

TIPO: TEXT TO IMAGE WITH TEXT PRESAMPLING FOR PROMPT OPTIMIZATION

Shih-Ying Yeh
National Tsing Hua University

ABSTRACT

TIPO (Text to Image with text presampling for Prompt Optimization) is an innovative framework designed to enhance text-to-image (T2I) generation by leveraging Large Language Models for automatic prompt engineering. By refining and extending user-provided prompts, TIPO bridges the gap between simple inputs and the detailed prompts required for high-quality image generation. Unlike previous approaches that primarily focus on extending prompts or utilize reinforcement learning without addressing the underlying reasons for suboptimal user inputs, TIPO introduces a method that optimizes prompts without the need for extensive RL tuning. Experimental results demonstrate TIPO’s effectiveness in improving aesthetic scores, reducing image corruption, and better aligning generated images with dataset distributions. These findings highlight the critical role of prompt engineering in T2I systems and open avenues for broader applications of automatic prompt refinement.

1 INTRODUCTION

Recent advancements in Text-to-Image (T2I) generative models have revolutionized creative applications (Saharia et al., 2022; Ramesh et al., 2021; 2022; Shi et al., 2020; Rombach et al., 2022; Podell et al., 2024; Sauer et al., 2024; Chen et al., 2024b;a; Li et al., 2024; Esser et al., 2024; black-forest labs, 2024). These models tend to perform better when provided with longer, more detailed prompts, which can describe specific elements like style, composition, and context. However, this reliance on highly descriptive inputs can be a significant bottleneck for generating high-quality images, especially for users who prefer or require simpler inputs. The need for intricate, nuanced prompts often results in a higher barrier to entry, limiting accessibility and ease of use for those unfamiliar with the complexities of prompt engineering. While previous works have attempted to achieve ‘better prompting’ by utilizing Large Language Models (LLMs), many ignore the root cause of why some prompts result in worse images and rely heavily on manually engineered prompts, which require substantial resources.

To address these challenges, we introduce **TIPO** (**T**ext to **I**mage with text presampling for **P**rompt **O**ptimization), an innovative framework designed to enhance T2I generative models. TIPO leverages LLMs to perform “Text Presampling” within the T2I inference pipeline, aiming to let text-to-image models reproduce a more accurate distribution of corresponding user prompts without additional manual engineering.

Unlike previous approaches that focus on extending prompts with manually collected prompt sets (AUTOMATIC, 2022; daspartho, 2022; succinctly, 2022), directly use LLM to modify the prompts (Lee et al., 2024; Zheng et al., 2024) or utilize reinforcement learning like method (Hao et al., 2023; crumb, 2023; Mañas et al., 2024) without addressing underlying issues, TIPO introduces a method that optimizes prompts efficiently.

Our work makes the following key contributions:

1. We introduce a general prompt optimization framework that is designed to work with any text-to-image model, providing a versatile solution for improving image generation across different architectures.

2. We provide comprehensive evaluation results using both standard and non-standard metrics. Our experiments demonstrate that TIPO’s prompt engineering method can improve image quality in terms of mathematical measures and user preferences, validating its effectiveness from multiple perspectives.

This paper presents TIPO’s theoretical foundations, implementation details, and experimental results demonstrating its effectiveness in improving aesthetic scores, reducing image corruption, and better aligning generated images with dataset distributions. Our findings highlight the critical role of prompt optimization in advancing T2I technology and open avenues for broader applications of automatic prompt refinement.

2 PREVIOUS WORKS

2.1 DIRECT PROMPT REFINEMENT THROUGH LLMs

With the advent of advanced Large Language Models (LLMs), several projects have leveraged their few-shot in-context learning capabilities to optimize prompts for Text-to-Image (T2I) models. For instance, CogView3 (Zheng et al., 2024) and the work by Lee et al. (2024) utilize GPT-J and Text Style Transfer (TST) techniques for prompt refinement. However, this approach relies heavily on the LLM’s inherent knowledge of visual content descriptions, which can be unpredictable and may not always align with specific T2I model requirements.

2.2 EXTENDING PROMPTS WITH MANUALLY COLLECTED DATA

In the open-source T2I community, a common approach involves using curated datasets of “good prompts” to fine-tune or train LLMs for prompt extension. (AUTOMATIC, 2022; succinctly, 2022; daspartho, 2022) These models are typically trained on image-text pairs collected from users of proprietary T2I services like Midjourney or DALL-E 3 . While this method is straightforward and somewhat effective, it often lacks the versatility to adapt to diverse T2I models beyond the one used in the original dataset collection.

2.3 IMAGE OUTPUT-BASED TUNING

Recent works have explored more sophisticated prompt optimization techniques based on image outputs:

2.3.1 REINFORCEMENT LEARNING APPROACHES

Promptist (Hao et al., 2023) introduced a two-stage framework combining supervised fine-tuning with reinforcement learning. The RL stage incorporates aesthetic and relevance scores as environmental feedback, enabling the model to learn optimal prompt refinement strategies.

2.3.2 ITERATIVE OPTIMIZATION

OPT2I (Mañas et al., 2024) proposed a backpropagation-free prompt optimization framework. This method uses scorers and LLMs to iteratively update input prompts, aiming to improve aesthetic quality and coherence.

These RL-based or RL-like approaches allow for prompt refinement tailored to specific model properties or user preferences. However, they often require extensive computational resources and may not generalize well across different T2I models or datasets.

2.4 LIMITATIONS OF PREVIOUS APPROACHES

While these methods have shown promise, they often face challenges such as:

- Dependence on specific T2I model architectures or datasets
- High computational costs for RL-based methods

- Difficulty in maintaining alignment with original user intent
- Limited generalization across diverse T2I models and datasets

TIPO addresses these limitations by introducing a more versatile and efficient approach to prompt optimization, leveraging the strengths of LLMs while maintaining broader applicability across different T2I models and datasets.

3 METHOD AND CONCEPT

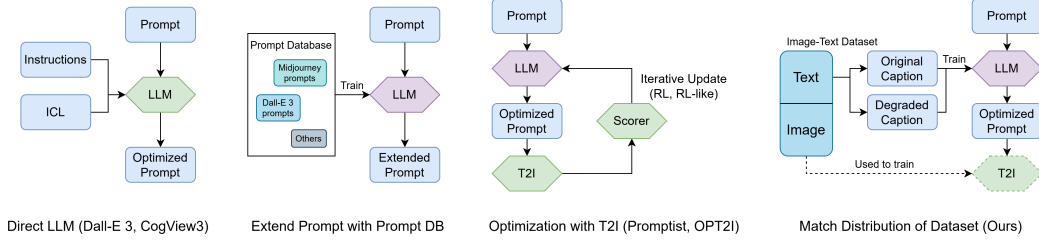


Figure 1: Comparison of prompt optimization approaches

3.1 CONCEPT

The core concept behind **TIPO** revolves around the relationship between prompt specificity and the quality and diversity of generated images. To understand TIPO’s approach, it’s helpful to first consider the limitations of existing methods and the intuition that led to our solution. As illustrated in Figure 1, previous approaches to prompt optimization face several challenges. Direct LLM methods like DALL-E 3 (Betker et al., 2023) and CogView3 (Zheng et al., 2024) rely heavily on the LLM’s pre-existing knowledge, which may not always align with specific T2I model requirements. Extending prompts with a LLM trained on specific prompt database can be effective but lacks adaptability across different T2I models or can result in worse performance. Optimization with T2I feedback as in Promptist (Hao et al., 2023) and OPT2I (Mañas et al., 2024) requires computationally expensive iterative processes and may not generalize well. In contrast, TIPO aims to match the distribution of the dataset used to train the T2I model, addressing these limitations by leveraging LLMs to generate optimized prompts that are both diverse and aligned with the T2I model’s training data distribution. We formalize the concept of TIPO as follows:

Let \mathcal{P} denote the set of all possible prompts, \mathcal{N} represent the space of Gaussian noise vectors, and \mathcal{I} denote the set of all possible images. A text-to-image model can be viewed as a mapping function:

$$f(p) : \mathcal{N} \rightarrow \mathcal{I}_p$$

where, for a given prompt $p \in \mathcal{P}$, the model maps noise vectors from \mathcal{N} to images in the subset $\mathcal{I}_p \subseteq \mathcal{I}$ corresponding to the prompt p .

Key Idea:

- **Simple Prompts:** Brief and general prompts correspond to broad distributions of possible outputs.
- **Detailed Prompts:** Longer and more specific prompts correspond to narrower distributions of outputs.

Formally, for a simple prompt p_s and a detailed prompt p_d that extends p_s :

$$f(p_d)(\mathcal{N}) \subseteq f(p_s)(\mathcal{N})$$

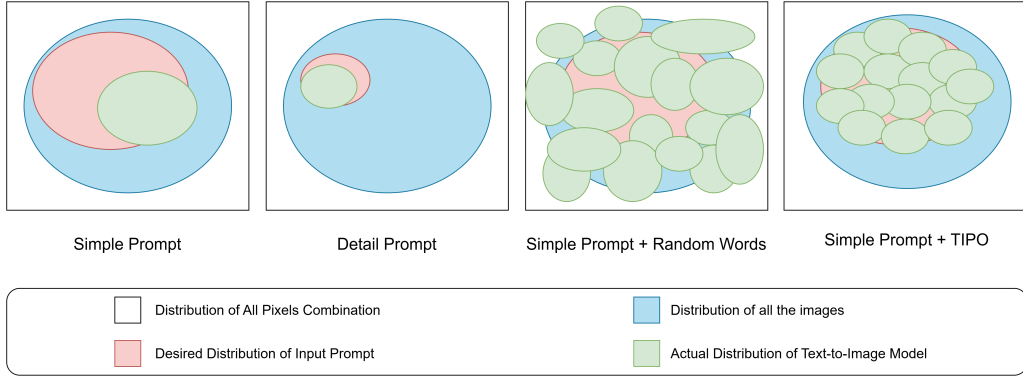


Figure 2: Visualization of prompt specificity and its impact on image generation distributions

Limitations of Simple Random Extensions: Directly adding random words to a simple prompt often leads to conflicting information, resulting in Out-Of-Domain inputs that cause the model to generate nonsensical images. Let p_{r_i} represent the i -th prompt created by adding random words to p_s . The resulting union of distributions fails to approximate the desired distribution:

$$\bigcup_{i=0}^n f(p_{r_i})(\mathcal{N}) \not\approx \mathcal{I}_{p_s}$$

This scenario is illustrated in Figure 2, where "Simple Prompt + Random Words" shows a scattered distribution extending beyond the bounds of the desired distribution or even the "image" distribution itself.

TIPO's Approach: TIPO introduces a novel approach to prompt optimization that aims to help text-to-image models achieve the same distribution as their training dataset, without relying on manually engineered prompts or Aesthetic Reinforcement Learning (RL).

Let p_{t_i} represent the i -th TIPO-generated prompt derived from a simple prompt p_s . The union of distributions from these prompts aims to approximate the desired distribution:

$$\bigcup_{i=0}^n f(p_{t_i})(\mathcal{N}) \approx \mathcal{I}_{p_s}$$

This approach, visualized in the "Simple Prompt + TIPO" scenario in Figure 2, allows for a broader exploration of the image space while maintaining relevance to the original prompt. It effectively balances diversity and coherence, leading to a set of images that better approximate the desired distribution of the input prompt.

Importantly, TIPO's design offers significant versatility:

- **Dataset Generalization:** A single TIPO model trained on dataset A can be used for all text-to-image models trained on dataset A. This allows for broad applicability across different model architectures and versions, as long as they share the same training dataset.
- **Caption Method Generalization:** More broadly, a TIPO model trained on a specific captioning method A can be applied to any text-to-image model trained on a dataset captioned using method A. This extends TIPO's utility across diverse datasets and models, as long as they share a common captioning approach.

By leveraging this approach, TIPO addresses the limitations of both overly simple and randomly extended prompts, providing a more effective and generalizable method for generating high-quality, diverse images that align with user intent. This generalization capability significantly enhances TIPO’s practical value, as it can be applied across a wide range of text-to-image models without the need for model-specific fine-tuning or manual prompt engineering.

3.2 TIPO FRAMEWORK

TIPO leverages Large Language Models (LLMs) to automatically extend and refine user-provided prompts. By generating detailed, content-rich prompts p_d from simpler user inputs p_s , TIPO ensures that the resultant prompts capture a more specific subset of possible outputs while maintaining alignment with the original user intent.

3.2.1 CONSTRUCTING p_s AND p_d

The construction of simple prompts (p_s) and detailed prompts (p_d) varies depending on the type of dataset and input modality. We consider two primary scenarios: tag-based datasets and natural language captions.

Tag-Based Captions For datasets like **danbooru2023**(Yeh, 2024b;a), tags are used to describe the content of images. In text-to-image models trained on such datasets, the concatenation of tags serves as the caption for each image.

- **Tag Sets:**
 - Let $T_n = \{t_1, t_2, t_3, \dots, t_n\}$ represent the complete set of tags for an image.
 - Derive the simple tag set $T_s = \{t_1, t_2, \dots, t_m\}$ where $m < n$.
 - The detailed tag set is $T_d = T_n$.
- **Prompts:**
 - Simple prompt: $p_s = \text{concat}(T_s)$
 - Detailed prompt: $p_d = \text{concat}(T_d)$

This approach ensures that p_d is a superset of p_s , aligning with the concept that detailed prompts provide a more specific description.

Natural Language Captions For datasets such as **GBC10M**(Hsieh et al., 2024) and **CoyoHD11M**(CaptionEmporium, 2024), natural language captions generated by models like LLaVA(Liu et al., 2023) are utilized. Defining p_s and p_d in this context involves additional considerations:

- **Short and Long Captions:**
 - Each image is associated with two types of captions:
 - * **Short Caption (p_s):** A brief, general description.
 - * **Long Caption (p_d):** An extended, detailed description.
 - **Note:** In this case, p_d is not a direct expansion of p_s but rather a paraphrased version that includes more details.
- **Single Long Caption:**
 - For images with a single long caption, split p_d into multiple sentences by identifying periods.
 - Generate p_s by retaining the first k sentences and omitting the last m sentences:

$$p_s = \text{concat}(\{\text{sentence}_1, \text{sentence}_2, \dots, \text{sentence}_k\})$$

$$p_d = \text{concat}(\{\text{sentence}_1, \text{sentence}_2, \dots, \text{sentence}_{k+m}\})$$

3.2.2 GENERATION PROCESS

To generate p_d from p_s , TIPO employs two distinct formats based on the relationship between p_s and p_d :

1. **When p_s is a Substring of p_d :**
 - **Format:** `<meta> <p_d>`
 - **Implementation:** In a causal language model, p_d is directly used after the `<meta>` token. The model learns to generate the subsequent tokens based on any substring of p_d .
2. **When p_s is Not a Substring of p_d :**
 - **Format:** `<meta> <p_s> <p_d>`
 - **Implementation:** Since p_s is not contained within p_d , both are included separately after the `<meta>` token to guide the generation process effectively.

3.2.3 HANDLING MULTIPLE INPUTS

In scenarios where both tag sequences (T_s) and natural language prompts (p_s) are present, TIPO processes each input type separately to maintain clarity and coherence.

Processing Logic:

1. **Isolate Input Types:** In each generation cycle, only one type of input is treated as the primary prompt (p_s), while the other types are considered as metadata.
2. **Sequential Generation:**
 - 2.1. **Step 1:** Use p_s (e.g., natural language prompt) with its corresponding metadata (e.g., aspect ratio, artists, etc.) to generate T_d from T_s .
 - 2.2. **Step 2:** Update the metadata with T_d and use p_s to generate the detailed natural language prompt p_d .

Example Workflow:

- **Initial Inputs:** p_s (natural language) and T_s (tags).
- **Generation 1:** Input `<meta> p_s T_s` to generate T_d from T_s .
- **Generation 2:** Input `<meta> T_d p_s` to generate p_d .
- **Aggregation:** Use T_d and p_d and metadata to construct final outputs.

This approach ensures that each input type is expanded and refined without interference, allowing TIPO to effectively capture the comprehensive distribution of possible outputs.

3.2.4 MATHEMATICAL FORMALIZATION

To encapsulate the generation process, consider the following formalization:

- **Prompt Expansion Function:**

$$\mathcal{E} : \mathcal{P}_s \times \mathcal{M} \rightarrow \mathcal{P}_d$$

where \mathcal{P}_s is the space of simple prompts, \mathcal{M} represents metadata (e.g., tags), and \mathcal{P}_d is the space of detailed prompts.

- **Generation Steps:**

$$p_d = \mathcal{E}(p_s, M)$$

where M could be either T_s or another form of metadata depending on the input type.

By iteratively applying the expansion function \mathcal{E} , TIPO systematically refines prompts to enhance the diversity and quality of generated images.

4 EXPERIMENTS SETUP

In this section, we detail the experimental setup used to evaluate the **TIPO** framework. We outline the formatting conventions for prompts and metadata, define the various tasks employed during training, and describe the models and datasets utilized in our experiments.

4.1 PROMPT AND METADATA FORMATTING

In **TIPO**, we define a simple and consistent format for the content of the `<meta>` token or the simple prompt p_s . The format is as follows:

`<Category>: <Content>`

For example, metadata categories include **artist**, **copyright**, **aspect ratio**, **quality**, and **year**. In certain cases, input tags or natural language (NL) prompts are treated as metadata as well, especially when the task involves generating tags or NL prompts conditioned on each other.

4.2 TASK DEFINITIONS AND TRAINING FORMATS

TIPO encompasses three primary tasks: extending tag sequences, extending NL prompts, and generating refined NL prompts. To facilitate these tasks, we define several specific task types and corresponding formatting methods during training:

1. **tag_to_long**: Use tags as metadata to generate a new NL prompt or extend a user-provided NL prompt.
2. **long_to_tag**: Use an NL prompt as metadata to extend a tag sequence.
3. **short_to_tag**: Use the simple prompt p_s as metadata to extend a tag sequence.
4. **short_to_long**: Use a user-input NL prompt as metadata to generate a refined detailed prompt p_d .
5. **short_to_tag_to_long**: Use a user-input NL prompt or tag sequence as metadata to generate a refined detailed prompt p_d .
6. **short_to_long_to_tag**: Use a user-input NL prompt or generated NL prompt as metadata to extend a tag sequence.
7. **tag_to_short_to_long**: Use user-input tags or NL prompts as metadata to generate a refined detailed prompt p_d .

By defining these task types, we can achieve special cases where, for instance, inputting a short description or a short tag sequence results in a full-size tag sequence and NL prompt in a single pass (i.e., generating T_d and p_d simultaneously).

4.3 TRAINING PROCEDURE

In our experiments, we ensure that each training pass generates a new prompt of a single type. This means that to generate an extended detailed prompt p_d , a refined prompt p_d , and an extended tag sequence T_d , the model requires at least three passes.

During training, for each dataset entry, we randomly choose one of the seven task types to apply. Additionally, the manner in which we split the simple prompt from the detailed prompt (p_s from p_d) and the simple tag set from the detailed tag set (T_s from T_d) is also randomly decided. This approach effectively increases the real dataset size beyond the number of entries present in the dataset due to the combinatorial possibilities introduced by the random task selection and prompt splitting.

4.4 MODEL ARCHITECTURE AND TRAINING DETAILS

We utilize the **LLaMA** architecture (Touvron et al., 2023a;b) with models of 200 million and 500 million parameters:

- **200M Model:** Pretrained on the **Danbooru2023**(Yeh, 2024b;a) and **GBC10M**(Hsieh et al., 2024) datasets for 5 epochs, then fine-tuned on the Danbooru2023, GBC10M, and **CoyoHD11M**(CaptionEmporium, 2024) datasets for 3 epochs, resulting in a total of approximately 40 billion tokens seen.
- **500M Model:** Pretrained on the Danbooru2023, GBC10M, and CoyoHD11M datasets for 5 epochs, resulting in a total of approximately 30 billion tokens seen.

Note: We count the tokens seen based on "non-padding tokens". With relatively short and varying range of data, the tokens seen here may be lower than expected.

4.5 DATASET AUGMENTATION AND EFFECTIVE SIZE

As mentioned previously, the randomization in task selection and prompt splitting effectively increases the real size of the training dataset. By generating multiple variations from a single dataset entry, the model is exposed to a wider range of inputs and outputs, enhancing its ability to generalize and perform the various tasks defined in **TIPO**.

5 EVALUATION RESULTS

All results presented in this section are tested on the **TIPO-200M** model.

5.1 GENERATION PROCESSES

In our experiments, we set up two distinct generation processes:

5.1.1 SHORT/TRUNCATED LONG TEST

- **Short Prompts:**
 - 10k short prompts randomly selected from **GBC10M**.
 - 10k short prompts randomly selected from **CoyoHD11M**.
- **Truncated Long Prompts:**
 - 10k long prompts randomly selected from **GBC10M**.
 - 10k long prompts randomly selected from **CoyoHD11M**.
 - Each long prompt is truncated to two sentences by splitting at periods.
- **TIPO-Enhanced Prompts:**
 - **TIPO + Short:** Apply the `short_to_long` task on short prompts, generating a total of 20k prompts.
 - **TIPO + Truncated Long:** Apply the `long_to_tag` task while forcing the model to expand the input long prompts, resulting in extended long prompts. (generated tags are ignored)

For the short/truncated long test, we generate one image from each prompt using the **SDXL-1.0-base** model(Podell et al., 2024).

5.1.2 SCENERY TAG TEST

In this test, we randomly select 32,768 entries from **Danbooru2023** that include the "scenery" tag. We set up the following inputs:

- **"Scenery" + Meta:**
 - Retain all metadata categories as described earlier.
 - Only include "scenery" as the content tag.
- **"Scenery" + Meta + TIPO:**
 - Use "scenery" + meta as input.

- Extend T_s (scenery) to T_d and generate p_d from T_d .

For the scenery tag test, we generate one image from each prompt using the **Kohaku-XL-zeta** model, which is a fine-tuned SDXL model on the Danbooru dataset(Touvron et al., 2023a).

5.2 EVALUATION METRICS

We employ the following metrics to evaluate the quality and alignment of the generated prompts and images:

5.2.1 AESTHETIC SCORE (HIGHER IS BETTER)

We compute the Aesthetic Score using the **Aesthetic Predictor V2.5**(discus0434, 2024). This metric is calculated on the short/truncated long test.

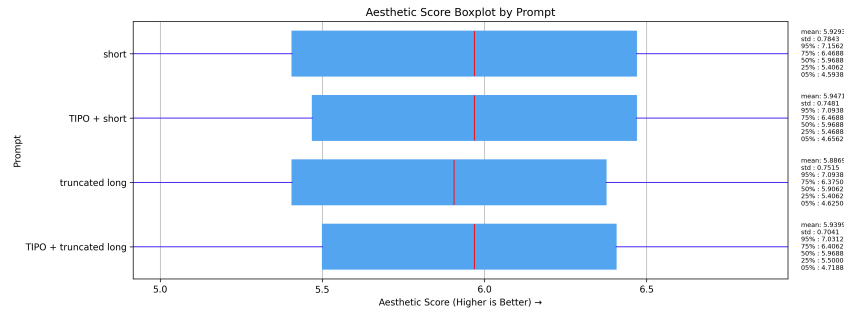


Figure 3: Aesthetic Score distribution. TIPO-generated prompts provide significantly higher values in the 25% quantile, indicating a higher lower bound in the aesthetic score. Both the median and mean values are also significantly higher.

5.2.2 AI CORRUPT SCORE (HIGHER IS BETTER)

The AI Corrupt Score is obtained from the **AICorruptMetrics**(narugo1992, 2023) in **sdeval**(narugo1992, 2024). This metric is trained on AI-generated images with human-annotated "corrupt or not" labels. A higher value indicates a higher likelihood of the image being "correct" or "complete".

This metric is calculated on the short/truncated long test.

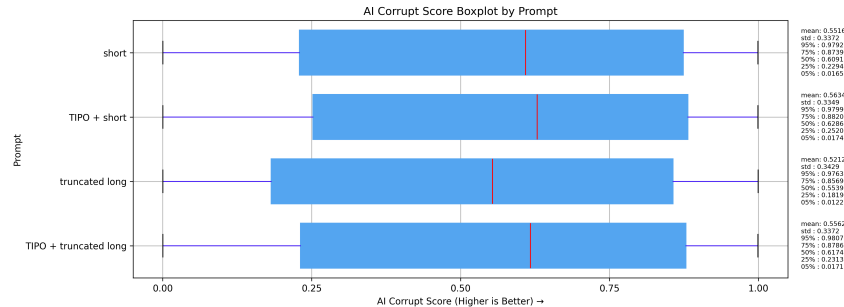


Figure 4: AI Corrupt Score distribution. TIPO-generated prompts achieve higher scores, indicating less corrupted images.

5.2.3 FRECHET DINO DISTANCE (FDD) ON SCENERY TAG TEST

Traditionally, the **Frechet Inception Distance (FID)**(Heusel et al., 2017) has been used to measure the distribution distance between datasets and generated images. However, recent works have shown that FID is not always aligned with human preferences(Stein et al., 2023). Therefore, we measure the Frechet Distance using **DinoV2** outputs across four different scales of the DinoV2 model(Crowson et al., 2024).

We use FDD on the Scenery Tag Test to demonstrate that when input prompts address a smaller distribution, the model struggles to generate images that reflect the true distribution. However, with **TIPO**, this issue is mitigated.

FDD Model	<meta> scenery only	<meta> scenery + TIPO
DinoV2 ViT-S	0.1917	0.1786
DinoV2 ViT-B	0.2002	0.1755
DinoV2 ViT-L	0.2017	0.1863
DinoV2 ViT-G	0.2359	0.2096

Table 1: Frechet Dino Distance (FDD) on Scenery Tag Test. Lower values indicate better alignment with the original dataset distribution.

As shown in Table 1, applying **TIPO** significantly improves FDD performance across all DinoV2 models. This implies that with **TIPO**, the generated images more closely match the original distribution in the dataset.

6 CONCLUSION

In this report, we introduced **TIPO**, a novel framework that enhances the quality of Text-to-Image (T2I) models through automatic prompt engineering. By leveraging Large Language Models (LLMs) to extend and refine user-provided prompts, **TIPO** effectively bridges the gap between simple user inputs and detailed, content-rich prompts. This approach ensures that the generated prompts are not only more specific but also maintain strong alignment with the original user intent.

6.1 KEY CONTRIBUTIONS

- Automatic Prompt Engineering:** We demonstrated the potential of **TIPO** in automating the process of prompt refinement. By transforming simple prompts (p_s) into detailed prompts (p_d), **TIPO** enhances the specificity and richness of the prompts, leading to higher-quality image generation.
- Versatile Task Framework:** **TIPO** encompasses a diverse set of tasks, including extending tag sequences, generating refined natural language prompts, and converting between different prompt formats. This versatility allows **TIPO** to handle various input modalities and dataset types effectively.
- Enhanced Alignment with T2I Datasets:** Our experiments revealed that improvements achieved through prompt modification can surpass the performance differences between different text-to-image model architectures. This finding underscores the critical importance of aligning user inputs with the underlying T2I dataset, highlighting that effective prompt engineering can significantly enhance model performance without necessitating architectural changes.

6.2 EXPERIMENTAL INSIGHTS

The experimental results validated the efficacy of **TIPO** across multiple metrics:

- Aesthetic Score:** **TIPO**-enhanced prompts consistently achieved higher aesthetic scores, indicating an improvement in the visual quality and appeal of the generated images.

- **AI Corrupt Score:** Higher AI Corrupt Scores for images generated on **TIPO**-generated prompts suggest that these images are more likely to be "correct" and "complete," reflecting better adherence to the desired content and structure.
- **Frechet Dino Distance (FDD):** **TIPO** significantly reduced the FDD score, demonstrating that the generated images more closely align with the original dataset distribution. This improvement highlights **TIPO**'s ability to help users generate images that are not only high in quality but also representative of the target data distribution.

6.3 IMPLICATIONS AND FUTURE WORK

The success of **TIPO** in improving T2I model outputs through prompt engineering opens several avenues for future research:

1. **Broader Application of Prompt Engineering:** Exploring **TIPO**'s applicability to other generative tasks, such as text generation or audio synthesis, could further demonstrate the versatility and impact of automatic prompt engineering.
2. **Integration with Interactive Systems:** Incorporating **TIPO** into interactive applications where users can iteratively refine prompts in real-time may enhance user experience and enable more precise control over generated content.
3. **Advanced Alignment Techniques:** Investigating more sophisticated methods for aligning user inputs with dataset distributions could further enhance the performance and reliability of generative models.

6.4 CONCLUSION

This study underscores the pivotal role of prompt engineering in the domain of Text-to-Image generation. By automating the refinement and extension of user prompts, **TIPO** not only improves the quality and specificity of generated images but also highlights the significance of aligning user inputs with the underlying data distributions. The findings suggest that strategic modifications to prompts can lead to substantial performance gains, potentially surpassing those achieved through architectural innovations alone. As generative models continue to evolve, frameworks like **TIPO** will be instrumental in unlocking their full potential and ensuring that they meet diverse user needs with precision and creativity.

REFERENCES

- AUTOMATIC. AUTOMATIC/promptgen-lexart. <https://huggingface.co/AUTOMATIC/promptgen-lexart>, 2022.
- James Betker, Gabriel Goh, Li Jing, † TimBrooks, Jianfeng Wang, Linjie Li, † LongOuyang, † JuntangZhuang, † JoyceLee, † YufeiGuo, † WesamManassra, † PrafullaDhariwal, † CaseyChu, † YunxinJiao, and Aditya Ramesh. Improving image generation with better captions. 2023. URL <https://api.semanticscholar.org/CorpusID:264403242>.
- black-forest labs. black-forest-labs/flux: Official inference repo for FLUX.1 models. <https://github.com/black-forest-labs/flux>, 2024.
- CaptionEmporium. CaptionEmporium/coyo-hd-11m-llavanext. <https://huggingface.co/datasets/CaptionEmporium/coyo-hd-11m-llavanext>, 2024.
- Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart- Σ : Weak-to-strong training of diffusion transformer for 4k text-to-image generation, 2024a. URL <https://arxiv.org/abs/2403.04692>.

- Junsong Chen, Jincheng YU, Chongjian GE, Lewei Yao, Enze Xie, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart- α : Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *The Twelfth International Conference on Learning Representations*, 2024b. URL <https://openreview.net/forum?id=eAKmQPe3m1>.
- Katherine Crowson, Stefan Andreas Baumann, Alex Birch, Tanishq Mathew Abraham, Daniel Z Kaplan, and Enrico Shippole. Scalable high-resolution pixel-space image synthesis with hourglass diffusion transformers. In *Forty-first International Conference on Machine Learning*, 2024. URL <https://openreview.net/forum?id=WRIn2HmtBS>.
- crumb. RLHF-Aesthetic Tuned model for prompt synthesis. <https://huggingface.co/crumb/bloom-560m-RLHF-SD2-prompter-aesthetic>, 2023.
- daspartho. daspartho/prompt-extend. <https://huggingface.co/daspartho/prompt-extend>, 2022.
- discus0434. discus0434/aesthetic-predictor-v2-5: SigLIP-based Aesthetic Score Predictor. <https://github.com/discus0434/aesthetic-predictor-v2-5>, 2024.
- 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. URL <https://arxiv.org/abs/2403.03206>.
- Yaru Hao, Zewen Chi, Li Dong, and Furu Wei. Optimizing prompts for text-to-image generation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=BsZNXD3a1>.
- 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. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf.
- Yu-Guan Hsieh, Cheng-Yu Hsieh, Shih-Ying Yeh, Louis Béthune, Hadi Pour Ansari, Pavan Kumar Anasosalu Vasu, Chun-Liang Li, Ranjay Krishna, Oncel Tuzel, and Marco Cuturi. Graph-based captioning: Enhancing visual descriptions by interconnecting region captions, 2024. URL <https://arxiv.org/abs/2407.06723>.
- Seunghun Lee, Jihoon Lee, Chan Ho Bae, Myung-Seok Choi, Ryong Lee, and Sangtae Ahn. Optimizing prompts using in-context few-shot learning for text-to-image generative models. *IEEE Access*, 12:2660–2673, 2024. doi: 10.1109/ACCESS.2023.3348778.
- Zhimin Li, Jianwei Zhang, Qin Lin, Jiangfeng Xiong, Yanxin Long, Xinchu Deng, Yingfang Zhang, Xingchao Liu, Minbin Huang, Zedong Xiao, Dayou Chen, Jiajun He, Jiahao Li, Wenyue Li, Chen Zhang, Rongwei Quan, Jianxiang Lu, Jiabin Huang, Xiaoyan Yuan, Xiaoxiao Zheng, Yixuan Li, Jihong Zhang, Chao Zhang, Meng Chen, Jie Liu, Zheng Fang, Weiyuan Wang, Jinbao Xue, Yangyu Tao, Jianchen Zhu, Kai Liu, Sihuan Lin, Yifu Sun, Yun Li, Dongdong Wang, Mingtao Chen, Zhichao Hu, Xiao Xiao, Yan Chen, Yuhong Liu, Wei Liu, Di Wang, Yong Yang, Jie Jiang, and Qinglin Lu. Hunyuan-dit: A powerful multi-resolution diffusion transformer with fine-grained chinese understanding, 2024. URL <https://arxiv.org/abs/2405.08748>.
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf.

- Oscar Mañas, Pietro Astolfi, Melissa Hall, Candace Ross, Jack Urbanek, Adina Williams, Aishwarya Agrawal, Adriana Romero-Soriano, and Michal Drozdal. Improving text-to-image consistency via automatic prompt optimization, 2024. URL <https://arxiv.org/abs/2403.17804>.
- narugo1992. Ai-corrupt score for anime images. https://huggingface.co/deepghs/ai_image_corrupted, 2023.
- narugo1992. deepghs/sdeval: Evaluation for stable diffusion model training. <https://github.com/deepghs/sdeval>, 2024.
- 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. URL <https://openreview.net/forum?id=di52zR8xgf>.
- Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8821–8831. PMLR, 18–24 Jul 2021. URL <https://proceedings.mlr.press/v139/ramesh21a.html>.
- Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022. URL <https://arxiv.org/abs/2204.06125>.
- 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 (CVPR)*, pp. 10684–10695, June 2022.
- Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 36479–36494. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/ec795aeadae0b7d230fa35cbaf04c041-Paper-Conference.pdf.
- Axel Sauer, Frederic Boesel, Tim Dockhorn, Andreas Blattmann, Patrick Esser, and Robin Rombach. Fast high-resolution image synthesis with latent adversarial diffusion distillation, 2024. URL <https://arxiv.org/abs/2403.12015>.
- Zhan Shi, Xu Zhou, Xipeng Qiu, and Xiaodan Zhu. Improving image captioning with better use of captions, 2020. URL <https://arxiv.org/abs/2006.11807>.
- George Stein, Jesse C. Cresswell, Rasa Hosseinzadeh, Yi Sui, Brendan Leigh Ross, Valentin Vilecroze, Zhaoyan Liu, Anthony L. Caterini, Eric Taylor, and Gabriel Loaiza-Ganem. Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=08zf7kT0oh>.
- succinctly. succinctly/text2image-prompt-generator. <https://huggingface.co/succinctly/text2image-prompt-generator>, 2022.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023a. URL <https://arxiv.org/abs/2302.13971>.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023b. URL <https://arxiv.org/abs/2307.09288>.

Shih-Ying Yeh. KohakuBlueleaf/HakuBooru: text-image dataset maker for anime-style images. <https://github.com/KohakuBlueleaf/HakuBooru>, 2024a.

Shih-Ying Yeh. KBlueLeaf/danbooru2023-webp-4Mpixel. <https://huggingface.co/datasets/KBlueLeaf/danbooru2023-webp-4Mpixel>, 2024b.

Wendi Zheng, Jiayan Teng, Zhuoyi Yang, Weihang Wang, Jidong Chen, Xiaotao Gu, Yuxiao Dong, Ming Ding, and Jie Tang. Cogview3: Finer and faster text-to-image generation via relay diffusion, 2024. URL <https://arxiv.org/abs/2403.05121>.

Appendix

A DATASET/RESOURCE

A.1 DANBOORU2023

The Danbooru2023 dataset(Yeh, 2024b;a) is an extensive collection of images and their corresponding tags, compiled from the Danbooru image board. This dataset includes images annotated with highly specific and detailed tags, providing a rich resource for training both the Text-to-Image (T2I) and Large Language Models (LLMs) involved in the TIPO framework. The dataset contains data up to image ID 7,349,999, encompassing a wide variety of visual content with granular annotations. These annotations allow for the creation of nuanced and precise prompts, ensuring that longer, more detailed prompts can indicate subsets of shorter prompts.

Key Characteristics:

- **Rich Annotations:** Danbooru2023’s detailed tags enable the differentiation of subtle variations in image content, such as "long hair" versus "very long hair," which is crucial for the specificity required in high-quality image generation.
- **Large Volume:** The dataset’s extensive size provides a robust foundation for training models, ensuring that they can learn from a diverse array of images and annotations.
- **Tag-Based Prompting:** Utilizing the detailed tags from Danbooru2023, the LLM can generate refined prompts that lead to more accurate and high-quality image generation by the T2I model.

A.2 GBC10M

The GBC10M dataset(Hsieh et al., 2024) is a large-scale collection of 10 million images sourced from CC12M, annotated using the Graph-Based Captioning (GBC) approach. Each image in GBC10M is represented by a graph where nodes correspond to object regions, compositions, and relations, and edges define the hierarchical relationships among them. Annotations are generated automatically through a pipeline that leverages pretrained multimodal large language models (MLLM) and object detection tools. The GBC structure enhances traditional image captions by providing both detailed descriptions and structural information, improving model performance in downstream tasks. All data is provided in JSON lines format, containing image URLs, bounding boxes, and caption annotations.

In TIPO, we only utilize the root node from GBC10M, since it provides both detail and short captions.

A.3 COYO HD 11M

The Coyo HD 11M dataset(CaptionEmporium, 2024) consists of 11.4 million high-density, high-definition images, with 22.8 million synthetic captions generated from the Coyo-700M dataset. The "HD" refers to both the high resolution of images (filtered to have at least 512 pixels on the shortest edge) and the high concept density, addressing the issues found in previous alt-text image pair datasets, which often included low-quality or low-concept images. Synthetic captions for this dataset were produced using the LLaMA-3 LLaVA-Next-8B model with post-processing for conciseness and clarity. This dataset provides high-quality visual and textual pairs, ideal for training advanced vision-language models.

In TIPO, we utilize the short/long captions, booru tags, and open images tags from Coyo HD 11M.

B IMPLEMENTATION DETAIL

In this appendix, we provide all the missing details about our dataset construction process that are not mentioned in sections 4.1 and 4.2.

B.1 TRAINING SETTINGS

	TIPO-200M stage1	TIPO-200M stage2	TIPO-500M
Architecture	LLaMA	-	LLaMA
Type	Pretrain	Finetune (from stage1)	Pretrain
Vocab Size	32013	-	32013
Hidden Dim	768	-	1280
Attention Heads	12	-	20
MLP Dim	2304	-	3840
Hidden Layers	20	-	20
Model Parameters	203M	-	508M
Max Learning Rate	2e-4	5e-5	2e-4
Optimizer	AdamW	AdamW	AdamW
betas	0.9, 0.98	0.9, 0.98	0.9, 0.98
weight decay	0.01	0.01	0.01
Dataset	GBC, Danbooru	Coyo, GBC, Danbooru	Coyo, GBC, Danbooru
max context length	512	1024	1024
global batch size	2048	2048	3584
Token Seen	22.625B	18.339B	31.274B
Hardware	4 x RTX3090	4 x RTX3090	8 x H100
Training Time (wall)	150 hour	270 hour	100 hour

Table 2: Training settings and details for TIPO-200M and TIPO-500M.

B.2 INFERENCE PIPELINE

The TIPO inference pipeline is designed to handle various input types and scenarios, combining different tasks to achieve refinement or expansion of both tag-based prompts and natural language prompts. Figure 5 illustrates this comprehensive workflow.

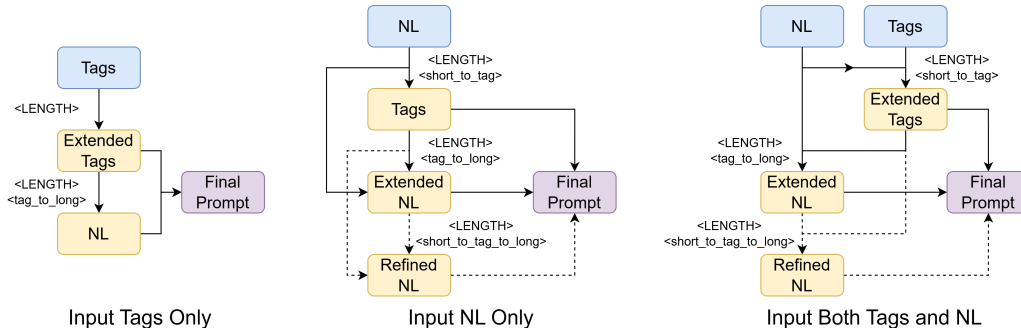


Figure 5: The inference workflow in TIPO, demonstrating how different task combinations are used to refine or expand tag-based and natural language prompts.

Our framework processes tags and natural language inputs separately, allowing for specialized handling of each input type. This flexible pipeline allows TIPO to adapt to various input scenarios, whether the user provides tags, natural language descriptions, or both. By leveraging different task combinations, TIPO ensures that both tag-based and natural language prompts are optimized, resulting in more detailed and effective inputs for text-to-image models.

C OUTPUT DISTRIBUTION TEST

In this section, we present sample images from the experiments described in Section 5 to visually demonstrate the improvements achieved by TIPO.

C.1 SHORT/TRUNCATED LONG TEST

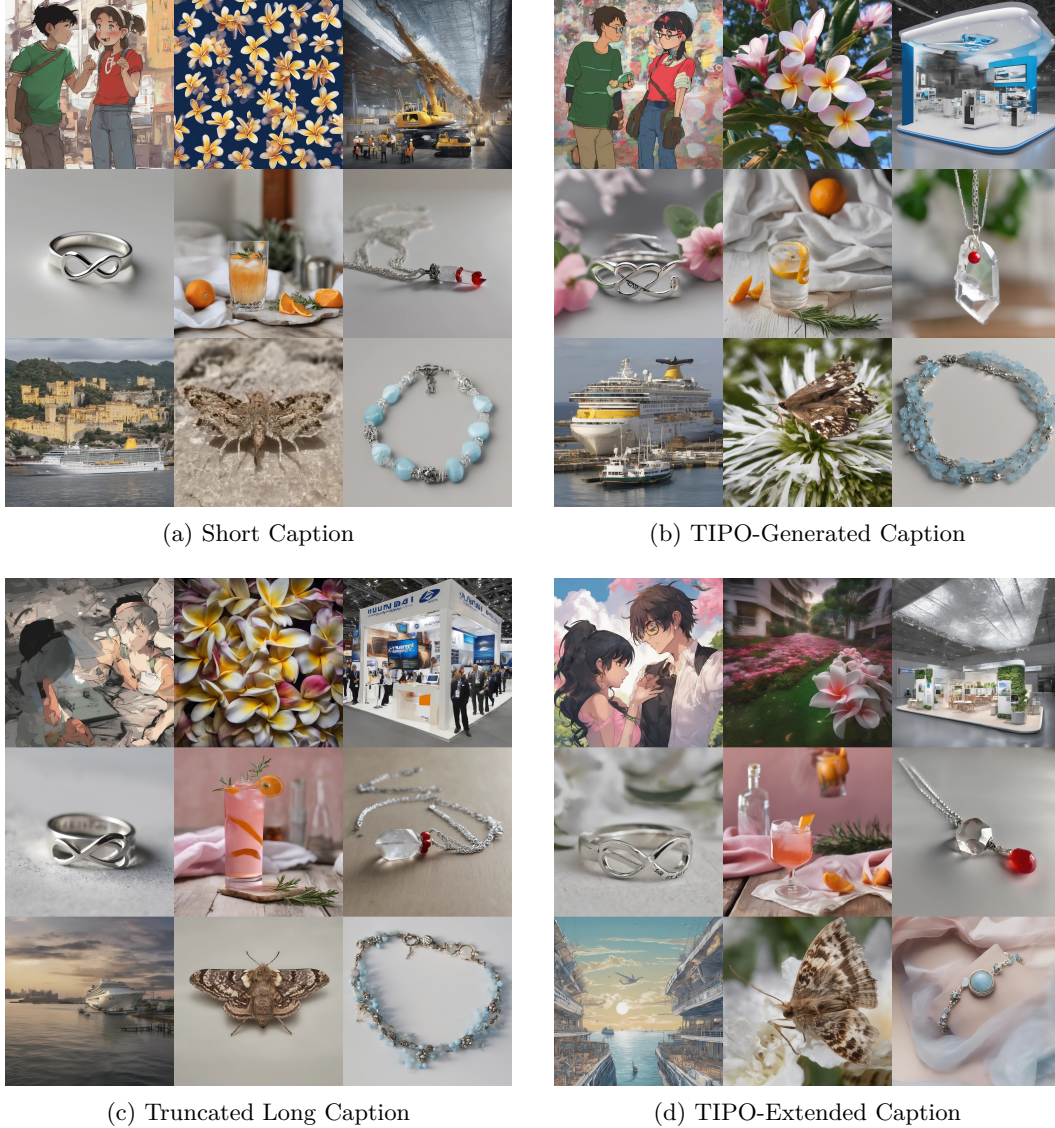


Figure 6: Comparison of generated images using original input (left) vs. TIPO-enhanced input (right)

Figure 6 illustrates the differences between short captions, truncated long captions, TIPO-generated captions, and TIPO-extended captions. The "short prompt" and "truncated long prompt" used in this experiment typically consist of 1-2 sentences, resulting in reasonably good quality outputs. However, the use of TIPO to refine or extend these prompts still yields noticeable improvements in aesthetics and overall quality.

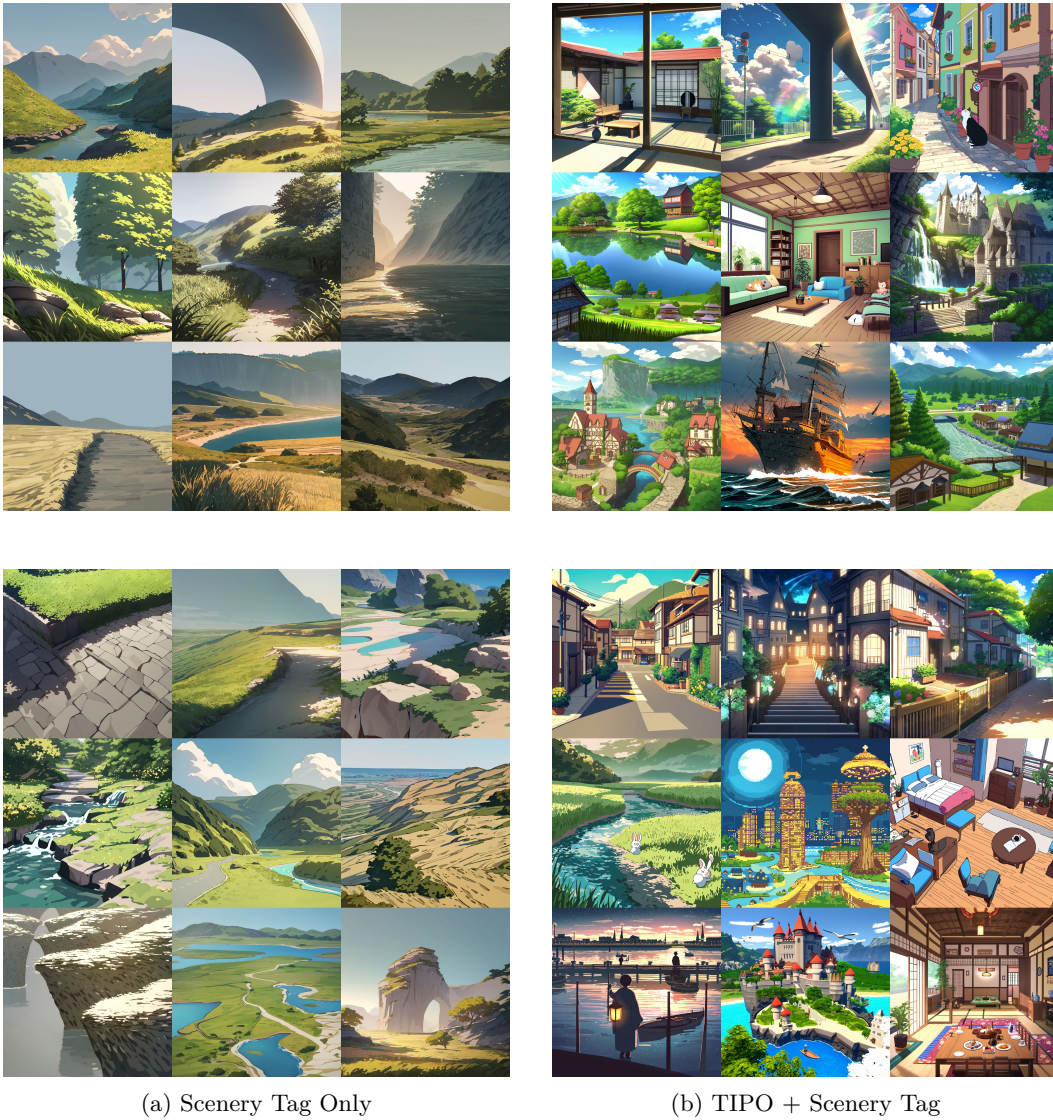


Figure 7: Comparison of generated images using simple input (left) vs. TIPO-enhanced input (right) for the scenery tag

C.2 SCENERY TAG TEST

Figure 7 demonstrates the difference in output diversity between simple input and TIPO-enhanced input for the scenery tag. As observed, TIPO significantly expands the range of generated sceneries, better reflecting the variety present in the Danbooru2023 dataset (Yeh, 2024b). The left column shows results from simple input (scenery tag only), while the right column illustrates the enhanced diversity achieved with TIPO-enhanced input.

D TIPO EXAMPLE

In this appendix, we provide some text example of TIPO's input and output.

TIPO Format template

User Input:

```
1girl, ciloranko, maccha (mochancc), ningen mame, ask (askzy), solo, masterpiece, absurdres,
newest, safe

A girl sits in a cozy cafe, cradling a cup of coffee in her hand
```

Formatted TIPO Input For Expand Tags:

```
meta: absurdres
rating: safe
artist: ciloranko, maccha (mochancc), ningen mame, ask (askzy)
quality: masterpiece, newest
aspect_ratio: 1.0
target: <|short|> <|short_to_tag|>
short: A girl sits in a cozy cafe, cradling a cup of coffee in her hand
tag: 1girl, solo
```

Formatted TIPO Output after Expand Tags and Expand Natural Prompt:

```
meta: absurdres
rating: safe
artist: ciloranko, maccha (mochancc), ningen mame, ask (askzy)
quality: masterpiece, newest
aspect_ratio: 1.0
target: <|short|> <|tag_to_long|>
tag: 1girl, solo, sitting, closed mouth, jewelry, long hair, looking at viewer, crossed legs,
plant, table, couch, bracelet, cup, smile, teacup, indoors, blue eyes, blonde hair, holding
long: A girl sits in a cozy cafe, cradling a cup of coffee in her hand. The cafe has large windows
with green plants on the walls and a wooden table in front of her. Behind her is a staircase
leading to another room. The overall atmosphere of the image is serene and inviting.
```

Formatted Output For Text-to-Image

```
1girl,
ciloranko, maccha \ (mochancc) \, ningen mame, ask \ (askzy) \,

solo, sitting, closed mouth, jewelry, long hair, looking at viewer,
crossed legs, plant, table, couch, bracelet, cup, smile, teacup,
indoors, blue eyes, blonde hair, holding,

A girl sits in a cozy cafe, cradling a cup of coffee in her hand. Behind
her is a staircase leading to another room. The cafe has large
windows with green plants on the walls and a wooden table in front
of her. The overall atmosphere of the image is serene and inviting.

masterpiece, newest, absurdres, safe
```



User Input



TIPO Output

Figure 8: An example formatted content we used for training and inference in TIPO.

TIPO Format template

User Input:

scenery, no humans, masterpiece, absurdres, newest, safe

Formatted TIPO Input For Expand Tags:

```
meta: absurdres
rating: safe
quality: masterpiece, newest
aspect_ratio: 1.0
target: <|long|>
tag: scenery, no humans
```

Formatted TIPO Output after Expand Tags and tag_to_long task:

```
meta: absurdres
rating: safe
quality: masterpiece, newest
aspect_ratio: 1.0
target: <|long|> <|tag_to_long|>
tag: scenery, no humans, storefront, motor vehicle, road sign, power lines, plant, railing, flower
    pot, vanishing point, outdoors, sign, potted plant, sidewalk, awning, tree, bicycle, window,
    railroad crossing, bush, building, utility pole, lamppost, shop, truck, traffic light, fence
    , chinese text, stairs, door, bicycle basket, town, day, streetcar (cafe), lamp, road
long: A small town with a variety of buildings and houses. the sky is blue and there are trees in
the background. on the left side of the image, there is an orange building with a sign that
reads "chinese restaurant". on the right side, there are several other buildings with
different types of shops and restaurants. in front of the buildings, there appears to be a
street with cars parked along the road.


in the center of the illustration, we can see a train crossing signal with two red lights and a
blue sky above it. there is also a yellow building with white walls and a green roof. on top
of the traffic light pole, there seems to be an air conditioning unit. the street is lined
with trees and bushes, and there is graffiti on the ground.
```

Formatted Output For Text-to-Image


```
scenery, no humans, storefront, motor vehicle, road sign, power lines,
    plant, railing, flower pot, vanishing point, outdoors, sign, potted
    plant, sidewalk, awning, tree, bicycle, window, railroad crossing,
    bush, building, utility pole, lamppost, shop, truck, traffic light,
    fence, chinese text, stairs, door, bicycle basket, town, day,
    streetcar \(\cafe\), lamp, road,
```

A small town with a variety of buildings and houses. the sky is blue and there are trees in the background. on the left side of the image, there is an orange building with a sign that reads "chinese restaurant". on the right side, there are several other buildings with different types of shops and restaurants. in front of the buildings, there appears to be a street with cars parked along the road. in the center of the illustration, we can see a train crossing signal with two red lights and a blue sky above it. there is also a yellow building with white walls and a green roof.

```
masterpiece, newest, absurdres, safe
```



User Input



TIPO Output

Figure 9: An example formatted content we used for training and inference in TIPO.