ADAPT: Attention Driven Adaptive Prompt Scheduling and InTerpolating Orthogonal Complements for Rare Concepts Generation

Supplementary Material

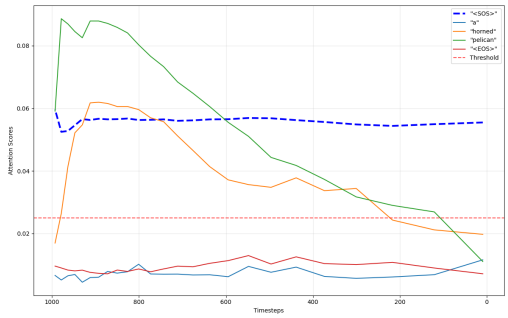


Figure 5. Attention response score $\mathcal{S}_{\text{Attn}}$ of target prompt "A horned pelican".

7. Algorithm Pseudocode for Adaptive Prompt Scheduling

Alg. 1 details the complete adaptive prompt scheduling procedure for ADAPT. The algorithm operates in two main stages: attention map scoring and prompt scheduling. In the attention map scoring stage, for timestep $(T-t)\%2 = 0$, we extract the maximum attention score $\mathcal{S}_{\text{Attn}}(y_{\text{tar}})$ over all tokens in the target prompt y_{tar}, excluding the $\langle$SOS$\rangle$ token. This quantifies the attention intensity for each rare concept at the current denoising step. If the number of transition counter P_{trans} is less than the total number of rare concepts m, we compute the top-k attention scores $s^{(k)}$ using the TopK operation. If the k-th highest attention score falls below the threshold τ_s, this indicates that the semantic meaning of the token is sufficiently saturated, triggering a prompt transition.

In the prompt scheduling stage, upon detecting low attention, we first increment the transition counter P_{trans} to track the scheduling progress. We then reconstruct the progressive prompt y_{prog} by replacing the P_{trans}-th frequent concept $y_f^{P_{\text{trans}}}$ with corresponding rare concepts $y_r^{P_{\text{trans}}}$.

If no transition occurs, y_{prog} defaults to y_{tar}. When $(T-t)\%2 \neq 0$, the progressive prompt y_{prog} is used without scoring. This alternation ensures gradual rare concept injection while maintaining global coherence.

8. Why except the $< SOS >$ token?

Fig. 5 illustrates attention score evolution for "A horned pelican." While semantic tokens exhibit saturation patterns converging to stable values, the <SOS> token (blue dashed line) maintains high scores throughout all timesteps. This occurs because CLIP's causal attention prevents <SOS> from receiving contextual information [27], making it a structural artifact rather than a semantic signal. Including <SOS> would prevent transitions in our stopping criterion as it never saturates, so we exclude it and compute $\mathcal{S}_{\text{Attn}}(y_{\text{tar}}) = \{z_i\}_{i=1}^{n}$ only over n semantic tokens.

9. Attention Map Aggregation and Scoring

Before computing attention scores, we aggregate attention maps from all MM-DiT transformer blocks in SD3. For each of the L transformer blocks, the attention mechanism produces an attention tensor of shape $[B, \text{num\_heads}, H, W, S]$, where B is the batch size, num_heads is the number of attention heads, $H \times W$ are spatial dimensions, and S is the sequence length.

We perform a two-stage aggregation process. First, within each block, we compute the arithmetic mean across all attention heads $\mathbf{A}_{\ell}^{\text{block}} = \frac{1}{|\text{num\_heads}|}\sum_{h=1}^{\text{num\_heads}} \mathbf{A}_{\ell,h}^{\text{block}}$. This reduces the shape from $[B, \text{num\_heads}, H, W, S]$ to $[B, H, W, S]$ for block ℓ.

Then we stack attention maps from all L blocks and compute their arithmetic mean $\mathbf{A}^c = \frac{1}{L}\sum_{\ell=1}^{L} \mathbf{A}_{\ell}^{\text{block}}$. This produces the final aggregated attention map $\mathbf{A}^c \in \mathbb{R}^{[B\times H\times W\times S]}$.

From the aggregated attention map $\mathbf{A}^c$, we extract attention scores for each token in the target prompt y_{tar}. For the i-th token, we compute

$$z_i = \max_{h,w} \mathbf{A}^c_{h,w,i}$$

where the max operation is applied over the spatial dimensions (h, w). This yields the attention response score set $\mathcal{S}_{\text{Attn}}(y_{\text{tar}}) = \{z_i\}_{i=1}^{n}$, where n is the number of tokens excluding the $\langle SOS \rangle$ token. The spatial maximum operation captures the strongest attention activation for each token across the entire spatial feature map, providing a robust measure of the model's focus intensity on that token during generation.

10. Attention Map Visualization

Attention-based Prompt Scheduling (APS) uses attention scores to determine optimal stop points. As shown in Fig. 6, the spatial attention map sharpens after certain steps, indicating generation completion of that specific token when $s^{(k)} < \tau_s$. Furthermore, we observe the spatial attention map pattern difference between the target and progressive

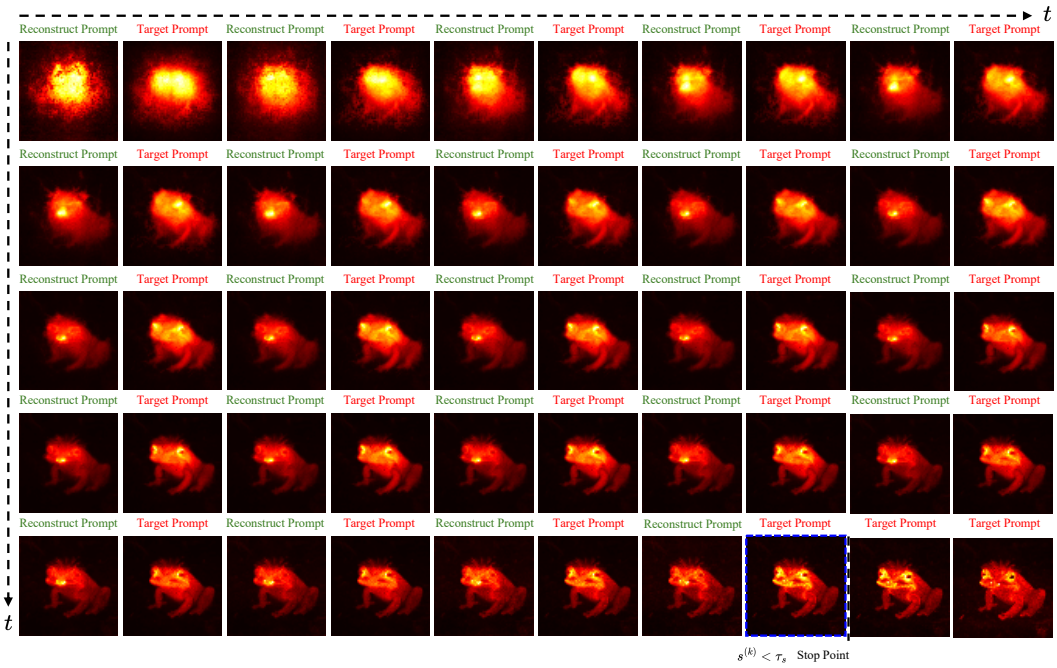


Figure 6. Attention map visualization for each step. The images are ordered sequentially from left to right across each row, starting from the top.

prompts. We find that this attention pattern difference hinders the convergence of the spatial attention score for the rare concept during the inference stage.

11. Attention Scores Visualization

Figure 13 illustrates the temporal evolution of attention scores S_{Attn} throughout the denoising process. For relatively simple rare concepts such as "A wooly alligator," we observe that the differentiating token "alligator" (which distinguishes the rare prompt from its frequent counterpart "A wooly animal") tends to exhibit slower convergence compared to shared tokens like "wooly." In such cases, the attention convergence pattern aligns with our intuition that rare semantic features require sustained focus during generation.

However, in complex multi-object or relational prompts, the convergence order becomes less predictable. Despite this variability, our APS approach using top-k attention scores consistently outperforms R2F's fixed stop points across all categories. This suggests that even without a strict correspondence between differentiating tokens and convergence order, attention scores provide a more semantically-aware scheduling signal than predetermined timestep ratios. By adaptively determining stop points based on observed attention dynamics rather than relying on fixed heuristics, APS better aligns prompt transitions with the actual generation process.

12. Further Visualization of Ablation Studies

Fig. 7 shows an ablation study on ADAPT hyperparameters for rare image generation, analyzing the effects of pooled embedding scale λ_{pool}, latent manipulation scale λ_{attr}, and attention threshold τ_s on generation quality.

Pooled embedding scale λ_{pool} modulates the rare-specific disentangled semantic direction in the pooled text embedding for "A bearded giraffe". The results show that $\lambda_{\text{pool}} = 0.3$ produces optimal results with clear beard generation while maintaining giraffe fidelity. When $\lambda_{\text{pool}} < 0.3$, the beard attribute is weakly expressed. However, when $\lambda_{\text{pool}} \geq 0.4$, image fidelity begins to degrade, and at $\lambda_{\text{pool}} \geq 0.7$, the rare-specific "bearded" attribute disappears entirely while the image becomes severely distorted, eventually reducing to a simple icon-like representation.

Latent manipulation scale λ_{attr} controls the attribute-specific disentangled semantic direction in the latent space for "A black white checkered crocodile." The optimal result occurs at $\lambda_{\text{attr}} = 0.15$, where the checkered pattern is clearly visible while preserving crocodile fidelity. When $\lambda_{\text{attr}} < 0.15$, the checkered pattern is insufficiently expressed. Conversely, when $\lambda_{\text{attr}} > 0.15$, the crocodile structure gradually deteriorates, and at $\lambda_{\text{attr}} \geq 0.4$, the "black white checkered" attribute completely vanishes.

Attention threshold τ_s determines the optimal stopping points S^i by measuring maximum spatial attention scores for "A mustachioed strawberry driving a banana shaped car

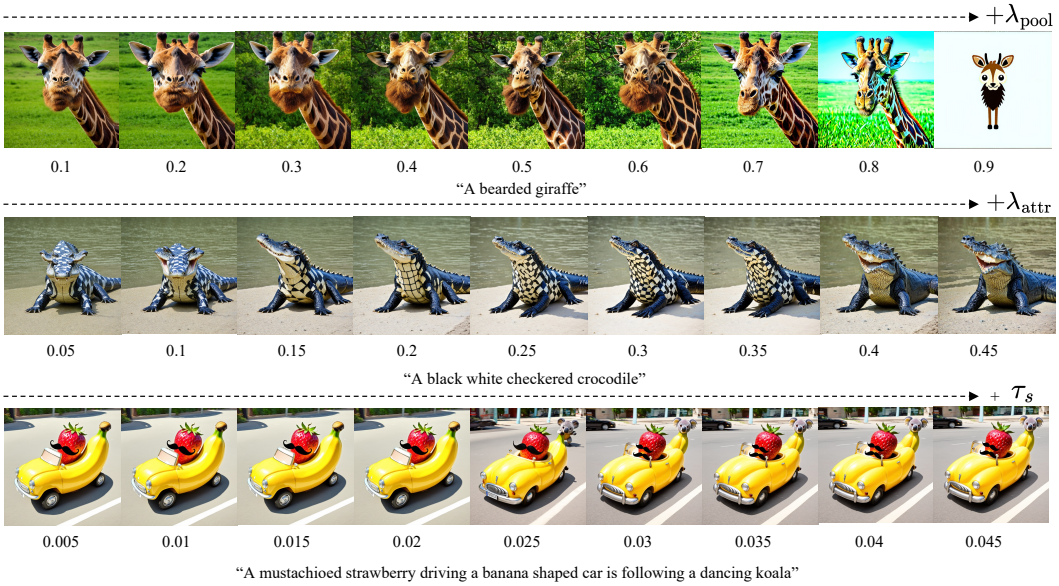


Figure 7. Further visualization on ADAPT hyperparameters for rare image generation. We vary the pooled embedding scale λ_{pool}, the latent manipulation scale λ_{attr}, and the attention threshold τ_s to analyze their effects on generation quality.

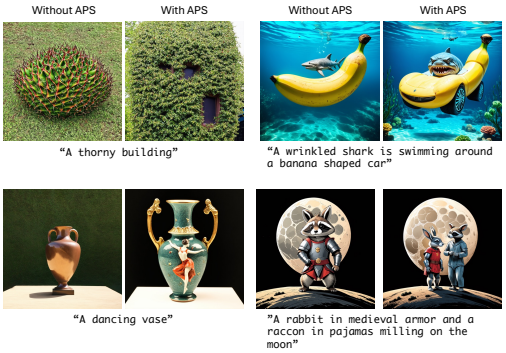


Figure 8. Ablation study with and without Attention-based Prompt Scheduling.

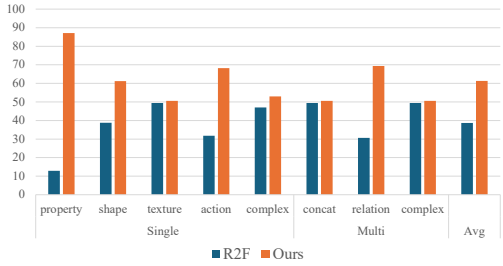


Figure 9. User study results for each category in the RareBench benchmark. Participants consistently preferred images generated by ADAPT over R2F, indicating stronger text-to-image alignment and rare concept fidelity.

is following a dancing koala." The results demonstrate that $\tau_s = 0.025$ achieves optimal generation with all elements properly generated. When $\tau_s < 0.025$, some objects (particularly the koala) fail to appear or are incompletely generated. When $\tau_s > 0.025$, object generation becomes truncated, with only partial elements (e.g., koala head only) being produced.

13. Impacts on Attention-based Prompt Scheduling

Fig. 8 shows the qualitative analysis of the Adaptive Prompt Scheduling (APS), comparing results with and without APS. Without APS, "A thorny building," "A dancing vase," "A wrinkled shark is swimming around a banana shaped car," and "A rabbit in medieval armor and a raccoon in pajamas milling on the moon," miss critical elements "building," "dancing," "banana shaped car," and "A rabbit" in the generated results. However, with APS, all prompt components are successfully incorporated. This demonstrates the superiority of APS that ensures comprehensive semantic coverage by adaptively scheduling attention across all prompt components.

14. User Study

Fig. 9 summarizes our user study evaluating human preferences for rare concept generation. We recruited 20 anonymous participants, each of whom compared 40 side-by-side image pairs generated by our method and R2F. The pairs were randomly drawn from RareBench, with 5 prompts

<Instructions for Image Preference Evaluation>

In this study, you will be shown a series of image pairs, each accompanied by a text prompt describing a rare or unusual concept. For each pair, please select the image that you believe best matches the given text prompt.

What to Consider

(1) Semantic accuracy: Does the image contain all objects and attributes mentioned in the prompt?
(2) Visual coherence: Is the rare concept visually well-integrated and believable?

Please focus on how well each image represents the described rare concept, even if such concepts don't exist in reality. The study consists of 40 prompts, selected from each category of RareBench. The entire process should only take about 20 minutes. Your input is extremely valuable and will help us understand how different image generation methods handle rare and unusual concepts.

Table 7. User Study Instructions.

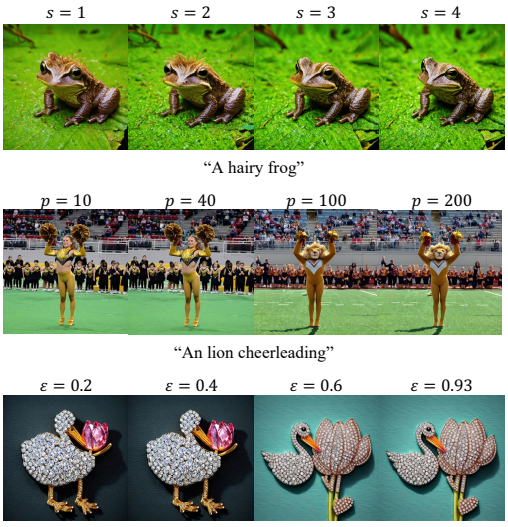


Figure 10. Qualitative analysis of adaptive weighting in PEM with varying parameters (s, p, ϵ) for scaling range, sharpness, and similarity threshold, respectively.

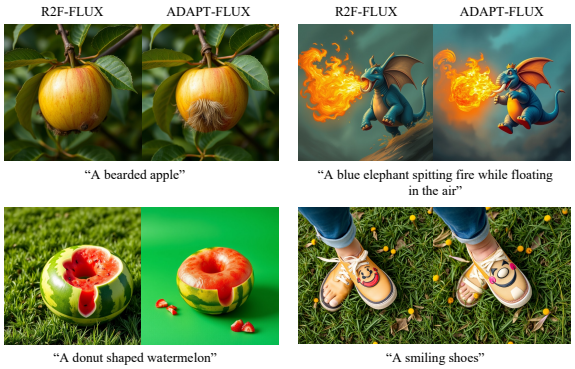


Figure 11. 4-step inference qualitative results of ADAPT combined with FLUX-schnell.

RareBench	Property	Shape	Texture	Action	Complex
FLUX	72.5	68.1	49.3	61.2	73.7
R2F (FLUX)	78.7	75.0	56.8	67.5	68.7
ADAPT (FLUX)	**81.9**	**80.6**	**61.3**	**70.0**	**77.5**

Table 8. 4-step inference quantitative results of ADAPT combined with FLUX-schnell.

sampled from each category to ensure balanced coverage. For every pair, participants selected the image that best matched the prompt based on two criteria—semantic accuracy and visual coherence—with image positions randomized to avoid bias. Overall, our method received 22.7% more preferences than R2F, demonstrating consistently stronger text-to-image alignment across categories. Tab. 7 provides the full evaluation instructions.

15. Qualitative Results on Adaptive Weighting

Fig. 10 shows the adaptive weighting hyperparameters used in the PEM. We follow the hyperparameter settings from [23]. When $s \geq 3$, the "hairy" feature in "A hairy frog" disappears. And when the $p \geq 100$, "lion" attribute appears on the "An lion cheerleading." For the threshold ϵ, values above 0.6 produce images with correct features of "a tulip made of diamonds."

16. ADAPT with FLUX-schnell

Here, we further propose an accelerated version of ADAPT integrated with FLUX-schnell, which requires only 4 or fewer steps to generate each image. We maintain only the core ideas of PEM and LSM, while following R2F's prompt scheduling for FLUX-schnell integration to handle short inference steps. Tab. 8 shows the 4-step inference results of ADAPT combined with FLUX-schnell on RareBench. While R2F applies the Composable method to pooled text

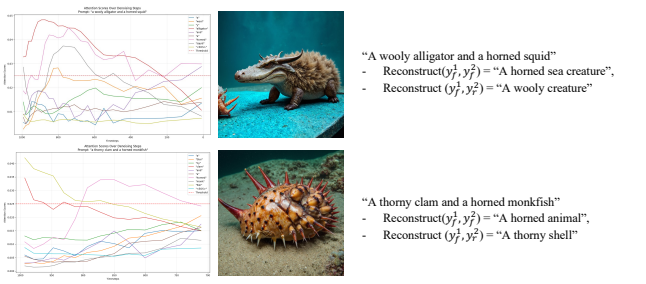


Figure 12. Illustration of failure cases. For the fair comparison, we use the same reconstructed prompts as R2F.

prompts, we employ PEM for pooled text embeddings. To inject attribute-specific guidance, we utilize LSM within the FLUX latent space. Fig. 11 shows the generated images of ADAPT combined with FLUX-schnell. Overall, ADAPT enhances text-to-image alignment while maintaining image quality, demonstrating that disentangled guidance is crucial for rare concept generation.

17. Further Visualization

Fig. 14 shows generated images on various seeds of ADAPT on RareBench. We randomly select eight prompts from RareBench's various categories and generate images using eight random seeds. Overall, most of the generated images are well-aligned with the input prompt, without compromising the fidelity and quality.

18. LLM Instruction for ADAPT

Tab. 10 presents the complete LLM instruction and in-context examples for ADAPT. To support LSM requirements, we incorporate **Context** fields corresponding to $\{y^i_{\text{attr}}\}^m_{i=1}$ in the example outputs.

The LLM decomposes input prompts into rare-to-frequent concept mappings with extracted contexts $\{y^i_r, y^i_f, y^i_{\text{attr}}\}^m_{i=1}$, enabling automatic rare concept identification, frequent concept generation, and attribute extraction in a one-shot LLM inference.

19. Frequent Prompt Selection Strategies.

To examine the sensitivity of ADAPT to the quality of frequent prompts, we generate three frequent prompt selection strategies on RareBench (Tab. 9). The *Human Generated* frequent prompt leverage expert-curated frequent prompts released by R2F authors; due to missing annotations for the complex category, we report the average over the remaining categories. For the *LLaMA3-8B-Instruct* [8] and *GPT-4o* configurations, we rely on LLM-generated frequent prompts produced according to our LLM instructions. Across all prompt selection strategies, ADAPT consistently outperforms R2F, demonstrating strong robustness

Models	Property	Shape	Texture	Action	Complex	Avg
SD3.0	49.4	76.3	53.1	71.9	65.0	63.1
Human Generated (R2F)	79.8	68.8	76.3	78.5	-	75.9
Human Generated (Ours)	**92.5**	**83.8**	**81.9**	**79.9**	-	**84.5**
LLaMA3 (R2F)	**81.9**	77.1	**76.3**	**78.8**	67.7	76.4
LLaMA3 (Ours)	78.8	**80.6**	**76.3**	78.1	**70.0**	**76.8**
GPT-4o (R2F)	89.4	79.4	81.9	80.0	72.5	80.6
GPT-4o (Ours)	**96.3**	**88.8**	**83.8**	**81.9**	**79.4**	**86.0**

Table 9. Experiments with different frequent prompt strategies.

to the choice and quality of frequent prompts. Notably, frequent prompts generated by GPT-4o lead to stronger results than human-curated prompts, suggesting that LLM-assisted prompt construction can provide richer or more semantically suitable anchors for rare concept generation.

20. Limitation

Fig. 12 shows that when reconstructed and target prompts differ greatly in length or semantics, the prompt schedule becomes challenging due to insufficient attention saturation. While our framework demonstrates robust performance across a wide range of prompt pairs, a possible extension is to generate frequent prompts that are semantically aligned and length-consistent with the target prompt, ensuring more stable scheduling.

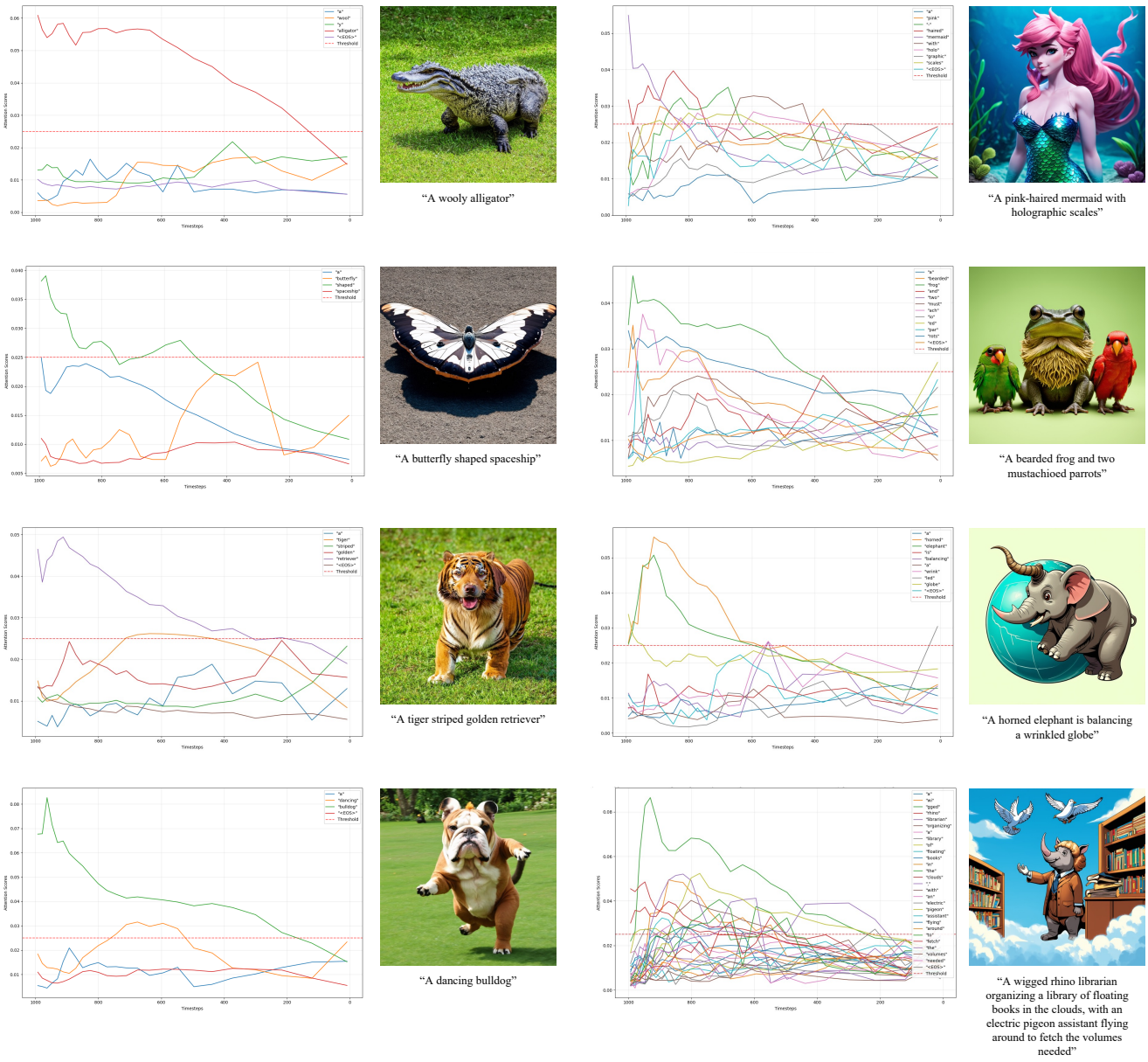
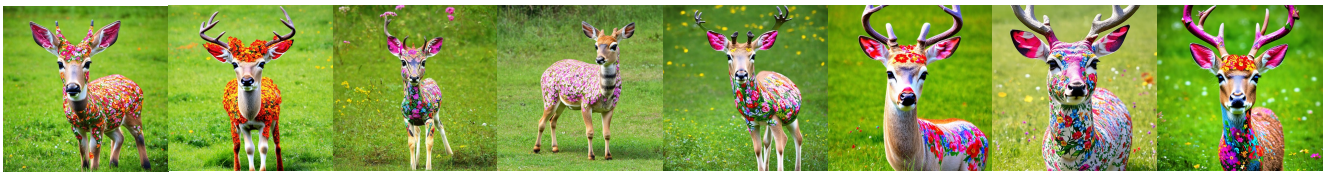


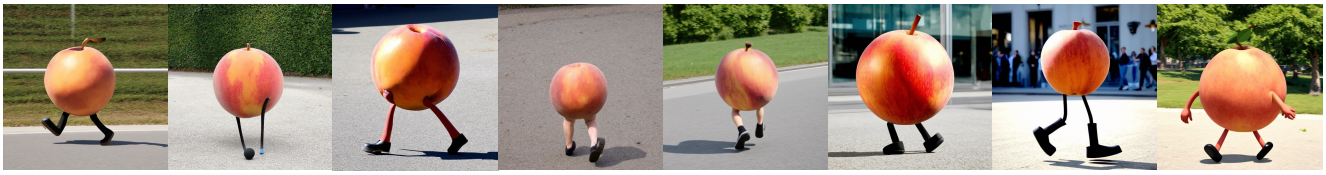
Figure 13. Visualization of S_{Attn} on each tokens and images.

Prompt: A horned pelican

Prompt: A donut shaped earth

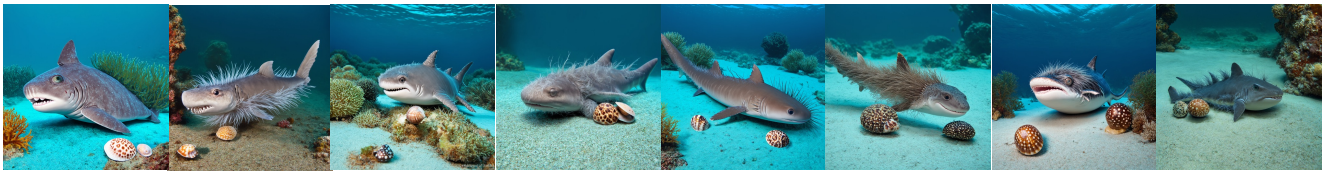


Prompt: A flower patterned deer



Prompt: A peach walking with legs

Prompt: A wrecked car made of clouds



Prompt: A hairy shark and two spotted clams

Prompt: A wooly alligator is guarding a banana shaped bottle

Prompt: A red bear and green bear fighting while sitting on a bench made of glass

Figure 14. Random seeds visualization results of ADAPT on RareBench. Images are generated from eight random prompts with eight random seeds.

<**System Prompt**>
You are a helper language model for a text-to-image generation program that aims to create images based on input text. The program often struggles to accurately generate images when the input text contains rare concepts that are not commonly found in reality. To address this, when a rare concept is identified in the input text, you should replace it with relevant yet more frequent concepts.

<**User Prompt**>
Extract rare concepts from the input text and replace them with relevant yet more frequent ones. Perform the following process step by step:
a. Identify and extract any rare concepts from the provided input text. If the text contains one or more rare concepts, extract them all. If there are no rare concepts present, do not extract any concepts. The extracted rare concepts should not overlap.
b. Given the rare concepts, replace each extracted rare concept with a more frequent concept. Specifically, split each rare concept into the main noun subject and the context, and replace the main noun subject with a more frequent noun subject that is likely to appear in the context of the original rare concept.
c. Generate a text sequence that starts from the text with replaced frequent concepts and ends with the text with the original rare concepts.
The output should follow the format of the examples below:

<**In-context Examples**>
Input: A peach made of glass
Output:
Num Rare Concepts: 1
a. Rare concept: A peach made of glass
b. A peach made of glass does not exist in reality, while the possibility of a pink sphere made of glass existing is much higher. Main noun subject: peach, Context: made of glass, Replaced frequent subject: pink sphere
c. A pink sphere made of glass BREAK A peach made of glass
Context: made of glass
Final Prompt Sequence: A pink sphere made of glass BREAK A peach made of glass

Input: A horned frog
Output:
Num Rare Concepts: 1
a. Rare concept: A horned frog
b. A horned frog does not exist in reality, while a horned animal does. Main noun subject: frog, Context: a horned, Replaced frequent subject: animal
c. A horned animal BREAK A horned frog
Context: a horned
Final Prompt Sequence: A horned animal BREAK A horned frog

Input: A horned lion and a hairy frog
Output:
Num Rare Concepts: 2
a. Rare concept: A horned lion
b. A horned lion does not exist in reality, while a horned animal does. Main noun subject: lion, Context: horned, Replaced frequent subject: animal
c. A horned animal BREAK A horned lion
AND
a. Rare concept: A hairy frog
b. A hairy frog does not exist in reality, while a hairy animal does. Main noun subject: frog, Context: a hairy, Replaced frequent subject: animal
c. A hairy animal BREAK A hairy frog
Context: horned AND a hairy
Final Prompt Sequence: A horned animal BREAK A horned lion AND A hairy animal BREAK A hairy frog

Table 10. Full LLM instruction for ADAPT to generate rare-to-frequent concept mappings.