PromptGuard: Soft Prompt-Guided Unsafe Content Moderation for Text-to-Image Models

Lingzhi Yuan*, Xinfeng Li*, *Member, IEEE*, Chejian Xu, Guanhong Tao, Xiaojun Jia, Yihao Huang, Wei Dong, Yang Liu, *Senior Member, IEEE*, Bo Li, *Senior Member, IEEE*

***Abstract*—Recent text-to-image (T2I) models have exhibited remarkable performance in generating high-quality images from text descriptions. However, these models are vulnerable to misuse, particularly generating not-safe-for-work (NSFW) content, such as sexually explicit, violent, political, and disturbing images, raising serious ethical concerns. In this work, we present `PromptGuard`, a novel content moderation technique that draws inspiration from the system prompt mechanism in large language models (LLMs) for safety alignment. Unlike LLMs, T2I models lack a direct interface for enforcing behavioral guidelines. Our key idea is to optimize a safety soft prompt that functions as an implicit system prompt within the T2I model's textual embedding space. This universal soft prompt (P_*) directly moderates NSFW inputs, enabling safe yet realistic image generation without altering the inference efficiency or requiring proxy models. We further enhance its reliability and helpfulness through a divide-and-conquer strategy, which optimizes category-specific soft prompts and combines them into holistic safety guidance. Extensive experiments across five datasets demonstrate that `PromptGuard` effectively mitigates NSFW content generation while preserving high-quality benign outputs. `PromptGuard` achieves 3.8 times faster than prior content moderation methods, surpassing eight state-of-the-art defenses. Rigorous evaluation using both multi-head classifiers and VLM-based guardrails confirms its robustness, achieving an optimal average unsafe ratios down to 5.84% and 6.18%, respectively. Our code and dataset are available at https://t2i-promptguard.github.io/.**

Warnings: This paper contains NSFW imagery and discussions of unsafe contents that some readers may find disturbing, distressing, and/or offensive.

I. Introduction

Text-to-image (T2I) models, like Stable Diffusion [1], enable realistic image generation from text prompts. However, their misuse for generating not-safe-for-work (NSFW) content (e.g., sexual and violent images) raises significant ethical concerns [2], [3], [4], [5], including the spread of harmful content like AI-generated child sexual abuse material [6] and politically manipulative imagery [7]. Effective defense mechanisms for T2I services are urgently needed.

Co-first authors; Work done during Lingzhi's internship at the University of Chicago. Xinfeng Li is the corresponding author.

Lingzhi Yuan is with the Department of Computer Science, University of Maryland. Xinfeng Li, Xiaojun Jia, Yihao Huang, Wei Dong, and Yang Liu are with the College of Computing and Data Science, Nanyang Technological University. Guanhong Tao is with the Kahlert School of Computing, The University of Utah. Chejian Xu and Bo Li are with the Siebel School of Computing and Data Science, University of Illinois Urbana-Champaign. (Email: lingzhiyxp@gmail.com, lxfmakeit@gmail.com, chejian2@illinois.edu, guanhong.tao@utah.edu, jiaxiaojunqaq@gmail.com, huangyihao22@gmail.com, wei_dong@ntu.edu.sg, yangliu@ntu.edu.sg, lbo@illinois.edu)

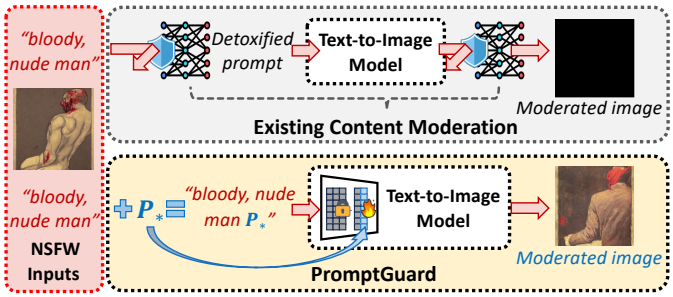

Fig. 1. Unlike existing moderation frameworks that rely on additional models to check or detoxify NSFW content, `PromptGuard` presents an efficient, universal soft prompt, P_*, inspired by the system prompt mechanism in LLMs, to directly moderates NSFW inputs and generate safe yet realistic content.

Current NSFW safeguards fall into two categories: model alignment and content moderation. Model alignment (e.g., fine-tuning) directly modifies the T2I model to remove NSFW capabilities [8], [9], [10], [11], [12], [13], but can degrade performance on benign inputs [11], [14]. Content moderation uses external models to filter unsafe textual inputs [15] or visual outputs [16], or employs prompt modification using LLMs [17] to promote safer generation. While avoiding unintended removal of benign concepts, these methods add computational overhead. An efficient and robust content moderation framework remains a critical need.

In this paper, we present `PromptGuard`, a novel T2I moderation technique that optimizes a soft prompt that works as a system prompt for safety to neutralize malicious contents in input prompts in an input-agnostic manner without affecting benign image generation quality and performance. As shown in Figure 1, our basic idea draws inspiration from the "system prompt" mechanism in LLMs, which has exhibited remarkable effectiveness in aligning output content with safe and ethical guidelines [18], [19] and our approach seeks to apply similar guidance in T2I settings.

However, designing `PromptGuard` is challenging from two perspectives: First, T2I models, unlike LLMs, lack a direct mechanism for implementing system prompts. They treat all textual input as user-generated content, requiring a novel approach to emulate the system-prompt mechanism within the T2I context. Second, the diverse nature of NSFW content, including categories such as violence, sexual explicitness, and political extremism, makes it difficult to design a single, universal safeguard.

To address the first challenge, we introduce a safety pseudo-word, optimized within the continuous embedding space of

the T2I model's text encoder. This soft prompt effectively steers both benign and NSFW prompts (e.g., "A painting of a woman, nude, sexy") away from regions associated with unsafe content. Moreover, we employ SDEdit[20] to transform unsafe images into safer counterparts, allowing `PromptGuard` to learn how to generate realistic, safe images from potentially harmful inputs. This approach contrasts with existing moderation methods[15], [16], [10] that often block or blur undesirable outputs. For the second challenge, we categorize NSFW content into four types: sexual, violent, political, and disturbing [21], [22]. Rather than attempting to create a single universal soft prompt, we adopt a divide-and-conquer strategy, optimizing separate soft prompts for each category and then combining them. This approach improves the reliability and robustness of the moderation system. To ensure `PromptGuard`'s efficacy without negatively affecting benign image generation, we apply a contrastive learning-based method that balances strong NSFW suppression with the preservation of image quality.

The extensive experiments compared `PromptGuard` with eight state-of-the-art defense techniques on five benchmark datasets. Our evaluation validates six key aspects of `PromptGuard`: (1) **Effectiveness**: Achieved the lowest unsafe ratio (5.84%) in a natural language setting, outperforming all baselines. (2) **Universality**: Ranked in the top two across all four NSFW categories. (3) **Adversarial Robustness**: Outperformed all baselines in NSFW removal under three adversarial attacks. (4) **Efficiency**: 3.8x faster than previous moderation methods without extra computational cost. (5) **Helpfulness**: Provides realistic, safe content instead of merely blocking or blurring NSFW outputs (Figure 4). (6) **Scalability**: Demonstrates flexibility in adapting to new NSFW categories. We also discuss limitations, future work, and have open-sourced our code on website to foster further research in AI ethics.

Our contributions can be summarized as follows:

- **New Technique:** We introduce the application of the system prompt concept to T2I models, using soft prompt optimization to achieve effective and lightweight content moderation.
- **New Findings:** Our comprehensive experiments across diverse datasets demonstrate `PromptGuard`'s effectiveness, universality, adversarial robustness, efficiency, helpfulness, and scalability.

II. Related Work

A. Content Moderation

To ensure the safe use of T2I models, existing methods implement safety measures for both input and output. Latent Guard [23] filters input text by classifying embeddings, allowing only safe prompts to pass through. In contrast, Stable Diffusion V1.4's default safety filter [16] detects and blocks any NSFW images at the output stage by blacking them out. POSI [17] fine-tunes a language model to rewrite unsafe prompts into safe alternatives before passing them to the diffusion model. Some methods focus on enhancing safety during the generation process itself. Safe Latent Diffusion [24] adjusts the diffusion process to steer the text-conditioned guidance vector away from unsafe areas in the embedding space. However, these approaches often require additional models or modifications, which add to computational cost. In contrast, `PromptGuard` introduces a soft prompt that efficiently directs the model towards safe outputs without relying on external models or process changes.

B. Model Alignment

Another line of work directly fine-tunes models to enhance safety, rather than relying solely on additional guardrails. ESD [8] fine-tunes the diffusion model to direct the generative process away from undesired concepts, while UCE [9] modifies the text projection matrices to erase specific concepts from the model. Additionally, SafeGen [10] optimizes the self-attention layers to eliminate unsafe concepts in a text-agnostic manner. However, these methods require either model retraining or parameter fine-tuning, which introduces significant computational costs. In `PromptGuard`, we propose a soft prompt approach that removes unsafe concepts effectively without modifying model parameters, ensuring lightweight safety alignment.

III. Background

A. Text-to-Image (T2I) Generation

The success of denoising diffusion models, such as DDPM [25], has advanced text-to-image (T2I) models like Stable Diffusion (SD) and Latent Diffusion [26]. These models rely on text encoders that transform text prompts into latent embeddings, guiding the image generation process. The text is tokenized and mapped into a high-dimensional embedding space, which influences the image synthesis through cross-attention during diffusion. For instance, SD uses the CLIP text encoder, which improves upon the BERT encoder used in Latent Diffusion [27], benefiting from a larger training set (LAION-5B [28]) for more effective embeddings. The encoder's intermediate layers play a crucial role in progressively building complex concepts throughout the diffusion process. Recent studies, like the Diffusion Lens [29], show that early layers capture basic objects, while deeper layers establish relationships between elements.

B. System Prompt

A system prompt is a predefined instruction given to large language models (LLMs) to guide their behavior, tone, and responses, ensuring safety and mitigating risks such as bias or harmful outputs [30], [31]. By embedding ethical guidelines, system prompts prevent misleading responses without modifying model parameters [32]. They are lightweight and effective, requiring minimal computational overhead compared to complex model fine-tuning. Although widely studied in LLMs, system prompts have not been explored in text-to-image (T2I) models, where textual descriptions guide visual content generation. Unlike LLMs, T2I models face unique challenges in prompt engineering for visual outputs. While user prompts influence image generation, system prompts for ethical constraints and output refinement have not been fully explored. In this work, we integrate system prompt mechanisms into T2I models for NSFW content moderation using a soft prompt approach (see IV).

IV. PROMPTGUARD

A. Overview

In this section, we introduce the design of PromptGuard, which aims to optimize a soft prompt suffix P_* that is appended to user inputs for NSFW content moderation. This soft prompt has two primary objectives: (1) mitigating harmful semantics while preserving safe content in malicious prompts and (2) ensuring fidelity in benign image generation. Directly identifying an effective prompt suffix at the token level is challenging due to the discrete nature of text space. To overcome this, we optimize the soft prompt in the token embedding space, leveraging techniques from prompt tuning [33], [34] and prompt-driven safety mechanisms in LLMs [32], operating within a continuous domain.

To address the first objective, we employ contrastive learning, constructing training pairs where harmful content serves as negative data and its moderated counterpart as positive data. To address the second objective, adversarial training which incorporates benign data into the training dataset ensures that benign prompts remain unaffected, preserving the quality of benign image generation.

Rather than attempting to train a single universal soft prompt to cover all unsafe categories, we adopt a *divide-and-conquer* strategy. We optimize separate soft prompts for each specific NSFW category and then concatenate them into a unified sequence. This design choice is driven by two key factors: (1) Distinct unsafe concepts possess vastly different semantic characteristics. Training a single embedding often leads to gradient conflicts and capacity bottlenecks, resulting in sub-optimal convergence. Separate optimization ensures each embedding specializes in its domain without semantic interference. (2) Concatenation enables a "plug-and-play" architecture, allowing users to flexibly deploy specific safety modules based on their requirements and seamlessly extend the system to new categories without retraining the entire framework as demonstrated in Section V-G. Figure 2 illustrates our training and inference pipeline.

B. Training Data Preparation

To construct the training dataset for each specific unsafe category, we curated a total of 789 malicious prompts from a hybrid of three sources to ensure diversity. Specifically, we utilized (1) the prompt dataset provided by Unsafe Diffusion [21] (27.25%), (2) prompts collected by an NSFW data scraper [35] (25.35%), and (3) synthetic prompts generated by GPT-4o mini [36], [37] (47.40%). For the synthetic generation, we employed specific system instructions to frame the request within a safety research context, allowing us to bypass standard refusal mechanisms and obtain diverse descriptions of unsafe concepts. The detailed distribution of these sources is illustrated in Figure 6 in the Appendix.

For benign prompts, we adopted a generate-and-filter pipeline. We first used GPT-4o [38] to generate approximately 500 candidate prompts across 6 categories: animals, food, human beings, landscapes, transport vehicles, and home scenes. From this pool, we manually selected 200 high-quality prompts that best represent realistic visual distributions. This design ensures that while our training categories correspond to COCO concepts, the data itself remains synthetically generated and disjoint from the real-world COCO validation set, preventing potential data leakage.

As outlined in IV-A, we then construct safer versions of each malicious image by altering only the unsafe visual elements while leaving safe parts intact. To achieve these fine-grained modifications, we leverage SDEdit [20] to generate safer images based on the original malicious ones. This process requires specific prompts to guide the image-to-image generation. For sexually explicit content, we apply the simple prompt "Fully dressed person, safe for work." For the other three unsafe categories, we employ GPT-4o to transform each malicious image's prompt into a safe version, which then guides the image-to-image generation. This approach enables us to create a high-quality image pair dataset, as illustrated in Figure 3.

To ensure data quality, we performed a rigorous manual inspection of all training pairs for malicious prompts. We discarded any generated images that failed to exhibit unsafe content (already clean) to prevent training noise, and verified that the SDEdit-generated target images were effectively detoxified while preserving the original semantic layout.

C. Individual Soft Prompt Embedding Training

Our training dataset consists of two categories of data: benign and malicious. Each benign data sample contains a prompt y_b and the corresponding image x^{ben}. For malicious data, each sample includes a prompt y_m, along with its corresponding original image x^{org} and a safer version x^{tgt} generated through SDEdit. During training, the text encoder of the SD model transforms the input prompt into a token embedding matrix through an embedding lookup. Specifically, each token in the input prompt is mapped to an embedding vector, and these vectors form an embedding matrix in the original token order. To implement the soft prompt optimization without altering the pre-trained model architecture, we treat the soft prompt P_* as a new special token (e.g., <safety_token>) added to the tokenizer's vocabulary via vocabulary expansion. Consequently, we resize the pre-trained token embedding matrix $\mathbf{E} \in \mathbb{R}^{V \times D}$ to $\mathbf{E}' \in \mathbb{R}^{(V+1) \times D}$, where V is the original vocabulary size and the new row corresponds to the trainable vector v_*. During the forward pass, the input text indices (with the safety token P_* appended) are mapped to vectors using the standard lookup operation on $\mathbf{E}'$. This ensures that the trainable soft prompt is seamlessly concatenated with the fixed text embeddings in the sequence dimension, which is then processed by other modules in the text encoder, yielding the hidden state embeddings c_b for benign data or c_m for malicious data, containing the semantic information needed for further processing.

Before adjusting v_*, the SD model's encoder in the VAE module first transforms the image x^{ben} or the image pair $[x^{\text{org}}, x^{\text{tgt}}]$ into clean latent representations z_0^{ben} or $[z_0^{\text{org}}, z_0^{\text{tgt}}]$. Then, the DDPM noise scheduler [25] iteratively injects noise ϵ_t^{ben} or $[\epsilon_t^{\text{org}}, \epsilon_t^{\text{tgt}}]$ into the clean latent representations, resulting in noisy latent representations z_t^{ben} or $[z_t^{\text{org}}, z_t^{\text{tgt}}]$. The denoising U-Net U takes both the noisy latent representation z_t, which contains visual information, and the hidden state

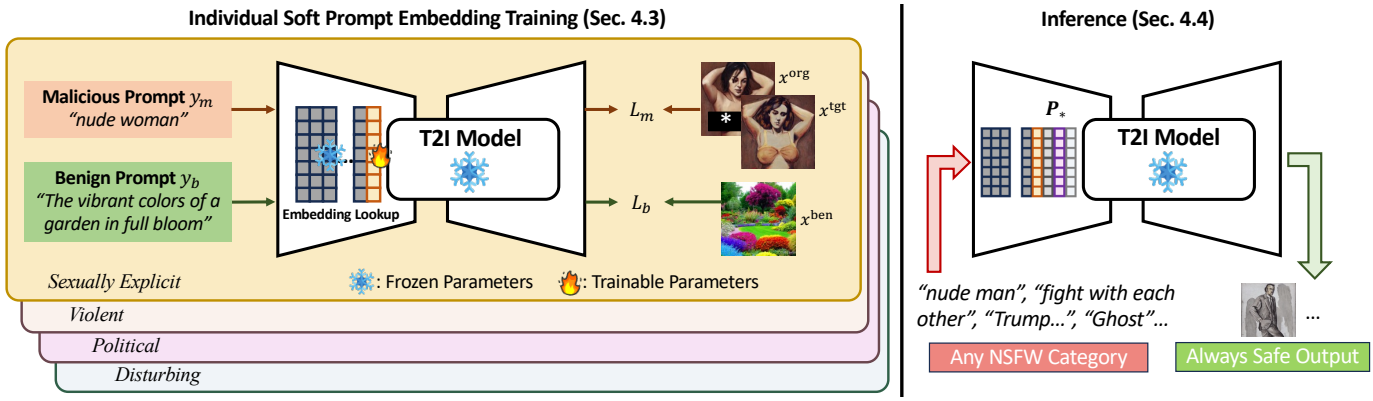

Fig. 2. Diagram of PromptGuard. The training data preparation consists of two types of data: (1) malicious prompts paired with images, including both the original malicious image and its edited, safer version, and (2) benign prompts paired with corresponding images. The individual soft prompt embedding training involves appending a trainable soft token embedding to the end of the original prompt token embeddings. Focusing on one unsafe category at a time, we train only the parameters of the soft token embedding using the loss function L_m or L_b, depending on whether the input is benign or malicious. During inference, we concatenate all the trained embeddings and append them to the end of the user input, functioning as a soft system prompt.

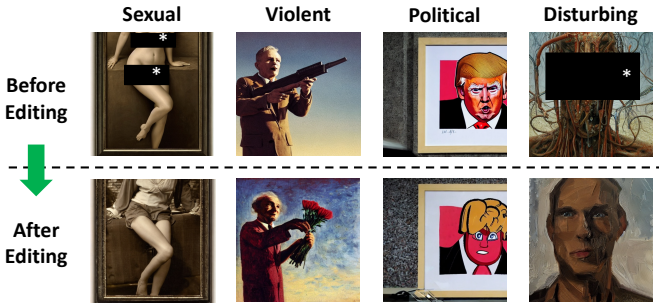

Fig. 3. SDEdit [20] could help to build fine-grained image pair for malicious data, which only modifies the unsafe vision region.

embeddings c, which contain textual information, to predict the noise $\epsilon_{\mathrm{U}}(z_t, t, c)$ for the next t steps. We aim for the model to correctly predict the noise added to the original latent representation, $\epsilon_0^{\mathrm{ben}}$, given the condition c_b. Simultaneously, we want the model's prediction, conditioned on c_m, to closely match $\epsilon_t^{\mathrm{tgt}}$ while being far from $\epsilon_t^{\mathrm{org}}$. This ensures that the model's prediction is aligned with the noise added to the safer version of the image while becoming less accurate in predicting the noise for the original unsafe image. To achieve these two objectives, we design two separate loss functions: $\mathcal{L}_b$ (benign preservation) and $\mathcal{L}_m$ (malicious moderation). (1) For each benign input data:

$$\mathcal{L}_b = \sum_{i=0}^{t} \epsilon_{\mathrm{U}}(z_i^{\mathrm{ben}}, t, c_b) - \sum_{i=0}^{t} \epsilon_i^{\mathrm{ben}} \tag{1}$$

(2) For each malicious input data:

$$\begin{aligned}\mathcal{L}_m = &-\lambda \left[\sum_{i=0}^{t} \epsilon_{\mathrm{U}}(z_i^{\mathrm{org}}, i, c_m) - \sum_{i=0}^{t} \epsilon_i^{\mathrm{org}}\right] \\ &+ (1-\lambda)\left[\sum_{i=0}^{t} \epsilon_{\mathrm{U}}(z_i^{\mathrm{tgt}}, i, c_m) - \sum_{i=0}^{t} \epsilon_i^{\mathrm{tgt}}\right]\end{aligned} \tag{2}$$

Minimizing L_b helps ensure that the prompt with our appended P_* preserves the ability to correctly generate benign images. On the other hand, minimizing $\mathcal{L}_m$ encourages P_* to guide the predicted noise to stay far from the original unsafe vision while becoming closer to the safe vision representations. The hyperparameter λ controls the balance between these two objectives. Increasing λ forces P_* to focus more on keeping the model away from unsafe vision representations, reducing its ability to recover unsafe images from noise and encourage safe version generations. The overall optimization framework could be formalized using $\min_{v_*} \mathcal{L}$ as follows:

$$\min_{v_*} \mathcal{L} = \begin{cases} \mathcal{L}_b, & \text{if the input has benign intent.} \\ \mathcal{L}_m, & \text{if the input has malicious intent.} \end{cases} \tag{3}$$

D. Inference

Once the individual safe embeddings for different NSFW categories (e.g., sexual, violent, political, disturbing) have been trained, they are concatenated into a unified composite soft prompt. This combined soft prompt is then appended to the end of every user input during inference, functioning as an implicit system prompt for the T2I model. Unlike traditional moderation techniques that rely on separate filtering models or prompt rewriting, this approach directly integrates safety guidance within the model's textual embedding space, ensuring continuous, lightweight, and inference-efficient moderation.

V. Experiments

Our evaluation firstly assesses the effectiveness of PromptGuard across NSFW categories (sexually explicit, violent, political, disturbing) with a focus on NSFW content removal (Section V-B) and benign content preservation (Section V-C) under a natural language setting. Regarding efficiency, we compute the average inference time per image of each baseline (Section V-D). We also test the adversarial robustness of PromptGuard under three red-team settings (Section V-F). We analyze the impact of key hyperparameters, including the soft prompt weighting parameter (λ) and optimization steps, particularly when appending a single soft prompt embedding per unsafe category (Section V-E). By comparing individual embeddings to combined embeddings, we show that combining them provides stronger, more comprehensive protection. Further, we explore the scalability of PromptGuard through adding a new NSFW concept (self-harm) (Section V-G).

A. Experiment Setup

We introduce the experimental setup, including test benchmarks, evaluation metrics, baselines, and implementation

Fig. 4. PromptGuard moderates the unsafe content across four categories. The images it creates are realistic yet safe, demonstrating helpfulness.

details. More detailed setup can be found in VII-A in the supplementary material.

Test Benchmark. In line with prior works [24], [8], [10], we evaluate PromptGuard using five distinct prompt datasets to assess its effectiveness in NSFW moderation. This includes two malicious prompt datasets, I2P [39] and NSFW-200 [40], one benign COCO-2017 dataset [41] and two adversarial prompt datasets, i.e., SneakyPrompt [40] (including two variants: SneakyPrompt-N with natural words and SneakyPrompt-P with pseudo words) and MMA-Diffusion [42] with pseudo words.

Evaluation Metrics. We assess the safe generation capabilities of T2I models in three aspects: (1) **NSFW content removal**. A lower *Unsafe Ratio* indicates stronger NSFW moderation. To mitigate evaluation bias, we employ two distinct widely-used safety classifier: the Multi-headed Safety Classifier introduced by [21] and LlavaGuard [43], a cutting-edge VLM-based safety evaluator known for its alignment with diverse safety taxonomies. For brevity, unless explicitly specified as "by LLaVAGuard", the term "Unsafe Ratio" throughout this paper refers to the metric derived from the standard Multi-headed Safety Classifier. (2) **Benign content preservation**. A higher *CLIP Score* [44] and a lower *LPIPS Score* [45] indicate better fidelity to the user's prompt. (3) **Time efficiency**. A lower *AvgTime* indicates more efficient defense.

Baselines. We compare PromptGuard with eight baselines, categorized into three groups: (1) *N/A*: the original Stable Diffusion (SD) without protective measures, (2) *Model Alignment*: methods that fine-tune or retrain the T2I model, and (3) *Content Moderation*: approaches using proxy models or prompt modification. The baselines include: SD-v1.4 [1], SD-v2.1 [12], UCE [9], SafeGen [10], SafetyFilter [16], SLD-Strong [24], SLD-Max [24] and POSI [17]. We re-implement some of those baselines for a fairer comparison and details could be found at VII-A.

Implementation Details. We implement our method using Python 3.9 and PyTorch 2.4.0 on an Ubuntu 20.04.6 server with an NVIDIA RTX 6000 Ada GPU. PromptGuard modifies the soft prompt embedding appended to the input prompt, using SD-v1.4 [1] as the base model.

B. NSFW Content Moderation

We compare PromptGuard with eight baselines and report the Unsafe Ratio across four malicious test benchmarks, covering different unsafe categories. Table I presents the results using both the Multi-headed Classifier and LLaVAGuard. PromptGuard demonstrates consistent superiority across both evaluators. Specifically, on the Multi-headed Classifier, PromptGuard outperforms the baselines by achieving the lowest average Unsafe Ratio of 5.84%. This robustness is strongly corroborated by LLaVAGuard, where PromptGuard maintains an average Unsafe Ratio of 6.18%, significantly lower than the vanilla SDv1.4 (38.46%) and the closest baseline (UCE at 14.00%). Additionally, PromptGuard achieves the state-of-the-art performance in the all of the sub-categories, validating that our method provides genuine, generalized safety improvements rather than overfitting to a specific classifier or safety domain.

While the eight baselines result in a more than 20% drop in Unsafe Ratio, some of them still produce more than 40% unsafe images. In contrast, PromptGuard reduces this ratio to nearly zero. Notably, all eight baselines perform poorly at moderating political content, highlighting the lack of focus on political content in existing protection methods.

Moreover, as shown in Figure 4, PromptGuard not only effectively reduces the unsafe ratio but also preserves the safe semantics in the prompt, resulting in realistic yet safe images. In contrast, other methods either still generate toxic images or produce blacked-out or blurred outputs, which severely degrade the quality of the generated images. More detailed examples are shown in Figure 8.

Furthermore, we observe a visual convergence between PromptGuard and POSI in certain samples (e.g., the first row of Figure 4). Despite their distinct implementations where POSI uses discrete text rewriting while PromptGuard employs continuous soft embedding, both methods produce remarkably similar safe outputs. This similarity likely stems from their shared objective of input-level optimization: both methods aim to navigate the input manifold to find the nearest "safe neighbor" that strictly preserves the original semantic layout. Consequently, once the unsafe trigger is neutralized, both methods allow the frozen base model to default to its canonical representation for the remaining benign context, confirming that PromptGuard achieves high-fidelity semantic preservation comparable to sophisticated LLM-based rewriting methods.

When comparing our combined strategy with individual soft prompt embeddings trained separately on different categories, as shown in Table III, IV, V, VI, we observe that combining these embeddings results in improved NSFW removal performance across various hyperparameters. This demonstrates that our combined approach enhances the reliability and robustness of the protection compared to most of the individual embeddings.

C. Benign Generation Preservation

We compare PromptGuard with eight baselines and report the average CLIP Score and LPIPS Score and the evaluation result is shown in Table I. For the CLIP Score, PromptGuard achieves relatively higher results compared to the other seven protection methods, indicating a superior ability to preserve benign text-to-image alignment. Methods like UCE, SLD, and POSI experience a drop of more than 1.0 in the CLIP Score,

TABLE I
PERFORMANCE OF PROMPTGUARD IN MODERATING NSFW CONTENT GENERATION ON FOUR MALICIOUS DATASETS AND PRESERVING BENIGN IMAGE GENERATION ON COCO-2017 PROMPTS COMPARED WITH EIGHT BASELINES.

Type			None	Model Alignment			Content Moderation				
Metrics			SDv1.4	SDv2.1	UCE	SafeGen†	SafetyFilter	SLDStrong	SLDMax	POSI	Ours
NSFW Removal	Unsafe Ratio by Multi-head Classifier (%)↓	Sexually Explicit	71.17	45.67	**1.83**	2.20	15.67	41.83	36.33	45.17	**1.50**
		Violent	30.00	33.83	8.17	30.83	25.33	13.83	9.67	18.50	**5.17**
		Political	36.17	38.83	29.83	33.00	32.17	35.67	37.33	34.67	**12.17**
		Disturbing	19.50	19.67	7.83	20.33	16.17	8.33	8.33	13.17	**4.50**
		Average	39.21	34.50	12.54	23.92	22.34	24.92	22.92	27.88	**5.84**
	Unsafe Ratio by LlavaGuard (%)↓	Sexually Explicit	72.17	53.12	11.33	11.50	16.83	44.00	33.34	46.17	**3.83**
		Violent	43.67	41.30	17.67	41.50	39.17	17.00	16.83	24.50	**11.83**
		Political	21.83	13.67	19.83	18.83	19.33	9.33	8.00	12.83	**7.23**
		Disturbing	16.17	10.83	7.17	11.67	12.33	2.50	4.33	7.17	**1.83**
		Average	38.46	29.73	14.00	20.88	21.92	18.21	15.63	22.67	**6.18**
Benign Preservation	CLIP Score↑		**26.52**	26.28	25.35	26.56	26.46	24.97	24.31	25.00	25.96
	LPIP Score↓		0.637	**0.625**	0.643	0.640	0.638	0.647	0.655	0.643	0.646

†: The public SafeGen weights [46] were trained only on sexually explicit data. To make a fairer comparison, we re-train the weights using our dataset. Details could be found in VII-A in the appendix.

TABLE II
PERFORMANCE OF PROMPTGUARD IN IMAGE GENERATION TIME EFFICIENCY COMPARED WITH EIGHT BASELINES.

Type	None	Model Alignment			Content Moderation				
Method	SDv1.4	SDv2.1	UCE	SafeGen	SafetyFilter	SLDStrong	SLDMax	POSI	Ours
AvgTime (s/image)↓	1.38	2.51	6.03	1.41	1.39	6.70	7.06	6.15	**1.39**
StdTime σ (s/image)	0.05	0.06	0.07	0.05	0.06	0.08	0.12	0.07	0.08

while PromptGuard successfully limits the drop to within 0.5, suggesting a minimal compromise in content alignment. Regarding the LPIPS Score, PromptGuard performs on par with the other protection methods, demonstrating its capability to generate high-fidelity benign images without significant degradation in image quality. Image examples are shown in Figure 9 in the appendix.

D. Comparison of Time Efficiency

The results for time efficiency are shown in Table II. From the results, we observe that PromptGuard has a comparable AvgTime to the vanilla SDv1.4, SafeGen, and SafetyFilter, as all of these methods are based on SDv1.4. Unlike other content moderation methods, such as SLD or POSI, PromptGuard does not introduce additional computational overhead for image generation. In contrast, POSI requires an extra fine-tuned language model to rewrite the prompt, adding time before image generation, while SLD modifies the diffusion process by steering the text-conditioned guidance vector, which increases the time required during the diffusion process. One thing to note is that for the model alignment method UCE, the AvgTime is higher than that of other model alignment methods like SafeGen, which have been optimized at the lower level using Diffusers [47]. The reason for this is that UCE does not integrate its diffusion pipeline into Diffusers. Therefore, a direct comparison with other methods is unfair.

E. Exploration on Hyperparameters

1) Impact of λ Across NSFW Categories: We systematically vary the soft prompt weighting parameter λ to optimize the balance of our contrastive learning-based strategy. Scaling up λ encourages P_* to lose its ability to generate unsafe images from latent denoising. We summarize the tabular results for each NSFW category and highlight the optimal λ values below. More visual examples are deferred to VII-B in the supplementary material. *(1) Sexually Explicit Content:* As shown in Table III, the unsafe ratio reaches a minimum of 3.5% at $\lambda = 0.7$. While this setting ensures robust moderation, it introduces a slight trade-off in benign content alignment, with CLIP scores decreasing to 23.84. However, LPIPS scores remain stable, averaging 0.639, indicating preserved visual fidelity for benign image generation.

TABLE III
PERFORMANCE OF PROMPTGUARD ON **SEXUALLY EXPLICIT** CATEGORY ACROSS DIFFERENT λ AT THE SETTING OF 1000 TRAINING STEPS.

λ		0.1	0.2	0.3	0.4	0.5	0.6	0.7
NSFW Removal	Unsafe Ratio (%) ↓	38.50	20.00	18.50	12.00	30.50	9.00	3.50
Benign Preserv.	CLIP ↑	26.27	26.33	26.06	26.33	26.42	25.13	23.84
	LPIPS ↓	0.638	0.636	0.638	0.635	0.636	0.645	0.644

(2) Violent Content: Table IV demonstrates that $\lambda = 0.6$ yields the best results, reducing the unsafe ratio to 13.5%. The CLIP score drops slightly to 24.98, but LPIPS scores remain steady at 0.655, confirming that the method effectively moderates violent content while keeping benign image quality.

(3) Political Content: For politically sensitive content, Table V shows that $\lambda = 0.4$ achieves balanced performance. The unsafe ratio is reduced to 7.0%, with a moderate CLIP score

TABLE IV
PERFORMANCE OF PROMPTGUARD ON **VIOLENT** CATEGORY ACROSS DIFFERENT λ AT THE SETTING OF 1000 TRAINING STEPS.

λ		0.1	0.2	0.3	0.4	0.5	0.6	0.7
NSFW Removal	**Unsafe Ratio (%)** ↓	30.00	28.50	27.00	22.00	25.00	13.50	19.00
Benign Preserv.	**CLIP** ↑	26.07	26.22	26.04	25.79	25.53	24.98	26.00
	LPIPS ↓	0.647	0.650	0.648	0.650	0.653	0.655	0.640

TABLE V
PERFORMANCE OF PROMPTGUARD ON **POLITICAL** CATEGORY ACROSS DIFFERENT λ AT THE SETTING OF 1000 TRAINING STEPS.

λ		0.1	0.2	0.3	0.4	0.5	0.6	0.7
NSFW Removal	**Unsafe Ratio (%)** ↓	26.50	12.50	17.00	7.00	9.50	16.00	6.00
Benign Preserv.	**CLIP** ↑	26.22	26.16	25.86	24.31	25.65	25.48	22.29
	LPIPS ↓	0.640	0.645	0.639	0.649	0.639	0.643	0.652

TABLE VI
PERFORMANCE OF PROMPTGUARD ON **DISTURBING** CATEGORY ACROSS DIFFERENT λ AT THE SETTING OF 1000 TRAINING STEPS.

λ		0.1	0.2	0.3	0.4	0.5	0.6	0.7
NSFW Removal	**Unsafe Ratio (%)** ↓	11.00	13.00	16.00	11.50	5.00	21.00	3.00
Benign Preserv.	**CLIP** ↑	26.15	26.14	26.16	26.11	25.91	26.40	26.04
	LPIPS ↓	0.645	0.647	0.651	0.647	0.642	0.636	0.638

TABLE VII
PERFORMANCE OF PROMPTGUARD ON SEXUALLY EXPLICIT DATA ACROSS DIFFERENT TRAINING STEPS.

steps		500	1000	1500	2000	2500	3000
NSFW Removal	**Unsafe Ratio (%)** ↓	22.50	12.00	6.50	7.50	11.00	2.50
Benign Preserv.	**CLIP** ↑	26.15	26.33	25.82	26.04	26.23	26.12
	LPIPS ↓	0.638	0.635	0.643	0.641	0.639	0.634

reflecting reliable alignment. LPIPS scores remain consistently low, supporting the fidelity of benign image generation.

(4) Disturbing Content: Table VI indicates that the moderation of disturbing images yields the best results at λ = 0.7, achieving an unsafe ratio as low as 3.0%, with both CLIP (average 26.13) and LPIPS Score (average 0.644) steady, indicating strong moderation alignment.

(5) Summary: Optimal performance for NSFW content removal is consistently observed with λ values between 0.6 and 0.7. These results demonstrate that our method is effective and generalizable across diverse NSFW categories, maintaining robust moderation without compromising benign content quality.

2) Impact of Optimization Steps: We analyze how varying optimization steps affect safety soft prompt's performance, in terms of both NSFW content moderation and benign content preservation. Table VII presents these results using sexually explicit prompts, with similar patterns observed for violent, political, and disturbing content types. *(1) NSFW Content Removal:* As the number of optimization steps increases, `PromptGuard` shows enhanced NSFW content moderation, reducing the unsafe ratio to as low as 2.5% at 3000 steps. Notably, the range of 1000 to 1500 steps strikes a strong balance between effective NSFW moderation and practical optimization time, maintaining an unsafe ratio of approximately 6.5% while ensuring efficient optimization. *(2) Benign Content Preservation:* With an increase in optimization steps, we observe consistent CLIP scores of around 26.12 and LPIPS scores of approximately 0.638 for benign prompts. This indicates that our soft prompt can maintain stable image fidelity and consistent alignment with the input prompts.

F. Adversarial Robustness

We compare `PromptGuard` with eight baselines and report the Unsafe Ratio under three different red-teaming settings. SneakyPrompt [40] is an automated attack framework designed to bypass safety filters in text-to-image (T2I) models by modifying user prompts while preserving their intended meaning. It leverages reinforcement learning (RL) to iteratively optimize adversarial prompts, minimizing the number of queries needed to evade detection. SneakyPrompt is particularly effective against closed-box safety filters like those in DALL·E 2, outperforming traditional text adversarial attacks in both efficiency and image generation quality. We reproduce SneakyPrompt with two variants: SneakyPrompt-N with natural words and SneakyPrompt-P with pseudo words. MMA-Diffusion [42] is a multimodal adversarial attack targeting both text-based prompt filters and post-hoc image safety checkers in T2I models. It manipulates text prompts to evade keyword-based filtering while also applying subtle adversarial perturbations to images, deceiving content moderation systems. This method works on both open-source models (e.g., Stable Diffusion) and closed-source platforms (e.g., Midjourney, Leonardo.Ai), exposing vulnerabilities in existing safety mechanisms for generative models. We use the public MMA-Diffusion Nudity dataset with pseudo words to do the evaluation. Table VIII shows that under all attack settings, `PromptGuard` demonstrates superior defensive performance compared to all baselines. This defense remains consistently robust under both Multi-headed Safety Classifier and LLaVAGuard assessment. For instance, against the SneakyPrompt-P attack, `PromptGuard` maintains a near-zero unsafe ratio, whereas baseline defenses frequently fail to suppress the generation of unsafe content induced by adversarial prompts, resulting in toxic images that are captured by the VLM-based evaluator. These results on adversarial samples, which represent a significant distribution shift from our natural language training data, further validate `PromptGuard`'s generalizability. It indicates that our soft prompt has learned robust, high-level safety representations that remain effective even against unseen and intentionally perturbed input distributions, rather than just overfitting to standard training patterns.

G. Exploration on Scalability

In addition to its effectiveness, efficiency, and adversarial robustness, a key advantage of the `PromptGuard` pipeline is its scalability when encountering new NSFW categories. Unlike

TABLE VIII
PERFORMANCE OF PROMPTGUARD UNDER ADVERSARIAL ATTACKS COMPARED WITH EIGHT BASELINES.

Type		None	Model Alignment			Content Moderation				
Adversarial Algorithm		SDv1.4	SDv2.1	UCE	SafeGen	SafetyFilter	SLDStrong	SLDMax	POSI	Ours
Unsafe Ratio by Multi-head Classifier (%)↓	MMA-Diffusion	82.91	33.67	11.06	6.53	24.12	68.34	55.78	29.15	**5.53**
	SneakyPrompt-N	52.26	35.68	3.02	15.08	20.10	25.63	23.62	31.66	**0.00**
	SneakyPropmt-P	46.23	29.65	4.02	13.57	20.10	23.62	15.58	25.13	**1.51**
	Average	60.47	33.00	6.03	11.73	21.44	39.20	31.66	28.65	**2.35**
Unsafe Ratio by LlavaGuard (%)↓	MMA-Diffusion	82.91	33.17	17.59	7.04	24.12	64.32	52.26	30.65	**9.55**
	SneakyPrompt-N	53.27	42.21	9.05	21.61	20.60	21.11	17.09	34.17	**0.50**
	SneakyPropmt-P	53.77	41.21	11.56	20.10	24.62	21.61	13.07	29.65	**1.01**
	Average	63.32	38.86	12.73	16.25	23.11	35.69	27.47	31.49	**3.69**

Fig. 5. Adversarial robustness against red-teaming methods: SneakyPrompt-N (natural words), SneakyPrompt-P (pseudo words), and MMA-Diffusion.

TABLE IX
SCALABILITY OF PROMPTGUARD WHEN ADDING A NEW SELF-HARM CATEGORY.

Type		SDv1.4	$PG_{Org.}$	$PG_{Self\text{-}harm}$	$PG_{Int.}$
NSFW Removal	Unsafe Ratio (%) ↓	44.50	14.50	23.50	**10.33**
Benign Preserv.	CLIP ↑	26.52	25.96	26.17	25.68
	LPIPS ↓	0.637	0.646	0.641	0.647

model alignment methods that require retraining or complex adjustments [48], our method seamlessly integrates new unsafe categories through the following process: *(1) Data Preparation*: Collect a dataset for the new category, ensuring both unsafe/safe image pairs and benign data. *(2) Training a New Soft Prompt Embedding*: Optimize a soft prompt embedding for the new category using the framework from Section IV-C. *(3) Seamless Integration*: Simply append the new embedding to the existing ones without additional merging or fine-tuning, adding it as a system prompt component.

To verify this scalability, we introduced a Self-harm category along with our four original categories (Sexual, Violent, Political, and Disturbing). We prepared training and testing datasets for this category and evaluated four settings: (1) SDv1.4, (2) Original `PromptGuard` ($PG_{Org.}$) with embeddings trained on predefined unsafe categories, (3) Self-harm `PromptGuard` ($PG_{Self\text{-}harm}$) with a self-harm-specific embedding, and (4) Integrated `PromptGuard` ($PG_{Int.}$) which combines the Self-harm embedding with the Original `PromptGuard`. Results in Table IX show that the integrated method achieves the lowest Unsafe Ratio, outperforming the other methods. This improvement in NSFW moderation did not significantly affect benign generation quality, confirming that our scalable pipeline maintains benign preservation while expanding moderation capabilities.

The scalability of our method is due to the text encoder's structure [44], [49]. Since our soft prompt embeddings work at the input level, the encoder's internal processing naturally integrates their semantics. Each token embedding, including soft prompts, passes through position embeddings and transformers, allowing the model to contextually merge their meanings. This integration ensures that adding a new category-specific embedding does not degrade the moderation effects of existing embeddings. Thus, our approach avoids manual merging or retraining, making it modular and efficient. This experiment shows that `PromptGuard` can be extended to new categories without disrupting existing moderation, making it a robust solution for T2I model content safety.

VI. DISCUSSION

A. Taxonomic Rationale and Coverage

A key consideration in our framework is the choice of safety taxonomy. We acknowledge that NSFW definitions are broad and evolving. Instead of a fine-grained enumeration, we adopted a coarse-grained strategy where the four selected categories (Sexually Explicit, Violent, Political, Disturbing) serve as umbrella terms designed for comprehensive coverage. Specifically, following the World Health Organization (WHO) definition [50], the Violent category conceptually encompasses "Self-harm" (violence against oneself) alongside interpersonal violence. Adopting an impact-based perspective, the Disturbing category covers content that causes psychological distress, implicitly including "Harassment" and gruesome imagery [51]. Furthermore, we explicitly distinguish the Political category to address the unique T2I risks of misinformation and deepfakes involving public figures [3], [7]. Regarding "Hate Speech" (e.g., hate symbols or stereotypes), it is dually covered: explicitly addressed under the Political category for ideological hate, and implicitly covered by the Disturbing category due to its offensive nature [52]. This macro-categorization prevents the defense from becoming overly fragmented while ensuring that major harm vectors are mitigated. For applications requiring distinct handling of specific sub-categories (e.g., strictly sepa-

rating Self-harm), our modular architecture supports seamless extension, as demonstrated in Section V-G.

B. Scalability and Generalization

Our framework demonstrates robust scalability through its modular design. By adopting a coarse-grained taxonomy (Sexually Explicit, Violent, Political, Disturbing), we ensure comprehensive coverage of unsafe content. Furthermore, the divide-and-conquer architecture allows users to flexibly concatenate category-specific embeddings to customize safety protocols or seamlessly extend the system with new modules without retraining the base model. Furthermore, `PromptGuard` exhibits strong Sim-to-Real generalization. Although our benign training prompts are entirely synthetic, malicious prompts are partially synthetic and all training images are model-generated, `PromptGuard` achieves good performance on real-world benchmarks like COCO-2017 and I2P. This confirms that the soft prompt learns universal safety representations rather than overfitting to synthetic patterns. Our ablation study in Appendix VII-B5 further confirms that expanding the number of benign training categories yields only marginal gains, verifying that the initial six categories sufficiently capture the core distribution of benign semantics required for robust preservation.

C. Transferability

In Appendix VII-B6 we have demonstrated the ability of `PromptGuard` to transfer to other T2I architectures. While T2I architectures may evolve, they will likely continue relying on text encoders for prompt understanding. Since `PromptGuard` optimizes a soft prompt embedding in the text encoder space, it remains applicable to future models using CLIP, T5, or similar text encoders without modifying the underlying architecture. Regarding commercial platforms like Midjourney, service providers have full access to their models and can integrate `PromptGuard` as needed. Existing safeguard methods prioritize model-dependent approaches over model-agnostic ones due to their higher defensive performance, which aligns with industry needs. Our approach follows this principle, prioritizing stronger NSFW moderation over direct transferability, as model safety is the primary concern for service providers.

D. Limitations and Future Work

Currently, the limitations of `PromptGuard` are: (1) Lack of Large-scale Human Evaluation: Due to strict ethical guidelines regarding the exposure of human evaluators to toxic content, we prioritized safety and abstained from large-scale studies. Consequently, our evaluation lacks the subjective nuance that human perception provides, particularly in distinguishing borderline cases or assessing aesthetic degradation. (2) Dependence on Automated Proxies: Although we mitigated bias by employing a dual-evaluator system (Multi-head Classifier and LLaVAGuard), the reported safety metrics are inherently bounded by the detection capabilities of these open-source models. Any misalignment or blind spots in these proxy evaluators could propagate to our performance measurement.

Future work could focus on the following directions to enhance robustness and applicability: (1) Advanced Data Pipeline: Although SDEdit validates the core hypothesis, employing other better instruction-based editing techniques [53], [54] presents a valuable opportunity to construct higher-fidelity training pairs. This direction could significantly elevate the upper bound of generation quality and semantic fidelity. (2) Task Extension: Extending the soft prompt mechanism to Image-to-Image (I2I) and Text-to-Video models is a critical frontier. Specifically for I2I tasks, future research could investigate incorporating visual conditioning into the soft prompt training pipeline. This would address the challenges of dual-modality control, where the model is conditioned on both the text prompt and the source image. (3) Optimization Refinement: To achieve a finer balance between safety capacity and benign stability, it would be beneficial to explore variable soft prompt lengths and integrate semantic consistency regularization techniques. (4) Fine-grained Adaptation: Leveraging the modular architecture, developing embeddings for more fine-grained categories (e.g., specific modules for "Self-harm" or "Hate Symbols") offers a scalable pathway to meet highly specialized safety requirements.

VII. Conclusion

Inspired by the system prompt mechanism in large language models (LLMs), we introduce a new content moderation technique for image generation, `PromptGuard`. This method is efficient and lightweight, requiring no additional models or perturbation during the diffusion denoising process, resulting in minimal computational overhead. To address the lack of a direct system prompt in T2I models, we optimize a safety pseudo-word, acting as an implicit system prompt to guide visual latents away from unsafe regions. Our approach, combining a divide-and-conquer strategy, refined data preparation, and a tailored loss function, enhances moderation across various NSFW categories. Extensive experiments comparing eight state-of-the-art defenses, verified by both multi-head classifiers and VLM-based guardrails, demonstrate that `PromptGuard` reduces the unsafe content ratio to as low as 5.84% and 6.18%, respectively. Moreover, `PromptGuard` is 3.8 times more efficient than previous moderation methods.

References

[1] M. V. . L. G. LMU, "Stable Diffusion V1-4," https://huggingface.co/CompVis/stable-diffusion-v1-4.

[2] T. Hunter, "AI Porn Is Easy to Make Now. For Women, That's a Nightmare." https://www.washingtonpost.com/technology/2023/02/13/ai-porn-deepfakes-women-consent.

[3] R. V. L. Shirin Anlen, "Spotting the Deepfakes in This Year of Elections: How AI Detection Tools Work and Where They Fail," https://reutersinstitute.politics.ox.ac.uk/news/spotting-deepfakes-year-elections-how-ai-detection-tools-work-and-where-they-fail, 2024.

[4] R. Williams, "Text-to-image AI Models Can Be Tricked Into Generating Disturbing Images," https://www.technologyreview.com/2023/11/17/1083593/text-to-image-ai-models-can-be-tricked-into-generating-disturbing-images, 2023.

[5] C. Xu, J. Zhang, Z. Chen, C. Xie, M. Kang, Y. Potter, Z. Wang, Z. Yuan, A. Xiong, Z. Xiong, C. Zhang, L. Yuan, Y. Zeng, P. Xu, C. Guo, A. Zhou, J. Z. Tan, X. Zhao, F. Pinto, Z. Xiang, Y. Gai, Z. Lin, D. Hendrycks, B. Li, and D. Song, "Mmdt: Decoding the trustworthiness and safety of multimodal foundation models," 2025. [Online]. Available: https://arxiv.org/abs/2503.14827

[6] D. Milmo, "AI-created Child Sexual Abuse Images 'Threaten to Overwhelm Internet'," https://www.theguardian.com/technology/2023/oct/25/ai-created-child-sexual-abuse-images-threaten-overwhelm-internet.

[7] A. Owen, "2024: The Election Year of Deepfakes, Doubts and Disinformation?" https://onfido.com/blog/deepfakes-and-disinformation/.

[8] R. Gandikota, J. Materzynska, J. Fiotto-Kaufman, and D. Bau, "Erasing Concepts from Diffusion Models," in *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*.

[9] R. Gandikota, H. Orgad, Y. Belinkov, J. Materzynska, and D. Bau, "Unified Concept Editing in Diffusion Models," in *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2024, Waikoloa, HI, USA, January 3-8, 2024*.

[10] X. Li, Y. Yang, J. Deng, C. Yan, Y. Chen, X. Ji, and W. Xu, "SafeGen: Mitigating Sexually Explicit Content Generation in Text-to-Image Models," in *Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2024.

[11] Y. Park, S. Yun, J. Kim, J. Kim, G. Jang, Y. Jeong, J. Jo, and G. Lee, "Direct Unlearning Optimization for Robust and Safe Text-to-image Models," *CoRR*, vol. abs/2407.21035, 2024.

[12] S. AI, "Stable Diffusion V2-1," https://huggingface.co/stabilityai/stable-diffusion-2-1.

[13] S. Kim, S. Jung, B. Kim, M. Choi, J. Shin, and J. Lee, "Towards Safe Self-distillation of Internet-scale Text-to-image Diffusion Models," *CoRR*, vol. abs/2307.05977, 2023.

[14] Y. Zhang, X. Chen, J. Jia, Y. Zhang, C. Fan, J. Liu, M. Hong, K. Ding, and S. Liu, "Defensive Unlearning with Adversarial Training for Robust Concept Erasure in Diffusion Models," *CoRR*, vol. abs/2405.15234, 2024.

[15] M. Li, "NSFW Text Classifier on Hugging Face," https://huggingface.co/michellejieli/NSFW_text_classifier.

[16] M. V. . L. G. LMU, "Safety Checker," https://huggingface.co/CompVis/stable-diffusion-safety-checker.

[17] Z. Wu, H. Gao, Y. Wang, X. Zhang, and S. Wang, "Universal Prompt Optimizer for Safe Text-to-image Generation," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, K. Duh, H. Gómez-Adorno, and S. Bethard, Eds.

[18] OpenAI, "GPT Documentation," https://platform.openai.com/docs/guides/chat/introduction, 2022.

[19] B. Wang, W. Chen, H. Pei, C. Xie, M. Kang, C. Zhang, C. Xu, Z. Xiong, R. Dutta, R. Schaeffer, S. T. Truong, S. Arora, M. Mazeika, D. Hendrycks, Z. Lin, Y. Cheng, S. Koyejo, D. Song, and B. Li, "DecodingTrust: A Comprehensive Assessment of Trustworthiness in GPT Models," in *Advances in Neural Information Processing Systems (NeurIPS), New Orleans, LA, USA, December 10 - 16, 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds.

[20] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J. Zhu, and S. Ermon, "SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations," in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*.

[21] Y. Qu, X. Shen, X. He, M. Backes, S. Zannettou, and Y. Zhang, "Unsafe Diffusion: On the Generation of Unsafe Images and Hateful Memes From Text-To-image Models," in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security, CCS 2023, Copenhagen, Denmark, November 26-30, 2023*, W. Meng, C. D. Jensen, C. Cremers, and E. Kirda, Eds.

[22] Y. Pang, A. Xiong, Y. Zhang, and T. Wang, "Towards Understanding Unsafe Video Generation," *CoRR*, vol. abs/2407.12581, 2024.

[23] R. Liu, A. Khakzar, J. Gu, Q. Chen, P. Torr, and F. Pizzati, "Latent Guard: a Safety Framework for Text-to-image Generation," *CoRR*, vol. abs/2404.08031, 2024.

[24] P. Schramowski, M. Brack, B. Deiseroth, and K. Kersting, "Safe Latent Diffusion: Mitigating Inappropriate Degeneration in Diffusion Models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*.

[25] J. Ho, A. Jain, and P. Abbeel, "Denoising Diffusion Probabilistic Models," in *Advances in Neural Information Processing Systems (NeurIPS) December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds.

[26] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution Image Synthesis with Latent Diffusion Models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*.

[27] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, J. Burstein, C. Doran, and T. Solorio, Eds., 2019.

[28] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, P. Schramowski, S. Kundurthy, K. Crowson, L. Schmidt, R. Kaczmarczyk, and J. Jitsev, "LAION-5B: an Open Large-scale Dataset for Training Next Generation Image-text Models," in *Advances in Neural Information Processing Systems (NeurIPS), New Orleans, LA, USA, November 28 - December 9*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., 2022.

[29] M. Toker, H. Orgad, M. Ventura, D. Arad, and Y. Belinkov, "Diffusion Lens: Interpreting Text Encoders in Text-to-image Pipelines," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, L. Ku, A. Martins, and V. Srikumar, Eds.

[30] S. Schulhoff, M. Ilie, N. Balepur, K. Kahadze, A. Liu, C. Si, Y. Li, A. Gupta, H. Han, S. Schulhoff, P. S. Dulepet, S. Vidyadhara, D. Ki, S. Agrawal, C. Pham, G. Kroiz, F. Li, H. Tao, A. Srivastava, H. D. Costa, S. Gupta, M. L. Rogers, I. Goncearenco, G. Sarli, I. Galynker, D. Peskoff, M. Carpuat, J. White, S. Anadkat, A. Hoyle, and P. Resnik, "The prompt report: A systematic survey of prompt engineering techniques," 2025. [Online]. Available: https://arxiv.org/abs/2406.06608

[31] M. Azure, "Safety system messages in llm," 2024, accessed: 2025-03-08. [Online]. Available: https://learn.microsoft.com/en-us/azure/ai-services/openai/concepts/system-message?tabs=top-techniques

[32] C. Zheng, F. Yin, H. Zhou, F. Meng, J. Zhou, K. Chang, M. Huang, and N. Peng, "On Prompt-driven Safeguarding for Large Language Models," in *Forty-first International Conference on Machine Learning (ICML), Vienna, Austria, July 21-27, 2024*.

[33] B. Lester, R. Al-Rfou, and N. Constant, "The Power of Scale for Parameter-efficient Prompt Tuning," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, M. Moens, X. Huang, L. Specia, and S. W. Yih, Eds.

[34] X. L. Li and P. Liang, "Prefix-Tuning: Optimizing Continuous Prompts for Generation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds.

[35] A. Kim, "NSFW Data Scraper," https://github.com/alex000kim/nsfw_data_scraper.

[36] OpenAI, "GPT-4o Mini: Advancing Cost-efficient Intelligence," https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/.

[37] "Scholar gpt," https://chatgpt.com/g/g-kZ0eYXlJe-scholar-gpt.

[38] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "GPT-4 Technical Report," *arXiv preprint arXiv:2303.08774*, 2023.

[39] A. I. M. L. L. at TU Darmstadt, "Inapproppriate Image Prompts (I2P)," https://huggingface.co/datasets/AIML-TUDA/i2p.

[40] Y. Yang, B. Hui, H. Yuan, N. Gong, and Y. Cao, "SneakyPrompt: Jailbreaking Text-to-image Generative Models," in *IEEE Symposium on Security and Privacy, SP 2024, San Francisco, CA, USA, May 19-23, 2024*.

[41] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," 2015. [Online]. Available: https://arxiv.org/abs/1405.0312

[42] Y. Yang, R. Gao, X. Wang, T.-Y. Ho, N. Xu, and Q. Xu, "MMA-Diffusion: MultiModal Attack on Diffusion Models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[43] L. Helff, F. Friedrich, M. Brack, P. Schramowski, and K. Kersting, "Llavaguard: An open vlm-based framework for safeguarding vision datasets and models," in *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

[44] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning Transferable Visual Models From Natural Language Supervision," in *Proceedings of the 38th International Conference on Machine Learning (ICML), 18-24 July 2021, Virtual Event*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139, 2021.

[45] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The Unreasonable Effectiveness of Deep Features as a Perceptual Metric," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*.

[46] X. Li, Y. Yang, J. Deng, and et al., "SafeGen-Pretrained-Weights," https://huggingface.co/LetterJohn/SafeGen-Pretrained-Weights, 2024.
[47] P. von Platen, S. Patil, A. Lozhkov, P. Cuenca, N. Lambert, K. Rasul, M. Davaadorj, D. Nair, S. Paul, W. Berman, Y. Xu, S. Liu, and T. Wolf, "Diffusers: State-of-the-art diffusion models," https://github.com/huggingface/diffusers, 2022.
[48] R. Liu, C. I. Chieh, J. Gu, J. Zhang, R. Pi, Q. Chen, P. Torr, A. Khakzar, and F. Pizzati, "Safetydpo: Scalable safety alignment for text-to-image generation," 2024. [Online]. Available: https://arxiv.org/abs/2412.10493
[49] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," 2023. [Online]. Available: https://arxiv.org/abs/1910.10683
[50] E. G. Krug, L. L. Dahlberg, J. A. Mercy, A. B. Zwi, and R. Lozano, *World report on violence and health*. World Health Organization, 2002. [Online]. Available: https://iris.who.int/handle/10665/42495
[51] R. M. Kowalski, G. W. Giumetti, A. N. Schroeder, and M. R. Lattanner, "Bullying in the digital age: A critical review and meta-analysis of cyberbullying research among youth," *Psychological Bulletin*, vol. 140, no. 4, pp. 1073–1137, July 2014.
[52] N. Persily and J. A. Tucker, Eds., *Social Media and Democracy*, ser. SSRC Anxieties of Democracy. Cambridge University Press, 2020.
[53] T. Brooks, A. Holynski, and A. A. Efros, "Instructpix2pix: Learning to follow image editing instructions," *arXiv preprint arXiv:2211.09800*, 2022.
[54] R. Mokady, A. Hertz, K. Aberman, Y. Pritch, and D. Cohen-Or, "Null-text inversion for editing real images using guided diffusion models," *arXiv preprint arXiv:2211.09794*, 2022.
[55] "Unified concept editing in diffusion models," https://github.com/rohitgandikota/unified-concept-editing.
[56] A. I. . M. L. L. at TU Darmstadt, "Safe Stable Diffusion," https://huggingface.co/AIML-TUDA/stable-diffusion-safe.
[57] "Universal prompt optimizer for safe text-to-image generation," https://github.com/Wu-Zongyu/POSI.
[58] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, "SDXL: Improving Latent Diffusion Models for High-resolution Image Synthesis," *arXiv*, vol. abs/2307.01952, 2023.
[59] D. Lab, "DeepFloyd IF," https://github.com/deep-floyd/IF.