Decoding Correlation-Induced Misalignment in the Stable Diffusion Workflow for Text-to-Image Generation

Yunze Tong Fengda Zhang Didi Zhu Jun Xiao Kun Kuang†
Zhejiang University
Hangzhou, Zhejiang, China
{tyz01, fdzhang, didi_zhu, junx, kunkuang}@zju.edu.cn

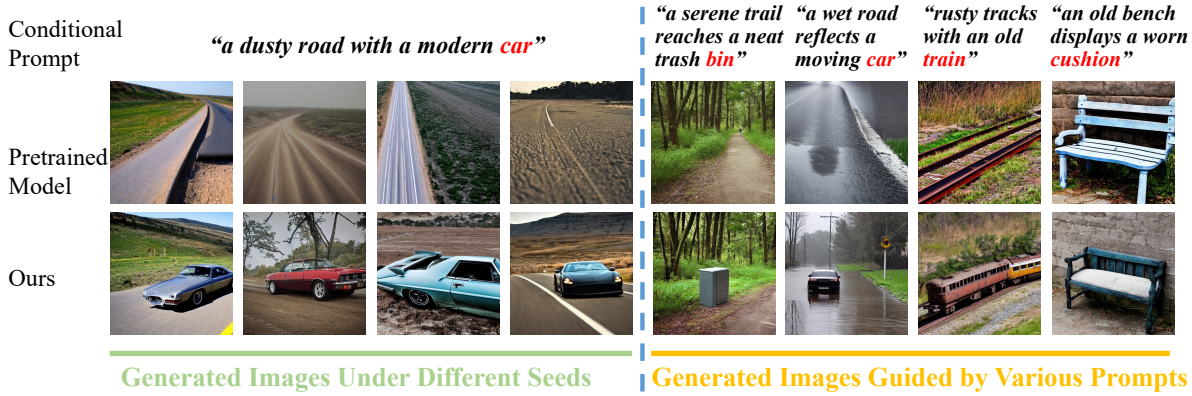


Figure 1. The comparison among images guided by various correlated texts. In the first row, columns one to four show that the pre-trained model fails to generate the object indicated by the red token in the prompt, even with different seeds. Columns five to eight further confirm that this issue occurs across various correlated objects. In contrast, our method could prevent object omission, as shown in the second row.

Abstract

The fundamental requirement for text-to-image generation is aligning the generated images with the provided text. With large-scale data, pre-trained Stable Diffusion (SD) models have achieved remarkable performance in this task. These models process an input prompt as text control, guiding a vision model to perform denoising operations that recover a clean image from pure noise. However, we observe that when there is correlation among text tokens, SD's generated images fail to accurately represent the semantics of the input prompt: simple yet crucial objects may be omitted, thereby disrupting text-image alignment. We refer to this problem as "object omission". Without additional external knowledge, previous methods have been ineffective at addressing this issue. To investigate this problem, we analyze the attention maps in SD and find that biased text representations mislead the visual denoising process when handling correlated tokens, impeding object generation. Moreover, we observe that even when two prompts share the same semantics, slight variations in token sequence significantly alter attention scores, consequently affecting the final generated images. Based on these findings, we propose a simple yet effective fine-tuning method that applies decorrelation to the self-attention maps in the text module, thus reducing dependencies between tokens. Our approach requires no external prior knowledge, is straightforward to implement, and operates solely on the text module of the SD model. Extensive experiments confirm that our method effectively alleviates the object omission problem under text correlations, thereby enhancing text-image alignment. ‡

1. Introduction

Text-to-image (T2I) generation involves creating images that reflect textual prompts, requiring a high degree of alignment between the text and the visual output. Stable Diffusion (SD) [29] has achieved remarkable performance in this area, yet it still faces challenges with text-image misalignment for certain prompts. Most previous methods incorporate external prior knowledge, such as human-provided reference images or auxiliary models, to refine image details and stylistic fidelity [6, 18, 26, 39, 43, 45, 46]. While these approaches appear promising, they rest on an implicit yet critical assumption: *the pre-trained model is capable of generating all objects referenced in the textual prompt.*

Strikingly, our findings reveal that this assumption does not hold when handling text prompts involving objects that are semantically corrrelated. To explore this issue, we gen-

† Corresponding author.
‡ Code: https://github.com/YunzeTong/DecorrelationDiffusion.

(a) an old train with rusty tracks (b) rusty tracks with an old train (c) an old train (d) rusty tracks

Figure 2. The comparison of generated image from pre-trained model under different prompts' control. Only Figure 2b fails to present all the objects in the given prompt due to correlation.

erated prompts using GPT-4 [1] and sampled corresponding images with SD. As shown in the first row of Figure 1, our analysis indicates that SD frequently fails to present all referenced objects in these cases, resulting in a text-image misalignment we refer to as the *object omission* problem. Since SD itself even fails to generate basic layouts for required semantics, previous methods focusing solely on detail and style optimization are insufficient to address the issue.

This study investigates how improper correlation causes misalignment in SD's workflow. SD consists of two modules: a text module that encodes the input prompt as text control, and a vision module that generates images through iterative denoising operations. Our experiments reveal that object omission primarily stems from biases in the text module, rather than vision module's incapability to present objects. For example, while the vision module can independently generate *"train"* and *"tracks"* (as shown in Figures 2c and 2d), the text module tends to produce correlated latent representations when processing prompts with correlated objects like *"rusty tracks with an old train"*. These correlated latents reflect token semantic overlap and bias the vision module toward one object. During denoising, SD prioritizes the dominant object, causing *"train"* omission in Figure 2b. In summary, **correlated text input disrupts the denoising process**, preventing certain object from being generated. For further analysis, please refer to Section 4.2.

We further investigate an intriguing observation in the text module's behavior: token order in a prompt directly influences whether all objects appear in the image. As shown in Figures 2a and 2b, swapping the positions of *"train"* and *"tracks"* in the prompt results in either the appearance or the omission of *"train"*. This effect arises from the self-attention mechanism, where different token arrangements modify how tokens attend to each other, leading to varying degrees of correlation reflected in attention scores. Higher attention scores imply stronger correlations between tokens, which may cause one token to be more semantically dominated by another, potentially biasing the generation and causing object omission as shown in Figure 2b. Conversely, a token order that induces moderate attention scores allows all objects to be generated consistently (see Table 1).

Motivated by this insight, we propose a simple yet effective method to avoid the object omission problem without relying on prior knowledge, thereby enhancing text-image alignment. Our approach involves reducing the attention scores from the highest correlated token to the target token representing the omitted object. We then select fine-tuned model checkpoints based on two criteria: (1) the extent of decorrelation achieved between the focused tokens, and (2) the preservation of interactions among irrelevant tokens. Finally, we improve the sampling guidance by incorporating both our fine-tuned model and the original pre-trained version to strengthen decorrelation. These adjustments effectively reduce correlation among key tokens while maintaining the attention patterns of unrelated ones.

To summarize, our contributions are listed as follows:

- We identify the issue that correlations between text tokens introduce *object omission* in T2I generation, which previous methods cannot address without prior knowledge.
- We conduct a comprehensive analysis and find that the problem stems from the text module in SD, which is typically derived from CLIP [27] and exhibits bias.
- We propose a simple yet effective method that adjusts self-attention maps in the text module without relying on external knowledge, thereby successfully generating omitted objects and enhancing text-image alignment.

2. Related Works

Diffusion Models. Diffusion models (DMs) [8, 15] have achieved success in text-to-image generation and other areas [34, 41]. When paired with a UNet or transformer backbone, DMs can sample a clean image progressively from a noisy input. Previous works [32] further described the estimated noise during the denoising process as the score, *i.e.*, the gradient of the log-likelihood of the transformed distribution. Latent diffusion models [29] operate in a compressed latent space, enabling efficien training on large-scale data. In addition, improvements in sampling techniques have also accelerated sampling speeds. For example, DDIM [31] applies a non-Markovian process, which enhances denoising efficiency.

Attention-based Methods for Diffusion Models. The attention mechanism [35] has proven to be highly effective in modeling complex input dependencies and is widely employed in DMs. The attention map, which represents the multiplication matrix of the query and key, contains rich representations and can partially reflect semantic consistency. As a result, many studies leverage this map for specific purposes. For example, self-attention maps in UNet are commonly used to construct desired spatial layouts or ensure the generation of specified objects [2, 4, 5, 16, 19, 22]. In addition, cross-attention maps are crucial for linking text information with visual pixels. Several methods [12, 26, 39, 44] enhance this connection by modifying the cross-attention modules, thereby improving the generation of images with desired properties for downstream tasks.

3. Preliminaries

3.1. Latent Diffusion Models

Latent Diffusion Models (LDMs) first compress images into latent representations using a pre-trained variational autoencoder (VAE). The diffusion process is then applied in the latent space for efficient modeling. Training a diffusion model consists of a forward and a backward process. The forward process adds Gaussian noises to the clean sample $\mathbf{x}_0$, resulting in a noisy sample $\mathbf{x}_t$. The distribution of $\mathbf{x}_t$ conditioned on $\mathbf{x}_0$ is given by $q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}}_t\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})$, where t represents the timestep and controls the noise scale via $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$. The backward process aims to recover $\mathbf{x}_0$ from the noisy sample $\mathbf{x}_t$ using $p(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_t(\mathbf{x}_t, t), \sigma_t^2\mathbf{I})$, where $\mu_t(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon)$, $\sigma_t^2 = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$, and $\beta_t = 1-\alpha_t$. Here, ϵ denotes the noise, which is the only uncertain variable. A neural network ϵ_θ is trained to predict ϵ, minimizing the objective: $\mathcal{L} = \mathbb{E}_{\mathbf{x}_0,t,\epsilon\sim\mathcal{N}(0,\mathbf{I})} \|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|_2^2$, for $t \in \{1, ..., T\}$. The sampling process begins with pure noise $\mathbf{x}_T$. The noise at different scales is progressively estimated and subtracted until the clean sample $\mathbf{x}_0$ is reconstructed. In addition, researchers often include the prompt $\mathbf{c}$ as an input to ϵ_θ to integrate text information during the denoising process, resulting in $\epsilon_\theta(\mathbf{x}_t, t, \mathbf{c})$.

3.2. Sampling Guidance

To generate samples with specific labels, classifier-free guidance (CFG) [14] is widely used to conduct efficient text control. CFG utilizes an unconditional denoising diffusion model $p_\theta(\mathbf{x}_t)$ parameterized by $\epsilon_\theta(\mathbf{x}_t, t)$, alongside a conditional model $p_\theta(\mathbf{x}_t|\mathbf{c})$ parameterized by $\epsilon_\theta(\mathbf{x}_t, t, \mathbf{c})$. A single neural network is used to parameterize both models, with a null token $\varnothing$ input for the class identifier in the unconditional model. The final score is given by:

$$\tilde{\epsilon}_\theta(\mathbf{x}_t, t, \mathbf{c}) = (1+w)\epsilon_\theta(\mathbf{x}_t, t, \mathbf{c}) - w\epsilon_\theta(\mathbf{x}_t, t). \quad (1)$$

Eq. 1 allows CFG to be easily adapted by modifying the details of the terms involved. Many works [2, 16, 20] have adapted it as a form of sampling guidance for various purposes. Regarding why CFG and its variants significantly improve image generation, recent works [20] demonstrated that CFG enhances image quality beyond simply applying conditional control. The improvement is also attributed to its adaptive truncation, which guides the sample towards higher-quality outputs. In this case, even an inferior version of the model itself could help guide sampling.

3.3. Attention Modules in Stable Diffusion

Stable Diffusion models [29] are a subset of latent diffusion models. In this study, we adopt the SD model as our backbone and focus on addressing issues within its workflow. This section provides a brief overview of the attention modules in SD, which will serve as a foundation for the problem analysis and proposed method in Sections 4 and 5. The Transformer architecture [35] has demonstrated the capability of attention mechanisms to model complex dependencies and semantic relationships across inputs. Three distinct attention blocks contribute to SD: (1) The self-attention map in the text encoder captures interactions between text-level tokens. Since SD's text encoder is typically derived from the CLIP model [27], it does not undergo the same pre-training process as the UNet in the vision model. Additionally, attention probabilities for each token are assigned only to tokens that precede it in the sequence. (2) The self-attention map in vision modules reflects the relationships between image features while preserving spatial information. Several methods [2, 16, 22], particularly for image-editing tasks, exploit it to refine the layout and achieve desired outcomes. (3) The cross-attention map plays a critical role in linking texts with image content, enabling SD to align prompt tokens with corresponding image areas [22]. Many studies [12, 26, 39, 44] leverage cross-attention maps to refine the semantics of generated images.

4. Problem Settings and Analysis

4.1. Misalignment Induced by Text Correlation

In the text-to-image generation task, the primary objective is to achieve alignment between the provided text descriptions and the generated images. As a result, many studies [2, 4, 23, 26, 30, 42, 44] have focused on enhancing this alignment by improving the detail of the generated images, such as using prior information [10, 38] to refine the style or spatial arrangement. These approaches often rely on a key assumption: the pre-trained Stable Diffusion (SD) model possesses the ability to generate coarse yet accurate layouts, particularly for noun objects specified in the prompt. Building on this assumption, these methods further achieve superior text-image alignment. However, we raise a critical question regarding the inherent capabilities of SD: Are the images produced by SD always capable of **correctly conveying the semantics** required by the provided prompt?

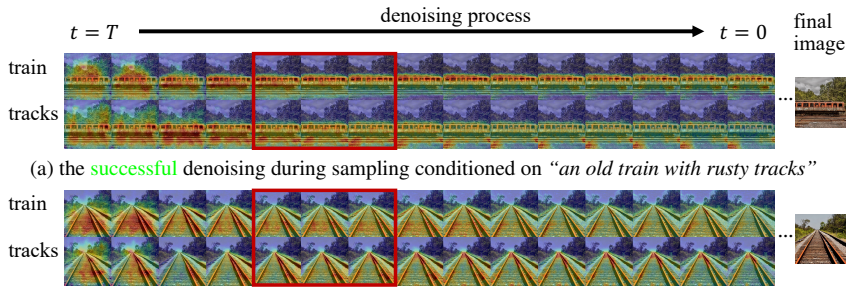
To evaluate whether the pre-trained SD model consistently captures the fundamental semantics of a given text, we conducted experiments using various prompts generated by GPT-4 [1]. These prompts were composed of common words and free from counterfactual scenarios. Despite this, we observed a recurring issue: in the first row of Figure 1, generated images often fail to fully depict even the key, simple words in the prompt. A notable feature of these problematic prompts is the presence of a word that is highly correlated with another word. In such cases, text-image alignment degrades significantly. In other words, **misalignment often involves with the correlation among input texts**. The most apparent manifestation of this issue is the omission of essential objects referenced in the prompt. Furthermore, we observed that for such prompts, the input text typ-

"train"	*"tracks"* (itself)	Ratio
0.0444	0.0769	$\frac{0.0444}{0.0769} \approx 0.57$

Table 1. The attention probability from a given token to *"tracks"* in *"an old train with rusty tracks"*.

"tracks"	*"train"* (itself)	Ratio
0.0611	0.0550	$\frac{0.0611}{0.0550} \approx 1.11(\uparrow)$

Table 2. The attention probability from a given token to *"train"* in *"rusty tracks with an old train"*.



(a) the successful denoising during sampling conditioned on *"an old train with rusty tracks"*

(b) the unsuccessful denoising during sampling conditioned on *"rusty tracks with an old train"*

Figure 3: The cross-attention probabilities with *"train"* (*"tracks"*) token at different noise levels. At the end of denoising process ($t = 0$), Figure 3a clearly demonstrates two objects while the cross-attention scores are similar and the object *"train"* has not been generated in Figure 3b.

ically exhibits a unidirectional correlation between its two nouns: one token is strongly correlated with another from a human perspective, while the reverse correlation does not always hold. For example, when prompted with *"rusty tracks with an old train"*, SD generates tracks but omits any reference to the train in Figure 2b. In real-world scenarios, trains are always associated with tracks because they cannot operate without them. However, tracks are not necessarily associated with trains; it is common for tracks to be empty, with no train present. We describe this relationship as a unidirectional correlation—it holds from one object to another but does not necessarily apply in the opposite direction.

In summary, we observe that misalignment often arises from correlations among input text tokens. In this study, we focus on one of its most challenging manifestations: the object omission problem caused by unidirectional correlation between two tokens in the input prompt. Without extra prior knowledge, previous methods are unable to address this issue, as the SD model itself fails to generate the fundamental semantics correctly. In such cases, techniques for refining details become ineffective. Our goal is to investigate how this problem arises and propose an approach to enhance text-image alignment without relying on prior knowledge.

4.2. Analysis based on the Attention Modules

SD consists of two modules: the text module and the vision module. The text module processes the input prompt and provides conditional control to the vision module, which then performs denoising operations. In this section, we review the workflow of SD to identify the root cause of the object omission problem referred in Section 4.1. We will then analyze and explain the underlying mechanism.

Is the vision module in the diffusion model unable to generate the specific missing object? The object omission issue arises when there is a correlation among prompt tokens. To explore whether this problem persists when there is no correlation, we investigate whether the incapability of SD itself leads to the failure. For each prompt, we split it into two separate prompts, each containing only one object, and conduct sampling. We observe that SD can generate each object successfully in separate images. For example, in Figures 2c and 2d, the words *"train"* and *"tracks"* are each accurately represented. This confirms that the vision module is capable of constructing essential objects, as otherwise, the object layout would not appear in any image.

Does the text module convey correlated semantics? The key difference between Figures 2a and 2b lies in the conditional control provided by the text input. Having established that the object omission issue does not stem from the vision module's limitations in SD, we now turn to the text module as the source of misalignment caused by correlation. To investigate whether the text module introduces such correlations into its representation, we conduct a quantitative analysis by examining its self-attention maps. The text module in our base SD model is derived from CLIP [27]. As explained in Section 3.3, its self-attention map is generated by multiplying the query and key matrices. The tokens are processed in an auto-regressive manner, with the attention maps of each token assigned only to tokens preceding it in the sequence. As a result, the values in each row form a complete probability distribution, reflecting the extent of correlation between different tokens.

We demonstrate the generation results of two prompts with the same meaning: *"rusty tracks with an old train"* and *"an old train with rusty tracks"*. Both prompts contain the same object nouns, *"train"* and *"tracks"*. However, as shown in Figure 2b, which is conditioned on *"rusty tracks with an old train"*, the model fails to generate the train. In contrast, Figure 2a correctly generates both the train and tracks, aligning well with the prompt. To investigate further, we extract the self-attention probabilities and examine the attention values between the tokens representing *"train"* and *"tracks"* in Table 1 and 2. From Table 2, we observe that the attention value from *"tracks"* to *"train"* is significantly higher than the attention from *"train"* to itself, indicating a unidirectional correlation that affects the model's performance. In the successful prompt, the attention from *"train"* to *"tracks"* is moderate, corresponding to the correct generation. In summary, unidirectional correlations influence the attention weights between tokens, intro-

ducing bias into the text representation of the prompt and ultimately causing the misalignment problem.

How does the bias in text module disturb the denoising process? In this part, we aim to reveal how the bias in text representation, which arises from unidirectional correlations in the prompt, affects the vision module's denoising process. We begin by observing the behavior of the cross-attention layers. As discussed in Section 3.3, cross-attention maps align the semantics of tokens with image pixels. We visualize the cross-attention maps for the two tokens, with pixels receiving higher attention assigned warmer colors. To illustrate the evolution of the attention process, we select several denoising timesteps, as shown in Figure 3. At the initial denoising stage ($t = T$), the cross-attention scores are randomly distributed, as the image is pure noise. As denoising progresses, the cross-attention scores for different tokens evolve. In Figure 3a, the attention on *"tracks"* and *"train"* gradually concentrates on different spatial areas of the image, as indicated by the red box. In contrast, in Figure 3b, the attention scores for the two tokens are almost identical. This is due to the correlation between the two tokens. Generating the framework for *"tracks"* inadvertently reflects the semantics of *"train"*, preventing the model from generating *"train"* independently. By the end of the denoising process ($t = 0$), Figure 3a shows both objects, while Figure 3b does not. In the latter case, the attention weights for *"train"* are nearly identical to those for *"tracks"*, illustrating how the correlation between tokens influences the denoising process. One object's semantics effectively replace the other's, leading to the omission of the latter.

The text module in SD introduces bias and outputs correlated representations for correlated tokens. This bias propagates into the vision module, leading to object omission. From the biased model's view, the representations of correlated tokens are inherently similar. Losing one object does not disrupt alignment because the other object still conveys the missing object's semantics. However, from a human perspective, this results in severe misalignment. Fortunately, we observe that simply changing the sequence of tokens, as shown in Tables 1 and 2, reduces the correlation and significantly improves text-image alignment. In the next section, we propose decorrelating tokens at the text level to enhance the image generation process.

5. Method

5.1. Decorrelation-based Fine-tuning

We have demonstrated that the bias introduced by token correlations presents challenges for text-image alignment. This bias cannot be alleviated without prior information if operations are limited to the vision modules of a diffusion model. However, we also show that the bias in the text encoder is often accompanied by improper attention weights, primarily due to one token being over-correlated with another. As a result, we focus on fine-tuning the self-attention blocks in the text module to address the misalignment issue.

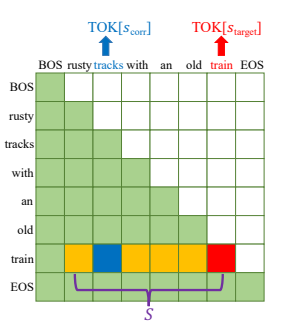


Figure 4. The illustration of an averaged self-attention map during fine-tuning.

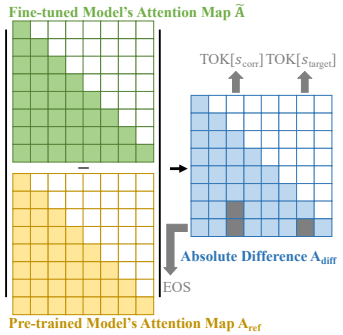


Figure 5. The computation process for A_{diff} during model selection.

We use TOK to denote the token list after tokenization, where TOK[0] refers to the BOS token. As shown in Figure 4, we extract the self-attention map from all heads and compute their mean, denoted as A. $A[i, j]$ represents the attention probability from the j-th token to the i-th token. s_{target} refers to the index of the token that aligns with the missing object (highlighted in the red block). We then define a set $\mathcal{S} = \{s \mid 1 \leq s \leq s_{\text{target}}\}$, which is shown in gold in Figure 4. $\mathcal{S}$ contains the indices of all tokens that precede the target token TOK[s_{target}], excluding the BOS token. Within $\mathcal{S}$, we select the token with the highest attention value on TOK[s_{target}] and denote it as TOK[s_{corr}]. In Section 4.2, we demonstrated that high attention values often indicate bias and correlation. Therefore, we reduce $A[s_{\text{target}}, s_{\text{corr}}]$ by applying the following loss function:

$$\mathcal{L}_{\text{train}} = A[s_{\text{target}}, s_{\text{corr}}] - \min(A[s_{\text{target}}, \mathcal{S}]). \quad (2)$$

We define Eq. 2 as a subtraction to prevent $A[s_{\text{target}}, s_{\text{corr}}]$ from becoming zero. Though Eq. 2 changes the original attention probabilities, it maintains the interaction among unfocused tokens as possible instead of forcing the specific attention probability to a constant value. As the fine-tuning progresses, the optimized model checkpoints reflect varying levels of decorrelation, from biased to debiased.

5.2. Model Selection

In Section 5.1, we defined an optimization process that reduces the unidirectional correlation between tokens by adjusting the self-attention weights. The next step is selecting the best checkpoint for final sampling. We denote the self-attention map of the fine-tuned text encoder as $\tilde{A}$. In this section, we propose a criterion to evaluate the fine-tuned model using $\tilde{A}$, which considers two main factors.

Decorrelation between the focused tokens should be achieved. To address the object omission problem caused by bias, the correlation between TOK[t_{corr}] and TOK[t_{target}] should be adequately reduced. We use the following crite-

rion to measure the extent of this correlation:

$$\mathcal{L}_{\text{corr}} = \log \frac{\widetilde{\mathrm{A}}[s_{\text{target}}, s_{\text{corr}}]}{\widetilde{\mathrm{A}}[s_{\text{target}}, s_{\text{target}}]}. \tag{3}$$

Smaller values of $\mathcal{L}_{\text{corr}}$ indicates a larger gap between the attention values, signifying more effective decorrelation.

The relationships among irrelevant tokens should be preserved and aligned with the pre-trained version as possible. This goal ensures that only the attention probabilities for the focused tokens are adjusted, while the relationships among other tokens remain intact. As a result, the text control sent to the vision module guides the generation of the missing object without disrupting the overall image structure. To assess the preservation, we use the attention map from pre-trained model A_{ref} as a reference. The absolute difference between the fine-tuned attention map and the reference is denoted as $\mathrm{A}_{\text{diff}} = |\widetilde{\mathrm{A}} - \mathrm{A}_{\text{ref}}|$. We then select the blue entries from A_{diff} in Figure 5 and denote them as $\mathrm{A}'_{\text{diff}}$. We exclude three specific gray entries , since they reflect the attention values for the focused tokens. The excluded entries are subject to change during fine-tuning and do not represent the behavior of irrelevant tokens. Finally, we measure preservation by taking the maximum value from $\mathrm{A}'_{\text{diff}}$:

$$\mathcal{L}_{\text{align}} = \max(\mathrm{A}'_{\text{diff}}). \tag{4}$$

We note that there is a trade-off between the two criteria. Modifying the attention probability between focused tokens inevitably affects the relationships among others. Therefore, we combine them into a single evaluation term:

$$\mathcal{L}_{\text{val}} = \mathcal{L}_{\text{corr}} + \tau \mathcal{L}_{\text{align}}. \tag{5}$$

We use $\mathcal{L}_{\text{val}}$ to evaluate each checkpoint during optimization, with τ controlling the balance between the two terms.

5.3. Sampling Guidance

After fine-tuning the models, we proceed to sample images. As introduced in Section 3.2, classifier-free guidance (CFG) is commonly used for sampling. Eq. 1 controls the generation process by varying the conditional input and the negative unconditional input. Some studies [2, 20] interpret CFG's improvement as truncation that pushes the sample towards higher-quality outputs. We follow their ideas for further decorrelation. To be specific, we use our fine-tuned model $\widetilde{\theta}$ to provide the conditional control $\epsilon_{\widetilde{\theta}}(\mathbf{x}_t, t, \mathbf{c})$, while using the pre-trained model for the negative embedding $\epsilon_\theta(\mathbf{x}_t, t)$. The predicted noise is given by:

$$\tilde{\epsilon}_{\theta,\widetilde{\theta}}(\mathbf{x}_t, t, \mathbf{c}) = (1+w)\epsilon_{\widetilde{\theta}}(\mathbf{x}_t, t, \mathbf{c}) - w\epsilon_\theta(\mathbf{x}_t, t). \tag{6}$$

The idea behind this approach is to increase the difference between the pre-trained text embedding and our fine-tuned version. The unconditional control from the biased pre-trained model acts as negative guidance, while the less biased fine-tuned model provides better generation direction. Therefore, our sampling guidance not only improves text control but also strengthens decorrelation during the denoising process by differentiating between the models.

6. Experiments

6.1. Experimental Settings

Evaluation Metrics. Since our focus is on generating images that meet text requirements under correlations, alignment serves as the primary evaluation criterion for our method. We assess alignment from two perspectives:

- *Text-image alignment*: We use BERTScore introduced by DDPO [3, 17]. A description is generated for an image using a vision-language model. BERT's recall metric [7] is then used to measure the semantic similarity between the original prompt and the generated description. For implementation, we use the instruction-tuned 7B Qwen2-VL [36] as the vision-language model and the DeBERTa xlarge model [11] to obtain the representations of both the prompt and description for BERTScore calculation.
- *Image-image alignment*: In addition to text-level evaluation, we also assess alignment at the visual level. First, we collect images from other sources and manually select those that align well with the prompt. We then compute the Fréchet Inception Distance (FID) [13] between these filtered images and our generated ones. For implementation, we use Emu3 [37] to generate candidate images for selection. Emu3's generation process is based on next-token prediction, differing from the denoising process of diffusion models. Using different evaluation (autoregressive) and evaluated (diffusion) models helps avoid overlaps in image styles and layouts, ensuring the evaluation focuses more on semantic alignment.

Compared Baselines. We compare our method with several baselines: the pre-trained model, PAG [2], Attend-and-Excite (AaE) [5], and DisenDiff [44]. The pre-trained model serves as a fundamental baseline. PAG modifies self-attention maps in the diffusion U-Net by replacing them with an identity matrix during sampling. AaE utilizes cross-attention units to achieve patch separation for tokens. DisenDiff is a fine-tuning-based method that uses cross-attention maps in the diffusion U-Net to disentangle different objects. These methods do not rely on extra prior information, such as external models [24], human-provided reference images [42], or auxiliary tasks [33, 39], aligning with the problem setting in Section 4.1.[1] Although these methods all involve attention modules, they target different attention maps and focus on distinct training or sampling

[1]For DisenDiff, we use images generated by the pre-trained model as training data to avoid introducing extra information.

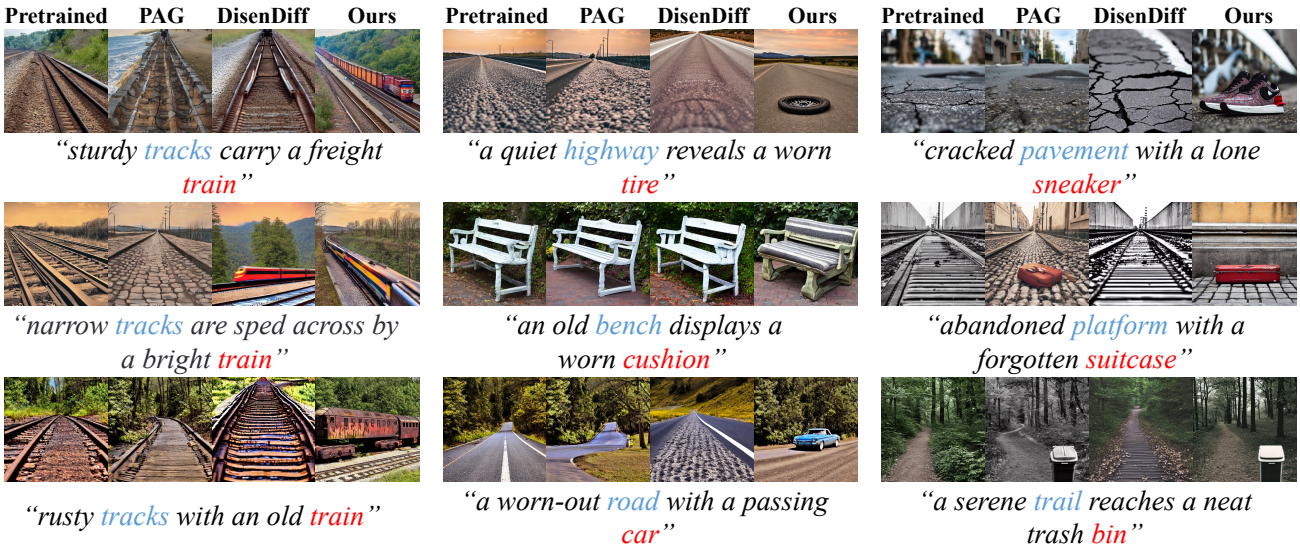


Figure 6. The qualitative comparison between generated images. The first column displays results for the same object pair with different decorations, while the remaining columns present different object pairs.

methods. Comparing our method with these baselines highlights the benefits of fine-tuning the self-attention module of the text encoder in diffusion models.

Implementation Details. Our experiments are conducted on an NVIDIA A100 GPU. We use Stable Diffusion V1.5[2] as the base model to generate 512 × 512 resolution images. The seed setting, which affects the initial noise during sampling, is kept the same when comparing image generation across different methods. For the implementation of our method, fine-tuning is performed on the self-attention maps extracted from `cond_stage_model.transformer.text_model.encoder.layers.11.self_attn`. The training consists of 20 epochs, with our validation criterion applied at each epoch to select the best model. Once the fine-tuned text encoder is obtained, we set $w = 7.5$ for sample guidance and $\tau = 5$ for the evaluation balance term. For other methods, we use the default hyperparameters recommended in their public repositories.

6.2. Comparison Results

Qualitative comparison. Figure 6 shows the generated images from our method and the baselines. There are two key observations. First, our approach, using a decorrelation strategy, successfully generates all the objects in the given prompt. In contrast, baseline methods often fail to generate at least one object due to unidirectional correlation, as discussed in Section 4. Second, the images generated by our method retain a layout structure similar to those from the pre-trained model. This confirms that our method achieves decorrelation without significantly altering the image structure. The vision module's capabilities in the pre-trained model are preserved. The only change in the samples is the successful reconstruction of the missing objects, further demonstrating that our $\mathcal{L}_{\text{align}}$ in Eq. 4 effectively preserves the relationships among irrelevant tokens.

Quantitative comparison. Table 3 shows the quantitative comparison for alignment. Higher BERTScore and lower FID indicate better text-image and image-image alignment. The overall comparison is presented in second to third columns. We collected 30 prompts generated by GPT-4 [1], each containing a correlated object pair. For each prompt, we generated 50 images and computed the average statistics. The results show that our method generalizes well to different correlated object pairs. Next, we examine whether our method can handle prompts with the same objects but different descriptions. For each correlated object pair, we created 3 different prompts by varying the verbs, adjectives, *etc*. We sampled 500 images for each prompt and computed the average for all 3×500 samples. The results in the forth to ninth columns of Table 3 show the effectiveness of our method across various prompt structures. Regardless of the variation in other words, our method successfully performs decorrelation to improve text-image alignment whenever unidirectional correlations are present.

User Study. Aside from model-based evaluation, we also conduct a human preference test. Two different prompts were selected for each object pair in Table 3, and images generated with five different seeds for each prompt were randomly chosen, resulting in a total of 3×2×5 comparisons. The results were randomly presented to the raters. Each comparison was rated on a 5-point scale, with 5 being the best. Participants evaluated two criteria: text-image alignment and image quality. The comparison results, shown in Table 4, demonstrate the overwhelming superiority of our method in text-image alignment, confirming that our samples align better with the prompts from a human perspec-

[2]https://huggingface.co/stable-diffusion-v1-5/stable-diffusion-v1-5

Methods	Overall		Object Pair 1		Object Pair 2		Object Pair 3	
	BERTScore ↑	FID ↓	BERTScore ↑	FID ↓	BERTScore ↑	FID ↓	BERTScore ↑	FID ↓
Pre-trained Model [29]	75.74 ±2.30	290.57	76.99 ±1.23	200.40	78.67 ±1.61	300.74	75.14 ±2.18	261.28
PAG [2]	75.41 ±2.23	276.77	76.74 ±1.50	188.46	78.12 ±1.83	278.86	74.72 ±2.26	235.27
AaE [5]	<u>75.97</u> ±1.69	<u>276.27</u>	76.94 ±1.26	<u>175.37</u>	78.98 ±0.95	<u>273.55</u>	<u>75.83</u> ±2.56	**155.90**
DisenDiff [44]	75.93 ±2.61	308.16	<u>77.07</u> ±1.05	217.59	<u>79.02</u> ±1.38	300.74	74.55 ±1.61	307.47
Ours	**76.35** ±2.44	**243.11**	**77.68** ±1.07	**137.29**	**79.68** ±1.49	**232.42**	**75.92** ±2.14	<u>170.79</u>

Table 3. The comparison for text-image alignment (denoted by BERTScore) and image-image alignment (denoted by FID). We omit the % for BERTScore and record the average standard deviation. The **bold** and <u>underline</u> denote the best and the second best results respectively.

tive. In addition, our images received high ratings for quality, indicating that decorrelation is achieved without sacrificing fidelity.

Method	Alignment ↑	Fidelity ↑
Pre-trained Model [29]	1.86 ±0.50	2.87 ±0.35
PAG [2]	1.81 ±0.51	2.81 ±0.47
DisenDiff [44]	1.98 ±0.75	2.53 ±0.57
Ours	3.60 ±0.56	3.79 ±0.40

Table 4. Text-image alignment and image fidelity evaluated by our user study. Higher rating implies better alignment with human.

6.3. Ablation Studies

6.3.1. The Effect of Our Sampling Guidance

In Section 5.3, we propose using our debiased model for conditional control and the pre-trained model for unconditional control. This approach aims to enhance decorrelation and guide the vision model towards better generation. The comparison with standard classifier-free guidance (CFG) is shown in Figure 7. As observed, CFG with the pre-trained biased model struggles to present all the objects in the left figures. The middle figures, generated by CFG with our fine-tuned model, produce some complete objects but fail to form valid spatial structures for others. In contrast, by applying our sampling guidance, which uses both models separately, decorrelation is enhanced, and all the figures on the right successfully generate all objects.

Figure 7. The comparison between the images generated w/ and w/o our sampling guidance.

6.3.2. The Sensitivity Analysis

In Section 5.2, we use Eq. 5 to select models. It consists of two terms: one representing the level of decorrelation among focused tokens and the other representing the preservation of relationships among irrelevant tokens. The balance between these terms is controlled by τ. We investigate the sensitivity of model selection to changes in τ. Note that $\mathcal{L}_{\text{val}}$ does not influence the fine-tuning process Different values of τ may still select the same model at certain stages. To test this, we use the prompt *"a peaceful trail stops at a trash bin"* and record the model epoch selected for different values of τ in Table 5. We could observe that our model selection strategy is insensitive to changes in τ.

τ	Epoch
1	12
3	11
5	11
7	11
10	9

Table 5. The ablation study on τ and its corresponding selected epoch.

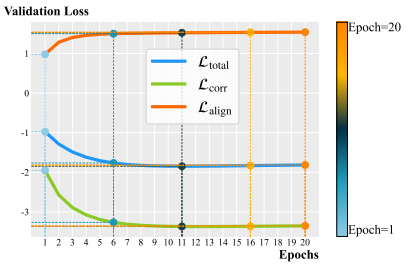


Figure 8. The loss curves of all the validation terms in Section 5.2.

We also plot the loss curve for all validation terms in Figure 8. As discussed in Section 5.2, there is a trade-off between $\mathcal{L}_{\text{corr}}$ and $\mathcal{L}_{\text{align}}$. As fine-tuning reaches the 11th epoch, we select the optimal model, which does not necessarily achieve the minimum loss for both terms. However, we observe that $\mathcal{L}_{\text{total}}$ follows a smooth trend, further confirming that our method is insensitive to τ.

7. Conclusion

In this paper, we analyze how token correlations induce text-image misalignment in Stable Diffusion (SD). Using the object omission problem as a case study, we examine how unidirectional correlation between tokens causes the omission of objects. We attribute this issue to the biased text representations generated by SD's text module. Under their biased guidance, the visual denoising process is misled, resulting in text-image misalignment. To address this issue, we propose a fine-tuning method based on the self-attention maps in the text module. Without any prior knowledge, our method could reconstruct objects that the pre-trained model fails to generate, thereby improving text-image alignment.

Acknowledgement

This work was supported in part by the National Key Research and Development Program of China (2024YFE0203700), National Natural Science Foundation of China (62376243, 62441617), "Pioneer" and "Leading Goose" R&D Program of Zhejiang (2025C02037), the Starry Night Science Fund of Zhejiang University Shanghai Institute for Advanced Study (SN-ZJU-SIAS-0010), and Key R&D Program of Zhejiang (2025C01128), a Fundamental Research Funds for the Central Universities (226-2025-00057). All opinions in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

We would also like to thank all anonymous reviewers for their insightful comments and helpful suggestions, which have significantly improved the quality of this work.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2, 3, 7

[2] Donghoon Ahn, Hyoungwon Cho, Jaewon Min, Wooseok Jang, Jungwoo Kim, SeonHwa Kim, Hyun Hee Park, Kyong Hwan Jin, and Seungryong Kim. Self-rectifying diffusion sampling with perturbed-attention guidance. In *European Conference on Computer Vision*, pages 1–17. Springer, 2024. 2, 3, 6, 8

[3] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023. 6

[4] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22560–22570, 2023. 2, 3

[5] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models, 2023. 2, 6, 8

[6] Omer Dahary, Or Patashnik, Kfir Aberman, and Daniel Cohen-Or. Be yourself: Bounded attention for multi-subject text-to-image generation, 2024. 1

[7] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 6, 14

[8] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2

[9] 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. 14

[10] Kairong Han, Kun Kuang, Ziyu Zhao, Junjian Ye, and Fei Wu. Causal agent based on large language model. *arXiv preprint arXiv:2408.06849*, 2024. 3

[11] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*, 2021. 6

[12] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 2, 3

[13] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 6

[14] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. 3

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2

[16] Susung Hong, Gyuseong Lee, Wooseok Jang, and Seungryong Kim. Improving sample quality of diffusion models using self-attention guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7462–7471, 2023. 2, 3

[17] Zijing Hu, Fengda Zhang, Long Chen, Kun Kuang, Jiahui Li, Kaifeng Gao, Jun Xiao, Xin Wang, and Wenwu Zhu. Towards better alignment: Training diffusion models with reinforcement learning against sparse rewards. *arXiv preprint arXiv:2503.11240*, 2025. 6

[18] Zijing Hu, Fengda Zhang, and Kun Kuang. D-fusion: Direct preference optimization for aligning diffusion models with visually consistent samples. *arXiv preprint arXiv:2505.22002*, 2025. 1

[19] Daqi Jiang, Hong Wang, Tan Li, Mohamed Amin Gouda, and Bin Zhou. Real-time tracker of chicken for poultry based on attention mechanism-enhanced yolo-chicken algorithm. *Computers and Electronics in Agriculture*, 237: 110640, 2025. 2

[20] Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself, 2024. 3, 6

[21] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024. 14

[22] Bingyan Liu, Chengyu Wang, Tingfeng Cao, Kui Jia, and Jun Huang. Towards understanding cross and self-attention in stable diffusion for text-guided image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7817–7826, 2024. 2, 3

[23] Zheqi Lv, Junhao Chen, Qi Tian, Keting Yin, Shengyu Zhang, and Fei Wu. Multimodal llm-guided semantic correction in text-to-image diffusion. *arXiv preprint arXiv:2505.20053*, 2025. 3

[24] Zheqi Lv, Tianyu Zhan, Wenjie Wang, Xinyu Lin, Shengyu Zhang, Wenqiao Zhang, Jiwei Li, Kun Kuang, and Fei Wu. Collaboration of large language models and small recommendation models for device-cloud recommendation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining, V.1, KDD 2025, Toronto, ON, Canada, August 3-7, 2025*, pages 962–973. ACM, 2025. 6

[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. In *The Twelfth International Conference on Learning Representations*, 2024. 13

[26] Tianhao Qi, Shancheng Fang, Yanze Wu, Hongtao Xie, Jiawei Liu, Lang Chen, Qian He, and Yongdong Zhang. Deadiff: An efficient stylization diffusion model with disentangled representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8693–8702, 2024. 1, 2, 3

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2, 3, 4, 11

[28] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023. 14

[29] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 2, 3, 8

[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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023. 3

[31] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2

[32] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. 2

[33] Yunze Tong, Junkun Yuan, Min Zhang, Didi Zhu, Keli Zhang, Fei Wu, and Kun Kuang. Quantitatively measuring and contrastively exploring heterogeneity for domain generalization. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, 2023. 6

[34] Yunze Tong, Fengda Zhang, Zihao Tang, Kaifeng Gao, Kai Huang, Pengfei Lyu, Jun Xiao, and Kun Kuang. Latent score-based reweighting for robust classification on imbalanced tabular data. In *Forty-second International Conference on Machine Learning*, 2025. 2

[35] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 2, 3

[36] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 6

[37] Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024. 6

[38] Anpeng Wu, Kun Kuang, Minqin Zhu, Yingrong Wang, Yujia Zheng, Kairong Han, Baohong Li, Guangyi Chen, Fei Wu, and Kun Zhang. Causality for large language models. *arXiv preprint arXiv:2410.15319*, 2024. 3

[39] Guangxuan Xiao, Tianwei Yin, William T Freeman, Frédo Durand, and Song Han. Fastcomposer: Tuning-free multi-subject image generation with localized attention. *International Journal of Computer Vision*, pages 1–20, 2024. 1, 2, 3, 6

[40] Zhongzhi Yu, Zheng Wang, Yonggan Fu, Huihong Shi, Khalid Shaikh, and Yingyan Celine Lin. Unveiling and harnessing hidden attention sinks: Enhancing large language models without training through attention calibration. *arXiv preprint arXiv:2406.15765*, 2024. 13

[41] Hengrui Zhang, Jiani Zhang, Balasubramaniam Srinivasan, Zhengyuan Shen, Xiao Qin, Christos Faloutsos, Huzefa Rangwala, and George Karypis. Mixed-type tabular data synthesis with score-based diffusion in latent space. In *The twelfth International Conference on Learning Representations*, 2024. 2

[42] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 3, 6

[43] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Scaling in-the-wild training for diffusion-based illumination harmonization and editing by imposing consistent light transport. In *The Thirteenth International Conference on Learning Representations*, 2025. 1

[44] Yanbing Zhang, Mengping Yang, Qin Zhou, and Zhe Wang. Attention calibration for disentangled text-to-image personalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4764–4774, 2024. 2, 3, 6, 8

[45] Canyu Zhao, Mingyu Liu, Wen Wang, Weihua Chen, Fan Wang, Hao Chen, Bo Zhang, and Chunhua Shen. Moviedreamer: Hierarchical generation for coherent long visual sequence. *arXiv preprint arXiv:2407.16655*, 2024. 1

[46] Canyu Zhao, Mingyu Liu, Huanyi Zheng, Muzhi Zhu, Zhiyue Zhao, Hao Chen, Tong He, and Chunhua Shen. Diception: A generalist diffusion model for visual perceptual tasks. *arXiv preprint arXiv:2502.17157*, 2025. 1