

A User-Friendly Framework for Generating Model-Preferred Prompts in Text-to-Image Synthesis

Nailei Hei¹, Qianyu Guo³, Zihao Wang⁴, Yan Wang^{1*}, Haofen Wang^{5*}, Wenqiang Zhang^{1,2,3*}

¹Shanghai Engineering Research Center of AI & Robotics, Academy for Engineering & Technology, Fudan University

²Engineering Research Center of Robotics, Ministry of Education, Academy for Engineering & Technology, Fudan University

³Shanghai Key Lab of Intelligent Information Processing, School of Computer Science, Fudan University

⁴Tongji University

⁵College of Design and Innovation, Tongji University

nlhei22@m.fudan.edu.cn, qyguo20@fudan.edu.cn, zihawang26@outlook.com

yanwang19@fudan.edu.cn, carter.whfcarter@gmail.com, wqzhang@fudan.edu.cn

Abstract

Well-designed prompts have demonstrated the potential to guide text-to-image models in generating amazing images. Although existing prompt engineering methods can provide high-level guidance, it is challenging for novice users to achieve the desired results by manually entering prompts due to a discrepancy between novice-user-input prompts and the model-preferred prompts. To bridge the distribution gap between user input behavior and model training datasets, we first construct a novel Coarse-Fine Granularity Prompts dataset (CFP) and propose a novel User-Friendly Fine-Grained Text Generation framework (UF-FGTG) for automated prompt optimization. For CFP, we construct a novel dataset for text-to-image tasks that combines coarse and fine-grained prompts to facilitate the development of automated prompt generation methods. For UF-FGTG, we propose a novel framework that automatically translates user-input prompts into model-preferred prompts. Specifically, we propose a prompt refiner that continually rewrites prompts to empower users to select results that align with their unique needs. Meanwhile, we integrate image-related loss functions from the text-to-image model into the training process of text generation to generate model-preferred prompts. Additionally, we propose an adaptive feature extraction module to ensure diversity in the generated results. Experiments demonstrate that our approach is capable of generating more visually appealing and diverse images than previous state-of-the-art methods, achieving an average improvement of 5% across six quality and aesthetic metrics. Data and code are available at <https://github.com/Naylenv/UF-FGTG>.

Introduction

Generative foundation models, including language models and text-to-image models, can be prompted to follow user instructions. Recent advancements in text-to-image synthesis such as Stable Diffusion (SD) (Rombach et al. 2022) and Midjourney (Holz 2023) have facilitated the generation of high-fidelity images based on text prompts. Concurrently, recent studies have revealed that prompt design plays a crucial role in determining the quality of the generated images (Ko et al. 2023; Liu, Qiao, and Chilton 2022). Adjusting the prompt to better reflect the user’s intentions can lead to superior results. This issue is particularly pronounced in text-to-image models, as the capacity of their text encoders is relatively limited, such as CLIP text encoder (Radford et al. 2021) in Stable Diffusion. Empirical observations have also shown that common user inputs are often insufficient to produce aesthetically pleasing images using current models (Hao et al. 2023).

*Corresponding author.
Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

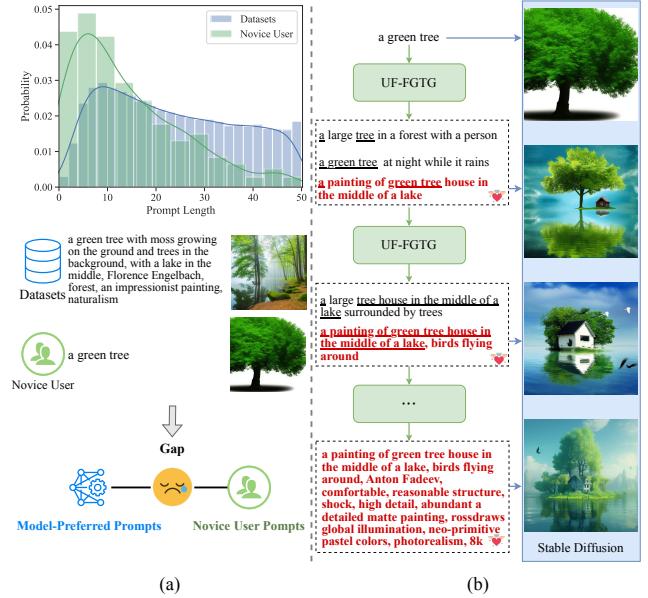


Figure 1: (a) We uncover an inconsistency in the word length distribution between prompts in the text-to-image training dataset and those provided by novice users, leading to a misalignment between model-preferred prompts and novice user prompts. (b) Our proposed UF-FGTG continually rewrites prompts, allowing users to select results of interest based on their needs until satisfied.

cial role in determining the quality of the generated images (Ko et al. 2023; Liu, Qiao, and Chilton 2022). Adjusting the prompt to better reflect the user’s intentions can lead to superior results. This issue is particularly pronounced in text-to-image models, as the capacity of their text encoders is relatively limited, such as CLIP text encoder (Radford et al. 2021) in Stable Diffusion. Empirical observations have also shown that common user inputs are often insufficient to produce aesthetically pleasing images using current models (Hao et al. 2023).

However, previous research has primarily focused on

manually designing prompts for specific text-to-image models (Liu and Chilton 2022; Pavlichenko and Ustalov 2023), typically adding some modifiers to the original prompts. While these studies have provided valuable insights, they are labor-intensive and only offer high-level suggestions, failing to offer personalized recommendations for users seeking specific aesthetics. Novice users, who lack experience in prompt writing and familiarity with relevant keywords, face significant challenges in achieving their desired results due to this. Therefore, it is essential to develop a method that can automatically rewrite prompts, thereby assisting novice users in generating model-preferred prompts.

As shown in Fig. 1(a), we analyze the probability distribution of prompt word lengths in text-to-image training datasets as compared to those actually used by novice users. Specifically, we employ DiffusionDB (Wang et al. 2022), a large-scale dataset frequently employed for training in text-to-image tasks, as our analysis dataset. Following (Wu et al. 2023), we use DiscordChatExporter (Holub 2022) to collect novice user prompts from the ‘*dreambot*’ channel on the Stable Diffusion Discord. Our analysis reveals a tendency among novice users to input short, coarse-grained prompts, contrasting with the long, fine-grained prompts used in model training. We believe that this gap results in a discrepancy between the intentions of novice users and the prompts that the model prefers.

Traditional methods for converting user-input prompts into model-preferred prompts rely on generative language models. However, existing generative language models such as GPT (Radford et al. 2019) and T5 (Raffel et al. 2020) are restricted to uni-modal text information during training, which constrains their ability to generate genuinely model-preferred prompts. To overcome this limitation and generate high-quality images in text-to-image tasks, the need for a multi-modal training framework is clearly highlighted.

To solve the above issues, we propose the **Coarse-Fine Granularity Prompts** dataset (CFP), a collection of 81,910 data instances from popular text-to-image community. Specifically, we refer to prompts in Lexica.art (Sanchez 2022) as fine-grained prompts. Then we generate corresponding images from the fine-grained prompts and use a summarization model (sshleifer 2021) to produce coarse-grained prompts, thereby creating a triplet dataset.

Building on our CFP dataset, we propose a novel **User-Friendly Fine-Grained Text Generation** framework (UF-FGTG) for automated prompt optimization. Specifically, we first propose a prompt refiner, which transforms coarse-grained prompts into fine-grained prompts. Secondly, we incorporate image-related loss functions in text-to-image tasks, ensuring the generated fine-grained prompts as model-preferred prompts. Thirdly, nearly every word in a text-to-image task prompt can find corresponding semantics in the generated image. However, many stylistic details of an image, particularly those described in short texts, are not represented. To prevent the generation in a fixed style, we propose an adaptive feature extraction module to ensure the diversity of the generated results. As shown in Fig. 1(b), our UF-FGTG continually refines prompts, enabling users to select outcomes of interest as per their requirements until satisfac-

tion is achieved. Through extensive experiments on the CFP dataset, we demonstrate the effectiveness of our proposed method on both quantitative and qualitative measures.

Our major contributions are as follows:

- We propose a novel Coarse-Fine Granularity Prompts dataset (CFP), a unique triplet dataset designed to bridge the gap between user behavior and model-preferred prompts. To the best of our knowledge, CFP is the first dataset that comprises fine-grained prompts with corresponding images, as well as coarse-grained prompts.
- We propose a novel training framework for text generation in text-to-image tasks, which transforms coarse-grained prompts into a fine-grained prompt feature space, named User-Friendly Fine-Grained Text Generation (UF-FGTG).
- We propose an adaptive feature extraction module that aligns prompt features with adaptive image features to prevent the generation of monotonous style results, ensuring diversity in the generated results.

Related Work

Text-to-Image Generative Models

Text-to-image generative models enable users to create images based on textual input. Researchers have explored a variety of architectures to enhance image quality, including autoregressive models (Ramesh et al. 2021), generative adversarial networks (GANs) (Sauer et al. 2023), and diffusion models (Rombach et al. 2022). Several subsequent works, including DALL-E-2 (Ramesh et al. 2022), GLIDE (Nichol et al. 2022), Imagen (Saharia et al. 2022), and Stable Diffusion (Rombach et al. 2022), have brought the magic of text-to-image generation to public attention. Among these models, Stable Diffusion stands out as an open-source model with an active user community. Although text-to-image models can generate impressive images, they demand high-quality input prompts. Novice users, lacking experience in prompt writing and unfamiliar with relevant keywords, may still struggle to generate the desired images.

Prompt Engineering

In the specific field of text-to-image generative models, prompt engineering research is nascent, which aims to use carefully selected and combined sentences to achieve a specific visual style in synthesized images (Oppenlaender 2023). The input texts, known as the prompts, direct the model to generate specific images. Manual prompt engineering is a natural method for optimizing prompts. Pavlichenko et al. (2023); Oppenlaender et al. (2023); Liu et al. (2022) explore how to find model-preferred prompts manually. While manual prompt engineering can lead to significant progress, the process of designing prompts requires time and experience, and may not always yield optimal results. Therefore, various methods have focused on automatically searching for prompts through mining (Jiang et al. 2020), parsing (Haviv, Berant, and Globerson 2021), text generation (Hao et al. 2023) and LLMs (Zhu et al. 2023; Chakrabarty et al. 2023). Additionally, many previous works

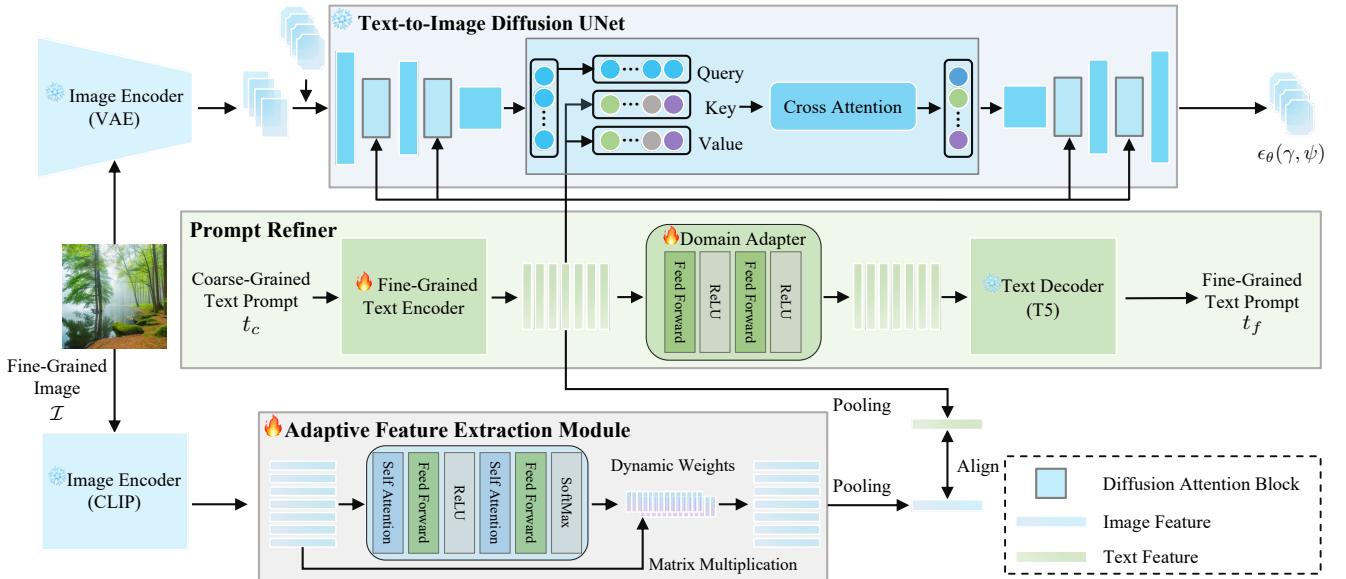


Figure 2: The architecture of our User-Friendly Fine-Grained Text Generation (UF-FGTG) Framework. The crux of the text generation network is the prompt refiner, which primarily comprises a fine-grained text encoder E_T and a text decoder D_E . The encoder E_T transforms coarse-grained prompt features into fine-grained prompt features. This process is supervised by fine-grained text T_F , a Stable Diffusion model ϵ_θ , and an adaptive feature extraction module N , ensuring the generated fine-grained prompts are not only model-preferred but also diverse. During inference process, only the prompt refiner is necessary.

have focused on gradient-based prompt learning methods, such as (Zhou et al. 2022; Wen et al. 2023). However, these gradient-based methods are not human-readable, which may hinder their application in human-AI collaboration.

Our work is closely related to prior research in the field of prompt engineering. For example, Google’s recent study (Hao et al. 2023) introduced a reinforcement learning-based approach to prompt training. However, their strategy is essentially a training methodology that can be applied to other models. Another related work by Wang et al. (2023) is limited by its reliance on manual design and may not be easily applicable to other domains. In this work, we propose a user-friendly prompt engineering framework that is data-driven and interpretable for text-to-image generation.

Coarse-Fine Granularity Prompts Dataset

Motivation. Existing datasets primarily depend on fine-grained prompts and corresponding images for model training (Wang et al. 2022). However, in real-world scenarios, users frequently input coarse-grained prompts, leading to a disparity between the model’s training and inference phases. Addressing this discrepancy to align native-user-input prompts with model-preferred prompts is crucial. To bridge this gap, we construct the Coarse-Fine Granularity Prompts dataset (CFP).

Fine-Grained Prompts Collection. We build a coarse-fine granularity prompts dataset based on Lexica.art (San-tana 2022), which consists of 81,910 fine-grained prompts filtered and extracted from user communities.

Coarse-Grained Prompts and Fine-Grained Images. For each fine-grained prompt obtained, following (Wang et al. 2022), we use Stable Diffusion-v2.1 (Rombach et al. 2022) to generate a corresponding image. The parameters used for image generation include “step”, “seed”, “height”, “width”, “CFG scale”, and “sampler”. Additionally, we employ BART (Lewis et al. 2020) as a summarization model (sshleifer 2021) to generate coarse-grained prompts of three different lengths: 1-5 tokens, 6-10 tokens, and 11-15 tokens. During training, one of these coarse-grained prompts is selected randomly.

Data Format. Finally, we obtain a total of 81,910 data instances, each consisting of one fine-grained prompt, one fine-grained image, and three coarse-grained prompts. We split 73,718 data pairs as the training set and 8,192 data instances as the testing set.

NSFW Contents. Following (Wu et al. 2023) and (Schuhmann et al. 2022), we observe that a small subset of data instances may contain NSFW (not safe for work) content. To avoid the potential harm caused by such content, we employ an NSFW detector (michellejiali 2022) to filter out fine-grained prompts that contain NSFW elements. We suggest using only the data with scores below 0.9, which amounts to a total of 79,447 data instances.

UF-FGTG Framework

Motivation. Existing text generation methods are unimodal, which can ensure the transformation of coarse-grained prompts into fine-grained ones, but cannot guarantee model-preferred prompts. To solve this issue, we

propose the User-Friendly Fine-Grained Text Generation framework (UF-FGTG), which has the capability to transform coarse-grained prompts into the feature space of fine-grained, model-preferred prompts, thereby generating high-quality images. More specifically, we propose a prompt refiner, which has the ability to transform coarse-grained prompts into fine-grained prompts. To ensure the generated prompts are model-preferred, we incorporate image-related supervision from Stable Diffusion. Additionally, we propose an adaptive feature extraction module that ensures diversity in the generated results.

Framework Overview

Fig. 2 presents an overview of our framework, specifically designed for prompt generation. We take the Stable Diffusion (Rombach et al. 2022) as an example to introduce our methodology. Our model is trained using triplet datasets, which we denote as $\mathcal{S} = \{(t_c, t_f, \mathcal{I})\}$. Here, t_c stands for coarse-grained prompts, while t_f represents fine-grained prompts. The symbol \mathcal{I} corresponds to the images that are associated with the fine-grained prompts.

The core of our framework is prompt refiner mainly composed of fine-grained text encoder and text decoder, which are designed to transform the input coarse-grained prompts into fine-grained prompts. For the first time, we incorporate image-related supervision from the Stable Diffusion process to generate model-preferred prompts, ensuring that the generated fine-grained prompts align with the model-preferred prompts. Additionally, we observe that while nearly every word in a prompt can find its corresponding semantics in the generated image, many stylistic details, particularly in short texts, are not adequately represented. This issue confines our UF-FGTG to generate images in a specific style. To tackle this problem, we propose an adaptive feature extraction module that aligns prompt features with adaptive image features, thereby ensuring diversity in the generated results.

During inference, the user inputs coarse-grained prompts and utilizes the prompt refiner within the framework to generate fine-grained prompts preferred by the model.

Text-to-Image Diffusion Model

Stable Diffusion. It consists of three main components: an autoencoder \mathcal{A} , a text time-conditional UNet denoising model ϵ_θ , and a CLIP fine-grained text encoder E_T . The autoencoder \mathcal{A} includes a VAE encoder \mathcal{E} and a VAE decoder \mathcal{D} , while the CLIP text encoder E_T accepts text prompts t as input. The encoder \mathcal{E} transforms an image $\mathcal{I} \in \mathbb{R}^{3 \times H \times W}$ into a lower-dimensional latent space in $\mathbb{R}^{4 \times h \times w}$, where $h = H/8$ and $w = W/8$. Conversely, the decoder \mathcal{D} carries out the inverse operation, decoding a latent variable into the pixel space.

Our goal is to generate fine-grained prompts t_f from the coarse-grained prompts t_c provided by the user while ensuring that the feature space can be understood by the UNet denoising model ϵ_θ , which is designed to generate fine-grained prompts that are model-preferred prompts. To achieve this, we refer to the ϵ_θ convolutional input as the spatial input γ (e.g., z_t) since convolutions preserve the spatial structure, and to the attention conditioning input as ψ

(e.g., $[\tau, E_T(T)]$). We train the text encoder E_T by minimizing the loss function defined as follows:

$$\mathcal{L}_{\text{mse}} = \mathbb{E}_{\mathcal{E}(\mathcal{I}), t_c, \epsilon \sim \mathcal{N}(0, 1), \tau} \|\epsilon - \epsilon_\theta(\gamma, \psi)\|_2^2, \quad (1)$$

where τ represents the diffusing time step, $\gamma = z_\tau$, z_τ is the encoded image $\mathcal{E}(\mathcal{I})$ where we stochastically add Gaussian noise $\epsilon \sim \mathcal{N}(0, 1)$, and $\psi = [\tau; E_T(t_c)]$.

Prompt Refiner

It consists of three components: a fine-grained text encoder E_T , a text decoder D_T , and a domain adapter Q . We utilize the CLIP model as the fine-grained text encoder and the T5 model (Raffel et al. 2020) as the text decoder to articulate our methodology.

Fine-Grained Text Encoder. CLIP (Radford et al. 2021) is a vision-language model that aligns visual and textual information within a shared embedding space. CLIP consists of a visual encoder E_V and a text encoder E_T . These encoders independently generate feature representations $E_V(\mathcal{I}) \in \mathbb{R}^n$ for an input image \mathcal{I} , and $E_T(L(t)) \in \mathbb{R}^n$ for the corresponding text t . Here, n represents the dimensionality of the embedding space in CLIP, and L denotes the embedding lookup layer that maps each tokenized word t to its respective token embedding in space \mathcal{W} .

In the original Stable Diffusion model, the CLIP text encoder only has the capability for text encoding. However, our fine-grained text encoder can transform the feature space from coarse-grained prompts t_c to model-preferred fine-grained prompts t_f , by concurrently employing the fine-grained prompt-related loss and the image-related loss supervision from the Stable Diffusion. In subsequent sections, we employ a language model to decode these features into human-readable prompts.

Text Decoder. The objective of our fine-grained text encoder is to transform coarse-grained prompt features into model-preferred fine-grained prompt features. Additionally, we implement a feature domain adapter Q , which utilizes a Multilayer Perceptron (MLP) to project CLIP text features onto the T5 text features space. Simultaneously, we employ the T5 model as a text feature decoder, denoted as D_T , to generate the final human-readable fine-grained prompts.

More specifically, the fine-grained text encoder is initialized using OpenCLIP (Cherti et al. 2023) derived from the Stable Diffusion model (Rombach et al. 2022). The text decoder D_T is initialized with a FLAN-T5 (Chung et al. 2022) pretrained generative language model. The fine-grained prompts t_f are utilized as training labels. The training objective is to minimize the log-likelihood by leveraging the teacher forcing technique (Williams and Zipser 1989):

$$\mathcal{L}_{\text{sft}} = \mathbb{E}_{(t_c, t_f) \sim \mathcal{S}} \log p(t_f | Q(E_T(t_c))). \quad (2)$$

Adaptive Feature Extraction Module

Through text-to-image model and prompt refiner in our framework, the fine-grained text encoder E_T transforms coarse-grained prompt features into model-preferred fine-grained prompt features. While nearly every word in a

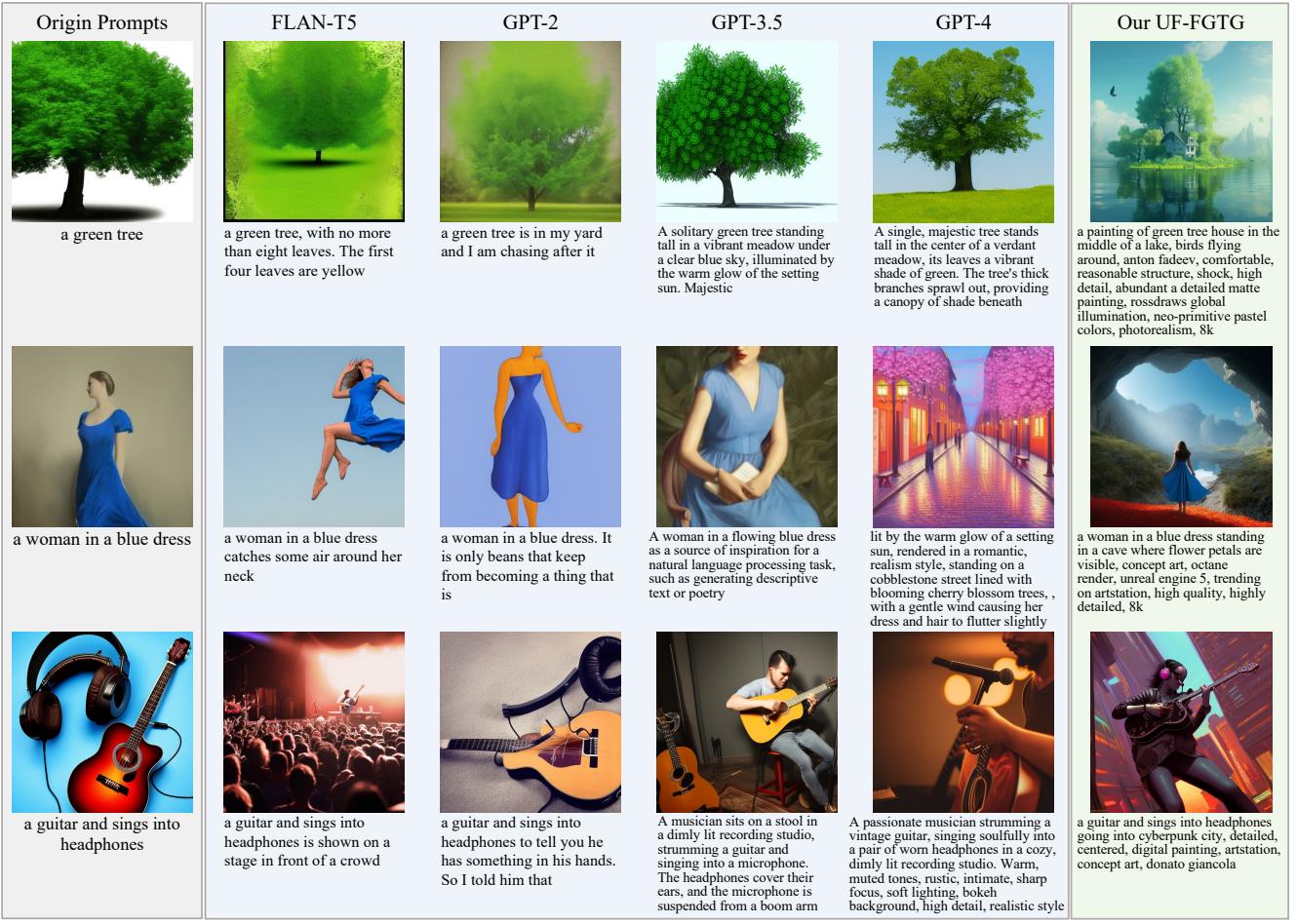


Figure 3: Comparison of prompts generated by FLAN-T5, GPT-2, GPT3.5, GPT-4, and our UF-FGTG, with corresponding images generated by Stable Diffusion-v2.1.

prompt can find its corresponding semantics in the generated image, many stylistic details in the image are difficult to reflect in the prompts, especially in short ones. For instance, a coarse-grained text such as “*a green tree*” might align with *a sunny forest scene* image, which, through direct training, could lead to generated results adhering to a uniform style, thereby reducing diversity. To ensure diversity in the generated results, we propose an adaptive feature extraction module that adaptively extracts image features.

Fig. 2 illustrates the architecture of our adaptive feature extraction module. This module aims to predict the soft dynamic weights of image representations. Specifically, we take the image \mathcal{I} extracted by the CLIP image encoder E_V as the input to the dynamic weight network. The dynamic weight network consists of self-attention layer, feed-forward layer, and ReLU activation functions. Ultimately, the dynamic weights w are obtained through a SoftMax function. These weights are then applied to weight pixel-wise features through matrix multiplication. This method enables the automatic learning of the most suitable and relevant representation from the image feature.

The features $E_T(t_c)$, which are extracted by the fine-

grained text encoder, and the features $\mathcal{N}(E_V(\mathcal{I}))$, generated by the CLIP image encoder and adaptive feature extraction module \mathcal{N} , have the same dimensions. As suggested by (Liang et al. 2023), we use the CLIP-Enhance loss function to evaluate the similarity between the prompt features and image features. The training objective is to minimize the CLIP-Enhance loss, where B represents the batch size:

$$\mathcal{L}_{\text{clip}} = -\frac{1}{B} \log \sum \frac{e^{\cos(E_T(t_c)_i, \mathcal{N}(E_V(\mathcal{I})))_i}}{\sum_{j \neq i} e^{\cos(E_T(t_c)_i, \mathcal{N}(E_V(\mathcal{I})))_j}}. \quad (3)$$

Loss Function

The overall loss function is a weighted sum of \mathcal{L}_{mse} , \mathcal{L}_{sft} and $\mathcal{L}_{\text{clip}}$. In our experiments, we set the trade-off hyperparameters α_1 and α_2 to 0.1.

$$\mathcal{L} = \mathcal{L}_{\text{mse}} + \alpha_1 \mathcal{L}_{\text{sft}} + \alpha_2 \mathcal{L}_{\text{clip}}. \quad (4)$$

Experiments

Experimental Setting

Implementation Details. We conduct our experiments on NVIDIA A100 GPUs. During training, we train the fine-

Model	NIMA -TID↑	MUSIQ -KonIQ↑	DB- CNN↑	TReS ↑	NIMA -AVA↑	MUSIQ -AVA↑
GPT-2	5.37	68.79	61.19	76.80	5.17	5.60
FLAN-T5	5.40	68.61	61.18	77.11	5.19	5.60
GPT-3.5	5.61	66.74	60.63	75.28	5.30	5.80
GPT-4	5.54	67.29	60.82	76.06	5.22	5.85
GPT-2*	5.62	69.55	62.40	80.81	5.32	5.81
FLAN-T5*	5.59	69.59	63.19	80.90	5.29	5.84
UF-FGTG*	5.73	69.74	65.21	83.34	5.48	5.97
<i>CFP-C</i>	5.46	68.76	62.05	78.06	5.25	5.66
<i>CFP-F</i>	5.81	70.15	66.97	84.18	5.62	6.07

Table 1: Image quality & aesthetic assessment. All methods first generate fine-grained prompts from coarse-grained prompts and then evaluate the generated images using Stable Diffusion-v2.1. “*” means trained on Coarse-Fine Granularity Prompts dataset (CFP). *CFP-C* and *CFP-F* denote the coarse-grained and fine-grained prompts in the CFP dataset.

grained text encoder, domain adapter, and adaptive feature extraction module on our CFP dataset for 100 epochs, using the AdamW optimizer (Loshchilov and Hutter 2018), a learning rate of 5e-5, and a batch size of 16.

In line with the Stable Diffusion-v2.1, our fine-grained text encoder is initialized with OpenCLIP (Cherti et al. 2023). The text decoder is initialized using FLAN-T5-base (Chung et al. 2022), while the image encoder employs the OpenCLIP that is paired with the fine-grained text encoder. This consistent approach to initialization ensures the compatibility and effectiveness of our proposed model.

Generation Strategy. In the subsequent experiments, we use the following default configuration: during prompt generation, we generate fine-grained prompts using 6-10 tokens of coarse-grained prompts. Following (von Platen 2020), we employ a strategy that combines Top-p and Top-K, which set p to 0.95 and K to 50. For the image generation phase, we utilize Stable Diffusion-v2.1, setting the CFG scale to 7, and perform 50 denoising steps using the Euler Ancestral sampler (Karras et al. 2022).

Qualitative Comparison

In Fig. 3, we visualize the generation results from various models, including GPT-2 (Radford et al. 2019), FLAN-T5 (Chung et al. 2022), GPT-3.5 (OpenAI 2023), and GPT-4 (OpenAI 2023). First, we rewrite the coarse-grained prompts, setting the max tokens to 20. Then, we generate images using Stable Diffusion-v2.1. Our method is capable of producing more visually appealing images. Furthermore, we note that traditional language models, including GPT-2 and FLAN-T5, struggle to comprehend the format of model-preferred prompts in text-to-image tasks. Even when we provide ChatGPT with a prompts format during generation, it still fails to produce satisfactory results. For example, in the case of “*a woman in a blue dress*”, GPT-4 modifies the original semantics, leading to a significant deviation in the generated result from the original content. In the majority of cases, GPT-2 and FLAN-T5 can only generate short

Model	NIMA -TID↑	MUSIQ -KonIQ↑	DB- CNN↑	TReS ↑	NIMA -AVA↑	MUSIQ -AVA↑
wo $\mathcal{L}_{\text{mse}, \text{clip}}$	5.48	67.23	62.34	78.21	5.21	5.64
wo \mathcal{L}_{mse}	5.53	68.65	64.01	80.92	5.35	5.73
wo $\mathcal{L}_{\text{clip}}$	5.68	69.32	64.74	82.24	5.41	5.89
UF-FGTG	5.73	69.74	65.21	83.34	5.48	5.97

Table 2: Impact of Stable Diffusion model (\mathcal{L}_{mse}) and adaptive feature extraction module ($\mathcal{L}_{\text{clip}}$) in our UF-FGTG.

text, even when we attempt to increase the maximum token count. In summary, these issues originate from language models’ limited grasp of image information in text-to-image tasks and their unfamiliarity with the structure of preferred prompts. Our method effectively addresses these issues.

Quantitative Comparison

Evaluation Metrics. Following (Li et al. 2023) and (Dinh, Nguyen, and Hua 2022), we quantitatively assess the image quality and aesthetic, using the non-reference metrics. Specifically, we choose NIMA (Talebi and Milanfar 2018), MUSIQ (Ke et al. 2021), DB-CNN (Zhang et al. 2020), and TReS (Golestaneh, Dadsetan, and Kitani 2022). We use NIMA-TID, MUSIQ-KonIQ, DB-CNN, and TReS for image quality assessment, while NIMA-AVA and MUSIQ-AVA are used for image aesthetic assessment. NIMA-AVA and MUSIQ-AVA are trained on AVA (Murray, Marchesotti, and Perronnin 2012), MUSIQ-KonIQ and DB-CNN are trained on KonIQ-10K (Hosu et al. 2020), and NIMA-TID is trained on TID2013 (Ponomarenko et al. 2013), following Py-IQA (Chen and Mo 2022).

Tab. 1 showcases the performance of various generative language models in image quality and aesthetic evaluation, with our method consistently surpassing other approaches across all six metrics, achieving an average improvement of 5%. This indicates that our method not only generates high-quality but substantial aesthetic images. Two primary reasons contribute to this phenomenon: (1) Previous uni-model text generation methods for fine-grained prompts neglect image information in text-to-image tasks, leading to sub-

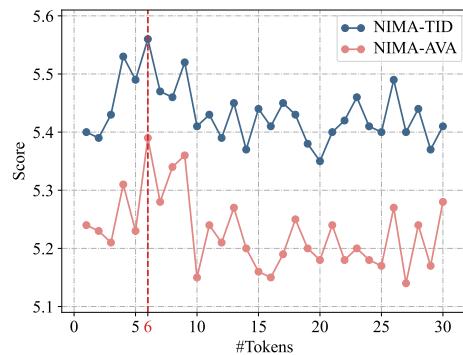


Figure 4: Ablation study on prompt length, showing both NIMA-TID and NIMA-AVA score as length increases.

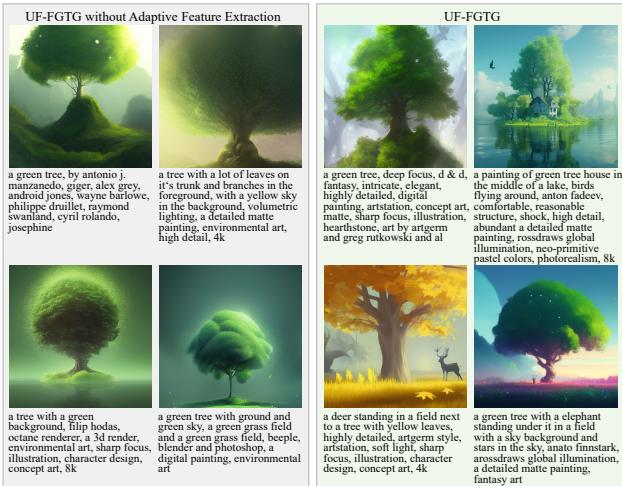


Figure 5: The adaptive feature extraction module enhances the diversity of the results. Origin prompts: “*a green tree*”.

par generative performance. Nonetheless, fine-tuning these methods with our CFP dataset yields enhanced results. (2) We observe that the text feature space in text-to-image tasks is merely a subset of the high-dimensional space employed in text generation. Our method maps the high-dimensional feature space in text generation into a low-dimensional one suitable for text-to-image tasks. This ensures that fine-grained prompts, derived from coarse-grained prompts, are model-preferred and correspond to high-quality images.

Ablation Study

Effect of Loss Functions. In Tab. 2, we study the model performance by varying its configuration. Our framework consists of three pipelines: text-to-image model, prompt refiner, and adaptive feature extraction module. It can be controlled by three loss functions, \mathcal{L}_{mse} , \mathcal{L}_{sft} , and \mathcal{L}_{clip} , to determine whether to add a particular pipeline. The results indicate that \mathcal{L}_{sft} and \mathcal{L}_{clip} are indispensable for text generation in text-to-image tasks.

Prompt Length. Following (Wen et al. 2023), we further ablate the optimal number of tokens. In Fig. 4, we observe that when using Stable Diffusion for image generation, longer prompts do not always lead to better image quality and aesthetic assessment. This phenomenon may be attributed to overfitting caused by longer prompts. Our experience indicates that additional prompts with a length of 6 produce the most generalizable performance.

Effect of Adaptive Feature Extraction Module. In Fig. 5, we demonstrate the increased diversity of generation results achieved by the adaptive feature extraction module. Without this module, the model tends to produce results with a singular style. However, by incorporating the adaptive feature extraction module, the model is capable of generating a variety of results. This diversity is attributed to our adaptive extraction of image features, which enhances the heterogeneity of the fine-grained prompt features.

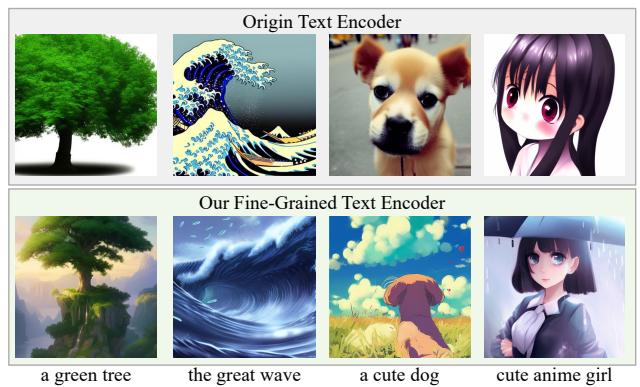


Figure 6: Origin text encoder vs our fine-grained text encoder in Stable Diffusion-v2.1.

Applications

Prompt Generation. The inference process for text generation is independent of its training phase. Our model proposes two recommended strategies for inference: (1) As shown in Fig. 1(b), the model generates three results concurrently, with each outcome further producing six tokens based on the preceding prompts, iterating this process until user satisfaction is achieved. (2) As shown in Fig. 3, the model aims to generate comprehensive prompts (setting max token to 20 or 50). Both strategies use the original Stable Diffusion model for image generation.

A Plug-and-Play Module in Stable Diffusion. Our method trains a fine-grained text encoder with the ability to map coarse-grained prompts to a fine-grained prompt feature space. As shown in Fig. 6, this allows it to fully supplant the encoding-only text encoder in the original Stable Diffusion model. Furthermore, we observe that when the input prompt extends to a certain length, the model tends to generate prompts such as “*4k resolution*”, “*highly detailed*” and “*best quality*”. Although these prompts are not semantically explicit, they can improve the quality of the generated images (Pavlichenko and Ustalov 2023). This suggests that our approach consistently projects any user-provided input prompt into a feature space aligned with fine-grained prompts, resulting in enhanced image generation quality.

Conclusion

In this paper, we propose the Coarse-Fine Granularity Prompts dataset (CFP), which enables the study of the gap between user behavior and model-preferred prompts. We also propose the User-Friendly Fine-Grained Text Generation (UF-FGTG) framework, which automatically translates user-input prompts into model-preferred prompts while incorporating image-related loss functions and an adaptive feature extraction module for improved diversity in generation results. Our experiments demonstrate that our method achieves state-of-the-art performance on both quantitative and qualitative measures, significantly advancing the field of text-to-image synthesis by providing a user-friendly method for automated prompt optimization.

Acknowledgments

This work was supported by Fundamental Research Funds for the Central Universities (No.22120230032), National Nature Science Foundation of China (No.62176185, No.62072112), Scientific and Technological Innovation action Plan of Shanghai Science and Technology Committee (No.22511102202), and China Postdoctoral Science Foundation (2023M730647, 2023TQ0075).

References

- Chakrabarty, T.; Saakyan, A.; Winn, O.; Panagopoulou, A.; Yang, Y.; Apidianaki, M.; and Muresan, S. 2023. I spy a metaphor: Large language models and diffusion models co-create visual metaphors.
- Chen, C.; and Mo, J. 2022. IQA-PyTorch: PyTorch Toolbox for Image Quality Assessment. [Online]. Available: <https://github.com/chaofeng/IQA-PyTorch>.
- Cherti, M.; Beaumont, R.; Wightman, R.; Wortsman, M.; Ilharco, G.; Gordon, C.; Schuhmann, C.; Schmidt, L.; and Jitsev, J. 2023. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2818–2829.
- Chung, H. W.; Hou, L.; Longpre, S.; Zoph, B.; Tay, Y.; Fedus, W.; Li, E.; Wang, X.; Dehghani, M.; Brahma, S.; et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.
- Dinh, T. M.; Nguyen, R.; and Hua, B.-S. 2022. TISE: Bag of Metrics for Text-to-Image Synthesis Evaluation. In *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Golestaneh, S. A.; Dadsetan, S.; and Kitani, K. M. 2022. No-reference image quality assessment via transformers, relative ranking, and self-consistency. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 1220–1230.
- Hao, Y.; Chi, Z.; Dong, L.; and Wei, F. 2023. Optimizing Prompts for Text-to-Image Generation. In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Haviv, A.; Berant, J.; and Globerson, A. 2021. BERTese: Learning to Speak to BERT. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 3618–3623.
- Holub, O. 2022. DiscordChatExporter. <https://github.com/Tyrrrz/DiscordChatExporter>.
- Holz, D. 2023. Midjourney alpha-release announcement on Discord. <https://www.midjourney.com/>.
- Hosu, V.; Lin, H.; Sziranyi, T.; and Saupe, D. 2020. KonIQ-10k: An Ecologically Valid Database for Deep Learning of Blind Image Quality Assessment. *IEEE Transactions on Image Processing*, 29: 4041–4056.
- Jiang, Z.; Xu, F. F.; Araki, J.; and Neubig, G. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8: 423–438.
- Karras, T.; Aittala, M.; Aila, T.; and Laine, S. 2022. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35: 26565–26577.
- Ke, J.; Wang, Q.; Wang, Y.; Milanfar, P.; and Yang, F. 2021. Musiq: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5148–5157.
- Ko, H.-K.; Park, G.; Jeon, H.; Jo, J.; Kim, J.; and Seo, J. 2023. Large-scale text-to-image generation models for visual artists’ creative works. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, 919–933.
- Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.
- Li, C.; Zhang, Z.; Wu, H.; Sun, W.; Min, X.; Liu, X.; Zhai, G.; and Lin, W. 2023. AGIQA-3K: An Open Database for AI-Generated Image Quality Assessment. *arXiv preprint arXiv:2306.04717*.
- Liang, Z.; Li, C.; Zhou, S.; Feng, R.; and Loy, C. C. 2023. Iterative Prompt Learning for Unsupervised Backlit Image Enhancement. In *ICCV*.
- Liu, V.; and Chilton, L. B. 2022. Design guidelines for prompt engineering text-to-image generative models. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, 1–23.
- Liu, V.; Qiao, H.; and Chilton, L. 2022. Opal: Multimodal image generation for news illustration. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, 1–17.
- Loshchilov, I.; and Hutter, F. 2018. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.
- michellejieli. 2022. NSFW Text Classifier. https://huggingface.co/michellejieli/NSFW_text_classifier.
- Murray, N.; Marchesotti, L.; and Perronnin, F. 2012. AVA: A large-scale database for aesthetic visual analysis. In *2012 IEEE conference on computer vision and pattern recognition*, 2408–2415. IEEE.
- Nichol, A. Q.; Dhariwal, P.; Ramesh, A.; Shyam, P.; Mishkin, P.; McGrew, B.; Sutskever, I.; and Chen, M. 2022. GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models. In *International Conference on Machine Learning*, 16784–16804. PMLR.
- OpenAI, O. 2023. GPT-4 Technical Report.
- Oppenlaender, J. 2023. A taxonomy of prompt modifiers for text-to-image generation. *Behaviour & Information Technology*, 1–14.
- Pavlichenko, N.; and Ustalov, D. 2023. Best prompts for text-to-image models and how to find them. In *Proceedings of the 46th International ACM SIGIR Conference on*

- Research and Development in Information Retrieval*, 2067–2071.
- Ponomarenko, N.; Ieremeiev, O.; Lukin, V.; Egiazarian, K.; Jin, L.; Astola, J.; Vozel, B.; Chehdi, K.; Carli, M.; Battisti, F.; et al. 2013. Color image database TID2013: Peculiarities and preliminary results. In *European workshop on visual information processing (EUVIP)*, 106–111. IEEE.
- Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.
- Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language Models are Unsupervised Multitask Learners.
- Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21(140): 1–67.
- Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; and Chen, M. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*.
- Ramesh, A.; Pavlov, M.; Goh, G.; Gray, S.; Voss, C.; Radford, A.; Chen, M.; and Sutskever, I. 2021. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, 8821–8831. PMLR.
- Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-Resolution Image Synthesis With Latent Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10684–10695.
- Saharia, C.; Chan, W.; Saxena, S.; Li, L.; Whang, J.; Denton, E. L.; Ghasemipour, K.; Gontijo Lopes, R.; Karagol Ayan, B.; Salimans, T.; et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494.
- Santana, G. 2022. Gustavosta/Stable-Diffusion-Prompts . Datasets at Hugging Face. <https://huggingface.co/datasets/Gustavosta/Stable-Diffusion-Prompts>. Accessed:2023-01-26.
- Sauer, A.; Karras, T.; Laine, S.; Geiger, A.; and Aila, T. 2023. StyleGAN-T: Unlocking the Power of GANs for Fast Large-Scale Text-to-Image Synthesis. volume abs/2301.09515.
- Schuhmann, C.; Beaumont, R.; Vencu, R.; Gordon, C.; Wightman, R.; Cherti, M.; Coombes, T.; Katta, A.; Mullis, C.; Wortsman, M.; et al. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35: 25278–25294.
- sshleifer. 2021. distilbart-cnn-12-6. <https://huggingface.co/sshleifer/distilbart-cnn-12-6>.
- Talebi, H.; and Milanfar, P. 2018. NIMA: Neural image assessment. *IEEE transactions on image processing*, 27(8): 3998–4011.
- von Platen, P. 2020. How to generate text: using different decoding methods for language generation with Transformers. <https://huggingface.co/blog/how-to-generate/>.
- Wang, Y.; Shen, S.; and Lim, B. Y. 2023. RePrompt: Automatic Prompt Editing to Refine AI-Generative Art Towards Precise Expressions. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, 1–29.
- Wang, Z. J.; Montoya, E.; Munechika, D.; Yang, H.; Hoover, B.; and Chau, D. H. 2022. Diffusiondb: A large-scale prompt gallery dataset for text-to-image generative models. *arXiv preprint arXiv:2210.14896*.
- Wen, Y.; Jain, N.; Kirchenbauer, J.; Goldblum, M.; Geiping, J.; and Goldstein, T. 2023. Hard Prompts Made Easy: Gradient-Based Discrete Optimization for Prompt Tuning and Discovery. In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Williams, R. J.; and Zipser, D. 1989. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2): 270–280.
- Wu, X.; Sun, K.; Zhu, F.; Zhao, R.; and Li, H. 2023. Better aligning text-to-image models with human preference. *arXiv preprint arXiv:2303.14420*.
- Zhang, W.; Ma, K.; Yan, J.; Deng, D.; and Wang, Z. 2020. Blind Image Quality Assessment Using A Deep Bilinear Convolutional Neural Network. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(1): 36–47.
- Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9): 2337–2348.
- Zhu, W.; Wang, X.; Lu, Y.; Fu, T.-J.; Wang, X. E.; Eckstein, M.; and Wang, W. Y. 2023. Collaborative Generative AI: Integrating GPT-k for Efficient Editing in Text-to-Image Generation.