

Replace in Translation: Boost Concept Alignment in Counterfactual Text-to-Image

Sifan Li¹ Ming Tao² Hao Zhao³ Ling Shao⁴ Hao Tang⁵

¹Liaoning University ²Nanjing University of Posts and Telecommunications ³Tsinghua University

⁴University of Chinese Academy of Sciences ⁵Peking University

haotang@pku.edu.cn

Abstract

Text-to-Image (T2I) has been prevalent in recent years, with most common condition tasks having been optimized nicely. Besides, counterfactual Text-to-Image is obstructing us from a more versatile AIGC experience. For those scenes that are impossible to happen in real world and anti-physics, we should spare no efforts in increasing the factual feel, which means synthesizing images that people think very likely to be happening, and concept alignment, which means all the required objects should be in the same frame. In this paper, we focus on concept alignment. As controllable T2I models have achieved satisfactory performance for real applications, we utilize this technology to replace the objects in a synthesized image in latent space step-by-step to change the image from a common scene to a counterfactual scene to meet the prompt. We propose a strategy to instruct this replacing process, which is called as Explicit Logical Narrative Prompt (ELNP), by using the newly SoTA language model DeepSeek to generate the instructions. Furthermore, to evaluate models' performance in counterfactual T2I, we design a metric to calculate how many required concepts in the prompt can be covered averagely in the synthesized images. The extensive experiments and qualitative comparisons demonstrate that our strategy can boost the concept alignment in counterfactual T2I.

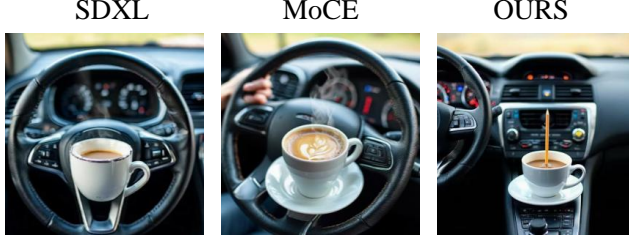
1. Introduction

Recent advancements in Text-to-Image (T2I) generation have revolutionized content creation [5, 21, 32, 34], enabling high-fidelity synthesis of images from textual prompts [10, 23, 25–28, 36, 39]. While state-of-the-art models excel at generating realistic scenes for common scenarios [4, 13, 16, 22, 24, 33], their performance degrades significantly when tasked with counterfactual generation—synthesizing images of improbable or physically implausible scenarios (e.g., "a cat astronaut riding a horse on the moon"). Such counterfactual

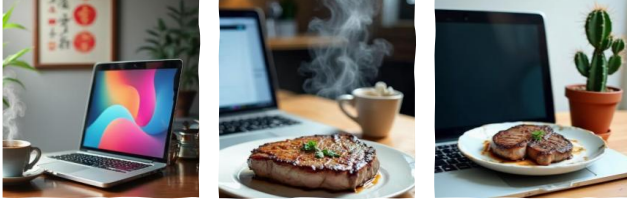
T2I demands not only visual plausibility but also strict adherence to concept alignment, where all specified entities must coexist coherently in the generated image [45]. Despite progress in controllable T2I frameworks [2, 17, 35, 44], existing methods struggle to balance multiple conflicting concepts, often omitting entities or blending them unrealistically [8, 19, 42]. This limitation hinders applications in creative design, educational visualization, and synthetic data generation, where complex scenes are essential [1, 15].

Current approaches to counterfactual T2I, such as causal disentanglement or fine-tuning with rare tokens, primarily address single-concept edits (e.g., swapping object attributes) but fail to scale to multi-entity scenarios [14, 20, 38]. For instance, generating a scene with five unrelated entities frequently results in partial omissions or semantic collisions [37, 45]. Furthermore, evaluation metrics for counterfactual tasks remain underdeveloped [9], with existing tools like Multi-Concept Disparity [45] limited to pairwise interactions. These gaps underscore the need for a systematic strategy to guide multi-concept integration and a robust evaluation framework to quantify alignment.

In this work, we propose **REPLACE IN TRANSLATION (RIT)**, a novel paradigm for enhancing concept alignment in counterfactual T2I. Our approach leverages the observation that counterfactual scenes can often be decomposed into iterative modifications of a plausible base scene. For example, transforming "a man riding a horse on grassland" into "a cat astronaut riding a horse on the moon" involves stepwise replacements: first altering the background ("grassland → moon"), then the subject ("man → cat astronaut"). To operationalize this intuition, we introduce the Explicit Logical Narrative Prompt (ELNP), a structured workflow that directs controllable T2I models (e.g., ControlNet [44]) to replace entities in latent space iteratively. We utilize the new powerful language model DEEPSEEK-R1 [3] to generate the ELNP instructions of steps to replace. At each step, a question block validates the presence of required concepts, preventing error propagation by reverting to prior states if checks fail. Additionally, we design two novel metrics—Multi-Concept



A pencil in a cup of coffee put beside a steering wheel.



A plate of steak on a laptop beside a cactus.



A cat astronaut riding a horse on the moon.



A bird inside a sneaker beside a hamburger and a lighter in front of a pillow.

Figure 1. The qualitative examples of some counterfactual text prompts with more than two entities. Both SDXL [25] and MoCE [45] cannot cover every concepts while our method succeeds to fully cover all the concepts mentioned in the prompts.

Variance and Targeted Entities Coverage—to evaluate alignment and completeness in multi-entity counterfactual scenes.

Our contributions can be summarized as follows:

- **Method:** We propose ELPN, a strategy for hierarchical entity replacement in latent space, coupled with iterative validation through question blocks, to enhance concept alignment in complex counterfactual scenes.
- **Benchmark:** We extend the LC-Mis dataset [45] to construct a comprehensive benchmark for multi-entity (2–5 concepts) and mixed-counterfactual scenarios, enabling systematic evaluation.

- **Metrics:** We devise Multi-Concept Variance and Targeted Entities Coverage to quantify alignment and completeness, addressing limitations in existing pairwise metrics.
- **Empirical Validation:** Extensive experiments demonstrate state-of-the-art performance across 2–5 entity tasks, with 91% entity coverage for 2-concept scenes and 45% for highly complex 5-concept scenarios, significantly outperforming baselines like SDXL [25], SD3-medium [7], DALL·E3 [27] and SANA-1.6B [39].

By bridging the gap between controllable synthesis and counterfactual reasoning, our work advances the frontier of T2I generation, enabling richer, more reliable creation of imaginative visual content. Code, datasets, and visual results will be made publicly available to support further research.

2. Related Works

This section reviews advancements in counterfactual text-to-image (T2I) generation and controllable T2I synthesis, focusing on their methodologies, challenges, and contributions to the field.

2.1. Counterfactual Text-to-Image

Counterfactual reasoning in T2I models addresses "what-if" scenarios by generating images that deviate from factual conditions while preserving logical consistency. A pivotal approach involves Generative Causal Models (GCMs), which disentangle class-agnostic features (e.g., pose) and class-specific attributes (e.g., object traits) to synthesize counterfactual samples. For instance, Zhang *et al.* [43] introduced a framework where replacing class attributes (e.g., "bird species") while retaining sample-specific features (e.g., posture) generates plausible counterfactual images. Their method employs contrastive and adversarial losses to enforce faithfulness between generated and original distributions, enabling applications in zero-shot and open-set recognition.

Recent work further integrates counterfactual faithfulness into diffusion models. For example, DreamBooth [30] fine-tunes pre-trained T2I models using rare token identifiers to bind specific subjects, ensuring that counterfactual edits (e.g., altering backgrounds or styles) preserve core identity features. This approach mitigates overfitting by combining self-generated regularization data with semantic alignment losses, enhancing robustness in multi-context generation [11].

2.2. Controllable Text-to-Image

Controllable text-to-image (T2I) generation [18] focuses on integrating diverse spatial, semantic, or stylistic conditions to guide image synthesis.

Multi-Control Frameworks. Methods like AnyControl [35] unify spatial signals (e.g., depth maps, sketches) and textual prompts through a Multi-Control Encoder. By alternating fusion and alignment blocks, it harmonizes complex relationships between conditions, addressing challenges

like input flexibility and semantic-textual compatibility. Similarly, ControlNet [44] extends T2I models by encoding auxiliary conditions (e.g., edges, segmentation maps) into latent representations, enabling fine-grained structural control.

Prompt Optimization and Style Adaptation. NeuroPrompts [29] automates prompt engineering using reinforcement learning, aligning user inputs with diffusion-model-friendly language. For style-aware generation, Magic Insert [31] combines LoRA-based style infusion with domain-adapted object insertion, enabling realistic drag-and-drop of subjects into stylized scenes while preserving identity.

Concept Editing and Personalization. EMCID [40] tackles large-scale concept editing (e.g., updating 1,000+ concepts) via dual self-distillation and closed-form weight adjustments, ensuring minimal quality degradation. DreamBooth [30] and Textual Inversion [31] personalize models by binding rare tokens to specific subjects, though challenges persist in preserving fine details like logos or text [11].

Sketch and Structure Guidance. Training-free methods like Latent Optimization [6] leverage cross-attention maps to align generated images with input sketches, bridging the gap between abstract drawings and photorealistic outputs.

While counterfactual methods excel in hypothetical generation, they often rely on disentanglement assumptions that may not hold for complex real-world data [11, 43]. Controllable T2I models, though versatile, face trade-offs between flexibility (e.g., supporting arbitrary condition combinations [35]) and computational efficiency. Future work may explore hybrid frameworks that unify causal reasoning with multi-modal control, enhancing both generalization and user-directed creativity.

3. Benchmark

In this section, we detail the process of constructing the datasets and designing the new evaluation metric for multiple entities counterfactual T2I, which addresses limitations in prior frameworks.

3.1. Data Construction

We develop datasets for counterfactual scenes involving more than two entities by extending the LC-Mis dataset [45], which originally targets textual prompts containing two concepts of entities that are statistically unlikely to co-occur in a single scene. Formally, we define an n -counterfactual problem as a text-to-image generation task where the input text specifies n distinct entities with minimal real-world co-occurrence likelihood. The LC-Mis dataset, by this definition, addresses 2-counterfactual problems.

To advance research in complex counterfactual reasoning, this work focuses on evaluating scenarios with $n > 2$ entities. Consequently, we augment the original dataset by generating concept combinations with higher cardinality

(e.g., 3, 5, and mixed-entity configurations). This extension enables systematic benchmarking of multi-entity alignment in counterfactual T2I generation, addressing limitations in prior datasets constrained to pairwise entity interactions.

3.1.1. Concept Distance

We first extract all the concepts contained in the old dataset and define them as a list $Concepts = \{concept_0, concept_1, \dots, concept_i, \dots, concept_n\}$. With the advances of large language models today, we give each two of the concepts ($concept_i$ and $concept_j$ where $i \neq j$) and ask the language model to calculate the probability p of the two given entities appearing in the same scene. The quantized probability $(1 - p)$ is named as the distance between two concepts, whereas p is lower, the two entities are more unlikely to appear in the same scene denoting a larger distance. If we have n concepts, we can obtain $n(n - 1)$ items of distance between any two of the concepts. When we want to construct a concept collection with m entities to tackle m -counterfactual problems, we can calculate the total distance of these m concepts. For each pair in these m concepts, we calculate the distance between them and sum every distance between each pair to represent the total distance. The number of pairs in m concepts should be $K = \binom{m}{2} = \frac{m!}{2!(m-2)!}$ and the total distance should be $D^m = \sum_{i \neq j}^K distance(concept_i, concept_j)$. Among all these n concepts, we can extract all the collections with m different concepts and name them collection- m . The total number of all collections- m should be $M = \binom{n}{m} = \frac{n!}{m!(n-m)!}$ in total. Thereafter, every collections- m takes its own total distance D_i^m . For example, if we want to construct a collection-3, we extract three different concepts from $Concepts$, sum all three distances between each two of them as the total distance D^3 .

3.1.2. Counterfactual Scene Text Generation

For one m -counterfactual problem, we should construct a dataset containing multiple prompts that describe a scene with m concepts of entities. For a *mixed*-counterfactual problem, we should construct a dataset containing prompts that describe scenes with different numbers of concepts of entities. As aforementioned, for m -counterfactual problem, we first extract all collections- m , calculate D^m for each of them, and then rank them by D^m in descending order where a higher rank means a more counterfactual scene. We pick the first r of them, if we want to generate r prompts that contain m concepts of entities. For all these picked collections that can be used to generate the most counterfactual scenes, we tell language models to connect the concepts in the collection with simple words or just commas and 'and'. For example, if we collect ['cat', 'hat', 'book'], the language model can generate *cat, hat and book* or *cat in hat sits beside a book*. We choose the latter pattern to generate prompts.

Compared to simple comma connection, the natural sentence descriptions are more close to real text-to-image. Mostly, we do not simply pile words to synthesize images.

Based on the 126 concept pairs in LC-Mis dataset. We construct subsets for 2, 3, 5 and mixed-counterfactual problems, named as dataset-2, 3, 5 and mixed. Each of them contains 120 concepts collections. For dataset-mixed, we pick 40 highest ranked collections in each of datasets-2, 3 and 5. We utilize DEEPSEEK as the language model to generate the counterfactual scene prompts for each collection in these datasets.

3.2. Quantize Counterfactual Scenes

Recent studies have proposed several evaluation metrics for counterfactual text-to-image (T2I) generation, such as Multi-Concept Disparity [45]. While this metric effectively evaluates scenes involving two entities, its applicability remains limited to pairwise counterfactual scenarios, necessitating extensions to address more complex configurations with three or more entities. Furthermore, inherent limitations in the metric—such as its inability to robustly handle multi-entity alignment and its susceptibility to misinterpretations of concept interactions—result in suboptimal performance when assessing T2I outputs for high-order counterfactual tasks. These shortcomings underscore the need to revise existing frameworks and develop a more comprehensive evaluation metric tailored to multi-concept scenarios.

3.2.1. Multi-Concept Variance

Multi-Concept Disparity is a metric designed to evaluate the alignment between text and images containing multiple concepts in a scene. When a specific score, such as *ClipScore* [12] denoted as S_c , is used to assess a particular feature of text-to-image (T2I) generation—for example, the fidelity of the generated image I to a given concept \mathcal{A} —the calculation can be expressed as:

$$S_c(I, \mathcal{A}) = \text{ClipScore}(I, \mathcal{A}). \quad (1)$$

Sole reliance on ClipScore is insufficient to accurately capture the correlation between the final image I and the entity \mathcal{A} . To address this limitation, the metric was refined to incorporate the ClipScore of the final image for both concepts and their associated textual descriptions generated by GPT-3.5. This revised approach can be formalized as follows:

$$S(I, \mathcal{A}) = \text{Max}(S_c(I, \mathcal{A}), S_c(I, \mathcal{A}_{\text{description}})). \quad (2)$$

This modification enhances the entity’s ClipScore while mitigating misalignment effects to a certain extent. Using this score S , the Multi-Concept Disparity denoted as \mathcal{D} can be formally calculated as follows:

$$\mathcal{D} = |S(I, \mathcal{A}) - S(I, \mathcal{B})|. \quad (3)$$

For 2-counterfactual problems, \mathcal{D} presents the difference in the scores of an image between two concepts. A larger \mathcal{D} denotes a higher likelihood that one concept within it becomes blurred.

Naive subtraction cannot be applied for more than two concepts. To expand Multi-Concept Disparity, we need to update the method how it calculates the difference. In reality, \mathcal{D} can be regarded as the score of balancing the two concepts. When \mathcal{D} turns lower, $S(I, \mathcal{A})$ is more similar to $S(I, \mathcal{B})$. Therefore, we can use biased variance to simulate this disparity score in more than two concepts, which is named Multi-Concept Variance. For n concepts, when we calculate a specific score like *ClipScore*, we use the process in Eq. (2), and calculate Multi-Concept Variance \mathcal{V}_n as:

$$\mathcal{V}_n = \frac{1}{\sqrt{n}} \sum_i^n (S(I, \text{concept}_i) - \bar{S})^2, \quad (4)$$

where \bar{S} denotes the mean value of all $S(I, \text{concept}_i)$. We denote the score with $\sim \text{ClipScore}$ to indicate what score is specifically used as S . For mixed-counterfactual problems, if there are k different kinds of counterfactual problems are mixed, and their total numbers are n_1, n_2, \dots, n_k , respectively, we can regard each kind of counterfactual subproblems as an independent event, calculate $\mathcal{V}_{n_1}, \mathcal{V}_{n_2}, \dots, \mathcal{V}_{n_k}$, and the mean value of them as the final variance score:

$$\mathcal{V}_{\text{mix}} = \frac{1}{k} \sum_{i=1}^k \mathcal{V}_{n_i}. \quad (5)$$

Thus, \mathcal{V}_n can reflect the balancing of more than two concepts. However, using \mathcal{D} and \mathcal{V}_n presents some deficits in evaluation. When a model fails to generate one or fewer than n entities in the image, the metric is increasing reflecting a worse performance. There is one condition that fails to meet our demand: the model fails to generate every required entity and the score will also be low which means good performance. To tackle this deficit, we devise the second new metric to evaluate the performance of actually covering entities other than the concepts balancing.

3.2.2. Targeted Entities

We devise a metric to simply calculate how many required concepts are covered in the generated image. For one prompt of n -counterfactual problems and its corresponding generated image, if the image covers k ($k \leq n$) concepts of entities, we can calculate its targeted score t as:

$$t = \frac{k}{n}. \quad (6)$$

If we use a model to generate N images for one prompt, we will calculate the Total Targeted score \mathcal{T}_n :

$$\mathcal{T}_n = \frac{\sum t}{N}. \quad (7)$$

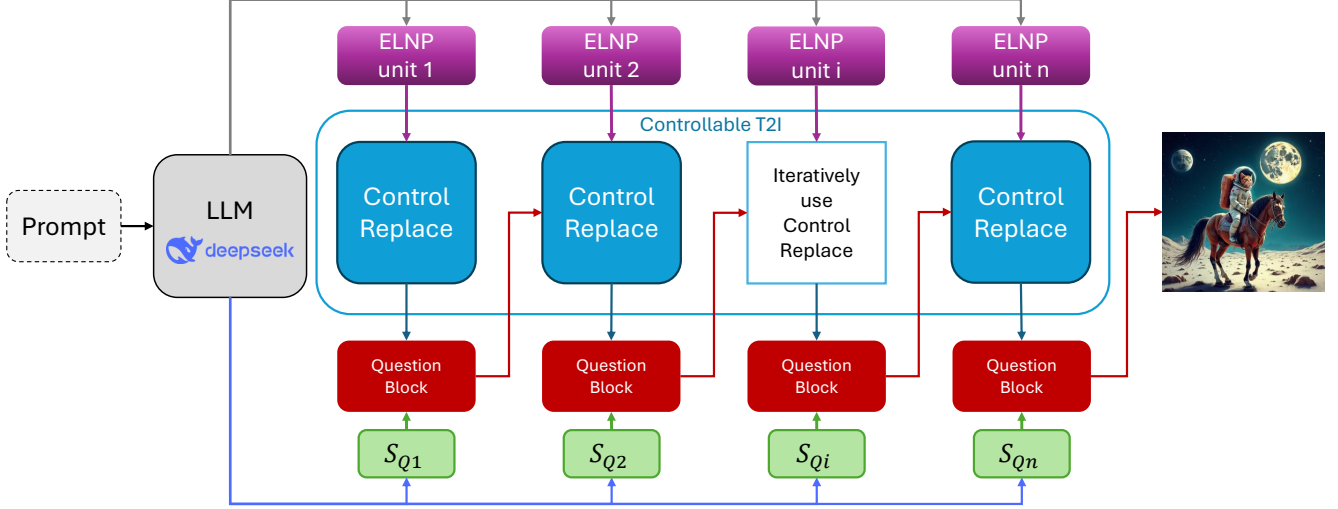


Figure 2. The overview of the ELNP and Question Blocks strategy for counterfactual T2I.

Thus, we can calculate total targeted score for mixed-counterfactual problems more simply. Whereas we can calculate t for every single image, we can just calculate the mean value of them as in Eq. (7) for a mixed problem.

4. Replace in Translation

In this section, we detail the strategy **REPLACE IN TRANSLATION (RIT)** that enhances counterfactual T2I generation. We adopt ControlNet [44] as the backbone framework for the replacement process, leveraging its open-source availability and demonstrated effectiveness in controllable T2I synthesis. The overall workflow can be seen in Fig. 2.

4.1. ELNP

Controllable text-to-image (T2I) generation has matured significantly in recent years. For counterfactual T2I, a natural approach is to replace entities in a common scene with those specified in the counterfactual text. A counterfactual scene can often be conceptualized as a variation of a common scene. For instance, while it is rare to see iced coke contained in a tea cup, this scenario can be derived from a basic outline, such as iced coke in a glass or tea in a tea cup. Given the capability of controllable T2I models to replace entities that share similarities, it is both intuitive and straightforward to substitute entities in the basic outline with their counterfactual counterparts.

Thus, the key challenge lies in identifying the corresponding common scene for a given counterfactual scenario and determining how to systematically replace entities in the common scene to achieve the desired counterfactual output.

To address this, we introduce a strategy called **EXPLICIT LOGICAL NARRATIVE PROMPT (ELNP)**. The process begins with defining the initial sentence as $Base_0$. After the

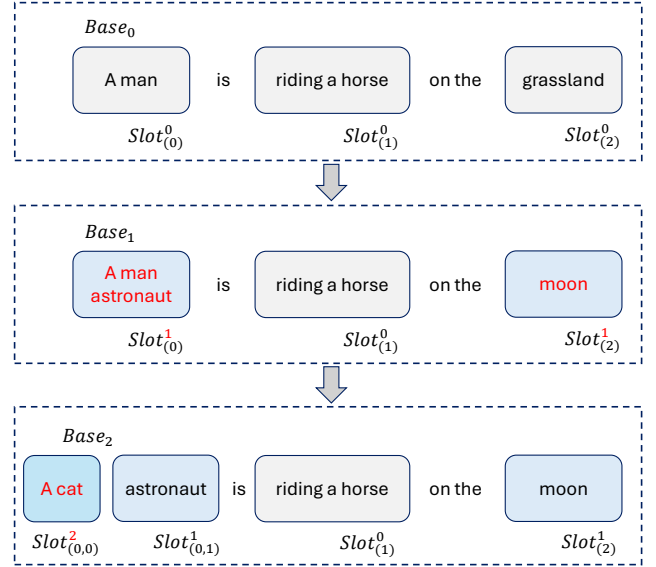


Figure 3. An example of replacement from original base to the target counterfactual text prompt: *A cat astronaut is riding a horse on the moon.*

first replacement step, $Base_0$ is updated to $Base_1$. This iterative process continues, and after k steps, the updated prompt is labeled as $Base_k$, with the corresponding generated image denoted as I_k . Within each $Base$, the term $Slot_v^n$ represents a position in the sentence where an entity is replaced. Here, n indicates the number of times the $Slot$ has been modified, and v serves as an identifier for the $Slot$ in the current step. The replacement of a $Slot$ may result in multiple new slots, forming a tree-like structure.

Once the ELNP is constructed, the steps are executed sequentially by the controllable T2I model. The model first

generates the basic common scene and then replaces the entities step-by-step according to the ELNP. Each iteration concludes when the question block confirms that all required entities up to that point have been successfully incorporated into the image. The intermediate image is then passed back into the latent space for the next iteration until all steps in the ELNP are completed.

4.2. ELNP Generation

There are three steps to ask DEEPSEEK to generate the final ELNP order of replacements for the whole T2I process. The first step is to extract the key entities in the prompt. The second step is to analyze the status of the key entities. The third step is to ask the DEEPSEEK-R1 to generate the proper order of replacement to realize the change from the ordinary scene to the target scene. The exemplars in the third step are based on the knowledge in the previous two steps. Although the third step is the main part of ELNP generation, the language model should interact through the previous two steps to understand the essence of generating ELNP. Straightly using a prompt in the third step will weaken the language models' understanding about ELNP and lead to a failure that gives meaningless response or incorrect replacement orders. Here we present the last step to generate ELNP in Fig. 4. The final output ordered series then become the instructions for the controllable T2I model to replace the entities within the image. The previous two steps generation prompts can be found in Appendix.

Although we use a three-step conversation strategy to generate ELNP, there are always some cases where language models cannot give effective results accordingly. Naturally, all counterfactual scene descriptions can be parsed by human experts, so we can let human experts finish this work. It is not automated, but the most reliable complementary way if we want to enhance the performance in the slightest by removing this disturbance of useless ELNP cases.

4.3. Questions to Ensure

To prevent model collapse, it is crucial to ensure that the initial scene contains the correct number of entities. Errors in entity count can occur at any stage of the process, prompting the need for a question-asking mechanism to verify both the number of entities and the completion of replacements during each iteration. Each question block includes a mandatory query to assess the number of distinct entity concepts present in the pixel space of the generated image. In addition to this essential count verification, the question block for each iteration includes queries to confirm whether all required entity concepts up to that point, as specified by the ELNP, have been successfully incorporated into the image.

For each question block, a threshold for the number of positive responses must be established. Setting the threshold too high (*e.g.*, requiring all questions to be answered

DEEPSEEK-R1

You are a prompt splitter. For a prompt that describes a bizarre scene, you should first figure out what is the most ordinary condition of the scene and provide an ordered series of replacing the objects in the ordinary condition to restore the original bizarre description. Give out the ordered series according to the two exemplars below.

...

Exemplar 1:
The bizarre scene:
{A cat is riding a horse running on the grassland.}

Its most ordinary condition:
{A man is riding a horse running on the grassland.}

The series to restore to the original description:
1: [(man, cat)]~[{A man is riding a horse running on the grassland.}, {A cat is riding a horse running on the grassland.}]

...

Exemplar 2:
The bizarre scene:
{A cat astronaut is riding a horse running on the moon.}

Its most ordinary condition:
{A man is riding a horse running on the grassland.}

The series to restore to the original description:
1: [(man, human astronaut), (grassland, moon)]~[{A man is riding a horse running on the grassland.}, {A human astronaut is riding a horse running on the moon.}]
2: [(human, cat)]~[{A human astronaut is riding a horse running on the moon.}, {A cat astronaut is riding a horse running on the moon.}]

...

Please give the ordered series of the bizarre scene: {A butterfly is flying over a cup of coke.}

Figure 4. The prompt template to generate ELNP: order of replacements. Based on the previous parsing toward entities in the exemplar, we can give clear instruction exemplars for DEEPSEEK to generate the ordered series instructing the controllable model to replace entities within the image step-by-step.

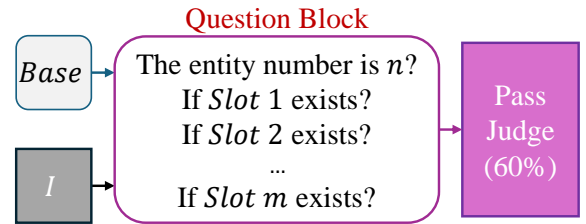


Figure 5. A question block. *I* is the intermediate image decoded from the latent space. Pass judge will check whether the positive answers cover no less than 60% of all questions in this block.

correctly) risks trapping the model in an infinite loop, while setting it too low may degrade performance. Empirically, a threshold of approximately 60% of the questions per step has been found to balance these concerns. Further details on the experiments used to determine this threshold can be found in the Appendix.

Besides the replacement orders and instructions, ELNP will also generate the questions for each period based on the *Slot*. The questions provided for the question block in one period are denoted as S_{Q_i} as shown in Fig. 2.

The state of the latent space from the previous iteration is preserved at each step. If the question block determines that the conditions for proceeding to the next iteration are not met, the latent space is restored to its previous state, and a new replacement is attempted to improve performance in the question block evaluation.

5. Experiments

To validate our strategy, we conduct experiments across a range of counterfactual scenarios, including 2, 3, and 5-entity tasks, as well as mixed-entity configurations. Beyond 2-counterfactual problems, the experiments with higher entity counts and mixed configurations demonstrate that our strategy achieves superior performance in counterfactual text-to-image generation, particularly in handling scenes with multiple concepts and complex conditions. Our approach, built on ControlNet [44], is compared against several state-of-the-art methods, including SDXL [25], MoCE [45], SD3-medium [7], DALL·E3 [27], and SANA-1.6B [39].

5.1. Setup

We exclusively fine-tune the controllable model using our proposed strategy, which significantly reduces the computational budget requirements compared to training a model from start. All experiments are conducted on a single NVIDIA A6000 GPU. We utilize DEEPSEEK-R1 [3] as the language model to generate ELNP and question blocks.

5.1.1. Dataset

To evaluate model performance across varying degrees of counterfactual complexity, we leverage the benchmark datasets introduced in Sec. 3—dataset-2, dataset-3, dataset-5, and dataset-mixed—designed for 2, 3, 5, and mixed-counterfactual scenarios, respectively.

5.1.2. Evaluation Metric

As we mentioned in Sec. 3, we calculate \mathcal{D}_C (\mathcal{D} with ClipScore [12] as S), \mathcal{D}_I (\mathcal{D} with ImageReward [41] as S) and \mathcal{T}_2 for 2-counterfactual problems. $\mathcal{V}_n(C)$ (\mathcal{V}_n with ClipScore as S), $\mathcal{V}_n(I)$ (\mathcal{V}_n with ImageReward as S) and \mathcal{T}_n are employed for n -counterfactual problems, while $\mathcal{V}_{mix}(C)$, $\mathcal{V}_{mix}(I)$ and \mathcal{T}_{mix} are employed for mixed-counterfactual problems. To facilitate easier observation, we establish all these metrics at a magnification of $10\times$, whereas the results are always at the level of 10^{-2} .

5.2. Two Entities

As shown in Tab. 1, for 2-counterfactual problems, we conduct experiments on dataset-2. All the metrics are calculated

Method	\mathcal{D}_C (\downarrow)	\mathcal{D}_I (\downarrow)	\mathcal{T}_2 (\uparrow)
SDXL [25]	1.09	1.21	0.80
MoCE [45]	0.55	0.79	0.86
SD3-medium [7]	0.66	0.75	0.84
DALL·E3 [27]	0.37	0.65	0.85
SANA-1.6B [39]	0.46	0.74	0.86
RIT (ours)	0.24	0.62	0.91

Table 1. The comparison on counterfactual T2I with 2 concepts of entities. The best results are in boldface.

to present that our strategy outperforms any other advanced text-to-image tasks.

Our method RIT achieves the lowest \mathcal{D}_C (0.24) and \mathcal{D}_I (0.62), indicating superior alignment between generated images and counterfactual prompts. The higher \mathcal{T}_2 (0.91) demonstrates that 91% of required entities are retained on average, outperforming baselines like DALL·E3 (0.85) and SDXL (0.80). This success stems from ELNP’s step-by-step latent replacement, which incrementally integrates concepts while minimizing conflicts. Notably, DALL·E3’s strong \mathcal{D}_I (0.65) suggests it excels in stylistic fidelity but struggles with multi-concept coverage, highlighting the trade-off between realism and alignment.

5.3. More Entities

As shown in Tab. 2, for 3-counterfactual problems and 5-counterfactual problems, we conduct experiments on dataset-3 and dataset-5. All the metrics present that our strategy outperforms any other advanced text-to-image tasks.

For 3-counterfactual problems, RIT maintains leading performance, with $\mathcal{V}_3(C) = 0.54$ and $\mathcal{V}_3(I) = 0.68$ outperforming SD3-medium (0.58 and 0.89) and SANA-1.6B (0.63 and 0.80). The smaller performance gap compared to 2-entity tasks reflects the inherent complexity of balancing more concepts. However, our question-block mechanism ensures iterative verification, enabling robust alignment even as complexity increases. The \mathcal{T}_3 score (0.84) further validates that 84% of entities are consistently preserved, a significant improvement over SDXL (0.64).

All methods exhibit performance degradation as concepts scale to five, but our approach remains superior, achieving $\mathcal{V}_5(C) = 0.95$ and $\mathcal{T}_5 = 0.45$. While even our method’s \mathcal{T}_5 is modest (45% coverage), it significantly outperforms SANA-1.6B (41%) and DALL·E3 (36%). This underscores the challenge of synthesizing highly counterfactual scenes with multiple entities, where latent-space replacements risk concept overlap or omission. The structured ELNP process mitigates this by enforcing gradual integration, but further improvements are needed for problems containing more concepts of entities.

We can conclude three features from the experiments of fixed-counterfactual problems.

Method	$\mathcal{V}_3(C) \downarrow$	$\mathcal{V}_3(I) \downarrow$	$\mathcal{T}_3 \uparrow$	$\mathcal{V}_5(C) \downarrow$	$\mathcal{V}_5(I) \downarrow$	$\mathcal{T}_5 \uparrow$	$\mathcal{V}_{mix}(C) \downarrow$	$\mathcal{V}_{mix}(I) \downarrow$	$\mathcal{T}_{mix} \uparrow$
SDXL [25]	1.69	1.83	0.64	1.78	2.02	0.12	1.48	1.69	0.48
MoCE [45]	0.93	1.17	0.72	1.67	1.95	0.32	1.08	1.39	0.56
SD3-medium [7]	0.58	0.89	0.78	1.24	1.59	0.35	0.86	1.29	0.55
DALL-E3 [27]	0.67	0.76	0.76	1.58	1.10	0.36	0.75	1.48	0.60
SANA-1.6B [39]	0.63	0.80	0.82	1.09	0.96	0.41	0.87	1.03	0.61
RIT (ours)	0.54	0.68	0.84	0.95	0.92	0.45	0.74	0.96	0.62

Table 2. The score evaluation on counterfactual T2I with 3, 5 and mixed numbers of concepts of entities. The best results are in boldface.

Scalability. Our method’s relative advantage grows with concept count (*e.g.*, 0.24 *vs.* 0.37 \mathcal{D}_C for two concepts, 0.95 *vs.* 1.09 $\mathcal{V}_5(C)$), suggesting its hierarchical replacement strategy scales better than end-to-end approaches.

Metric Sensitivity. The disparity between \mathcal{D}_C and \mathcal{D}_I *e.g.*, DALL-E3’s lower \mathcal{D}_I , implies that ClipScore and ImageReward capture different aspects of alignment (semantic *vs.* stylistic), necessitating multi-metric evaluation.

Failure Modes. Cases with low \mathcal{T}_n *e.g.*, SDXL’s $\mathcal{T}_5 = 0.12$, often involve concept "collisions" in latent space, where one entity dominates or replaces others. Our question blocks partially address this by reverting to prior states if checks fail, ensuring incremental progress.

These results validate that ELNP and iterative verification enhance concept alignment in counterfactual T2I, particularly for complex, multi-entity scenes. Future work could explore hybrid architectures to further improve scalability and coverage.

5.3.1. Qualitative Comparison

Some counterfactual T2I results generated by SDXL [25], MoCE [45] and ours are presented in Fig. 1. Among all these examples, SDXL and MoCE fail to cover all the concepts of entities in the prompt. In the first row, our method RIT is the only one painting the pencil, and putting the coffee cup at a relatively possible position. The other two put the coffee cup on the steering wheel which is an odd position. In the second row, SDXL only paints the laptop, as MoCE adds the steak, while ours cover all the entities including the cactus. In the third row, our method is the only one to manifest the head of the astronaut showing that it is a cat. In the last row, the models are faced with a complex prompts with five concepts. Our method manifests all the five concepts of entities and displays them in a natural layout, while the other two models tend to omit the sneaker.

5.4. Mixed Numbers of Entities

As shown in Tab. 2, for mixed-counterfactual problems, we conduct experiments on dataset-mixed.

RIT achieves state-of-the-art results across all metrics for mixed-counterfactual problems, with the lowest variance scores ($\mathcal{V}_{mix}(C) = 0.70$, $\mathcal{V}_{mix}(I) = 0.80$) and the highest targeted coverage $\mathcal{T}_{mix}(0.80)$. This demonstrates ro-

bust alignment and consistency when handling prompts with varying entity counts (2–5 concepts). In contrast, baseline methods exhibit significant performance degradation. For instance, SDXL struggles with high variance ($\mathcal{V}_{mix}(C) = 1.50$) and poor coverage $\mathcal{T}_{mix}(0.50)$, indicating its inability to balance or retain multiple conflicting concepts. Even stronger baselines like SANA-1.6B ($\mathcal{T}_{mix}(0.75)$) lag behind our approach, highlighting the effectiveness of ELNP’s iterative replacement strategy.

Thereafter, we can validate the effectiveness of the key insights of our method.

Hierarchical Latent Editing. The ELNP framework’s step-by-step entity replacement mitigates concept collisions in mixed scenarios. By decomposing complex prompts into sequential edits *e.g.*, first replacing "grassland" with "moon," then "man" with "cat astronaut", RIT ensures gradual adaptation in latent space, preserving coherence even as counterfactual complexity increases.

Question Blocks as Stabilizers. The 60% threshold in question blocks prevents over-commitment to partial replacements. For example, if an intermediate step fails to include 3/5 required entities, the model reverts and retries, avoiding error propagation. This mechanism explains the higher \mathcal{T}_{mix} compared to end-to-end approaches like DALL-E3.

Metric Behavior. The gap between $\mathcal{V}_{mix}(C)$ and $\mathcal{V}_{mix}(I)$ (0.70 *vs.* 0.80) reflects differing sensitivities: ClipScore as in $\mathcal{V}_{mix}(C)$ penalizes semantic misalignments *e.g.*, missing "astronaut", while ImageReward as in $\mathcal{V}_{mix}(I)$ critiques stylistic inconsistencies *e.g.*, unrealistic lunar textures. RIT balances both aspects effectively.

While our approach outperforms baselines, the absolute $\mathcal{T}_{mix} = 0.80$ (80% coverage) reveals room for improvement. Failures often occur in prompts with 5+ entities, where latent-space conflicts lead to partial omissions *e.g.*, a "cat astronaut riding a horse on the moon" might exclude the "book" from the prompt. This suggests that while ELNP improves scalability, highly dense counterfactual scenes still require better disentanglement mechanisms.

6. Conclusion

Replacing step-by-step from a normal image is a natural idea for counterfactual T2I. With adding question check

mechanism, this idea makes a boost of performance on multiple entities counterfactual T2I. We define the counterfactual T2I problems concentrated on entity numbers and design a method to generate descriptions for counterfactual scenes to construct datasets for specific concept numbers. A new evaluation metric is also devised addressing the deficit in existing metrics. The strategy RIT is finally proposed to address the problems with ELNP and question blocks, boosting T2I models' alignment of the counterfactual text and the generated image by replacing entities from a corresponding ordinary scene step-by-step. Similar ideas can be used on other controllable models for counterfactual T2I in the future.

References

- [1] Dana Arad, Hadas Orgad, and Yonatan Belinkov. Refact: Updating text-to-image models by editing the text encoder, 2024. 1
- [2] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions, 2023. 1
- [3] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojuan Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuan Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanbiao Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yudian Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. 1, 7
- [4] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. 1
- [5] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, and Jie Tang. Cogview: Mastering text-to-image generation via transformers, 2021. 1
- [6] Sandra Zhang Ding, Jiafeng Mao, and Kiyoharu Aizawa. Training-free sketch-guided diffusion with latent optimization, 2024. 3
- [7] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024. 2, 7, 8
- [8] Songwei Ge, Taesung Park, Jun-Yan Zhu, and Jia-Bin Huang. Expressive text-to-image generation with rich text, 2025. 1
- [9] Siqi Gu, Chunrong Fang, Quanjun Zhang, and Zhenyu Chen. Actesting: Automated cross-modal testing method of text-to-image software, 2025. 1
- [10] Inhwa Han, Serin Yang, Taesung Kwon, and Jong Chul Ye. Highly personalized text embedding for image manipulation by stable diffusion, 2023. 1
- [11] Xingzhe He, Zhiwen Cao, Nicholas Kolkin, Lantao Yu, Kun Wan, Helge Rhodin, and Ratheesh Kalarot. A data perspective on enhanced identity preservation for diffusion personalization, 2024. 2, 3
- [12] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning, 2022. 4, 7
- [13] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. 1
- [14] Maxwell Jones, Sheng-Yu Wang, Nupur Kumari, David Bau, and Jun-Yan Zhu. Customizing text-to-image models with a single image pair, 2024. 1
- [15] Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. Text-to-image generation grounded by fine-grained user attention, 2021. 1
- [16] Zeqiang Lai, Xizhou Zhu, Jifeng Dai, Yu Qiao, and Wenhai Wang. Mini-dalle3: Interactive text to image by prompting large language models, 2023. 1
- [17] Minhyeok Lee and Junhee Seok. Controllable generative adversarial network, 2019. 1
- [18] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip H. S. Torr. Controllable text-to-image generation, 2019. 2
- [19] Youwei Liang, Junfeng He, Gang Li, Peizhao Li, Arseniy Klimovskiy, Nicholas Carolan, Jiao Sun, Jordi Pont-Tuset,

- Sarah Young, Feng Yang, Junjie Ke, Krishnamurthy Dj Dvijotham, Katie Collins, Yiwen Luo, Yang Li, Kai J Kohlhoff, Deepak Ramachandran, and Vidhya Navalpakkam. Rich human feedback for text-to-image generation, 2024. 1
- [20] Kevin Lin, Zhengyuan Yang, Linjie Li, Jianfeng Wang, and Lijuan Wang. Designbench: Exploring and benchmarking dall-e 3 for imagining visual design, 2023. 1
- [21] Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. Generating images from captions with attention, 2016. 1
- [22] Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models, 2021. 1
- [23] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models, 2022. 1
- [24] William Peebles and Saining Xie. Scalable diffusion models with transformers, 2023. 1
- [25] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis, 2023. 1, 2, 7, 8
- [26] Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. Mirrorgan: Learning text-to-image generation by redescription, 2019.
- [27] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation, 2021. 2, 7, 8
- [28] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 1
- [29] Shachar Rosenman, Vasudev Lal, and Phillip Howard. Neuroprompts: An adaptive framework to optimize prompts for text-to-image generation, 2024. 3
- [30] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation, 2023. 2, 3
- [31] Nataniel Ruiz, Yuanzhen Li, Neal Wadhwa, Yael Pritch, Michael Rubinstein, David E. Jacobs, and Shlomi Fruchter. Magic insert: Style-aware drag-and-drop, 2024. 3
- [32] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022. 1
- [33] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. 1
- [34] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models, 2023. 1
- [35] Yanan Sun, Yanchen Liu, Yinhao Tang, Wenjie Pei, and Kai Chen. Anycontrol: Create your artwork with versatile control on text-to-image generation, 2024. 1, 2, 3
- [36] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning, 2018. 1
- [37] Chao Wang, Giulio Franzese, Alessandro Finamore, Massimo Gallo, and Pietro Michiardi. Information theoretic text-to-image alignment, 2025. 1
- [38] Xiaoshi Wu, Yiming Hao, Manyuan Zhang, Keqiang Sun, Zhaoyang Huang, Guanglu Song, Yu Liu, and Hongsheng Li. Deep reward supervisions for tuning text-to-image diffusion models, 2024. 1
- [39] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. Sana: Efficient high-resolution image synthesis with linear diffusion transformers, 2024. 1, 2, 7, 8
- [40] Tianwei Xiong, Yue Wu, Enze Xie, Yue Wu, Zhenguo Li, and Xihui Liu. Editing massive concepts in text-to-image diffusion models, 2024. 3
- [41] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation, 2023. 7
- [42] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. Scaling autoregressive models for content-rich text-to-image generation, 2022. 1
- [43] Zhongqi Yue, Tan Wang, Hanwang Zhang, Qianru Sun, and Xian-Sheng Hua. Counterfactual zero-shot and open-set visual recognition, 2021. 2, 3
- [44] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 1, 3, 5, 7
- [45] Juntu Zhao, Junyu Deng, Yixin Ye, Chongxuan Li, Zhijie Deng, and Dequan Wang. Lost in translation: Latent concept misalignment in text-to-image diffusion models, 2024. 1, 2, 3, 4, 7, 8

Replace in Translation: Boost Concept Alignment in Counterfactual Text-to-Image

Supplementary Material

7. Appendix

7.1. ELNP Prompt Templates

To prepare for the third step of generating ELNP, we need to provide the previous two steps for DEEPSEEK-R1 to enhance its understanding on ELNP process. Here we present the prompts of the first step: extracting the key entities in the given prompt in Fig. 6 and the second step: analyze the status of these key entities in Fig. 7.

DEEPSEEK-R1
You are a sentence parser, please extract all the objects in the prompt according to the exemplar below: "
Exemplar: Prompt: {A cat astronaut is riding a horse running on the moon.} All the objects: {cat, astronaut, horse, moon} "
Please give all the objects in the Prompt: {A butterfly is flying over a cup of coke.}

Figure 6. The prompt template to generate ELNP: extract key entities. In this example, we ask DEEPSEEK-R1 to process *a butterfly is flying over a cup of coke*.

DEEPSEEK-R1
You are a sentence parser and status analyzer, please extract all the objects in the prompt and the status of each object according to the exemplar below: "
Exemplar: Prompt: {A cat astronaut is riding a horse running on the moon.} All the objects: {cat, astronaut, horse, moon} "
The status: { cat: [astronaut, riding a horse], astronaut: [cat, riding a horse], horse: [being ridden, running], moon: [there is a horse running on it] } "
Please give all the objects and the status in the Prompt: {A butterfly is flying over a cup of coke.}

Figure 7. The prompt template to generate ELNP: analyze the status of key entities.

7.2. Question Blocks

When the language model gives the key entities as in Fig. 6, the full questions S_Q should be naturally generated for question blocks. After the ELNP order of replacements is generated, the questions set for each period can be generated naturally. Toward the exemplar in Fig. 4, the questions in each step should be as in Fig. 8.

For question blocks, if we set the threshold of the pass check to less or greater than 60%, the performance will be clearly weakened. There are two strategies to set the

QUESTION BLOCKS
The entity number is {3}?
If a {horse} exists in the image?

The entity number is {3}?
If a {horse} exists in the image?
If a {astronaut} exists in the image?

The entity number is {3}?
If a {horse} exists in the image?
If a {astronaut} exists in the image?
If a {cat} exists in the image?

The entity number is {3}?
If a {horse} exists in the image?
If a {astronaut} exists in the image?
If a {cat} exists in the image?
If a {moon} exists in the image?

Figure 8. The questions for each period can be generated from the key entities and the order.

threshold-all steps the same and lowering the threshold as the generation proceeds. We conduct the experiments to demonstrate that 60% is the best threshold. We utilize 2, 3, 5 and mixed-counterfactual problems experiment on our model with pass check threshold set to 20%, 40%, 60%, 80%, 100% and dynamic threshold. For dynamic threshold, the value in the first step is set to 100% and lowered by 20% each step. For 5-entity tasks, no more than seven times replacing happens, which means lowering down by 20% each step makes the threshold cover a wide range from 100% to 0. The experimental results are shown in Tabs. 3 and 4.

Threshold	\mathcal{D}_C (\downarrow)	\mathcal{D}_I (\downarrow)	\mathcal{T}_2 (\uparrow)
20%	1.98	1.92	0.32
40%	1.89	1.79	0.36
60%	0.24	0.62	0.91
80%	1.82	1.68	0.46
100%	-	-	-
dynamic	0.86	0.72	0.77

Table 3. The performance on 2-counterfactual problems under different threshold values in question blocks' pass check. The best results are in boldface. We can see fixed 60% is the best choice and dynamic method outperforms many fixed ways. The 100% method turns to collapse in loop with no results.

Threshold	$\mathcal{V}_3(C) \downarrow$	$\mathcal{V}_3(I) \downarrow$	$\mathcal{T}_3 \uparrow$	$\mathcal{V}_5(C) \downarrow$	$\mathcal{V}_5(I) \downarrow$	$\mathcal{T}_5 \uparrow$	$\mathcal{V}_{mix}(C) \downarrow$	$\mathcal{V}_{mix}(I) \downarrow$	$\mathcal{T}_{mix} \uparrow$
20%	1.87	1.74	0.29	2.01	1.78	0.26	1.91	1.83	0.27
40%	1.56	1.48	0.49	1.56	1.42	0.30	1.66	1.53	0.39
60%	0.54	0.68	0.84	0.95	0.92	0.45	0.74	0.96	0.62
80%	0.75	0.88	0.72	1.36	1.39	0.42	1.05	1.16	0.43
100%	-	-	-	-	-	-	-	-	-
dynamic	0.92	0.86	0.68	1.46	1.33	0.28	1.44	1.32	0.51

Table 4. The performance on 3, 5 and mixed-counterfactual problems under different threshold values in question blocks’ pass judge. 100% setting leads to collapses in loop in all situations. The best results are in boldface. We can see fixed 60% is the best choice while dynamic method outperforms many fixed ways.