Meaning. Following [24, 28], we first adopt the Universal Sentence Encoder [12] to encode the adversarial sentence and target sentence into high dimensional vectors, and then their cosine similarity score are calculated to measure the semantic similarity.

Sentence Naturality. The adversarial examples should look natural, similar to normal sentences in the real world. In our experiment, we calculate the perplexity (PPL) of the adversarial examples with GPT-2 [41], which is trained on a large corpus of real-world sentences. PPL measures how likely the model is to generate the input text sequence, representing the fluency of the adversarial examples. Generally, the sample with lower PPL is more natural.

5.3. Evaluation Results

White-box Attacks. The results for white-box attacks are shown in Table 1. It is seen that RIATIG succeeds in generating all adversarial examples in DM-GAN and DF-GAN, and can succeed in generating 9 out of 10 samples in AttnGAN when we examine top-3 recall results. The recall precision drops when we only examine top-1 recall results. This phenomenon might be caused by the image generation quality of the target model, since the DM-GAN and AttnGAN achieve an average R-precision of 85.47% [52] and 88.56% [58] on COCO datasets. Furthermore, a low semantic distance means there is a low semantic relevance between the adversarial text and the text used to generate the target images. The low PPL means our adversarial text is fluent and natural as compared to the normal text. On the other hand, due to the limited-size and monolingual datasets that these three models are trained on, the baselines can not succeed in generating adversarial examples.

Black-box Attacks. The results for black-box attacks are shown in Table 2. RIATIG achieves a high R-precision as compared to the baseline methods. HomoSubs and TextFooler are designed for untargeted attacks, and cannot be easily adapted for targeted attacks. HiddVocab does not provide a reliable method for finding adversarial examples and can achieve only a 40% success rate against DALL·E 2. Though MacPromp and EvoPromp can achieve a higher

| a hoesse is standing on gra ss | Zeedonk is on the side of a hill | a tran is passing through a villa ge | a man kite braoding in the Napatree Point next to Dane Dacus | the viartters are on the center of oeacn | a colck is flying in the blue phoebe | a chinkapin is on the fiel d | a blue dolube decker bus brightly colored kites in florescent orange |

Figure 4. Examples of our adversarial attack results against DALL·E mini, DALL·E 2, and Imagen.

Table 1. The results of the white-box attacks against AttnGAN, DM-GAN, and DF-GAN.

Model	R-1 ↑	R-3 ↑	Semantic Distance ↓	PPL ↓
AttnGAN [52]	8/10	9/10	0.24	522.63
DM-GAN [58]	8/10	10/10	0.27	420.16
DF-GAN [49]	9/10	10/10	0.34	558.81

Note: R-1 and R-3 represent the R-precision when R=1 and R=3, respectively. PPL represents the perplexity of sentences.

precision rate, due to their loosely formulated linguistic methods, MacPromp and EvoPromp do not always guarantee to find an adversarial example. In addition, MacPromp adopts the compositional prompts by combining two semantically similar words resulting in a higher semantic distance. For example, it combines creepy and spooky into *creepooky* and uses the sentence *A very creepooky person* to generate a person that looks creepy and spooky. As a result, the semantic meaning between the adversarial examples and the target is still very close, making them highly perceptible. Also, these generated words can break the fluency and naturalness of the sentences, resulting in a high PPL. For EvoPromp, since it relies on broad morphological similarity, the semantic difference is quite low. However, the crafted sentences can seem meaningless, again breaking the naturality of the sentences, e.g., using prompt *rygamera pultris* to generate the images of fishes.

5.4. Ablation Study

Effectiveness of Mutation Strategy. To evaluate the effectiveness of our proposed mutation strategy, we conducted a comparative analysis with a random mutation approach, which employed insert, swap, and delete operations. Table 3 presents the results of our experiments, indicating that our mutations significantly outperformed random mutations, particularly in black-box settings.

Influence of Target Images. To further analyze the im-

Table 2. The results of the black-box attacks against DALL·E mini, DALL·E 2, and Imagen.

Model	Methods	R-1 ↑	R-3 ↑	Sem. Dist. ↓	PPL ↓
DALL·E mini [43]	HiddVocab [15]	3/10	3/10	0.17	5284.78
	MacPromp [36]	8/10	8/10	0.46	6814.61
	EvoPromp [36]	6/10	6/10	0.14	5662.97
	HomoSubs [48]	0/10	0/10	0.09	1115.84
	TextFooler [24]	0/10	0/10	0.18	1848.41
	RIATIG	**9/10**	**10/10**	0.27	**420.16**
DALL·E 2 [42]	HiddVocab [15]	4/10	4/10	0.16	5162.71
	MacPromp [36]	9/10	9/10	0.39	4931.31
	EvoPromp [36]	7/10	7/10	0.17	6139.39
	HomoSubs [48]	0/10	0/10	0.06	981.69
	TextFooler [24]	0/10	0/10	0.18	2624.88
	RIATIG	**10/10**	**10/10**	0.27	872.03
Imagen [45]	HiddVocab [15]	2/10	2/10	0.18	5073.74
	MacPromp [36]	6/10	6/10	0.42	5694.73
	EvoPromp [36]	4/10	4/10	0.16	5607.66
	HomoSubs [48]	0/10	0/10	0.08	1103.53
	TextFooler [24]	0/10	0/10	0.16	2231.10
	RIATIG	**10/10**	**10/10**	0.23	**704.09**

Note: Sem. Dist. represents semantic distance.

Table 3. Attack performance using random mutation strategy.

Model	R-1 ↑	R-3 ↑	Sem. Dist. ↓	PPL ↓
DM-GAN	7/10	8/10	0.14	867.93
DF-GAN	1/10	2/10	0.12	882.80
DALL·E mini	0/10	0/10	0.17	2111.82

Table 4. The mean and variance of evaluation metrics under different selections of target images.

Model	R-1 ↑	R-3 ↑	Sem. Dist. ↓	PPL ↓
DF-GAN	9.2(±0.42)/10	9.8(±0.42)/10	0.31±0.04	596.33±297.02
DALL·E mini	9.7(±0.48)/10	9.9(±0.32)/10	0.28±0.03	475.83±180.53

pact of target image selection on attack performance, we conducted ten experiments of the sample training process, with each experiment utilizing a different target image that

Table 5. The attack performance by using different substitution methods against DF-GAN and DALL·E mini.

Model	Sub. Meth.	R-1 ↑	R-3 ↑	Sem. Dist. ↓	PPL ↓
DF-GAN [49]	Word2vec [34]	**9/10**	**10/10**	0.336	**558.81**
	Glove [34]	8/10	9/10	**0.331**	836.96
	WordNet [35]	5/10	7/10	0.358	580.49
DALLE mini [43]	Word2vec [34]	**9/10**	**10/10**	**0.268**	**420.16**
	Glove [34]	7/10	9/10	0.311	797.63
	WordNet [35]	4/10	6/10	0.368	522.34

Note: Sub. Meth. represents substitution method.

Table 6. The attack performance of 50 additional experiments.

Model	R-1↑	R-3↑	Sem. Dist.↓	PPL↓
DF-GAN	42/50	46/50	0.22	838.28
DALL·E mini	45/50	47/50	0.34	640.78

shared the same semantic meaning. The results, as presented in Table 4, indicate that our attacks demonstrated high robustness across varying target images.

Robustness of Example Generation. One important property of RIATIG is that the adversarial text can be reliably associated with specific visual concepts when used as a prompt. We evaluated the sample robustness of DALL·E mini, DALL·E 2, and Imagen. Specifically, we feed the same adversarial text to the target model ten times and calculated the average R-precision. The results show that RIATIG can achieve a 100% R-3.

Different Substitution Methods. In our mutation strategy, one of the most important parts is the semantic word substitution, which is essential for optimizing the fitness scores and improving the sample quality. To identify a suitable approach for substitution, we evaluate three different approaches, namely Glove models [38], WordNet models [35] and Word2vec models [34]. We evaluate two target generation models (DF-GAN [49] and DALL·E mini [43]) in both white-box and black-box settings. For all three approaches, we select the substitute word from the top 30 results. For each target model and substitution approach, we start with ten identical benign text and target images and generate ten adversarial samples accordingly to evaluate the performance. The results are shown in Table 5. It is seen that Word2vec performs better than Glove and WordNet with respect to R-precision and PPL for the attack against DF-GAN, and Word2vec performs better than the others with respect to all three performance metrics when attacking DALL·E mini. We hypothesize that it is probably caused by the larger semantic vocabulary of Word2vec.

Attack Performance with More Evaluations. To comprehensively evaluate the scalability of our attack, we trained 50 additional samples, each with distinct sources and targets. The results of this larger-scale evaluation are in Table 6, showing consistent performance of our attack strategy.

6. Discussion

Security Impacts. To prevent the malicious use of text-to-image generation models such as spreading misinformation (e.g., deepfake) [18], generating biased and inappropriate contents (e.g., stereotyping, pornography, violence) [10, 47], content moderation filters are deployed to filter harmful inputs. However, as we demonstrated, it is possible to craft a completely unrelated sentence to generate target images with black-box knowledge. Thus, it is important to consider the security risks of latest AI models.

Possible Defense. There are several possible defenses. First, a rule-based text filter could be deployed to remove redundant spaces and wrong words before generating images. To evaluate this defense, we tested the adversarial text using Grammarly and found that 20% of the samples could bypass the text filtering. For example, the sentence *"a fruit stand display with bananas and starlit skies"* was able to evade detection because this rule-based tool does not inspect the semantic meaning of the sentence. Second, one may consider using an image filter to prevent harmful images from being displayed to users. However, training such an image filter may require non-trivial efforts in collecting a large number of datasets. Moreover, the scope of harmful images can sometimes be difficult to define, such as with deep fake images. Another possible defense mechanism lies in the enhancement of the text-to-image generation models by adversarial training, which is one of the most effective defensive approaches against adversarial examples in the image and text domains [19, 24]. The model can be trained using both the original and adversarial text sharing the same associated images as the original text. However, it requires non-trivial efforts when facing a huge volume of adversarial examples as the modern text-to-image generation models are based on large-scale training datasets (e.g., Imagen [45] trains on about 460M image-text pairs).

7. Conclusion

In this paper, we propose an adversarial attack against the text-to-image generation models. We propose a two-stage attack framework to craft an effective adversarial example, where we find a workable adversarial example in the first stage, followed by a sample quality fine-tuning in the second stage. Through extensive evaluation, we can generate highly natural and semantically unrelated examples, allowing for successful attacks. We hope this work will raise awareness of potential security risks and aid the development of more effective defenses.

References

[1] Attngan implementation. `https://github.com/taoxugit/AttnGAN`. Accessed: 2022-10-01.

[2] Dall·e 2 laion implementation. `https://github.com/LAION-AI/dalle2-laion`. Accessed: 2022-10-01.

[3] Dall·e mini implementation. `https://github.com/borisdayma/dalle-mini`. Accessed: 2022-10-01.

[4] Df-gan implementation. `https://github.com/tobran/DF-GAN`. Accessed: 2022-10-01.

[5] Dm-gan implementation. `https://github.com/MinfengZhu/DM-GAN`. Accessed: 2022-10-01.

[6] Imagen implementation. `https://github.com/cene555/Imagen-pytorch`. Accessed: 2022-10-01.

[7] Leet speak cheat sheet. `https://www.gamehouse.com/blog/leet-speak-cheat-sheet/`. Accessed: 2022-09-30.

[8] Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. Generating natural language adversarial examples. *arXiv preprint arXiv:1804.07998*, 2018.

[9] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 387–402. Springer, 2013.

[10] Abeba Birhane, Vinay Uday Prabhu, and Emmanuel Kahembwe. Multimodal datasets: misogyny, pornography, and malignant stereotypes. *arXiv preprint arXiv:2110.01963*, 2021.

[11] Clemens-Alexander Brust and Joachim Denzler. Not just a matter of semantics: the relationship between visual similarity and semantic similarity. *arXiv preprint arXiv:1811.07120*, 2018.

[12] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*, 2018.

[13] Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*, 2013.

[14] Bobby Chesney and Danielle Citron. Deep fakes: A looming challenge for privacy, democracy, and national security. *Calif. L. Rev.*, 107:1753, 2019.

[15] Giannis Daras and Alexandros G Dimakis. Discovering the hidden vocabulary of dalle-2. *arXiv preprint arXiv:2206.00169*, 2022.

[16] Thomas Deselaers and Vittorio Ferrari. Visual and semantic similarity in imagenet. In *CVPR 2011*, pages 1777–1784. IEEE, 2011.

[17] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[18] Mary Anne Franks and Ari Ezra Waldman. Sex, lies, and videotape: Deep fakes and free speech delusions. *Md. L. Rev.*, 78:892, 2018.

[19] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[20] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Rezende, and Daan Wierstra. Draw: A recurrent neural network for image generation. In *International conference on machine learning*, pages 1462–1471. PMLR, 2015.

[21] Chuan Guo, Jacob Gardner, Yurong You, Andrew Gordon Wilson, and Kilian Weinberger. Simple black-box adversarial attacks. In *International Conference on Machine Learning*, pages 2484–2493. PMLR, 2019.

[22] John H Holland. Genetic algorithms. *Scientific american*, 267(1):66–73, 1992.

[23] Rowan T Hughes, Liming Zhu, and Tomasz Bednarz. Generative adversarial networks–enabled human–artificial intelligence collaborative applications for creative and design industries: A systematic review of current approaches and trends. *Frontiers in artificial intelligence*, 4:604234, 2021.

[24] Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8018–8025, 2020.

[25] Sourabh Katoch, Sumit Singh Chauhan, and Vijay Kumar. A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*, 80(5):8091–8126, 2021.

[26] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, pages 5583–5594. PMLR, 2021.

[27] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip Torr. Controllable text-to-image generation. *Advances in Neural Information Processing Systems*, 32, 2019.

[28] Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. Textbugger: Generating adversarial text against real-world applications. *arXiv preprint arXiv:1812.05271*, 2018.

[29] Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. Deep text classification can be fooled. *arXiv preprint arXiv:1704.08006*, 2017.

[30] 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 *European conference on computer vision*, pages 740–755. Springer, 2014.

[31] Han Liu, Zhiyuan Yu, Mingming Zha, XiaoFeng Wang, William Yeoh, Yevgeniy Vorobeychik, and Ning Zhang. When evil calls: Targeted adversarial voice over ip network. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 2009–2023, 2022.

[32] Giulio Lovisotto, Henry Turner, Ivo Sluganovic, Martin Strohmeier, and Ivan Martinovic. {SLAP}: Improving physical adversarial examples with {Short-Lived} adversarial perturbations. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 1865–1882, 2021.

[33] Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. Generating images from captions with attention. *arXiv preprint arXiv:1511.02793*, 2015.

[34] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[35] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[36] Raphaël Millière. Adversarial attacks on image generation with made-up words. *arXiv preprint arXiv:2208.04135*, 2022.

[37] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

[38] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[39] Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. Mirrorgan: Learning text-to-image generation by redescription. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1505–1514, 2019.

[40] 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 *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[41] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[42] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[43] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.

[44] Graham Rawlinson. The significance of letter position in word recognition. *IEEE Aerospace and Electronic Systems Magazine*, 22(1):26–27, 2007.

[45] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

[46] Mert Bulent Sariyildiz, Julien Perez, and Diane Larlus. Learning visual representations with caption annotations. In *European Conference on Computer Vision*, pages 153–170. Springer, 2020.

[47] Ramya Srinivasan and Kanji Uchino. Biases in generative art: A causal look from the lens of art history. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 41–51, 2021.

[48] Lukas Struppek, Dominik Hintersdorf, and Kristian Kersting. The biased artist: Exploiting cultural biases via homoglyphs in text-guided image generation models. *arXiv preprint arXiv:2209.08891*, 2022.

[49] Ming Tao, Hao Tang, Fei Wu, Xiao-Yuan Jing, Bing-Kun Bao, and Changsheng Xu. Df-gan: A simple and effective baseline for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16515–16525, 2022.

[50] Rohan Taori, Amog Kamsetty, Brenton Chu, and Nikita Vemuri. Targeted adversarial examples for black box audio systems. In *2019 IEEE security and privacy workshops (SPW)*, pages 15–20. IEEE, 2019.

[51] Umut Topkara, Mercan Topkara, and Mikhail J Atallah. The hiding virtues of ambiguity: quantifiably resilient watermarking of natural language text through synonym substitutions. In *Proceedings of the 8th workshop on Multimedia and security*, pages 164–174, 2006.

[52] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1316–1324, 2018.

[53] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022.

[54] Han Zhang, Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. Cross-modal contrastive learning for text-to-image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 833–842, 2021.

[55] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 5907–5915, 2017.

[56] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1947–1962, 2018.

[57] Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D Manning, and Curtis P Langlotz. Contrastive learning of medical visual representations from paired images and text. *arXiv preprint arXiv:2010.00747*, 2020.

[58] Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5802–5810, 2019.