Leipzig University
Institute of Computer Science
Degree Programme Digital Humanities, B.Sc.

Manipulating Embeddings of Stable Diffusion Prompts to Control Image Compositionality

Bachelor's Thesis

Dinara Imambayeva

1. Referee: Prof. Dr. Martin Potthast

Submission date: July 29, 2024

Declaration

Unless otherwise indicated in the text or references, this thesis is entirely the product of my own scholarly work.

Leipzig, July 29, 2024

.....
Dinara Imambayeva

Abstract

Due to the rapid development, diffusion models have achieved state-of-the-art results in the field of image generation. Among these, Stable Diffusion has become state-of-the-art open-source model with the ability to produce high-quality images. Despite the benefits the model offers, it still provides limited control over the image output. This affects particularly compositionally complex prompts containing multiple objects,which often result in a number of visual issues within the generated images. Such visual problems as missing objects, attribute leakage, incorrect number of objects, or incorrect attribute binding induces the user experience of lower quality and user intent remains unsatisfied. Adopted the idea of incorporating linguistic structure for improvement the image compositionality presented by the Structured Diffusion Guidance approach, we present three variations of the Simplified Structured Diffusion approach with additional pooling that addresses the problem of image compositionality and user control by Stable Diffusion. We assessed our approaches on both Stable Diffusion and Stable Diffusion XL and on two different datasets. The evaluation showed differences between the three approach variations and overall better performance for Simplified Structured Diffusion without pooling. Nevertheless, we could not achieve the results of baseline methods. Therefore, we address the limitations of our approach by presenting further experiments and introduce possible future research avenues.

Contents

Chapter 1

Introduction

In the last few years, the field of generative modelling has been undergoing rapid evolution, and diffusion models have been identified as a powerful class of techniques for generating high-quality images. Among these, Stable Diffusion [Rombach et al., 2022] stands out due to its robustness and capacity to produce detailed and diverse visual content. The widespread use and open availability of the Stable Diffusion model made it the state-of-the art open-source model for text-to-image generation. Its ability to produce an infinite variety of high-quality text-guided images made it popular among researchers as well as among regular users. Despite the success that Stable Diffusion has achieved, providing precise control over image composition remains a significant challenge [Feng et al., 2023, Liu et al., 2022]. The model faces a difficult task to effectively capture and reproduce the user's vision of the image. Therefore, there are two crucial points to consider. Firstly, the user must effectively convert their ideas in a prompt. Then, the model must produce an image that incorporates all the concepts and details defined by the user in order to give the user the desired result. Thus, the alignment of text and image is of significant importance with regard to the user experience. However, compositionally complex prompts that combine multiple objects present a major challenge for the model, as can be seen in Figure 1.1. In particular, the generation of an image for compositionally complex prompts demands from the model an understanding of individual linguistic concepts of prompts, as well as the comprehension of the entire scene simultaneously [Feng et al., 2023].

The crucial aspect for successful composition in generated images is correct attribute binding. In the first instance, it is essential to grasp the linguistic structure of the prompt, which is crucial for the generation of objects with correct attributes [Feng et al., 2023]. Therefore, implications of failed compositionality result in numerous visual problems that are shown in Figure 1.1. Missing objects, attribute leakage, incorrect number of objects, failed text

Figure 1.1: `a picture of a white cat and a black cat` produced by Stable Diffusion 1.5 with different seeds

visualization, and erroneous attribute bindings are the most common compositional failures that Stable Diffusion encounters. All of these issues lead to a lack of user control over the generated output, which can result in user frustration and loss of interest. The challenge of identifying effective textual prompts that facilitate the appropriate text-image alignment has given rise to the field of prompt engineering [Sun et al., 2023]. Although this development has led to the formation of a significant online community dedicated to the sharing of useful prompts and prompt engineering techniques, the method is not particularly user-friendly, as it often requires a considerable number of attempts to achieve the desired results. In this context, it is important to underscore the necessity for the development of additional approaches that address this problem by providing users with an option that requires minimal effort while still achieving the desired outcome.

Furthermore, it is necessary to note, that during work on this thesis, the newer Stable Diffusion XL 0.9 [Podell et al., 2023] was released. The new model has been released with a modified architecture, which enables it to generate high-resolution images and to exhibit enhanced performance compared to previous versions of Stable Diffusion. Although, Podell et al. [2023] shows what a significant overall improvement the model achieves, the Figure 1.2 illustrates that generating images with such a prompt as `a picture of a white cat and a black cat` is still challenging for the model with regard to correct attribute-binding. These examples demonstrate that the generation of images with compositionally complex prompts remains a challenging task, even for the recently released Stable Diffusion XL and remains a significant factor that affects the user experience. Therefore, the research of this thesis encompasses the Stable Diffusion XL as well.

Motivated by the concept of using the linguistic structure of the prompt for embedding manipulation presented in Structured Diffusion Guidance by Feng et al. [2023], we propose a modified method for improving the image composition. Our approach adopts the idea of manipulation of CLIP embeddings

Figure 1.2: `a picture of a white cat and a black cat` produced by Stable Diffusion XL with different seeds

according to the linguistic structure of the prompt from Structured Diffusion with additional manipulation step using pooling and is therefore referred to as Simplified Structured Diffusion. Accordingly, our approach investigates the role of CLIP embeddings in Stable Diffusion with regard to the compositionality problem. Furthermore, we seek to determine the ability of our approach to confer the same advantage as that observed in Structured Diffusion, without the need to manipulate cross-attention layers. The objective of this thesis is, therefore, to address the aforementioned composition problem in images generated by Stable Diffusion and to assess the Simplified Structured Diffusion approach.

The following content constitutes the structure of this thesis: Firstly, the theoretical understanding of the technical principles of diffusion models, with a particular focus on the Stable Diffusion and CLIP embeddings, is presented in Chapter 2. Subsequently, in Chapter 3, the studies that address the same compositional problem of the model will be described. Afterwards, the methodological framework of the Simplified Structured Diffusion will be presented in Chapter 4. In the following chapter, the undertaken experiments and results will be presented. Finally, a summary of our approach and results will be provided in the conclusion.

Chapter 2

Theoretical Background

This chapter provides an overview of the theoretical background of diffusion models, with a particular focus on the Stable Diffusion and Stable Diffusion XL models. The understanding of the theoretical background and key concepts is important for highlighting the possible underlying reasons of such compositional problems in output images, such as attribute binding failure and object missing. Furthermore, it is essential to grasp these concepts in order to comprehend the selected and proposed methods presented in this thesis.

Stable Diffusion refers to the latent diffusion models that operate in a low-dimensional latent space, which enables to reduce computational effort during the training process. Therefore, this chapter begins with an explanation of what a latent diffusion model is and what the processes behind diffusion are. Then, the underlying architecture of the models is closely examined, showing the differences between the two models and the consequent influence on the generated output and overall performance. Subsequently, the CLIP encoder used in the Stable Diffusion models is examined in detail. In particular, the CLIP embedding generation process is closely observed, as it plays a crucial role in the proposed Simplified Structured Diffusion approach. Then, the image generation process within Stable Diffusion and Stable Diffusion XL is considered. Finally, the cross-attention conditioning mechanism used in both models and the pooling method are presented.

2.1 Latent Diffusion Models

Diffusion probabilistic models [Sohl-Dickstein et al., 2015] are generative likelihood-based models that succeed in many tasks as image synthesis, colorization, inpainting and other [Rombach et al., 2022]. Diffusion models arose from the principles of non-equilibrium statistical physics and are crucial for the development of the Stable Diffusion model. The main idea of diffusion is to

destroy the data distribution structure systematically and gradually within an iterative process and then, to train the model how to reconstruct the data structure during a reverse diffusion process [Sohl-Dickstein et al., 2015] (see Figure 2.1). These two main processes within the diffusion models are realized through the implementation of two Markov chains: forward and reverse, which will be considered in more detail in this section.

Compared to diffusion models trained in a pixel space, which require high computational effort, latent diffusion models are trained in a lower-resolution latent space. This is an efficient, lower-dimensional space that is equivalent to the data space, but excludes high-frequency, imperceptible details [Rombach et al., 2022]. Consequently, latent diffusion models use a variational autoencoder [Kingma and Welling, 2014] to encode images into a latent space and then decode them back from latent into a pixel space. The variational autoencoder is trained separately from the diffusion process and therefore, the compressive operations are separated from a generative learning phase (Rombach, 2022). Another advantage of process separation is that the variational autoencoder needs to be trained only once and can then be used to train multiple diffusion models, further reducing training time [Rombach et al., 2022].Overall, the crucial change from pixel to latent space introduced in the latent diffusion models, as well as the separation of compression and diffusion process, drastically reduce computational complexity while still producing high-resolution images compared to previous methods operating in pixel space [Rombach et al., 2022].

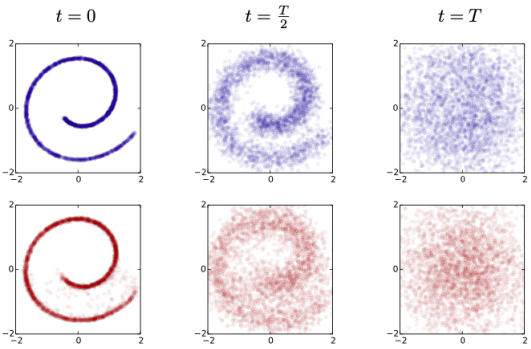


Figure 2.1: Forward process (from left to right at the top) and reverse process (from right to left at the bottom) at time steps $t = 0, t = T/2, t = T$, reproduced from Sohl-Dickstein et al. [2015]

Forward Process

The forward Markov chain features the forward diffusion process in latent diffusion models. It iteratively adds some noise to the initial image data at

each time step t to convert complex data distribution into a simple, controllable distribution, a Gaussian distribution [Ho et al., 2020, Sohl-Dickstein et al., 2015].

The input data distribution, denoted as $q(x_0)$, is then turned to $q(x_T)$ in the last timestep T by the diffusion process, whereby the resulting data distribution x_T contains only Gaussian noise, $x_T \sim \mathcal{N}(0, I)$. Applying the chain rule of probability, the diffusion process can be represented as an approximate posterior $q(x_{1:T}|x_0)$ [Ho et al., 2020]. Therefore, the generation of each x_t in every timestep t is conditioned by its predecessor x_{t-1} and is noted as $q(x_t|x_{t-1})$.

$$q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}) \tag{2.1}$$

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I) \tag{2.2}$$

The noise added to the image within each timestep is controlled by a noise scheduler. The noise scheduler calculates the amount of noise to be added within each timestep t according to a variance schedule $\beta_1, ..., \beta_T$, where the parameter β defines the diffusion rate within the forward process. This diffusion rate β can be learned by reparameterization or, in some cases, can be set to a constant value [Ho et al., 2020, Sohl-Dickstein et al., 2015]. Nevertheless, the reparameterization trick, described in Ho et al. [2020], enables to calculate the noise for each timestep t without using all the predecessing x_ts for calculation.

Reverse Process

Otherwise, the reverse Markov chain learns a finite-time denoising mechanism to reconstruct the initial data by converting a simple Gaussian distribution back into a complex target distribution. This process is also known as a generative diffusion process [Sohl-Dickstein et al., 2015] and is a reverse of the forward process. Therefore, to learn the reverse process of data reconstruction, the model p_θ must be trained for obtaining conditional probabilities $p_\theta(x_{t-1}|x_t)$ for each timestep t.

The learning succeeds on estimating mean $\mu_\theta(x_t, t)$ and covariance $\sum_\theta(x_t, t)$ functions within each timestep t [Sohl-Dickstein et al., 2015]. That means, the p_θ predicts noise to be removed for particular timestep t. The predicted noise is then subtracted from x_t and the process is repeated for each timestep up to x_0.

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t) \tag{2.3}$$

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{2.4}$$

2.1.1 Denoising Architecture

The underlying denoising architecture for reverse process of the denoising diffusion probabilistic model implemented by Ho et al. [2020] uses a U-Net architecture presented by Ronneberger et al. [2015], which is illustrated in Figure 2.2. Moreover, this architecture enables connection with an attention mechanism, which makes realization of the conditioning mechanism possible [Rombach et al., 2022].

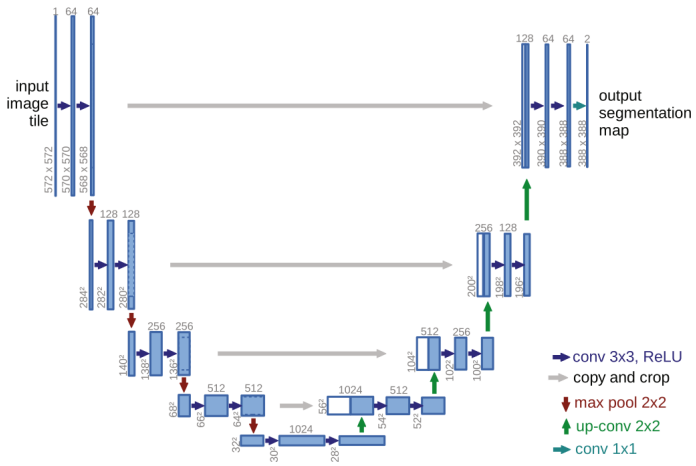


Figure 2.2: U-Net architecture, reproduced from Ronneberger et al. [2015]

The U-Net architecture was firstly introduced for the ISBI challenge for segmentation of neuronal structures in electronic microscopic stacks in 2015 and this implemented approach won the challenge. Since then, U-Net architecture is widely used for visual computation tasks.

Essentially, the U-Net architecture is based on the "fully convolutional network" [Long et al., 2014]. The authors of the U-Net supplemented some pooling layers with upsampling layers and established a U-shaped network that works analogous to an autoencoder. In the contracting path it acts like an encoder and downsamples the image, reducing the resolution up to a "bottleneck" at the bottom. Whereby the expansive path of the U-Net represents a decoder that upsamples the compressed hidden representation of the image back to its

original dimensions. The downsampling part of the U-Net is designed as a typical convolutional network (CNN) with convolution layers including a rectified linear unit (ReLu) and max pooling operation layers. In contrast, the upsampling decoder path includes extra residual connections to the downsampling encoder, which is not typical for an autoencoder architecture. This innovation prevents the model from losing contextual information during downsampling process and enables the model to create high resolution images and localize objects within the image [Ronneberger et al., 2015].

Stable Diffusion Denoising Architecture

The neural backbone of the Stable Diffusion is a modified time-conditional U-Net, which works as a noise predictor for the reverse diffusion process. In order to accept noised latents at specific timestep t, the information about current timestep should be additionally transferred. For this sake, the network uses transformer sinusoidal position embedding [Vaswani et al., 2017] in every downsampling and upsampling block [Erdem, 2023, Ho et al., 2020]. As the U-Net predicts the noise that should be removed from the latent within each timestep, a scheduler algorithm computes the samples with subtracted predicted noise and controls the noise level at each timestep [Patil et al., 2022, Wong, 2024]. The U-Net takes a created noised latent of size 64×64 as input and downsamples it twice in each level in the contracting path until it reaches the "bottleneck" at the bottom. In the expansive path, the network upsamples the latents back to the original size.

As shown in Figure 2.3, the U-Net network of Stable Diffusion model is comprised of ResNet blocks [He et al., 2016] and transformer blocks [Patil et al., 2022, Tian et al., 2023]. Thereby, the transformer blocks implement both self-attention and cross-attention layers, that are crucial for the conditioned image generation process. The self-attention layer learns attention across the image and enables saving the spatial structure shape throughout the whole image generation process [Liu et al., 2024, Tian et al., 2023]. Whereby the cross-attention layer learns attention between image and text prompt enabling image generation according to a given text prompt [Liu et al., 2024, Tian et al., 2023].

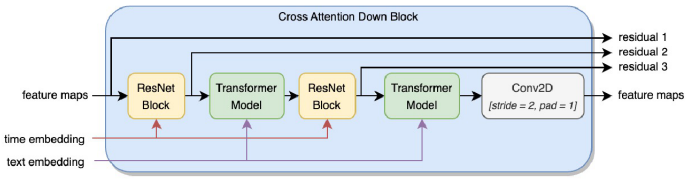


Figure 2.3: Cross-attention downblock module from the time-conditional U-Net architecture of Stable Diffusion, reproduced from Voetman et al. [2023]

Stable Diffusion XL Denoising Architecture

The U-Net architecture of the Stable Diffusion XL model slightly differs in comparison to the previous Stable Diffusion models. It has a three times larger U-Net backbone with more transformer blocks and removed lowest downsampled level [Podell et al., 2023]. Moreover, the model uses bigger latents of size 128×128, which are four times as large than latents used in Stable Diffusion [Podell et al., 2023]. Nevertheless, the number of U-Net parameters increased from 860M for Stable Diffusion to 2.6B in Stable Diffusion XL [Podell et al., 2023]. All these novelties enable the model to produce high-resolution images with better fidelity and better user satisfaction in comparison to the previous Stable Diffusion models [Podell et al., 2023].

Aside from that, the authors of Stable Diffusion XL presented additional refinement stage that boosts performance of the model [Podell et al., 2023].

2.2 CLIP Text Encoder

In order to enable generation of images based on textual descriptions, it is essential firstly to transform text into an appropriate embedding space. Therefore, Stable Diffusion model uses the frozen CLIP (Contrastive Language-Image Pretraining) [Radford et al., 2021] text encoder for this task. CLIP is a pre-trained visual and language model that has been trained on a vast dataset of 400 million image-text pairs. It is used for tasks such as defining image-text similarity and zero-shot image classification [huggingface].

The CLIP text encoder architecture is based on the GPT-2 [Radford et al., 2019] architecture and refers to a transformer [Vaswani et al., 2017]. It is a decoder only transformer with 12 layers and 8 attention heads, which operates on a lower-cased byte pair encoding of the text [Radford et al., 2021]. The attention layer represents a masked multi-head self-attention layer, which uses a causal mask to enable using of pre-trained language model as an additional input to the model [Radford et al., 2021].

The model is trained using a contrastive objective and learns how to predict correct image-text pairs among others. In the pre-training phase, both image and text encoders are trained simultaneously to maximize the cosine similarity between image and text embeddings of correct pairs while minimizing the cosine similarity of incorrect pairs [Radford et al., 2021]. While testing, the text encoder generates a zero-short linear classifier by using the names of the target dataset's classes [Radford et al., 2021]. Thereby, CLIP learns a multimodal embedding space, where image and text are represented by embeddings of the same size and could be compared. For mapping both image and text embeddings into the multimodal embedding space, both embeddings should be firstly projected through a linear layer [Radford et al., 2021].

2.2.1 CLIP Embeddings Generation

This chapter takes a closer look at the process of CLIP embeddings generation. Thus, the approach introduced in this thesis concentrates on the modification of CLIP embeddings to improve image compositionality, therefore, it is crucial to understand the process of embedding generation. The following text is based on the work of Howard [2022].

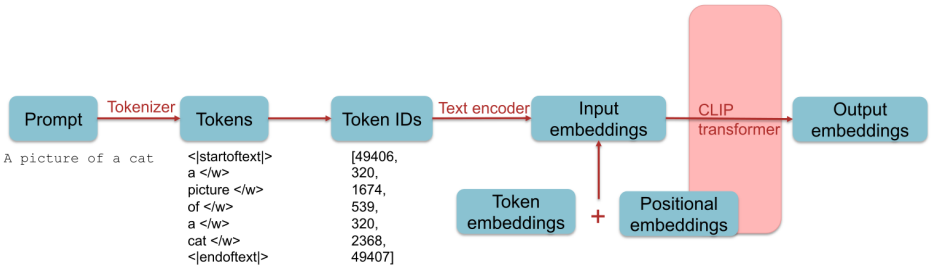


Figure 2.4: CLIP embeddings generation process

As shown in Figure 2.4, the process of embedding generation within the CLIP encoder starts with a tokenization step. To this end, a CLIP tokenizer is applied to the prompt. Furthermore, throughout this process two special tokens, start token [SOS] or [BOS] and end token [EOS], are added at the beginning and at the end of the sequence. These special tokens serve a functional purpose in the subsequent processing stages. Therefore, the start token is a crucial and obligatory element, as it signals the beginning of the prompt for the model. For this reason, the start token cannot be modified. This is a pivotal consideration for approaches utilising CLIP embeddings modifications as the approach presented in this thesis. The impact of the start token can be observed in Figure 2.5.

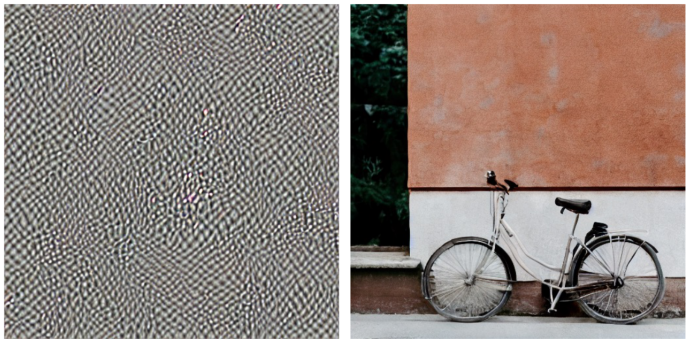


Figure 2.5: Comparison between images generated with Stable Diffusion 1.5. Left image: applying pooling over all tokens including start token. Right image: pooling over all tokens excluding start token

While using a tokenizer, it is possible to determine the total length of the prompt by specifying a "padding" argument, which is set to the maximum length of 77 tokens by default. This makes it possible to use prompts with the length of up to 75 tokens, which is restricted to control the computational effort [Radford et al., 2021]. In the case when the prompt contains less tokens than 75, the empty places are filled with padding tokens. For this purpose, the CLIP tokenizer uses the [EOS] token as a padding token. After the tokenization step, all tokens are translated to the corresponding token ids, which are stored in the vocabulary. Therefore, the vocabulary used by the CLIP text encoder contains 49152 tokens, where start and end tokens feature their own token ids.

In the next step the input embedding will be generated. For this purpose, the encoder uses token and positional layers to generate token and positional embeddings, respectively. Therefore, for token embedding creation each token is mapped to a 768-dimensional vector. Thus, the token embeddings are represented by a tensor of a shape [1,77,768]. Subsequently, positional embeddings of the same shape are added to the token embeddings, which give the model information about the position of each token in the sequence. After both embeddings are summarized, the generated input embedding can be fed to a CLIP transformer to get the output embedding, which is the sequence of hidden states of the model's last layer output that is further used for conditioning within the Stable Diffusion. Due to use of causal mask in the multi-head self-attention layer in CLIP encoder, the generated output embeddings are cumulative. It means each token position in the embedding also includes information of the preceding tokens. Due to this fact, the information about attributes of the preceding objects is also contains in the following objects and could lead to the failed attribute binding in the image. This feature can be crucial for failed composition in the generated images of Stable Diffusion [Feng

et al., 2023].

2.3 Image Generation Process With Stable Diffusion

Text-to-image generation process within Stable Diffusion contains three main components: prompt encoding with a frozen CLIP Vit-L encoder, a diffusion process realized by a time-conditioned U-Net and image decoding with a variational autoencoder (see Figure 2.6).

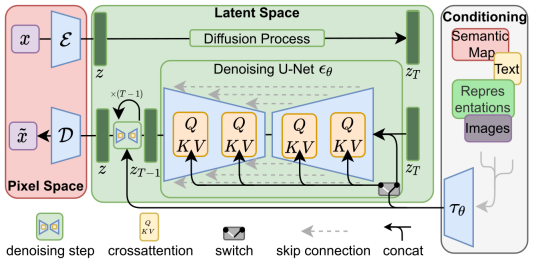


Figure 2.6: Conditioned latent diffusion model architecture, reproduced from Rombach et al. [2022]

A variational autoencoder used in Stable Diffusion is an autoencoder constructed of an encoder $\mathcal{E}$ that encodes the image $x \in \mathbb{R}^{H \times W \times 3}$ from pixel space to its lower-dimensional latent representation $z \in \mathbb{R}^{h \times w \times 3}$, whereby $z = \mathcal{E}(x)$ and a decoder $\mathcal{D}$ that reconstructs the generated image $\tilde{x} = \mathcal{D}(z)$ back to the pixel space.

Stable Diffusion enables different inputs as a conditioning element, such as a text prompt, an image, or a semantic map [Rombach et al., 2022]. In this case the Stable Diffusion represents a conditional latent diffusion model (LDM) with a conditional denoising autoencoder of form $\epsilon_\theta(z_t, t, y)$ where ϵ_θ is a denoising autoencoder represented by a time-conditioned U-Net, which predicts the noise to be removed from z_t. Then, z_t is the latent for timestep t and y is a conditional input such as a text prompt, which is pre-processed by an encoder τ_θ. Therefore, the model training objective is defined as followed:

$$L_{LDM} := \mathbb{E}_{\mathcal{E}(x),y,\epsilon\sim\mathcal{N}(0,1),t}\left[||\epsilon - \epsilon_\theta(z_t, t, \tau_\theta(y))||_2^2\right] \tag{2.5}$$

In case of the text-to-image generation process the model requires two inputs: a text prompt and a seed. The prompt should be firstly encoded by an encoder τ_θ, which is realized by a frozen CLIP ViT-L encoder that transforms the prompt into a CLIP embedding, which was described in section 2.2. With

a given seed the model creates a normally distributed random latent map with a size $\mathbb{R}^{h \times w \times c}$, which is 64×64. The U-Net denoising autoencoder takes the created latent and uses it as a z_T, which is the starting point for the diffusion process. The network along with a scheduler begin iteratively predicting noise within each timestep t that should be removed from the latents until $t = 0$ (see section 2.1). The prediction is conditioned by a given CLIP embedding that is used in cross-attention in both upsampling and downsampling transformer blocks of the U-Net.

Finally, the denoised latent z_0 is then decoded by a decoder of a variational autoencoder $\mathcal{D}$ back to a pixel space and given as an output.

2.3.1 Cross-Attention

The attention mechanism is a key component in neural network architectures since the transformer architecture was introduced by Vaswani et al. [2017]. The attention mechanism enables models to dynamically assign different weights to different parts of the input sequence, emphasizing the most relevant information for each step of processing. For this sake, the attention function outputs the weighted sum of the values by using calculated weights assigned to each value [Vaswani et al., 2017]. The input for the function is a set of query and a key-value pairs, which are all vectors.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2.6}$$

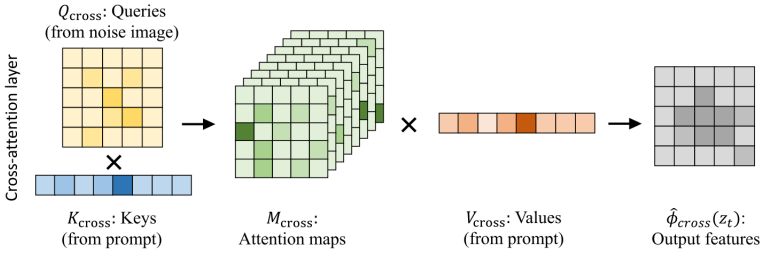


Figure 2.7: Cross-attention layer in Stable Diffusion, reproduced from Liu et al. [2024]

In order to enable conditioning on the given text within the image generation process of Stable Diffusion, cross-attention layers are used in the model (see Figure 2.7). In this case, the keys and values are derived from the CLIP prompt embedding, whereby the query is provided by an image latent, which is initialized with a given seed. Due to this mechanism the model is trained to

predict the residual noise in order to get the desired image defined by a given prompt.

2.3.2 Image Generation With Stable Diffusion XL

The image generation process within the Stable Diffusion XL model is essentially identical to that of the Stable Diffusion model. The most important difference between the two models, which is also crucial for the approach presented in this thesis, is the prompt embedding generation step. In comparison to the Stable Diffusion model, the Stable Diffusion XL uses two pre-trained CLIP encoders: the CLIP ViT-L text encoder, which is also used in the Stable Diffusion model, and the CLIP ViT-bigG [Ilharco et al., 2021]. The CLIP ViT-bigG is a more powerful text encoder, which was trained on a large dataset of 2B samples and achieved a zero-shot top-1 accuracy of 80.1% on ImageNet-1k. Furthermore, ViT-bigG's tokenizer encodes tokens into larger vectors that represent more features. It maps each token id to a vector of length 1280, hence the final output embedding of the encoder has the size of [1,77,1280].

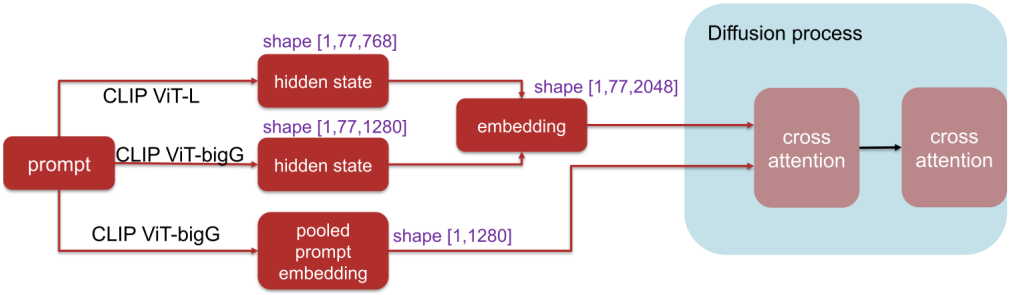


Figure 2.8: Prompt embedding generation in Stable Diffusion XL

The process of prompt embedding generation for Stable Diffusion XL begins with the prompt encoding stage. Consequently, the prompt is encoded with both CLIP text encoders. Another notable difference between the two models is that, in contrast to Stable Diffusion, Stable Diffusion XL employs the outputs of penultimate hidden states, rather than the outputs of the final hidden state. Subsequently, the outputs of the two CLIP encoders are then concatenated along the channel axis to construct an embedding of the shape [1, 77, 2048] (see Figure 2.8) [Podell et al., 2023].

Furthermore, one of the outputs of the CLIP ViT-bigG encoder is used as an additional conditioning input for Stable Diffusion XL. This embedding is located in the multimodal embedding space of the CLIP encoder and is

constructed through application of a projection layer to the pooler output of CLIPTextModel. Due to the linear projection the final embedding has a shape of [1,1280]. Finally, the described outputs of both CLIP embeddings are used as a conditioning mechanism in the cross-attention layers of the Stable Diffusion XL model.

2.4 Pooling

Pooling is a technique employed in the field of natural language processing (NLP) that captures the meaning of an entire sequence rather than focusing on individual tokens [Leys, 2022]. Consequently, the pooling operation is defined as a compression of token-level representation to a single representation, which still preserves the meaning of the entire sequence [Leys, 2022]. This technique is used for such tasks as sentence pooling, next sentence prediction, semantic similarity or sentiment analysis [Leys, 2022]. There are several types of pooling techniques, each of which is defined by the aggregation function used for compression. One of the most popular pooling techniques is mean pooling, which involves averaging the contextual token embeddings. Mean-square pooling is similar to mean pooling but involves additional division by the square of the number of elements [Leys, 2022]. Additionally, there are min and max pooling approaches, which utilize the maximal or minimal value from the sequence.

One of the most prominent applications of the pooling technique in pre-trained language models is BERT [Devlin et al., 2019]. It is a bidirectional transformer that is capable of solving a multitude of NLP tasks, including sentiment analysis, text generation, text prediction, question answering and summarization [Muller, 2022]. In a similar manner, the application of this technique at the token embedding level in Stable Diffusion can influence the image output with regard to the composition problem, which was initially investigated by Smith [2023].

Chapter 3

Related Work

This chapter covers previous research addressing to the problem of image compositionality within diffusion models and particularly within Stable Diffusion. Within the arose popularity of diffusion models, different image generation guidance techniques were introduced.

Firstly, an overview of different approaches that address compositional problem within image generation process with Stable Diffusion will be presented. Afterwards, the Structured Diffusion Guidance proposed by Feng et al. [2023] is presented because it presents a base idea of our approach.

Previous approaches addressed to the compositional problems within diffusion models follow different techniques. Fan et al. [2023] and Lee et al. [2023] use human feedback and reinforcement learning for fine-tuning the pre-trained diffusion model like Stable Diffusion. Both approaches collect human feedback on generated images, which is then used for training a reward function. Furthermore, this function is used for diffusion model optimization by maximizing the reward-weighted likelihood. As a result, the model is trained to produce more desirable images for users with better compositionality Lee et al. [2023]. However, this approach requires obtaining of labeled data and additional model training, which demand time and effort.

Another approach, recently introduced by Hertz et al. [2022] presents a text-driven image editing. A prompt-to-prompt editing framework is based on the manipulation of cross-attention layers controlled by prompt manipulation in user interface. The underlying process works by identifying which pixels are associated with which tokens and allows controlled attention to the selected tokens. Therefore, this approach enables selective editing of objects as well as change of the image style by remaining the image composition. However, in order to edit the image, the user should firstly use a suitable prompt, which initially generates an image with correct objects [Hertz et al., 2022]. That makes the approach less suitable for producing images from compositional

challenging prompts.

An approach introduced by Gandikota et al. [2023] focuses on using Concept Sliders, which are plug-and-play low-rank adaptors for better control over image generation process and post-hoc image editing. The authors presented sliders for different concepts as weather, age, styles, expressions as well as composition up to 50 different sliders. The approach requires single inference phase and focuses on enhancing overall image composition as well as on fixing the problem of hand distortions, which is also known for Stable Diffusion and Stable Diffusion XL. Although concentrating on enhancing the overall image compositionality and presenting improvement of generated images, this approach does not fully address the problem of failed attribute-binding or missing objects directly.

Another auspicious method, which also uses fine-tuning with low-rank adaptors as a part of the presented technique, is DreamSync, an approach introduced by Sun et al. [2023]. This approach is based on the recent finding about vision-language models that can identify discrepancies between text prompt and generated image via using visual question answering [Hu et al., 2023]. Therefore, the method introduced by Sun et al. [2023] firstly generates the images within the pre-trained diffusion model. Afterwards two visual language models are applied for alignment measurement between image and text via question answering and for anesthetics measurement. After selecting the best images low-rank adaptors are iteratively applied for the model fine-tuning. This method showed enhancement in both semantic alignment and aesthetics within Stable Diffusion 1.4 and Stable Diffusion XL [Sun et al., 2023]. Nevertheless, the method is limited by the chosen diffusion model. For this reason, such composition problems as correct depicting of attribute-objects is still challenging.

In their study, Liu et al. [2022] introduce Composable Diffusion, which generate images by combining multiple outputs from a pre-trained diffusion model. Each output is designed to capture distinct image elements, subsequently merged using compositional operators to produce a cohesive image. However, it is noted that this technique frequently faces challenges in creating realistic compositions of multiple objects. Moreover, the method is restricted to using only two operators: conjunction and negation.

To provide better image compositionality in generated images by Stable Diffusion, Chefer et al. [2023] introduced their method Attend-and-Excite, which operates on the cross-attention layer of pre-trained the Stable Diffusion model. The method concentrates on attending all tokens to some image patch in the cross-attention and intensifies the appearance of the object by strengthen the activation. This approach could show improvements in image fidelity and compositionality.

3.1 Structured Diffusion

In parallel with the growing popularity of diffusion models, numerous approaches have emerged for enhancing text image alignment and compositionality in the generated images. One such approach is the methodology introduced by Feng et al. [2023], called Structured Diffusion Guidance. The approach incorporates linguistic properties of the prompts within the diffusion guidance process for better attribute-binding and consequently better image compositionality and image-text alignment.

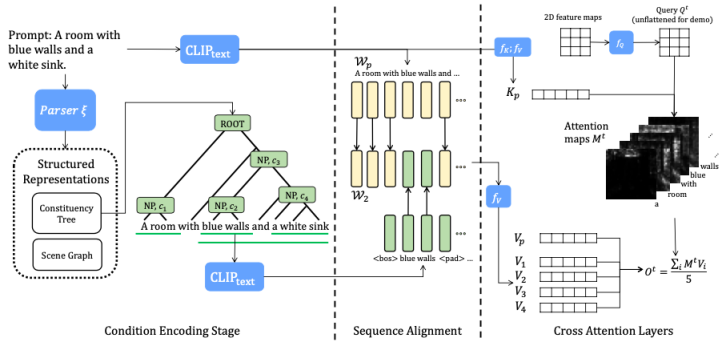


Figure 3.1: Design of Structured Diffusion Guidance, reproduced from Feng et al. [2023]

The methodology contains two major steps. Firstly, the linguistic structure of the prompt is explored. For this step, the approach allows to use either constituency trees or scene graphs. The obvious benefit of using those approaches is providing attribute-object pairs without extra computational cost [Feng et al., 2023]. For instance, the authors use language parser for building a constituency tree, which provide all noun phrases from all hierarchical levels $C = \{c_1, c_2, ..., c_k\}$ and their spans for further processing. After this step, the prompt and the noun phrases are encoded separately by a CLIP encoder. Afterwards, all calculated embeddings W_i between <bos> and <pad> are inserted into W_p. The modified embedding of $\overline{W_p}$ is then used to generate key K_p for cross-attention layers and to compute attention maps M_t (see the Figure 3.1). Moreover, the encoded embeddings for each noun phrase are used to compute values V_i, which are then used to calculate the O_t, an output of a cross-attention layer for timestep t. The authors could achieve 5-8% advantage in comparison to state-of-the-art Stable Diffusion model in user comparison studies.

Chapter 4

Methodology

This chapter is devoted to present our methodology for enhancing attribute-binding and compositionality of images generated by Stable Diffusion and Stable Diffusion XL models. It builds on the linguistic approach of the Structured Diffusion Guidance proposed by Feng et al. [2023], integrating an additional mean-square pooling technique at the embedding level presented by Smith [2023] and described in the section 2.4. The primary objective of the selected methodology is to investigate the impact of linguistic properties applied exclusively at the embedding level, without any manipulation of the cross-attention layers. As observed by Feng et al. [2023], the most significant impact on the compositionality failure in the generated images is attributed to the cumulative nature of the CLIP embeddings. This outcome is the result of the implementation of causal attention masks in CLIP for prompt encoding, which in turn results in the semantic blending of all previous tokens in the embedding. Furthermore, this blending process ultimately leads to the observed deficiencies in the compositional quality of the generated images [Feng et al., 2023]. For this reason, our methodology focuses solely on the embedding level manipulation.

4.1 Simplified Structured Diffusion

The Structured Diffusion Guidance approach was chosen as the basis of our methodology because it addresses the problem of failed attribute-binding and missing objects, which are the most common composition errors in the images generated by Stable Diffusion. Our method is referred to as a Simplified Structured Diffusion because it focuses solely on prompt embedding manipulation and does not alter the cross-attention layers of the model, as it is done in the original approach. Instead of that, we introduce a technique of CLIP prompt embedding manipulation via separate noun phrase encoding and additional pooling step. The focus on the CLIP embeddings is referred to the

fact of cumulative nature of the embeddings due to usage of causal attention mask, which could be a potential cause of failed attribute binding in the images generated by Stable Diffusion [Feng et al., 2023].

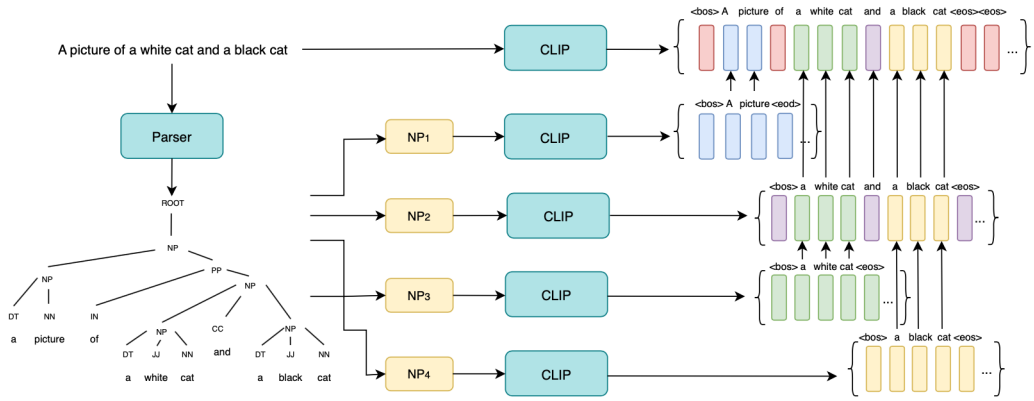


Figure 4.1: Simplified Structured Diffusion: prompt embedding construction

We adopt the idea of separate encoding of noun phrases from Structured Diffusion Guidance. This approach should intensify the connection between attribute and object and reduce the negative outcomes caused by the cumulative nature of CLIP embeddings. The separate encoding of noun phrases ensures that there is no semantic blending with the previous tokens that lead to failed attribute-binding. The application of an additional mean-square pooling of noun phrases allows to link attribute and object even more strongly together. Nevertheless, in contrast to the Structured Diffusion approach, where the authors alter the cross-attention layers, we manipulate the prompt embedding itself. In the first step, the linguistic structure of the prompt should be defined. For this sake, we use stanza language parser [Qi et al., 2020] and NLTK tree package [Bird et al., 2009], both open-source toolkits for NLP. The tools are used for parsing and building constituency tree for each prompt. Therefore, in the first step of the Simplified Structured Diffusion the language parser ξ provides a set of all noun phrases in prompt from different hierarchical levels $\mathcal{C} = [c_P, c_1, c_2, \ldots c_n]$ and the list of corresponding spans $\mathcal{S} = [s_P, s_1, s_2, \ldots, s_n]$.

After the noun phrase extraction, the encoding phase starts. In this phase, the prompt c_P and all extracted noun phrases c_i are encoded into CLIP embeddings separately.

$$\mathcal{W} = [W_P, W_1, W_2, ..., W_n] \tag{4.1}$$

$$\mathcal{W}_P = CLIP(c_P) \tag{4.2}$$

$$\mathcal{W}_i = CLIP(c_i), i = 1, ..., n \tag{4.3}$$

Afterwards, the encoded noun phrases W_is are added back into the main prompt W_P replacing original values (see Figure 4.1 and Algorithm 1). For this step, the vectors from the noun phrase embedding of each noun phrase between [SOS] and [EOS] are extracted and, then, added into the corresponding position in the main prompt using extracted spans $\mathcal{S}$. In case of complex noun phrases including sub-noun phrases, the encoding begins from the lowest level. With this approach, we ensure that encoded noun phrases do not contain any semantic information from the predecessing tokens. Finally, the manipulated embedding $\overline{W_P}$ is used as an input for Stable Diffusion for image generation.

Algorithm 1 Simplified Structured Diffusion

Input: Prompt P, language parser ξ, trained diffusion model ϕ
Output: image x

Retrieve noun phrases set $C = [c_P, c_1, c_2, ..., c_n]$ by traversing $\xi(P)$
$W_P \leftarrow CLIP(c_P), W_i \leftarrow CLIP(c_i)$;
for each W_i in $[W_n, W_{n-1}, ..., W_1]$ **do**
    pool(W_i) ▷ additional pooling step
    add W_i into W_P ▷ embedding noun phrase into prompt
end for
Feed $\overline{W_P}$ to ϕ to generate x

According to the different underlying processes of embedding generation for Stable Diffusion XL, the described method is adapted. Since the model uses two CLIP encoders, the whole procedure is also applied to both CLIP embeddings W_P and W'_P. After the embedding generation step and the addition of noun phrases to both CLIP prompt embeddings, both embeddings are concatenated along the channel axis to produce a single embedding of the given size, which is denoted as $\overline{W_{P(XL)}}$ (see Algorithm 2). Furthermore, since the Stable Diffusion XL model requires two conditioning inputs, we also investigated the possible approach with the additional manipulation of the pooled prompt embedding E_P. Nevertheless, our empiric tests did not show any improvement by additional manipulation of the pooled prompt embedding. Therefore, the final version of the Simplified Structured Diffusion approach for Stable Diffusion XL contains no manipulation of the pooled prompt embedding (see Algorithm 2).

Algorithm 2 Simplified Structured Diffusion for Stable Diffusion XL

Input: Prompt P, language parser ξ, trained diffusion model ϕ
Output: image x

Retrieve noun phrases set $C = [c_P, c_1, c_2, ..., c_n]$ by traversing $\xi(P)$
$W_P \leftarrow CLIP(c_P), W_i \leftarrow CLIP(c_i)$;
for each W_i in $[W_n, W_{n-1}, ..., W_1]$ **do**
    pool(W_i) ▷ additional pooling step
    add W_i into W_P ▷ adding noun phrase into prompt
end for

$W'_P \leftarrow CLIP2(c_P), W'_i \leftarrow CLIP2(c_i)$;
$E_P \leftarrow CLIP2(c_P)$ ▷ generate pooled prompt embedding
for each W'_i in $[W'_n, W'_{n-1}, ..., W'_1]$ **do**
    pool(W'_i) ▷ additional pooling step
    add W'_i into W'_P ▷ adding noun phrase into prompt
end for

Concatenate $\overline{W_P}$ and $\overline{W'_P}$ to $\overline{W_{P(XL)}}$
Feed $\overline{W_{P(XL)}}$ and E_P to ϕ to generate x

4.2 Pooling

The following step engages an additional embedding manipulation via pooling. As there are few different pooling techniques, we chose one of the most frequently used, the mean-square pooling. The pooling technique is applied again over the extracted noun phrases aiming the amplification of attribute-binding. Furthermore,as shown in Figure 4.2, we explored two different implementation types of pooling. Firstly, the pooling of noun phrases is implemented as a single vector on the first position of the noun phrase in the prompt embedding. The remaining positions of the noun phrase in the prompt embedding are set to zero. This technique is adopted from Smith [2023]. The second type of implementation is realized by embedding the pooled vector over each position of the noun phrase in the CLIP embedding of the main prompt. This technique, assigned as "poolnz" for no-zero-pooling, was additionally chosen as the alternative without setting vectors to zero and avoiding possible implications from it.

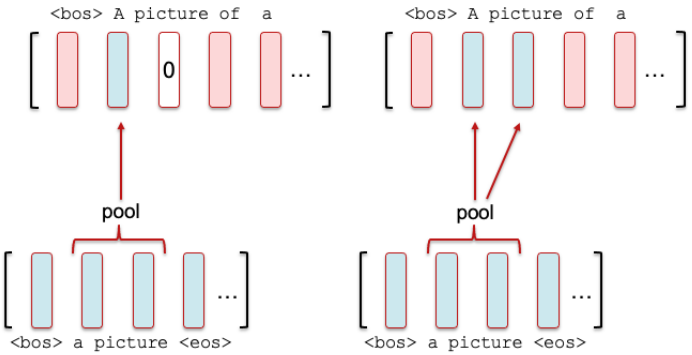


Figure 4.2: Two pooling techniques applied in the method

Chapter 5

Experiments and Results

The principal objective of this chapter is to present a comprehensive analysis of the experiments conducted to evaluate the performance of the three proposed variants of the Simplified Structured Diffusion approach. These methods are subjected to a detailed comparison with both Stable Diffusion models, namely Stable Diffusion 1.5 and Stable Diffusion XL 0.9. The objective of the evaluation is to test the thesis hypothesis. Specifically, we aim to ascertain whether embedding level manipulation based on the linguistic structure of the prompt can yield results that are as effective as those achieved by the baseline methodology, Structured Diffusion as presented by Feng et al. [2023]. Moreover, the analysis of the experiments is designed to identify the relative strengths and weaknesses of each approach, as well as to provide a potential for improvement.

This chapter begins with an outline of the experimental setup, including a description of the datasets and the evaluation metrics employed. Subsequently, a comprehensive account and assessment of the findings will be provided. In conclusion, an additional study will be presented with the intention of identifying potential avenues for future research and optimization.

5.1 Experimental Setup

For our experimental setup, we generate images using both Stable Diffusion and Stable Diffusion XL as baseline methods. To test our approach, we firstly generate images with Simplified Structured Diffusion without pooling technique and afterwards, the method is augmented by an additional pooling step with both pooling methods described in the chapter 4. Those three approach variations are applied to both Stable Diffusion and Stale Diffusion XL models, as they both address the problem of failed composition. To ensure the replicability and comparability of the methods, all image generation processes for a given prompt are initialized with a generated pseudo-random seed.

5.2 Dataset

In order to conduct an effective evaluation of our approach, it is essential that the chosen dataset meets specific criteria. In order to highlight the compositional problem the prompts must be challenging for Stable Diffusion. Therefore, we chose two existing datasets to evaluate our methodology. Firstly, we use ABC-6K dataset presented by Feng et al. [2023]. This dataset contains 3.2K prompts from the MSCOCO [Lin et al., 2014] dataset, which was created for object detection, segmentation, key-point detection, and captioning. The selected prompts contain multiple objects with at least two different colors and therefore are challenging for compositional depiction. For better contrast caption, the authors Feng et al. [2023] added contrastive prompt with replaced colors for each prompt expanding the dataset up to 6.4K prompts or 3.2K prompt pairs. Nevertheless, upon a detailed analysis of the dataset, it became evident that the dataset contained duplicates. Therefore, after removing the duplicates the total number of prompts was reduced to 6321.

Alongside, we use a second dataset CC-500 [Feng et al., 2023], which contains 588 prompts with exact two objects and two different colors of form ‘a red bench and a green car”. Although this prompt pattern seems simple, as shown by Feng et al. [2023], they are actually challenging for Stable Diffusion. As both datasets are also used by authors Feng et al. [2023] for evaluation of the Structured Diffusion Guidance, we chose them to be able to make a better comparison between the two approaches. Furthermore, the disparate structure of the prompts in both datasets provides a more comprehensive understanding of the potential and limitations of the implemented approaches.

5.3 Evaluation Metrics

With the growth and proliferation of different approaches to improve image generation through diffusion models, the number of evaluation metrics has also increased. For evaluation we choose two techniques. Firstly, we use evaluation method called Decompositional-Alignment-Score (DA-Score) presented by Singh and Zheng [2023]. This evaluation approach is based on a visual question answering model [Hu et al., 2023] [Singh and Zheng, 2023]. The DA-Score method initially identifies subprompts, which are then used to construct a disjoint assertion set. For instance, for the prompt `a monkey eats a blue apple` the following subprompts would be extracted: `a monkey`, `eats an apple` and `a blue apple`. Those subprompts are then transformed into assertions: `The image shows a monkey`, `The monkey eats an apple` and `The apple is blue`. Subsequently, the assertions are transformed into questions, which are then answered by a visual question answering model: `Does the`

`image show a monkey?`, `Does the monkey eat an apple?` and `Is the apple blue?` According to the given answer the model assigns a score between 0.0 and 1.0 for each question. Finally, the scores are combined to give the final evaluation score for text-image alignment.

Although the DA-Score demonstrates a higher correlation with human ratings in comparison to the state-of-the-art alignment metric [Singh and Zheng, 2023], we elected to corroborate the alignment through the use of human ratings as well. In order to ascertain the efficacy of this approach, 100 randomly selected prompts from both datasets were evaluated for both models. The evaluation metrics are based on those used for evaluation of Structured Diffusion [Feng et al., 2023]. With regard to the image-text alignment, the annotator is required to respond to the following question: "Which image demonstrates better image-text alignment?" In the case of the evaluation of image fidelity, the following question was posed: "Regardless of the accompanying text, which image is more realistic and natural?" As we explored three distinct variations of the Simplified Structured Diffusion, the user should compare each method variance with the baseline method separately.

5.4 Experimental Results For Simplified Structured Diffusion

Three variations of Simplified Structured Diffusion were applied. To assess our methodology we chose two different datasets with compositional challenging prompts described in section 5.2. In order to gain a comprehensive overview and analysis of our methodology, we applied all three variations of the Simplified Structured Diffusion on both datasets within both Stable Diffusion models. The first variation of the Simplified Structured Diffusion, labeled as "SSD", does not include a pooling step, while the other two include different versions of additional mean square pooling and labeled as "pool" for pooling technique assigning zero values and "poolnz" for no-zero pooling that were described in section 4.2.

(a) baseline **(b)** SSD **(c)** SSD+pool **(d)** SSD+poolnz

Figure 5.1: Selected results for the prompt `a brown bird and a red cow` generated by SDXL using the SSD variants

(a) baseline **(b)** SSD **(c)** SSD+pool **(d)** SSD+poolnz

Figure 5.2: Selected results for the prompt `a blue backpack and a red car` generated by SD1.5 using the SSD variants

(a) baseline **(b)** SSD **(c)** SSD+pool **(d)** SSD+poolnz

Figure 5.3: Selected results for the prompt `a bearded man wearing a yellow dress shirt and white tie` generated by SDXL using the SSD variants

(a) baseline **(b)** SSD **(c)** SSD+pool **(d)** SSD+poolnz

Figure 5.4: Selected results for the prompt `a beautiful green horse pulling a white carriage` generated by SDXL using the SSD variants

In order to evaluate the efficacy of our methodologies, we present a summary of the obtained scores in the Figure 5.5. The figure depicts four plots with calculated DA-Scores, wherein each plot illustrates the data distribution for each baseline method and all three approaches of Simplified Structured Diffusion within a single plot. Accordingly, four plots provide an overview of the scores for each Stable Diffusion model, calculated for both datasets.

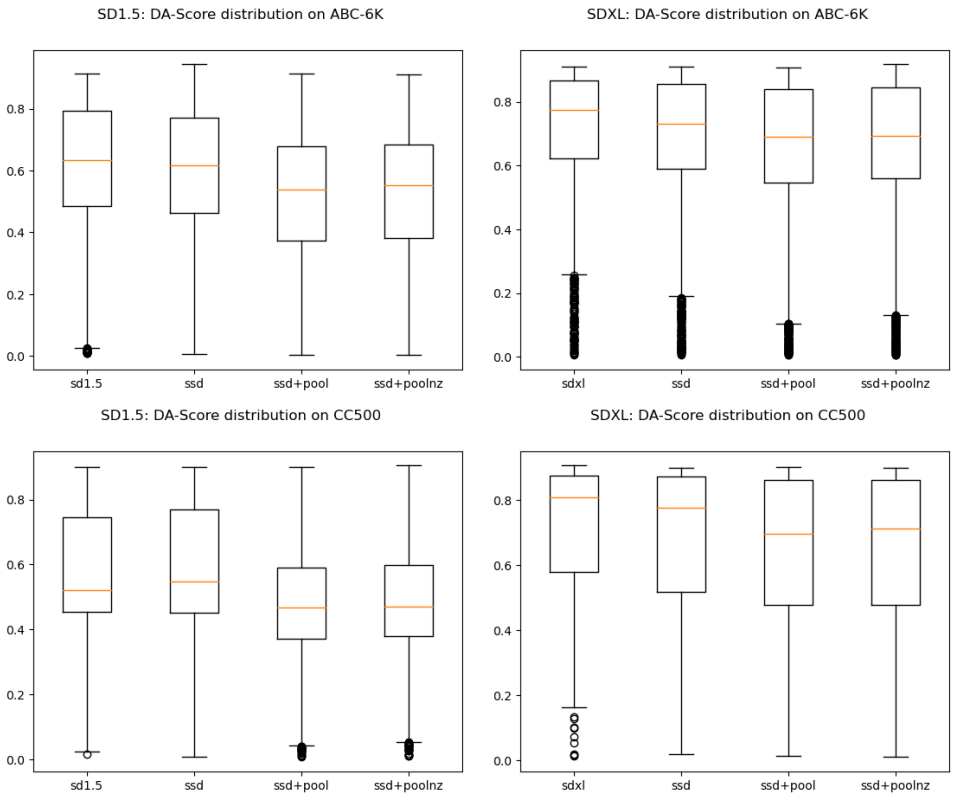


Figure 5.5: DA-Score distribution for both SD1.5 and SDXL models on ABC-6K and CC500 datasets

Firstly, we take a closer look at the differences between both baseline models. As illustrated in Figure 5.5, the score distributions of the two models exhibit significant discrepancies. The initial observations show that Stable Diffusion XL model performs higher scores and better overall score distribution for both datasets as the Stable Diffusion 1.5. This can be attributed to the fact that the model's innovative architectural design incorporates the use of two distinct CLIP encoders, which already enables the model to generate images with a higher degree of text-image alignment.

It should be noted, however, that the discrepancies between the two baseline methods on the two datasets are not identical. While, the median values

of Stable Diffusion XL have slightly equal value for both datasets, the results for Stable Diffusion 1.5 show a larger difference between the datasets. Consequently, the discrepancies between the models on a single dataset are proportionally different. While the median value difference between Stable Diffusion 1.5 and Stable Diffusion XL for for the CC500 dataset is approximately 0.3 points, the ABC-6K dataset shows a much smaller discrepancy of only 0.1 points. Furthermore, the overall data distribution for Stable Diffusion XL differs between the two datasets, with the CC500 dataset showing a wider range of values than the ABC-6K dataset. It is clearly seen by the value of minimum score and by the interquartile range. While, the minimum score on the ABC-6K lies between 0.2 and 0.3, the minimum for the CC500 is under 0.2. These observations illustrate the significant influence of prompt structure on the resulting output, indicating that the CC500 dataset comprises prompts that are compositionally more challenging than those in the ABC-6K. It is also essential to consider the size of the dataset, which may have contributed to these outcomes.

Another point to emphasize is that not only the baseline scores vary between the models but that also all forms of Simplified Structured Diffusion produce superior scores when applied to Stable Diffusion XL. Nevertheless, the most significant discrepancy can be observed among SSD methods implementing two pooling techniques. Once more, the results obtained from the CC500 dataset demonstrate the most pronounced discrepancy among both the median value and the interquartile range, when comparing the two pooling techniques applied on Stable Diffusion XL and Stable Diffusion 1.5. This is in a manner similar to that observed with the baseline method. As this trend is observable for both datasets, we can conclude that all three approaches of SSD are dependent on the data model.

The following section of the evaluation will focus on comparing of the three SSD approaches in order to provide an assessment of their relative efficacy. The results depicted on Figure 5.5 show that the baseline method consistently outperformed all of the SSD methods. The sole exception is the scores achieved by the SSD approach applied on the CC500 dataset. The results demonstrate a slight increase in the median value, reaching 0.55 compared to the 0.52 observed in the baseline model. Additionally, the interquartile range exhibits a higher upper border as the baseline method. This assessment was also corroborated by the manual evaluation of the 100 prompts for the image-text alignment parameter on the CC500 dataset, as illustrated in the Figure 5.6 (see Appendix A for complete visualization of manual evaluation). Furthermore, the SSD method without pooling demonstrates the overall better performance compared to the other two approaches with applied additional pooling techniques. In contrast, the two pooling approaches exhibited the lowest score

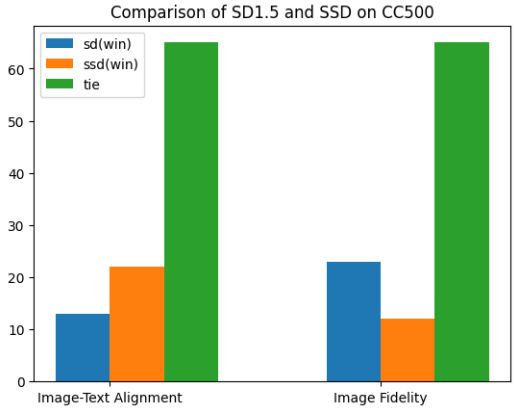


Figure 5.6: Manual evaluation of SSD on SD1.5 compared to baseline method on CC500

results with regard to both median value and data distribution. Nevertheless, the score discrepancies with the SSD and the baseline method are least when applied on Stable Diffusion XL model. Furthermore, among the both SSD with additional pooling steps, the approach with no-zero pooling shows slightly better results as the variance that assigns zero values.

Based on the observations we made within our experiments and by analyzing the score results, we conclude that the effectiveness of our methods is not uniform. Thus, the SSD approach without additional pooling step showed the overall better results among the three variations of the approach. Consequently, our findings did not substantiate the hypothesis that the supplementary pooling step enhances the SSD methodology, particularly with regard to the attribute-binding parameter of the images. Moreover, despite the fact that in some cases our approach achieves better text alignment and attribute-binding, as it can be seen in the Figure 5.1 and Figure 5.4, none of our methods have been able to surpass the performance of the baseline method and achieve the same results as the Structured Diffusion.

The observed results may be attributed to the cumulative nature of the CLIP embeddings, which was not sufficiently addressed by our approach. As demonstrated in Smith [2023], not only the prompt tokens but also the padding tokens have a great impact on the generated image. One potential avenue for further investigation would be the amplification of the SSD approach on the padding tokens as well.

5.5 Further Embedding Manipulations

5.5.1 Interactive Manipulation Approach

In addition to the primary study, an alternative approach was also investigated. The objective of this supplementary study is to address the attribute-binding problem by once more using the linguistic structure of the prompt in conjunction with the iterative user interaction approach. The principle concept is based on the features of word vectors investigated by Mikolov et al. [2013]. More exact, the possibility to sum and subtract word vectors relying on the semantic relationships between them. For instance, the following vector calculations "`king`" - "`man`" + "`woman`" will result in "`queen`". The same approach is applied to CLIP embeddings as a preprocessing step within the image generation process on Stable Diffusion.

Figure 5.7: Generated with SD1.5

Firstly, the calculations over entire CLIP embeddings excluding start token were explored. Figure 5.7 shows that this approach is applicable for CLIP embeddings that encode a single object. The sum and subtraction operations over entire embedding, excluding the start token, allow attribute assignment and object replacement. These features can be used to correct attribute-binding failures and object loss in generated images. Moreover, we explored normalized and non-normalized variants of calculations and can observe differences between both approaches. However, although some images show improvements through the application of normalized operations, no consistent improvement was found overall in relation to normalized version of embedding operations, which can be observed on the Figure 5.8 and Figure 5.9.

Figure 5.8: `the red car - the red` + `the blue`; left picture: not normalized calculation; right picture: normalized calculation; generated with SD1.5

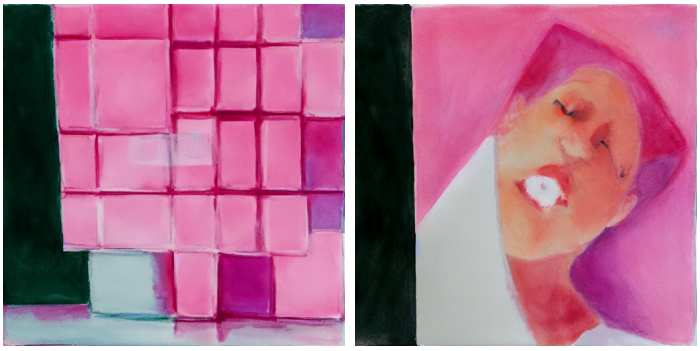


Figure 5.9: `red cube - red` + `pink`, left image: not normalized approach; right image: normalized approach; generated with SD1.5

Though, the calculations over complete embedding give promising results, they do not provide sufficient control over the output and do not allow manipulation of embeddings with more than one object. Therefore, we explored another variation of this method. The main difference to aforementioned overall embedding manipulation is applying calculations targeted at the selected vectors within the embedding. The main objective of this methodology is to enhance or diminish the appearance of particular attributes and objects and therefore, to enable the user to have more control over the output. However, this approach requires more precise instructions to identify which object-attribute pair requires correction. Thus, we devised a potential solution by utilizing the iterative user feedback. We model the prototype of possible solution via user interface, which is demonstrated in Figure 5.10.

As the first step of the method, the user is asked to enter two values, a prompt and optionally a seed (see step 1 in Figure 5.10). These inputs are then used for generating an image via Stable Diffusion. In case, the user does

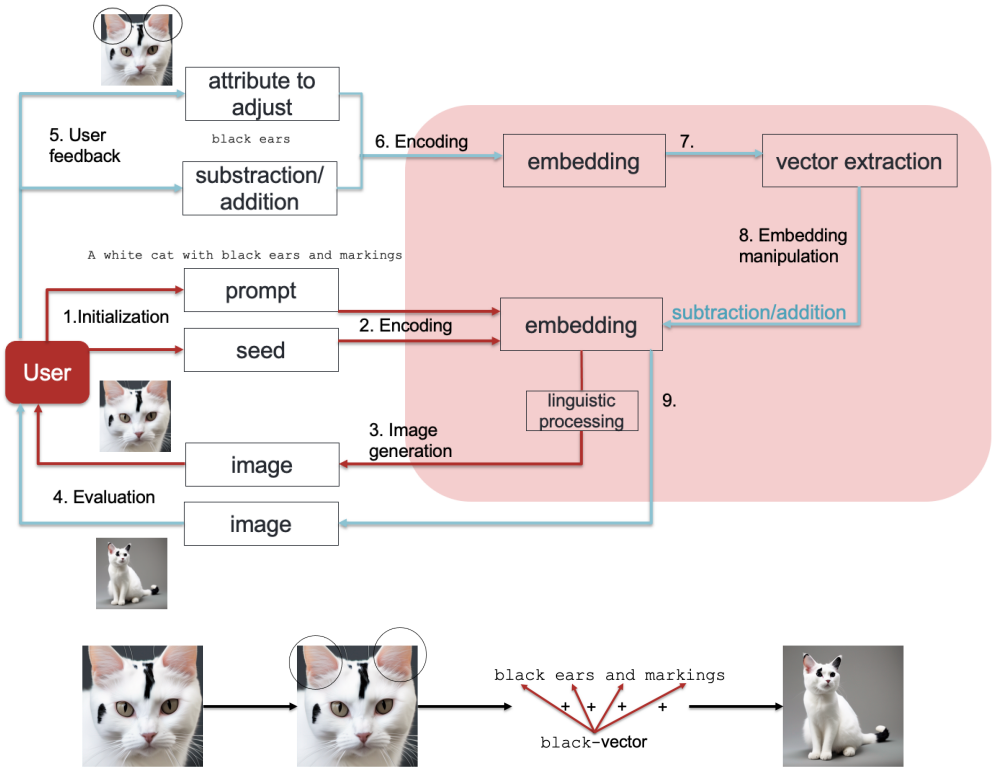


Figure 5.10: Illustration of the prototype of user interface for targeted embedding manipulation approach; images generated with Stable Diffusion XL

not enter a seed value, a pseudo-random seed will be generated and saved. In accordance with the given prompt, a CLIP embedding is generated. This step follows with the processing of the linguistic structure of the prompt for extracting object-attribute pairs. It is the same approach, which was used in Structured Stable Diffusion and described in chapter 4. Subsequently, the generated CLIP embedding and the seed are used for image generation, which is then presented to the user. According to the user's evaluation, the user decides whether the image requires corrected considering the attribute-binding feature. In the provided example in Figure 5.10 the user has identified a discrepancy between the generated image and the given prompt `a white cat with black ears and markings` because the cat in the generated image has ears of wrong color. Therefore, the generated image shows attribute-binding issues and fails to meet user expectations. After the user evaluation, which is step 4 in the illustration, the user can determine which attributes should be adjusted. In order to facilitate this process, the user is provided with an overview of the extracted object-attribute pairs from the prompt and is afforded to determine

whether the attribute should be intensified or weakened. According to the given answers of the user, the initial CLIP embedding is modified and the new image is generated. Then the process of evaluation and adjustment can be repeated until the user gets the desired output. Nevertheless, the user has also the opportunity to change the seed and start the process from the beginning.

The presumably advantages of this approach include the ability to exert greater control over the image generation process, which can enhance user satisfaction and reduce the time required for user-model interaction until the desired result is achieved. However, it is crucial to note, that the presented approach is still in the prototype phase, which has not be implemented now. Therefore, it is an auspicious method, which can be closer investigated in the future works.

Chapter 6

Conclusion

The advancement of diffusion models has reached a significant level of sophistication, resulting in the generation of high-quality outputs. Stable Diffusion is one of the most popular models for text-to-image generation, which enables user to unfold the creativity and to generate high-quality images. Nevertheless, even the latest version of Stable Diffusion XL still exhibits deficiencies when processing compositionally complex prompts, which has a direct impact on the user experience. Consequently, a multitude of methodologies address this issue and propose disparate approaches for its improvement. In this thesis, we investigated three variations of a novel approach, Simplified Structured Diffusion with additional pooling, which is based on the method introduced by Feng et al. [2023]. The objective of this thesis was to assess this approach and to investigate whether a modification at the embedding level only could yield results comparable to those of Structured Diffusion without modifying the cross-attention layers of the model.

In order to evaluate the proposed methodology and to test the underlying hypothesis, we conducted a comparative analysis of the scores obtained through the application of our method to the Stable Diffusion 1.5 and Stable Diffusion XL models and evaluated with DA-Score evaluation metrics. To facilitate a comprehensive overview and analysis, three variations of the proposed approach were applied to two distinct datasets on both models. The findings revealed that the three approaches exhibited disparities in their performance. Accordingly, the SSD approach without an additional pooling step demonstrated superior outcomes among the alternative approach variations on both models and both datasets. Nevertheless, it has not yet achieved a comprehensive enhancement in text-image alignment and image composition in comparison to the Stable Diffusion models. Consequently, the desired advancement of the introduced approach was not achieved and the results have not yield results shown by Structured Diffusion Guideline. As a result, the

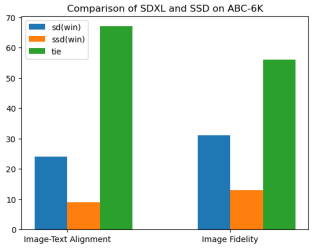
anticipated advancement of the introduced approach was not achieved, and the outcomes did not reach the level of those demonstrated by Structured Diffusion. One potential explanation for these findings is the accumulative nature of CLIP embeddings. To address this issue, we propose an extension of our SSD method on the pooling tokens as a potential avenue for future research.

Furthermore, we presented an additional approach addressing the aforementioned image composition issue. This approach leverages the linguistic structure of the prompt for targeted embedding manipulation via vector arithmetic and incorporates iterative user feedback for embedding adjustment. We proposed a prototype of a user interface and presented advantages of the method. Subsequent steps would entail the implementation of the interface and a user study for evaluation purposes. This could represent a further avenue for research to cover further needs in user-friendly interaction for image generation process. Since the CLIP encoder remains one of the most popular and widely used language models for text encoding, this research direction can maintain the research field not only within Stable Diffusion models, but also in other models that use CLIP encoder.

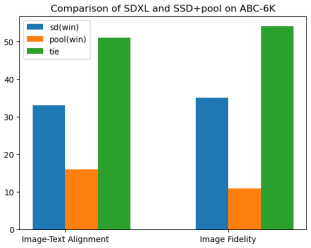
In this thesis we addressed the problem of failed image compositionality whithin images generated by Stable Diffusion and its impact on the user experience. Our study showed that it is not a trivial problem which still requires more research and deeper investigation. One of the most significant research points is the exploration of the root cause of the problem, which is the cumulative nature of CLIP embeddings used for text encoding in Stable Diffusion models. Furthermore, the necessity for the development of further approaches addressing this issue and providing users with an option that requires minimal effort while still achieving the desired outcome and impact the overall model-user experience is evident.

Appendix A

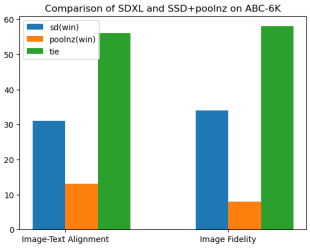
Evaluation



(a) SDXL vs. SSD

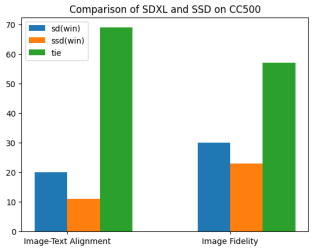


(b) SDXL vs. SSD+pool

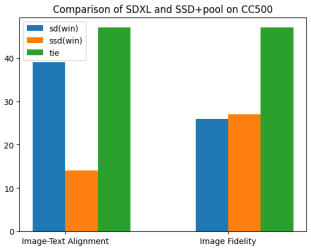


(c) SDXL vs. SSD+poolnz

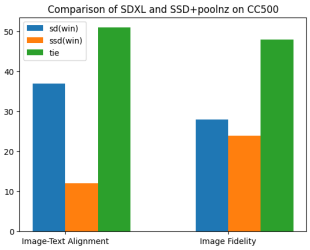
Figure A.1: Manual evaluation of SDXL on ABC-6K



(a) SDXL vs. SSD

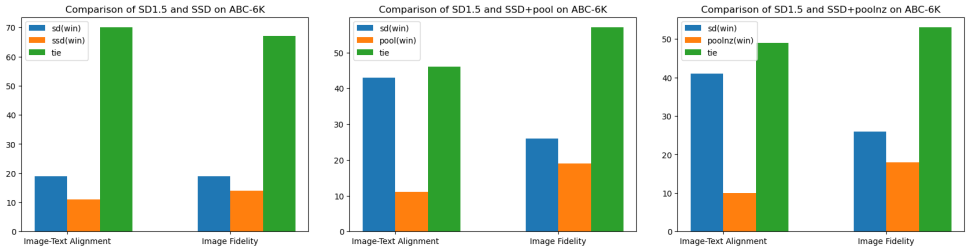


(b) SDXL vs. SSD+pool



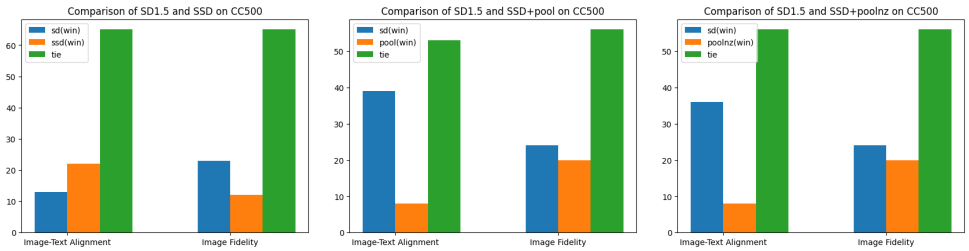
(c) SDXL vs. SSD+poolnz

Figure A.2: Manual evaluation of SDXL on CC500



(a) SD1.5 vs. SSD **(b)** SD1.5 vs. SSD+pool **(c)** SD1.5 vs. SSD+poolnz

Figure A.3: Manual evaluation of SD1.5 on ABC-6K



(a) SD1.5 vs. SSD **(b)** SD1.5 vs. SSD+pool **(c)** SD1.5 vs. SSD+poolnz

Figure A.4: Manual evaluation of SD1.5 on CC500

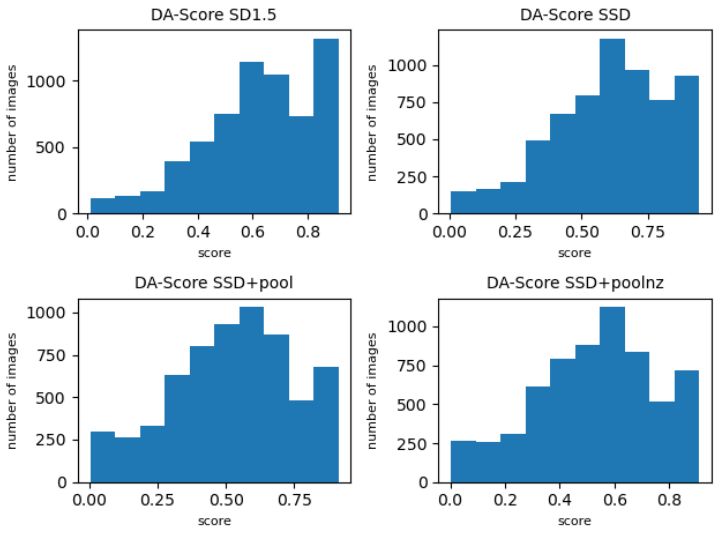


Figure A.5: DA-Score data distribution for SD1.5 on ABC-6K

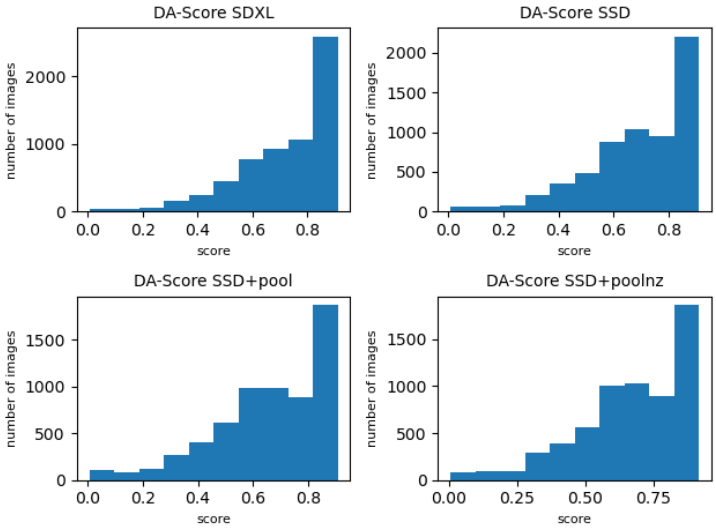


Figure A.6: DA-Score data distribution for SDXL on ABC-6K

Bibliography

Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. O'Reilly, 2009. ISBN 978-0-596-51649-9. URL http://www.oreilly.de/catalog/9780596516499/index.html.

Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *CoRR*, abs/2301.13826, 2023. doi: 10.48550/ARXIV.2301.13826. URL https://doi.org/10.48550/arXiv.2301.13826.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/V1/N19-1423. URL https://doi.org/10.18653/v1/n19-1423.

Kemal Erdem. Step by step visual introduction to diffusion models. https://erdem.pl/2023/11/step-by-step-visual-introduction-to-diffusion-models, 2023. Accessed: 08.07.2024.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Reinforcement learning for fine-tuning text-to-image diffusion models. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*,

2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/fc65fab891d83433bd3c8d966edde311-Abstract-Conference.html.

Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun R. Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/pdf?id=PUIqjT4rzq7.

Rohit Gandikota, Joanna Materzynska, Tingrui Zhou, Antonio Torralba, and David Bau. Concept sliders: Lora adaptors for precise control in diffusion models. *CoRR*, abs/2311.12092, 2023. doi: 10.48550/ARXIV.2311.12092. URL https://doi.org/10.48550/arXiv.2311.12092.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.90. URL https://doi.org/10.1109/CVPR.2016.90.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *CoRR*, abs/2208.01626, 2022. doi: 10.48550/ARXIV.2208.01626. URL https://doi.org/10.48550/arXiv.2208.01626.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html.

Jeremy Howard. Stable diffusion deep dive. https://github.com/fastai/diffusion-nbs/blob/master/Stable%20Diffusion%20Deep%20Dive.ipynb, 2022. Accessed: 08.07.2024.

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A. Smith. TIFA: accurate and interpretable text-to-image faithfulness evaluation with question answering. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 20349–20360. IEEE, 2023. doi: 10.1109/ICCV51070.2023.01866. URL https://doi.org/10.1109/ICCV51070.2023.01866.

huggingface. Clip. `https://huggingface.co/docs/transformers/model_doc/clip#clip`. Accessed: 08.07.2024.

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. URL `https://doi.org/10.5281/zenodo.5143773`.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL `http://arxiv.org/abs/1312.6114`.

Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback. *CoRR*, abs/2302.12192, 2023. doi: 10.48550/ARXIV.2302.12192. URL `https://doi.org/10.48550/arXiv.2302.12192`.

Mathias Leys. The art of pooling embeddings. `https://medium.com/ml6team/the-art-of-pooling-embeddings-c56575114cf8`, 2022. Accessed: 08.07.2024.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pages 740–755. Springer, 2014. doi: 10.1007/978-3-319-10602-1_48. URL `https://doi.org/10.1007/978-3-319-10602-1_48`.

Bingyan Liu, Chengyu Wang, Tingfeng Cao, Kui Jia, and Jun Huang. Towards understanding cross and self-attention in stable diffusion for text-guided image editing. *CoRR*, abs/2403.03431, 2024. doi: 10.48550/ARXIV.2403.03431. URL `https://doi.org/10.48550/arXiv.2403.03431`.

Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B. Tenenbaum. Compositional visual generation with composable diffusion models. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XVII*, volume 13677 of *Lecture Notes in Computer Science*,

pages 423–439. Springer, 2022. doi: 10.1007/978-3-031-19790-1_26. URL https://doi.org/10.1007/978-3-031-19790-1_26.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1411.4038, 2014. URL http://arxiv.org/abs/1411.4038.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In Yoshua Bengio and Yann LeCun, editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013. URL http://arxiv.org/abs/1301.3781.

Britney Muller. Bert 101 state of the art nlp model explained. https://huggingface.co/blog/bert-101, 2022. Accessed: 08.07.2024.

Suraj Patil, Pedro Cuenca, Nathan Lambert, and Patrick von Platen. Stable diffusion with diffusers. https://huggingface.co/blog/stable_diffusion, 2022. Accessed: 08.07.2024.

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. *CoRR*, abs/2307.01952, 2023. doi: 10.48550/ARXIV.2307.01952. URL https://doi.org/10.48550/arXiv.2307.01952.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A python natural language processing toolkit for many human languages. *CoRR*, abs/2003.07082, 2020. URL https://arxiv.org/abs/2003.07082.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021. URL http://proceedings.mlr.press/v139/radford21a.html.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 10674–10685. IEEE, 2022. doi: 10.1109/CVPR52688.2022.01042. URL https://doi.org/10.1109/CVPR52688.2022.01042.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015. URL http://arxiv.org/abs/1505.04597.

Jaskirat Singh and Liang Zheng. Divide, evaluate, and refine: Evaluating and improving text-to-image alignment with iterative VQA feedback. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/dfd0bd56e8a6f82d1619f5d093d5f9ca-Abstract-Conference.html.

Adam M. Smith. Promptmanipulation. https://colab.research.google.com/drive/1izMKdvBMfThVSRp8Tg4Fc1eiGVP0NKZP?usp=sharing, 2023. Accessed: 08.07.2024.

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 2256–2265. JMLR.org, 2015. URL http://proceedings.mlr.press/v37/sohl-dickstein15.html.

Jiao Sun, Deqing Fu, Yushi Hu, Su Wang, Royi Rassin, Da-Cheng Juan, Dana Alon, Charles Herrmann, Sjoerd van Steenkiste, Ranjay Krishna, and Cyrus Rashtchian. Dreamsync: Aligning text-to-image generation with image understanding feedback. *CoRR*, abs/2311.17946, 2023. doi: 10.48550/ARXIV.2311.17946. URL https://doi.org/10.48550/arXiv.2311.17946.

Junjiao Tian, Lavisha Aggarwal, Andrea Colaco, Zsolt Kira, and Mar González-Franco. Diffuse, attend, and segment: Unsupervised zero-shot segmentation using stable diffusion. *CoRR*, abs/2308.12469, 2023. doi: 10.48550/ARXIV.2308.12469. URL https://doi.org/10.48550/arXiv.2308.12469.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Roy Voetman, Maya Aghaei, and Klaas Dijkstra. The big data myth: Using diffusion models for dataset generation to train deep detection models. *CoRR*, abs/2306.09762, 2023. doi: 10.48550/ARXIV.2306.09762. URL https://doi.org/10.48550/arXiv.2306.09762.

Andrew Wong. Stable diffusion samplers: A comprehensive guide. https://stable-diffusion-art.com/samplers/, 2024. Accessed: 08.07.2024.