ProTIP: Probabilistic Robustness Verification on Text-to-Image Diffusion Models against Stochastic Perturbation

Yi Zhang[1], Yun Tang[1], Wenjie Ruan[2], Xiaowei Huang[2], Siddartha Khastgir[1], Paul Jennings[1], and Xingyu Zhao[1](✉)

[1] WMG, University of Warwick, Coventry CV4 7AL, U.K.
{yi.zhang.16,yun.tang,s.khastgir.1,paul.jennings,xingyu.zhao}@warwick.ac.uk
[2] Computer Science Department, University of Liverpool, Liverpool L69 3BX, U.K.
w.ruan@trustai.uk; xiaowei.huang@liverpool.ac.uk

Abstract. Text-to-Image (T2I) Diffusion Models (DMs) excel at creating high-quality images from text descriptions but, like many deep learning models, suffer from robustness issues. While there are attempts to evaluate the robustness of T2I DMs as a *binary* or *worst-case* problem, they cannot answer how robust *in general* the model is whenever an adversarial example (AE) can be found. In this study, we first formalise a *probabilistic* notion of T2I DMs' robustness; and then devise an *efficient* framework, ProTIP, to evaluate it with *statistical guarantees*. The main challenges stem from: *i)* the high computational cost of the image generation process; and *ii)* identifying if a perturbed input is an AE involves comparing two output *distributions*, which is fundamentally harder compared to other DL tasks like classification where an AE is identified upon misprediction of labels. To tackle the challenges, we employ *sequential analysis with efficacy and futility early stopping rules* in the statistical testing for identifying AEs, and *adaptive concentration inequalities* to dynamically determine the "just-right" number of stochastic perturbations whenever the verification target is met. Empirical experiments validate ProTIP's effectiveness and efficiency, and showcase its application in ranking common defence methods.

Keywords: Diffusion Models, Probabilistic Robustness, Safe AI

1 Introduction

Recent advancements in Text-to-Image (T2I) Diffusion Models (DMs), including state-of-the-arts (SOTA) like DALL-E 3 [3], Imagen [44], Parti [60] and Stable Diffusion [42], enable the generation of high-quality images from text prompts. However, studies [10,11,67] have shown that small perturbations in textual input can substantially degrade the performance of T2I DMs. Fig. 1 illustrates how minor prompt changes can lead to substantial differences in generated images, raising concerns about model robustness in downstream applications [5, 32]. As such, a pivotal question arises: *how can we systematically evaluate and verify (when a specific verification target is provided) the robustness of T2I DMs?*

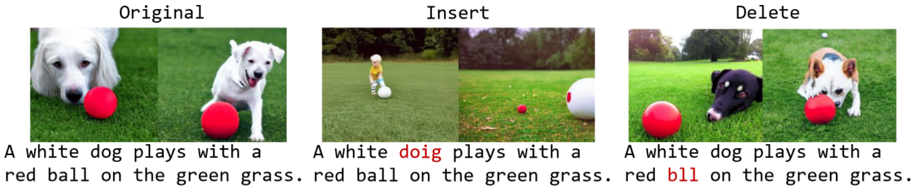

Original
Insert
Delete
A white dog plays with a red ball on the green grass.
A white doig plays with a red ball on the green grass.
A white dog plays with a red bll on the green grass.

Fig. 1: Examples illustrating perturbations applied to the prompt for Stable Diffusion.

The lack of robustness in T2I DMs is unsurprising, given similar issues in the broader realm of Deep Learning (DL) [14, 48]. Generally, robustness refers to the model's consistent decision-making despite small input perturbations. A small perturbation that changes the prediction is termed an Adversarial Example (AE). Over the past decade, numerous studies have attempted to frame DL robustness evaluation as a *binary* or *worst-case* problem, addressing questions like "if AEs exist (in a small local region of the original input)?" or "what is the closest AE to the original input?" [6, 23, 61]. Recently, emerging studies are adopting a *probabilistic view*, formulating robustness verification as a statistical inference problem [7, 8, 22, 50, 51, 55, 64]. Such a probabilistic robustness notion is arguably of more practical interest than binary/worst-case ones, because it provides an *overall* evaluation of how robust the model is whenever an AE can be found [22, 55] and accepts residual risks that are more realistic to achieve [51, 64]. We concur with this view and argue that T2I DMs also necessitate probabilistic robustness verification, which, to the best of our knowledge, is absent in SOTA.

In this paper, we define the probabilistic robustness of T2I DMs against *stochastic* perturbations and introduce ProTIP, an efficient framework for verifying this robustness. Unique challenges arise due to the generative nature of DMs, including computational insensitivity of the generation process and the difficulty of identifying AEs which entails comparing distributional differences of images. Thus, we use *sequential analysis with early stopping rules* and *adaptive concentration inequalities* to determine the "just-right" number of perturbations.

In summary, key contributions of this paper include:

a) **Problem formulation:** For the first time, we formulate the probabilistic robustness verification problem for T2I DMs against stochastic perturbation.

b) **Efficient solution:** To solve the formulated problem, we develop an efficient framework, ProTIP, which incorporates several sequential analysis methods to dynamically determine the sample size and thus enhance the efficiency.

c) **Open-source repository:** A public repository at `https://github.com/wellzline/ProTIP/` containing the codes, datasets, models and experiments.

2 Preliminaries and Related Work

2.1 Text-to-Image Diffusion Models

Generative AI has thrived in the multi-modal field, with DMs excelling in applications like video generation [58], image reconstruction [49], and T2I generation [3]. DMs, probabilistic models using noise injection and reverse sampling, offer control through guidance, enabling complex tasks like T2I generation [19] [26] [46]. Models such as Stable Diffusion [42] and Imagen [44], trained on large text-image datasets, produce high-quality images from text descriptions. Commercial products like DALL-E 3 [3] and Midjourney [1] demonstrate remarkable T2I capabilities. See Appendix A for more details on T2I DMs.

2.2 Deep Learning Robustness

DL models are notoriously unrobust to small perturbations [6, 23, 61]. While definitions of robustness vary in literature, they share a common intuition that a DL model's decision should remain invariant against small perturbations on a given input—typically it is defined as all inputs in a region η have the same prediction label, where η is a small norm ball (in a L_p-norm distance) of radius γ around an input x. A perturbed input (e.g., by adding noise on x) x' within η is an AE if its prediction label differs from x.

In the Safe AI community, DL robustness has been *the* property in the spotlight. Many studies on evaluating DL robustness have been done, framing the problem in different ways. In Fig. 2, we summarise 4 common ways of formulating the problem, inspired by [8]. Earlier works, such as [13, 25, 43], formulate the verification problem as a binary question by asking if any AEs can be found within a given input norm-ball of a specified radius, cf. Fig. 2(a). Such "binary robustness" can be normally evaluated in two ways [25, 43, 59]: either through reachability algorithms that aim to determine the lower and upper bounds of the output within the input norm-ball η, using layer-by-layer analysis; or by solving it using SAT/SMT solvers as a variety of constraint-based programming problems. Fig. 2(b) poses a similar yet different question: what is the maximum radius of η such that no AE exists within it? Intuitively, it is finding the "largest safe perturbation distance" for input x [2, 35, 56, 57]. While in Fig. 2(c), it evaluates the model's robustness by introducing adversarial attacks to cause the maximum prediction loss in the specified norm ball η. It is often applied in *adversarial training* to enhance the model's robustness to resist attacks [33, 52].

The three aforementioned methods all estimate the robustness of the model by detecting the presence of AE or *the* AE that gives the maximum loss/safe-radius, the so called *deterministic* robustness [64]. As argued by [55], they suffer from two major drawbacks: *i)* they fail to convey *how* robust the model is whenever an AE is found; *ii)* they pose scalability challenges when the model is large. Thus, recent works [7,8,21,22,50,51,55,64] develop a *probabilistic* view, by defining robustness as the *proportion* of AEs inside the norm-ball η, cf. Fig. 2(d). This probabilistic notion is arguably more practical, because: *i)* binary/worst-case robustness focusing on extreme cases is neither necessary nor realistic, especially when the model is large; knowing the proportion of AEs is more relevant; *ii)* since all practical applications have acceptable levels of risk, it suffices to demonstrate that the violation probability is below a required threshold, rather than confirming it to be exactly zero. Without loss of generality (WLOG), we illustrate probabilistic robustness using a DL classification task as:

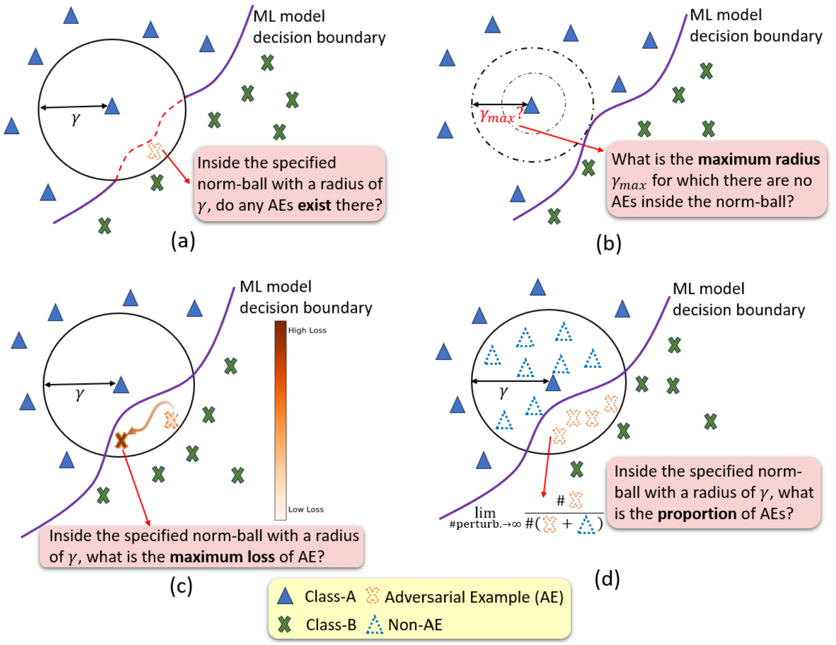

ML model decision boundary
γ
Inside the specified norm-ball with a radius of γ, do any AEs **exist** there?
(a)
ML model decision boundary
γmax?
What is the **maximum radius** γmax for which there are no AEs inside the norm-ball?
(b)
ML model decision boundary
High Loss
Low Loss
γ
Inside the specified norm-ball with a radius of γ, what is the **maximum loss** of AE?
(c)
ML model decision boundary
γ
lim #perturb.→∞ # / #(+)
Inside the specified norm-ball with a radius of γ, what is the **proportion** of AEs?
(d)
Class-A
Adversarial Example (AE)
Class-B
Non-AE

Fig. 2: Four common formulations of robustness verification in DL—binary (a), worst-case (b & c), and probabilistic (d) robustness.

Definition 1 (Probabilistic Robustness). *For a DL classifier f that takes input $x \in \mathcal{X}$ and returns a prediction label, the probabilistic robustness of an input x in a norm ball of radius γ, denoted as $B(x, \gamma)$, is:*

$$R(x, \gamma) := \int_{x' \in B(x,\gamma)} I_{\{f(x')=f(x)\}}(x') Pr(x') \, dx' \tag{1}$$

where $I_{\mathcal{S}}(x)$ is an indicator function—it is equal to 1 when $\mathcal{S}$ is true and equal to 0 otherwise; $Pr(\cdot)$ is the local distribution of inputs representing how perturbations x' are generated, which is precisely the "input model" used by [55, 56].

In this paper, we re-frame such a generic definition of probabilistic robustness for T2I DMs and provide an efficient solution to its verification.

2.3 Robustness of Text-to-Image Diffusion Models

Despite variations of definitions, the key to evaluating and improving the model's robustness is detecting AEs. The approaches to this end are often referred to as "adversarial attacks" via input perturbations. Although such perturbations are commonly referred to as "attacks", they are *not necessarily malicious actions of attackers* [66]. They may also represent natural sensor white noise [17] or benign human errors that follow a *stochastic* generation process. Note that in this work, we adopt such a generalised terminology of AEs to represent *small and random* perturbations. For T2I models, such perturbations can be classified into character-level, word-level, sentence-level or multi-level, depending on the granularity of input perturbations. Recent studies [10,34] show that T2I DMs are very sensitive to black-box attacks like text perturbation. In [11,67], T2I DMs are shown to be vulnerable to realistic human errors (e.g., typos, glyphs, phonetic errors), exposing significant robustness issues due to weak text encoders.

These studies focused only on *deterministic* robustness (e.g., maximised prediction loss). To the best of our knowledge, there is no dedicated exploration into the *probabilistic robustness* of T2I models when they are subject to *stochastic perturbations*, and our ProTIP is the first verification framework for this problem, backed by statistical guarantees. Moreover, ProTIP addresses unique challenges arising from the generative characteristics of T2I DMs, by adopting sequential analysis and adaptive concentration inequality to improve the verification efficiency, details of which are provided in the next section.

3 Method: ProTIP

3.1 Problem Statement

A T2I DM that takes a text input $x \in \mathcal{X}$ and generates an image $y \in \mathcal{Y}$ essentially characterises the conditional distribution $Pr(Y \mid X = x)$[3], i.e., the T2I DM is a function $f : \mathcal{X} \to \mathcal{D}(\mathcal{Y})$ where $\mathcal{D}$ represents the space of all possible distributions over the image set $\mathcal{Y}$. Accordingly, the general probabilistic robustness Def. 1 needs to be adapted for T2I DMs as:

Definition 2 (Probabilistic Robustness of T2I DMs). *For a T2I DM f that takes text inputs X and generates a conditional distribution of images $Pr(Y \mid X)$, the probabilistic robustness of the given input x is:*

$$R_M(x, \gamma) = \sum_{x' : s(T(x), T(x')) \geq \gamma} I_{\{Pr(Y|X=x)=Pr(Y|X=x')\}}(x') Pr(x') \tag{2}$$

where T denotes the CLIP [41] model's text encoder, s is a similarity measurement function (e.g., cosine similarity), γ denotes a given threshold on similarity. While I is an indication function as defined in Def. 1, its value now depends on whether the output distributions before and after the perturbation differ. $Pr(x')$ indicates the probability that x' is the next perturbed text generated randomly.

Intuitively, Def. 2 suggests that $R_M(x, \gamma)$ is the expected probability that the output image distribution remains unchanged for a random perturbation x' that preserves a similar semantic meaning to x (ensured by $s(T(x), T(x')) \geq \gamma$). A "frequentist" interpretation of $R_M(x)$[4], following the gist of Fig. 2(d), is: it is the *limiting relative frequency* of perturbations for which the output distribution is preserved, in an infinite sequence of independently generated perturbations.

[3] As usual, we use capital letters to denote random variables and lower case letters for their specific realisations; $Pr(X)$ is used to represent the distribution of variable X.

[4] Notation-wise, we omit γ from R_M where γ represents a hyperparameter thereafter.

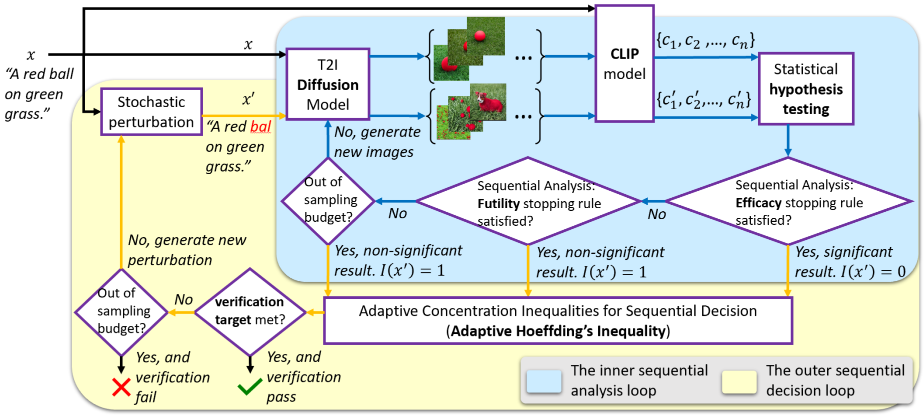

x
"A red ball on green grass."
x
Stochastic perturbation
x'
"A red bal on green grass."
T2I Diffusion Model
CLIP model
$\{c_1, c_2, ..., c_n\}$
$\{c'_1, c'_2, ..., c'_n\}$
Statistical hypothesis testing
No, generate new images
Out of sampling budget?
No
Sequential Analysis: Futility stopping rule satisfied?
No
Sequential Analysis: Efficacy stopping rule satisfied?
Yes, non-significant result. $I(x') = 1$
Yes, non-significant result. $I(x') = 1$
Yes, significant result. $I(x') = 0$
No, generate new perturbation
Out of sampling budget?
No
verification target met?
Adaptive Concentration Inequalities for Sequential Decision (Adaptive Hoeffding's Inequality)
Yes, and verification fail
Yes, and verification pass
The inner sequential analysis loop
The outer sequential decision loop

Fig. 3: Workflow of ProTIP.

Definition 3 (Verification Target). *The probabilistic robustness of the T2I DM f when processing an input x cannot be less than a certain lower bound b_l, with sufficient confidence $1 - \sigma$, i.e.:*

$$Pr(R_M(x) \geq b_l) \geq 1 - \sigma \tag{3}$$

where the pair (b_l, σ) is the given verification target.

To determine if a verification target is met for the input x, we propose the ProTIP workflow in Fig. 3, addressing the following questions: *i)* How to generate stochastic perturbations x' (i.e., the implementation of $Pr(x')$ in Eq. (2)); *ii)* Given a perturbed input x', how to determine if its new output distribution is significantly different to the original one (i.e., the implementation of $I(x')$ in Eq. (2)); *iii)* How to do decision makings based on statistical evaluations of $R_M(x)$ over a sequence of perturbations (i.e., even if we implemented both $Pr(x')$ and $I(x')$ in Eq. (2), the true $R_M(x)$ is still unknown due to the fact that we cannot exhaustively take the sum of their product over all possible perturbations x'; thus, we can only estimate it from a finite sample of perturbed inputs).

While the aforementioned question *i)* is established for which we adopt SOTA methods, e.g., [36], to generate stochastic perturbations on text (cf. Sec. 3.2), questions *ii)* and *iii)* are relatively challenging. Because, both $Pr(Y \mid X = x)$ and $Pr(Y \mid X = x')$ are unknown non-parametric distributions that require sampling images from them (which is costly for T2I DMs) to determine if they are significantly different. To tackle, we propose the *sequential analysis with early stopping rules* and *adaptive concentration inequality* for the last two questions, corresponding to the "inner loop" and "outer loop" of Fig. 3, respectively.

3.2 Stochastic Perturbation Generation

While ProTIP can cope with any *text perturbations generated stochastically*, WLOG, we study five *character-level* text perturbation methods from [36]. First, we set the perturbation rate like 10%, indicating that 10% of the words in the sentence will be perturbed. Then, one of the five perturbation methods (insert, substitute, swap, delete, and keyboard error) shown in Table 2 of Appendix B will be randomly selected. Using insertion perturbation as an example, a letter from the 26 English alphabets is randomly selected and inserted into a randomly chosen position within a word in the sentence. This process is repeated until the perturbed sentence reaches a 10% perturbation rate, ensuring it's not a duplicate before stopping. Taking "*A white dog plays with a red ball on the green grass*" as an example, after insertion perturbation, we get the perturbed sentence "*A white daog plays with a red ball on the green grass*". More details and examples about each text perturbation method can be found in Table 2 of Appendix B.

As Def. 1, the intuition behind robustness suggests that the outputs of a DL model should remain invariant to perturbations on the inputs, while the inputs, before and after the perturbation, should be deemed very similar—thus, Def. 1 restricts the perturbation in a norm-ball $B(x, \gamma)$ with a small radius γ. Such perturbation distance control is easily achievable in Computer Vision where the inputs (either raw pixel values or the latent space feature values) can be normalised and projected into a continuous input space. However, for DL models take text inputs (like T2I DMs), the discrete nature of text make it more challenging to study robustness [29] as a single character change may completely alter the semantics meaning. E.g., "a red ball on ..." and "a red bell on ..." have totally different semantic meanings with only one character perturbed. Thus, to study the robustness of T2I DMs, first we need to ensure that the perturbed text x' has the same semantics to the original text x. Following the idea from [11,31,47,62,63,67] for the same problem, we employ CLIP [41] as the model for text encoding, leveraging CLIP's ability to represent image and text information while preserving their relationships.

Therefore, as per Def. 2, we use CLIP's text encoder T to extract the original embeddings $T(x)$ and perturbed sentence embeddings $T(x')$ to compute the similarity score $s(T(x), T(x'))$ (e.g., by cosine similarity). Only when the similarity score is greater than a given threshold γ, we deem x' is a valid perturbation, where γ can be determined by the method developed in [29] (which yields the maximum perturbation distance in the continuous embedding space that preserves the semantics meaning for a given text). In this way, we ensure that only valid perturbations (sharing similar semantics as the original text) are used for robustness evaluation in ProTIP.

3.3 Indication of Adversarial Examples

For a given perturbed input x', whether it is an AE depends on if the T2I DM output distribution $Pr(Y \mid X = x')$ is significantly different from $Pr(Y \mid X = x)$. In this section, we describe the details of how ProTIP solves the question as a sequential, two-sample statistical testing problem, shown as the "inner loop" in Fig. 3. Although there are techniques capable of *measuring* the distance between two given distributions, e.g., the KL divergence [27] and the Maximum Mean

Discrepancy [16], they are not a *test* and thus they do not provide a statistically principled way to determine *whether or not* the distributions are different.

Statistical hypothesis testing for indicating AE. First, a set of images is generated for the perturbed input x', denoted as $\{y'_1, \dots, y'_n\}$, i.e., a set of independent and identically distributed (i.i.d.) samples from the distribution $Pr(Y \mid X = x')$. Off-the-shelf two-sample test tools do not perform well for high-dimensional data like images [16]. Thus, similar to [11], we adapt CLIP Score to evaluate the correlation between a generated caption for an image and the actual content of the image, which has been found to be highly correlated with human judgement [18]. The metric is formally defined as [18]:

$$CLIPScore(x, y) := max(100 \cdot cos(T(x), V(y)), 0) \tag{4}$$

which corresponds to the cosine similarity between visual CLIP embedding $V(y)$ for an image y and textual CLIP embedding $T(x)$ for a caption x. We refer readers to Appendix B for details on the CLIP Score. For each image y'_i, we calculate a set of i.i.d. "CLIP scores" $\{c'_1, \dots, c'_n\}$ by measuring $c'_i := CLIPScore(x, y'_i)$. A CLIP score c'_i represents the *coexistence likelihood* of the image y'_i and the original text x. We repeat the same calculation for each image y_i generated from the original text x, collectively forming another set of CLIP scores $\{c_1, \dots, c_n\}$.

Intuitively, a higher c'_i indicates a "less adversarial" x' [9,11]. Thus, if x' is not an AE, then the statistics on the group of c'_is should not be significantly smaller than the group of c_is, for which we can do an *one-sided statistical hypothesis testing* with the following null and alternative hypotheses:

- $\mathcal{H}_0$: There is no difference in the two groups of CLIP scores.
- $\mathcal{H}_1$: The CLIP scores of the c'_i group is smaller than the c_i group.

While there are established statistical testing methods, e.g., the t-test, u-test, and their variants [28], that can be employed to answer the above question, their applicability depends on whether the data distribution meets certain assumptions. Our ProTIP is compatible with any applicable statistical tests in this step, cf. later Sec. 4 for our choice in the experiments. We can now implement the AE indicator function in Def. 2 as:

$$I(x') = \begin{cases} 1, & \text{accept the null hypothesis } \mathcal{H}_0 \\ 0, & \text{reject the null hypothesis } \mathcal{H}_0 \end{cases} \tag{5}$$

Sequential analysis with early stopping rules. In contrast to conventional DL tasks like classification, where detecting an AE involves an "one-off" test to see if the prediction label changes, determining if a perturbation affects the T2I DM requires *multiple generations* and comparing the *distributional differences* between two sets of images generated before and after the perturbation. This shift in the evaluation approach leads to a significantly increased computational workload. That said, we employ sequential analysis with early stopping rules [24,39,53] for the hypothetical test in Eq. (5). Instead of collecting all data with the maximum sample size (fixed at the design time) and then analysing the data *a single time by the end*, sequential analysis conducts *interim* analyses during data collection, so that we can prematurely stop data collection at an interim analysis upon rejecting or accepting the null hypothesis. Thanks to early stopping, sequential designs will, on average, require fewer samples [28, Chap. 10].

In ProTIP, we adopt sequential analysis with both *Efficacy* and *Futility* stopping rules, by controlling the α (Type I error, false positive when a null hypothesis is incorrectly rejected) and the β (Type II error, false negative when the null hypothesis is accepted and it is actually false), respectively. Intuitively, the Efficacy stopping rule says, if the analysis reveals a statistically significant result at some interim stage, data collection can be terminated because we reject $\mathcal{H}_0$ when observing the p-value $p < \alpha_k$ (where α_k is the cumulative Type I error rate at the k-th interim analysis, controlled by alpha-spending functions [28]). On the other hand, we may also stop for futility, which means it is either impossible or very unlikely for the final analysis to yield $p < \alpha$. This can be implemented by controlling the Type II error rate across interim analysis using a beta-spending function. We refer readers to [28, Chap. 10] for more details on the topic.

Specifically, we employ the R package RPACT [54] to perform the sequential analysis by setting Pocock type alpha/beta spending functions [15] to determine the sample size and the thresholds for efficacy/futility early stopping rules. Details are presented in Sec. 4 and Appendix D

3.4 Decision-making for Verification

While we establish a method of indicating AEs for *a given perturbation x'* in the last section, to make the decision if a given verification target (cf. Def. 3) is met we need to estimate R_M over *a population of perturbations*, for which we propose the "outer loop" in Fig. 3 and explain details in this section.

Concentration inequalities [4], such as Chernoff inequality, Azuma's bound, and Hoeffding's inequality, represent important statistical methods widely employed in ensuring dependable decision-making with probabilistic guarantees. Specifically, in probability theory, Hoeffding's inequality [20] provides a bound on the probability that the sum of bounded independent random variables deviates from its expected value by more than a certain amount. In ProTIP, for an input x, the T2I DM's probabilistic robustness $R_M(x)$ is the proportion of AEs over a population of all possible perturbations on x. Normally this population is very large (if not infinite)[5], thus the ground truth of $R_M(x)$ can only be estimated as the sample mean of samples drawn from the population. Irrespective of the sample size, there will inevitably be discrepancies between the sample mean and the ground truth population mean. Nevertheless, Hoeffding's inequality allows us to establish tight probabilistic bounds on this error (cf. Appendix C

[5] The population size depends on the length of the original text x and perturbation parameters like perturbation rate. Note, ProTIP can also cope with extreme (and simpler) cases in which the perturbation population is small and its members can be enumerated, e.g., when x is a single word with one character to be changed.

for more on Hoeffding's inequality). A limitation of Hoeffding's inequality is its requirement for a *predetermined, process-independent* sample size. However, in most circumstances, we generally have no idea how many samples are sufficient to verify the model, a priori. Thus, it is common to allocate a maximised sample size that accommodates the budget limit which may result in unnecessary waste.

Inspired by [64, 65], our ProTIP employs an "adaptive" version of Hoeffding's inequality, in which the sample size itself is a variable. This enables the sampling process to stop as soon as the "just right" number of perturbations has been tested for making the verification decision. Moreover, the following theorem proves that the theoretical guarantee on the estimation errors is preserved:

Theorem 1 (Adaptive Hoeffding's Inequality). *We know $I(x'_i)$ is a binary 0–1 random variable. Let $\hat{\mu}_I^{(n)} = \frac{1}{n}\sum_{i=1}^{n} I(x'_i)$ (i.e., the sample mean). Also let J be a random variable on $\mathbb{N}\cup\{\infty\}$, and $\varepsilon(\sigma, n) = \sqrt{\frac{0.6\cdot\log(\log_{1.1} n+1)+1.8^{-1}\cdot\log(24/\sigma)}{n}}$, then we have:*

$$Pr\left(|\hat{\mu}_I^{(J)} - R_M| \leq \varepsilon(\sigma, J)\right) \geq 1 - \sigma \tag{6}$$

where R_M is the true population mean of $I(x'_i)$, and σ is a given confidence level.

The proof for Theorem 1 is presented in [64], which adapts the more general proof in [65] for the binary 0–1 variables I_i and also rearranges terms to align with the form of the original Hoeffding's inequality. We further prove two corollaries in Appendix C to better explain our later experimental results: One compares the tightness of the two bounds derived from the original and adaptive Hoeffding's inequality; the other concerns their monotonicity with respect to n and σ.

In ProTIP, we directly apply Eq. (6) by sequentially increasing the perturbation number J from 1 to j_{max} (the maximum sampling budget). Given the verification target (b_l, σ) (cf. Def. 3), whenever the *empirically estimated lower bound* on $R_M(x)$ yielded by Eq. (6) (after rearranging the inequality) is greater or equal to the specified requirement b_l (i.e., $\hat{\mu}_I^{(J)} - \varepsilon(\sigma, J) \geq b_l$), we stop generating new perturbations and assert "pass" for the verification. Otherwise, when $J = j_{max}$, ProTIP asserts "fail" with an estimated robustness.

4 Experiments

In our experiments, we study three versions of the widely acclaimed and open-source[6] Stable Diffusion (SD) model [42], SD-V1.5, SD-V1.4 and SDXL-Turbo. While different versions of SD models have the same structure, the higher version is further trained based on the previous version; and SDXL-Turbo is based on Adversarial Diffusion Distillation [45]. We use the MS-COCO dataset [30], a comprehensive collection designed for common computer vision tasks including captioning. The dataset comprises 328,000 images with captions, from which we

[6] We run experiments locally on our own servers for efficiency, as ProTIP generates a large number of queries to the model under verification. Thus, we exclude experiments on non-open source, commercial models, e.g., DALLE-3 and Midjourney.

randomly select captions as prompts for the T2I DM and then apply the stochastic perturbation method discussed in Sec. 3.2 to generate perturbed inputs.

All experiments are conducted on NVIDIA GeForce RTX 3090 GPUs, Python 3.11, PyTorch 2.1.1. We use the off-the-shelf R package [40] to design group sequential analysis parameters, cf. Appendix D for details. All our models, data, source code and experimental results are publicly accessible at our Github site.

4.1 Effectiveness of ProTIP

In order to evaluate the effectiveness of ProTIP, i.e., how accurate our ProTIP is, we need to know the ground truth probabilistic robustness $R_M(x)$ for the given prompt x. However, as articulated in Sec. 3.1, the ground truth of $R_M(x)$ can never be known (as it would require exhaustively testing all possible perturbations). Consequently, we can only *approximate the ground truth* by using a significantly larger number of samples than what would normally be used in ProTIP to demonstrate accuracy, which is arguably a common practice in statistical inference, e.g., [8,22,55]. Specifically, we conduct the following two steps: *i)* estimate the ground truth of $I(x')$ for a given perturbation x' by generating a large number of images; *ii)* then estimate the ground truth $R_M(x)$ using a large number of x's with their ground truth $I(x')$. For step 1, we stop generating the images for the x' until the distribution of its CLIP scores "converge" (i.e., the shape of the CLIP score distribution shows no significant changes when new images are generated). We then check whether the two CLIP score distributions follow a normal distribution. If yes, we execute a t-test, otherwise a u-test. In step 2, we generate $1,000$ perturbations, which is a much larger number than the adaptive number of perturbations (around $50 \sim 350$) used in ProTIP. Then the original Hoffeding's inequality (which yields a tighter error bound than the adaptive version used in ProTIP for the same number of perturbations, cf. Corollary 1 in Appendix C) is applied to approximate the ground truth $R_M(x)$.

As shown in Fig. 4(a,b), for each given input x, we conduct 2 sets of comparative experiments for different perturbation rates. In each experiment, we run ProTIP for 3 different confidence levels $1 - \sigma$ (coloured solid lines), and set the verification target to 0.8 (solid black line), while the dashed red line represents the (approximated) ground truth. ProTIP would stop whenever the verification target is met (i.e., the intersection point of the coloured and black lines). But for illustration, we also plot the assessment result after the intersection point, represented by dotted lines.

For a given prompt, Fig. 4(a) illustrates its robustness estimation with 10% perturbation rate, where the estimates based on different confidence levels all converge to the ground truth. The noticeable gap between the approximated ground truth and our results is expected since ProTIP focuses on the lower bound estimation (cf. Eq. (3)), being conservative. Such gap can be reduced when more perturbations are generated for a more accurate estimation (cf. Corollary 2 in Appendix C). For the case of Fig. 4(a), we observe that ProTIP only requires $70 \sim 100$ perturbations to achieve the given verification target $(0.8, \sigma)$.

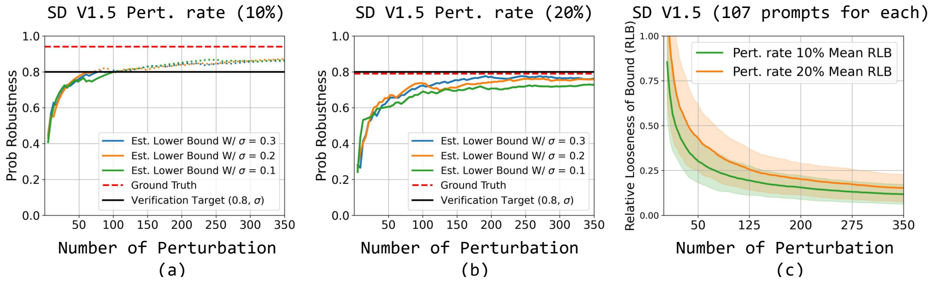


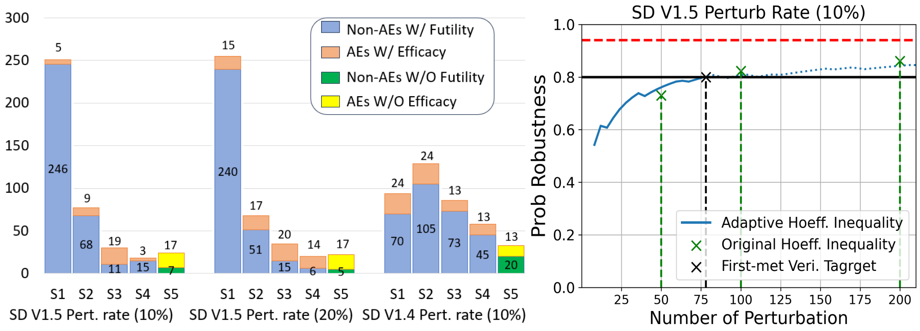
Fig. 4: ProTIP results for a given prompt, with different confidence levels $1-\sigma$ (a)(b); Mean & Std. (shared area) of RLB over 107 prompts (c).

Fig. 4(b) presents the results with an increased perturbation rate of 20%. As expected, the ground truth robustness is lower, as well as the ProTIP results—it asserts "fail" reflects that the ground truth is lower than the verification target. In Fig. 4(c), we introduce the "relative looseness of bound" (defined as $\frac{|GT-\hat{R}|}{GT}$ where GT is the (estimated) ground truth and $\hat{R}$ is the ProTIP result, and present their statistical variations over 107 randomly selected prompts for two settings. We observe the relative errors converge to the theoretical tolerable error ε in Eq. (6). See Appendix E for resutls of more prompts and SDXL-Turbo.

4.2 Efficiency of ProTIP

The efficiency of ProTIP manifests firstly in identifying AEs—the "inner loop" of Fig. 3. As the aforementioned sequential statistical testing method, following the parameter settings of Appendix D, WLOG we divide it into 5 stages to conduct interim analysis. Fig. 5(a) illustrates the number of perturbations identified as Non-AEs/AEs at each interim analysis stage, for a given prompt processed by SD V1.5 with perturbation rates of 10% and 20%, as well as by SD V1.4 with a 10% perturbation rate. For V1.5 with 10% rate, we observe that, out of the total 400 perturbations, 347 are identified as Non-AEs. Thanks to the futility stopping rule, 246 of these Non-AEs can be identified at stage 1, saving 4 times the cost of generating images compared to what would be required without early stopping rules (i.e., test at the final stage 5). Considering other stages as well, in total it reduces the computational cost by detecting 97% of Non-AEs before reaching the final stage 5. Similarly, 68% of AEs are detected early due to the efficacy stopping rule. With a 20% perturbation rate and an older version, the total number of AEs increased and Non-AEs decreased, yet overall, the effect of two early stopping rules in the statistical testing for indicating Non-AEs/AEs remains evident. While Fig. 5(a) demonstrates the results for only one prompt, statistics for 36 randomly selected prompts are presented in Table 1.

The other efficient aspect of ProTIP is demonstrated by introducing the adaptive concentration inequalities to dynamically determine the number of perturbations, cf. the "outer loop" of Fig. 3. To compare with the original Hoeffding's



(a) (b)

Fig. 5: (a) Number of perturbations (out of 400) identified as Non-AEs/AEs at each interim stage (S) in the sequential hypothesis testing. (b) ProTIP with the (adaptive) sample size of 78 vs. Hoeffding's inequality with fixed sample size 50, 100, and 200.

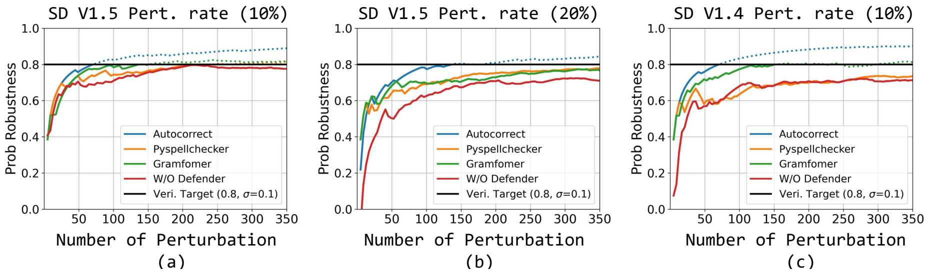
Table 1: The mean and std. of perturbations identified as Non-AE/AE by the two early stopping rules before the final stage 5, over 36 randomly selected prompts.

Model	Pert. rate	Stage 1		Stage 2		Stage 3		Stage 4	
		Efficacy	Futility	Efficacy	Futility	Efficacy	Futility	Efficacy	Futility
SD V1.5	10%	47 ± 32	121 ± 70	22 ± 16	82 ± 35	20 ± 14	43 ± 28	14 ± 12	20 ± 14
	20%	83 ± 48	99 ± 67	38 ± 20	54 ± 26	23 ± 15	34 ± 19	19 ± 9	19 ± 15
SD V1.4	10%	48 ± 29	100 ± 66	25 ± 16	86 ± 38	19 ± 17	45 ± 29	16 ± 10	23 ± 18
SDXL Turbo	10%	64 ± 44	136 ± 76	21 ± 14	73 ± 32	15 ± 13	34 ± 23	12 ± 8	19 ± 14

inequality which requires a predetermined, process-independent sample size, we have to do "what if" calculations by assuming the sample size used by it. Fig. 5(b) presents the ProTIP results and the (original) Hoeffding's inequality with sample sizes of 50, 100, and 200 (dotted cross in green). With sample size 50, Hoeffding's inequality asserts an incorrect verification result "fail", while ProTIP correctly verifies it, although with more samples of 78. This is non-surprising as the error bound is bigger when with limited samples (cf. Corollary 2 in Appendix C). Thus no practitioners would apply Hoeffding's inequality with such small sample size, rather allocate a much larger sample size for a small estimation error. The case with sample size 200 is replicating this scenario, in which both methods make the correct verification decision while ProTIP saves 122 perturbations. For the case of sample size 100, indeed both methods yield similar results (and Hoeffding's inequality is slightly better, cf. Corollary 1 in Appendix C) with similar numbers of samples. However, in practice, we never know such "just right" sample size a priori when applying Hoeffding's inequality, highlighting ProTIP's superiority in efficiency.

4.3 Application of ProTIP for Ranking Defence Methods

As a robustness assessment tool, a natural use case of our ProTIP is to rank common defence methods for T2I DMs. For character-level perturbations that



(a) (b) (c)

Fig. 6: Probabilistic robustness estimated by ProTIP with/without defence methods.

are perceivable and semantic, scrutinising the input is a direct and universally applicable defence. Thus, we study three commonly used misspelling checking tools: Python Autocorrect 0.3.0 [12], Pyspellchecker [37], and Gramformer [38] for error correction on the perturbed inputs before inputting them into T2I DMs.

In Fig. 6, for an example prompt, all three defence tools may improve the model's resistance to perturbations in inputs, compared to the case without using any defenders (the red curve). Notably, Autocorrect demonstrates superior overall performance compared to the other two tools. Such observations are further confirmed by the box plots in Fig. 7 over a set of 36 randomly selected prompts, ranking their performance. Cf. Appendix E for results of more prompts/models.

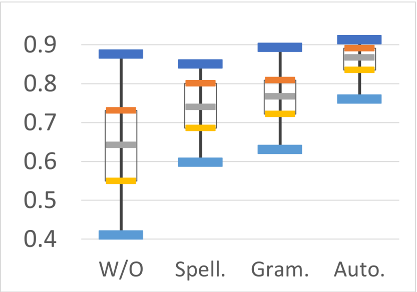


Fig. 7: Box plot of the probabilistic robustness with & without defenders, over 36 randomly selected prompts.

5 Conclusion

In this work, for the first time, we formalise the definition of probabilistic robustness for T2I DMs and then establish an efficient framework, ProTIP, for evaluating it. As a black-box verification method, ProTIP is based on principled statistical inference approaches. It incorporates sequential analysis with early stopping rules in hypothesis testing when identifying AEs, along with adaptive concentration inequalities to dynamically adjust the number of stochastic perturbations needed to make the verification decision. Empirical experiments have substantiated the effectiveness and efficiency of ProTIP. Finally, we demonstrate a use case of ProTIP to rank various commonly used defence methods, highlighting its versatility and applicability.

Acknowledgments

This work is supported by the British Academy's Pump Priming Collaboration between UK and EU partners 2024 Programme [PPHE24/100115], UKRI Future Leaders Fellowship [MR/S035176/1], UK Department of Transport [TETI0042], and Transport Canada [T8080-220112]. The authors would like to thank the WMG centre of HVM Catapult for providing the necessary infrastructure for conducting this study. XH's contribution is supported by the UK EPSRC through End-to-End Conceptual Guarding of Neural Architectures [EP/T026995/1]. YZ's contribution is supported by China Scholarship Council. We thank Hongchen Gao for his valuable comments on our work.

References

1. Midjourney. `https://www.midjourney.com/`
2. Aminifar, A.: Universal adversarial perturbations in epileptic seizure detection. In: 2020 Int. Joint Conference on Neural Networks (IJCNN). pp. 1–6. IEEE (2020)
3. Betker, J., Goh, G., Jing, L., TimBrooks, Wang, J., Li, L., LongOuyang, JuntangZhuang, JoyceLee, YufeiGuo, WesamManassra, PrafullaDhariwal, CaseyChu, YunxinJiao, Ramesh, A.: Improving image generation with better captions
4. Boucheron, S., Lugosi, G., Massart, P.: Concentration inequalities - a nonasymptotic theory of independence. In: Concentration Inequalities (2013)
5. Carlini, N., Farid, H.: Evading deepfake-image detectors with white-and black-box attacks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. pp. 658–659 (2020)
6. Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A., Mukhopadhyay, D.: A survey on adversarial attacks and defences. CAAI Transactions on Intelligence Technology **6**(1), 25–45 (2021)
7. Cohen, J., Rosenfeld, E., Kolter, Z.: Certified adversarial robustness via randomized smoothing. In: Proc. of the 36th Int. Conf. on Machine Learning. PMLR, vol. 97, pp. 1310–1320. PMLR (2019)
8. Dong, Y., Huang, W., Bharti, V., Cox, V., Banks, A., Wang, S., Zhao, X., Schewe, S., Huang, X.: Reliability Assessment and Safety Arguments for Machine Learning Components in System Assurance. ACM TECS **22**(3) (2023)
9. Du, C., Li, Y., Qiu, Z., Xu, C.: Stable diffusion is unstable. 37th Conference on Neural Information Processing Systems (2023)
10. Fort, S.: Pixels still beat text: attacking the openai clip model with text patches and adversarial pixel perturbations. Stanislav Fort [Internet] **5** (2021)
11. Gao, H., Zhang, H., Dong, Y., Deng, Z.: Evaluating the robustness of text-to-image diffusion models against real-world attacks. arXiv preprint arXiv:2306.13103 (2023)
12. Gao, J., Lanchantin, J., Soffa, M.L., Qi, Y.: Black-box generation of adversarial text sequences to evade deep learning classifiers. In: 2018 IEEE Security and Privacy Workshops (SPW). pp. 50–56. IEEE (2018)
13. Gehr, T., Mirman, M., Drachsler-Cohen, D., Tsankov, P., Chaudhuri, S., Vechev, M.: Ai2: Safety and robustness certification of neural networks with abstract interpretation. In: IEEE symposium on security and privacy (SP). pp. 3–18 (2018)
14. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: In Proc. of 3rd Int. Conf. on Learning Representations (2015)

15. Gordon Lan, K., DeMets, D.L.: Discrete sequential boundaries for clinical trials. Biometrika **70**(3), 659–663 (1983)
16. Gretton, A., Borgwardt, K.M., Rasch, M.J., Schölkopf, B., Smola, A.: A kernel two-sample test. The Journal of Machine Learning Research **13**(1), 723–773 (2012)
17. Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., Song, D.: Natural adversarial examples. In: Proc. of the IEEE/CVF conf. on computer vision and pattern recognition. pp. 15262–15271 (2021)
18. Hessel, J., Holtzman, A., Forbes, M., Le Bras, R., Choi, Y.: CLIPScore: A reference-free evaluation metric for image captioning. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. pp. 7514–7528. Association for Computational Linguistics (Nov 2021)
19. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
20. Hoeffding, W.: Probability inequalities for sums of bounded random variables. The collected works of Wassily Hoeffding pp. 409–426 (1994)
21. Huang, W., Zhao, X., Banks, A., Cox, V., Huang, X.: Hierarchical Distribution-Aware Testing of Deep Learning. ACM Trans. Softw. Eng. Methodol. **33**(2) (2023)
22. Huang, W., Zhao, X., Jin, G., Huang, X.: Safari: Versatile and efficient evaluations for robustness of interpretability. In: Proc. of the IEEE/CVF Int. Conf. on Computer Vision (ICCV). pp. 1988–1998 (2023)
23. Huang, X., Kroening, D., Ruan, W., et al: A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. Computer Science Review **37**, 100270 (2020)
24. Jennison, C., Turnbull, B.W.: Group sequential methods with applications to clinical trials. CRC Press (1999)
25. Katz, G., Barrett, C., Dill, D.L., Julian, K., Kochenderfer, M.J.: Reluplex: An efficient smt solver for verifying deep neural networks. In: 29th Int. Conf. Computer Aided Verification (CAV'17). pp. 97–117. Springer (2017)
26. Koller, D., Friedman, N.: Probabilistic graphical models: principles and techniques. MIT press (2009)
27. Kullback, S., Leibler, R.A.: On Information and Sufficiency. The Annals of Mathematical Statistics **22**(1), 79 – 86 (1951)
28. Lakens, D.: Improving Your Statistical Inferences. `https://lakens.github.io/statistical_inferences/` (2022)
29. Li, L., Ren, K., Shao, Y., Wang, P., Qiu, X.: Perturbscore: Connecting discrete and continuous perturbations in NLP. In: Findings of the Association for Computational Linguistics: EMNLP 2023. pp. 6638–6648 (2023)
30. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV'14. pp. 740–755. Springer (2014)
31. Liu, H., Wu, Y., Zhai, S., Yuan, B., Zhang, N.: Riatig: Reliable and imperceptible adversarial text-to-image generation with natural prompts. In: IEEE/CVF Conf. on Computer Vision and Pattern Recognition. pp. 20585–20594 (2023)
32. Lyu, L.: A pathway towards responsible ai generated content. In: Proc. of the 32nd Int. Joint Conference on Artificial Intelligence (IJCAI23). pp. 7033–7038 (2023)
33. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. stat **1050**, 9 (2017)
34. Maus, N., Chao, P., Wong, E., Gardner, J.R.: Black box adversarial prompting for foundation models. In: The 2nd Workshop on New Frontiers in Adversarial Machine Learning (2023)

35. Moosavi-Dezfooli, S.M., Fawzi, A., Fawzi, O., Frossard, P.: Universal adversarial perturbations. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1765–1773 (2017)
36. Morris, J.X., Lifland, E., Yoo, J.Y., Grigsby, J., Jin, D., Qi, Y.: Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP. In: Conference on Empirical Methods in Natural Language Processing (2020)
37. Norvig, P.: pyspellchecker: A spell checker for python. GitHub repository (2024)
38. Prithivida: Gramformer: A library for a family of algorithms to detect, highlight and correct grammar errors. GitHub repository (2021)
39. Proschan, M.A., Lan, K.G., Wittes, J.T.: Statistical monitoring of clinical trials: a unified approach. Springer Science & Business Media (2006)
40. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2021)
41. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML'21. pp. 8748–8763. PMLR (2021)
42. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
43. Ruan, W., Huang, X., Kwiatkowska, M.: Reachability analysis of deep neural networks with provable guarantees. In: Proc. of the 27th Int. Joint Conference on Artificial Intelligence (IJCAI'18). pp. 2651–2659 (2018)
44. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. Advances in Neural Information Processing Systems **35**, 36479–36494 (2022)
45. Sauer, A., Lorenz, D., Blattmann, A., Rombach, R.: Adversarial diffusion distillation. arXiv preprint arXiv:2311.17042 (2023)
46. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: Int. Conf. on machine learning. pp. 2256–2265. PMLR (2015)
47. Struppek, L., Hintersdorf, D., Kersting, K.: Rickrolling the artist: Injecting backdoors into text encoders for text-to-image synthesis. In: Proceedings of the IEEE/CVF Int. Conf. on Computer Vision. pp. 4584–4596 (2023)
48. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. In: In Proc. of 2nd Int. Conf. on Learning Representations (2014)
49. Takagi, Y., Nishimoto, S.: High-resolution image reconstruction with latent diffusion models from human brain activity. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14453–14463 (2023)
50. TIT, K., Furon, T., Rousset, M.: Gradient-informed neural network statistical robustness estimation. In: Proc. of The 26th Int. Conf. on Artificial Intelligence and Statistics. vol. 206, pp. 323–334. PMLR (2023)
51. Wang, B., Webb, S., Rainforth, T.: Statistically robust neural network classification. In: Uncertainty in Artificial Intelligence. pp. 1735–1745. PMLR (2021)
52. Wang, Y., Ma, X., Bailey, J., Yi, J., Zhou, B., Gu, Q.: On the convergence and robustness of adversarial training. In: ICML'19. pp. 6586–6595. PMLR (2019)
53. Wassmer, G., Brannath, W.: Group sequential and confirmatory adaptive designs in clinical trials, vol. 301. Springer (2016)
54. Wassmer, G., Pahlke, F.: rpact: Confirmatory adaptive clinical trial design and analysis (2022)

55. Webb, S., Rainforth, T., Teh, Y.W., Kumar, M.P.: A statistical approach to assessing neural network robustness. In: Int. Conf. on Learning Representations (2019)
56. Weng, L., Chen, P.Y., Nguyen, L., Squillante, M., Boopathy, A., Oseledets, I., Daniel, L.: Proven: Verifying robustness of neural networks with a probabilistic approach. In: Int. Conf. on Machine Learning. pp. 6727–6736. PMLR (2019)
57. Weng, T.W., Zhang, H., Chen, P.Y., Yi, J., Su, D., Gao, Y., Hsieh, C.J., Daniel, L.: Evaluating the robustness of neural networks: An extreme value theory approach. In: Int. Conf. on Learning Representations (2018)
58. Wu, J.Z., Ge, Y., Wang, X., Lei, S.W., Gu, Y., Shi, Y., Hsu, W., Shan, Y., Qie, X., Shou, M.Z.: Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In: Proceedings of the IEEE/CVF Int. Conf. on Computer Vision. pp. 7623–7633 (2023)
59. Xiang, W., Tran, H.D., Johnson, T.T.: Output reachable set estimation and verification for multilayer neural networks. IEEE Tran. on Neural Networks and Learning Systems **29**(11), 5777–5783 (2018)
60. Yu, J., Xu, Y., Koh, J.Y., Luong, T., Baid, G., Wang, Z., Vasudevan, V., Ku, A., Yang, Y., Ayan, B.K., Hutchinson, B.C., Han, W., Parekh, Z., Li, X., Zhang, H., Baldridge, J., Wu, Y.: Scaling autoregressive models for content-rich text-to-image generation. Trans. Mach. Learn. Res. **2022** (2022)
61. Yuan, X., He, P., Zhu, Q., Li, X.: Adversarial examples: Attacks and defenses for deep learning. IEEE Tran. on neural networks and learning systems **30**(9), 2805–2824 (2019)
62. Zhai, S., Dong, Y., Shen, Q., Pu, S., Fang, Y., Su, H.: Text-to-image diffusion models can be easily backdoored through multimodal data poisoning. In: Proceedings of the 31st ACM Int. Conf. on Multimedia. pp. 1577–1587 (2023)
63. Zhang, C., Wang, L., Liu, A.: Revealing vulnerabilities in stable diffusion via targeted attacks. arXiv preprint arXiv:2401.08725 (2024)
64. Zhang, T., Ruan, W., Fieldsend, J.E.: Proa: A probabilistic robustness assessment against functional perturbations. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 154–170. Springer (2022)
65. Zhao, S., Zhou, E., Sabharwal, A., Ermon, S.: Adaptive concentration inequalities for sequential decision problems. In: NurIPS. vol. 29 (2016)
66. Zhao, X., Huang, W., Schewe, S., Dong, Y., Huang, X.: Detecting Operational Adversarial Examples for Reliable Deep Learning. In: 51st Annual IEEE/IFIP Int. Conf. on Dependable Systems and Networks. DSN'21, IEEE/IFIP (2021)
67. Zhuang, H., Zhang, Y., Liu, S.: A pilot study of query-free adversarial attack against stable diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2384–2391 (2023)