

CLIP is All You Need for Human-like Semantic Representations in Stable Diffusion

Cameron Braunstein^{1,2}, Mariya Toneva^{2,3}, and Eddy Ilg³

¹ Saarland University, Saarbrücken Germany
`braunstein@cs.uni-saarland.de`

² MPI for Software Systems, Saarbrücken Germany

³ University of Technology Nuremberg, Nuremberg Germany

Abstract. Latent diffusion models such as Stable Diffusion achieve state-of-the-art results on text-to-image generation tasks. However, the extent to which these models have a semantic understanding of the images they generate is not well understood. In this work, we investigate whether the internal representations used by these models during text-to-image generation contain semantic information that is meaningful to humans. To do so, we perform probing on Stable Diffusion with simple regression layers that predict semantic attributes for objects and evaluate these predictions against human annotations. Surprisingly, we find that this success can actually be attributed to the text encoding occurring in CLIP rather than the reverse diffusion process. We demonstrate that groups of specific semantic attributes have markedly different decoding accuracy than the average, and are thus represented to different degrees. Finally, we show that attributes become more difficult to disambiguate from one another during the inverse diffusion process, further demonstrating the strongest semantic representation of object attributes in CLIP. We conclude that the separately trained CLIP vision-language model is what determines the human-like semantic representation, and that the diffusion process instead takes the role of a visual decoder.

Keywords: Diffusion · Probing · Generative Modelling · Alignment

1 Introduction

Text-to-image generation has undergone a recent, rapid advancement [7]. In particular, diffusion models produce state-of-the-art results on image generation conditioned on a text prompt [34]. However, despite their success and the ability to steer the generation with text, the internal workings of diffusion models are not interpretable, and it is vastly unclear if the representations learned by such models align with human judgment. Work in the NLP domain has shown promising results that models aligned with human judgment can have improved performance, and are inherently more interpretable [1, 4], motivating us to study human alignment with vision models. In this work, we make a significant step towards better understanding text-to-image diffusion models by analyzing how their internal representations align with human perception.

For our analysis, we leverage well-known probing techniques [6]. We use the MTurk dataset [49], which consists of object nouns that serve as prompts, paired with ratings of over 200 attributes associated to the objects. To probe whether the attributes are present similarly in the intermediate representations of the image generation process of Stable Diffusion [40], we train linear ridge regressions to predict the attribute ratings. Since they constitute a simple mapping, their ability to accurately predict attributes tells us the extent to which the intermediate representations of Stable Diffusion align with human judgment.

Our analysis reveals that the representations present in Stable Diffusion have the strongest alignment at the final layers of the CLIP [35] text encoder. This comes as a surprise, as it indicates that semantic understanding comes mostly from the pretrained CLIP model and not from the reverse diffusion process. Instead, the reverse diffusion process can be seen as a visual decoding of the representation provided by CLIP.

We provide a detailed analysis of how well certain groups of semantic attributes align with human annotation and which groups of attributes are represented well. Finally, we investigate how well Stable Diffusion can disentangle such attributes. To summarize, the contributions of this work are as follows:

- We apply probing techniques to a text-to-image diffusion model, and show that these techniques are effective in the task of object attribute prediction for Stable Diffusion as an example of a generative model.
- We demonstrate that the semantic understanding in Stable Diffusion comes from CLIP instead of the diffusion model, and show that the reverse diffusion process acts as a visual decoding.
- We provide a detailed analysis and show that object attributes extracted from CLIP align very well with human judgment, and furthermore, that CLIP is able to disentangle attributes which humans tend to closely associate.

2 Related Work

2.1 Text-to-Image Generative Models

Text-to-image generative models have undergone rapid advancement in recent history, and for a comprehensive overview we refer to [55]. Notable models, including DALL-E [37], DALL-E-2 [36], eDiff-I [5], Imagen [42], and GigaGAN [20], produce impressive results, but are closed source and do not allow an analysis of the models. Open source alternatives are plentiful, including DiT [33], GLIDE [30], and several generations of Kadinsky models [3].

From these open source options, we chose Stable Diffusion [40] as a suitable reference work, and conduct our evaluations exclusively on this architecture. Stable Diffusion was trained on general-purpose data and a wide domain of images. It distinguishes itself from similar models by already having a rich literature on investigations on its interpretability (see Sec. 2.2), which we seek to extend in our

own work. Diffusion models work by drawing a data sample from a simple distribution (typically Gaussian noise), and denoising it into a sample from a more complex distribution, in this case, the distribution of plausible images [26, 34]. Unlike earlier diffusion works [46], which sample images in the RGB space, Stable Diffusion is the seminal work for performing the reverse diffusion process in a more efficient latent space that is obtained from a pre-trained VAE. To produce images conditioned on text, a method for bridging the language and image modalities is required. Many state-of-the-art models [5, 20, 25, 30, 36, 40] including Stable Diffusion achieve this using Contrastive Language-Image Pre-training (CLIP) [35], which consists of a language and a vision encoder that are trained to produce similar encodings for a caption and corresponding image on large amounts of unlabeled data.

2.2 Prior Investigations of Explaining Stable Diffusion and CLIP

Prior works have investigated the interpretability of Stable Diffusion and of CLIP in isolation. For the most part, works that study the interpretability of Stable Diffusion or similar models do so with image editing or more accurate text conditioning [8, 16, 17, 21, 25, 56] as the primary goal. One of these works, one that is closely related to ours is by Park *et al.* [32], which explored manipulating Stable Diffusion’s image latents for image editing. Their work is inspired by Kwon *et al.* [22], in which they demonstrate that diffusion models have a semantically interpretable latent space, and can adjust the latent in space to produce an intentional change in the semantics of the output image. In contrast, our work investigates latent interpretability by comparing it directly to human rated attributes. Our work is thus a novel investigation into the explainability of Stable Diffusion.

Existing works have investigated CLIP’s compositionality in multi-word prompts [24, 38, 51, 54], and CLIP’s ability to rate image aesthetics [15]. Schiappa *et al.* investigate CLIP’s relational, attribute, and contextual understanding [43]. But, in their investigation, they look at whether CLIP can misunderstand attributes if they are passed as adjectives in the text prompt. Our investigation is novel for several reasons: unlike previous works, we investigate the CLIP text encoder’s alignment with human perception of semantics on a single object. This is a more challenging task, as we are not querying CLIP with the attribute directly with the text prompt. An additional novelty is that we put CLIP’s understanding of attributes into greater context by searching for alignment with directly with human perception.

2.3 Interpretability with Probing

Few works focus purely on interpreting Stable Diffusion’s generative process, notably, Tang *et al.* [50] analyze where different words of a text prompt are expressed in an image. Our focus is on the interpretation of intermediate latent representations, to see if the model has a human-like understanding of the objects it generates. We use *probing* as a tool to interpret the model.

Probing is a simple, effective technique for measuring AI model alignment with human perception [29]. It emerged in the NLP domain with works by Köhn [23], Gupta *et al.* [11], and Shi *et al.* [45], but relevant works in the computer vision field are by Alain and Bengio [2], and Muttenthaler *et al.* [29], which explored network alignment with humans on tasks such as odd-one-out classification and image classification. Our work applies these interpretability techniques to the novel domain of latent diffusion models, and explores alignment on the sophisticated task of object attribute understanding, which has not been done previously in computer vision. We elaborate on the technical details of probing in Sec. 3.2. Our work is a unique contribution to the interpretability of Stable Diffusion and especially CLIP.

3 Method

In Sec. 3.1, we create notation to precisely label the intermediate representations created by Stable Diffusion, which we use in our analyses. Sec. 3.2 explains how model-human alignment is quantified with the technique of *probing*, by first introducing general mathematical notation, and then plugging in our notation from Sec. 3.1. Finally, the concept of *entanglement*, a measure for quantifying whether the model disambiguates between related attributes, is introduced in Sec. 3.3.

3.1 Background Notation

We analyze intermediate representations from Stable Diffusion during text-to-image generation, which relies on CLIP for conditioning the generation on text prompts. The pipeline utilizes the CLIP ViT-L/14 architecture, which consists of a tokenizer, followed by 12 hidden encoder blocks. We denote the output of the CLIP text encoder from text input $\mathbf{T}$ as

$$\text{CLIP}(\mathbf{T}), \quad (1)$$

with $\text{CLIP}_l(\mathbf{T})$ specifying CLIP’s output at hidden layer l . Latent feature map generation in Stable Diffusion is realized through a time-conditional U-Net [41] architecture. The initial latent feature map is initialized as Gaussian noise ϵ . The U-Net is then applied repeatedly 50 times, where it receives the latent feature map generated by the previous iteration and features from $\text{CLIP}(\mathbf{T})$ injected into it at various levels via cross attention as input. We examine internal representations of the U-Net at every iteration, both at the bottleneck and the output. Work from Kwon *et al.* [22] suggests that the bottleneck has more easily interpretable semantics, however, we also note that features from $\text{CLIP}(\mathbf{T})$ are inserted before the bottleneck, and thus it may be strongly influenced by CLIP. We write the bottleneck output at iteration k as:

$$\text{Diff-Bot}_k(\text{CLIP}(\mathbf{T})), \quad (2)$$

and the U-Net output at iteration k as:

$$\text{Diff-Out}_k(\text{CLIP}(\mathbf{T})), \quad (3)$$

where we denote the final generated latent feature map with $\text{Diff-Out}(\text{CLIP}(\mathbf{T}))$. The generation is overall conditioned on $\mathbf{T}$ and random noise, and the functions Diff-Bot_k and Diff-Out_k produce different outputs if the noise is changed. We use this to create multiple samples for a given input $\mathbf{T}$. The latent feature map is passed into a decoder Dec to get the final output image $\text{Dec}(\text{Diff-Out}(\text{CLIP}(\mathbf{T})))$. An illustration of this architecture can be found in Fig. 1.

3.2 Measuring Alignment

We probe Stable Diffusion to understand the alignment between internal representations of a text-to-image generative model and human perception. In the following, we introduce our probing mechanism and then explain how it can be used to measure alignment. The input to the probing is a set of stimuli $\{s_i | i \in 1, \dots, n\}$, which in our case are text prompts that correspond to objects. We then pass the stimuli s_i to the network and extract the outputs from different inner layers of CLIP, as well as at the bottleneck and output of the U-Net after each iteration of the reverse diffusion process. In general, we want to attach different probes to all of these intermediate representations to understand where certain attributes are present. However, in the following, we will first concentrate the discussion on a single probe of a network f and define the output for stimuli s_i as x_i :

$$x_i = f(s_i). \quad (4)$$

s_i is also presented to humans to capture their rating $y_{i,j} \in \mathbb{R}$ across $j \in 1, \dots, m$ response classes. Probing argues [6] that the neural network and human are aligned for response class j , if there is a simple model (called a *probe*), that can effectively predict $y_{i,j}$ from x_i , because a successful probe implies that the necessary information to predict $y_{i,j}$ is readily available in the neural network’s representation.

In our case, responses $y_{i,j}$ are scalar values that represent human annotator ratings and we can use a linear model as a probe. We denote the model weights as β_j and c_j , and calculate the predicted scalar value:

$$\hat{y}_{i,j} = x_i^T \beta_j + c_j. \quad (5)$$

We write the vector of responses across all text prompts for a single attribute j as Y_j , and the predictions across all text prompts as $\hat{Y}_j$:

$$Y_j = (y_{1,j}, y_{2,j}, \dots, y_{n,j}), \hat{Y}_j = (\hat{y}_{1,j}, \hat{y}_{2,j}, \dots, \hat{y}_{n,j}). \quad (6)$$

To assess the performance of this model in the same units as the scalar $y_{i,j}$, we compute the root mean squared error (RMSE) across all stimuli:

$$\text{RMSE}(Y_j, \hat{Y}_j) = \sqrt{\frac{(Y_j - \hat{Y}_j)^T (Y_j - \hat{Y}_j)}{n}}. \quad (7)$$

Leveraging ridge regression to estimate β_j and c_j can make the linear model more robust to unseen x_i [14, pp.60], and we therefore formulate the following ridge regression problem:

$$\beta_j, c_j = \operatorname{argmin}_{\beta_j, c_j} \{ \alpha_j \cdot \beta_j^T \beta_j + n \cdot \operatorname{RMSE}((Y_j, \hat{Y}_j))^2 \}, \quad (8)$$

where $\alpha_j \geq 0$ is a scaling hyperparameter on a regularization term, used to keep entries of β_j from becoming too large and overfitting. Following existing conventions to improve the robustness of the regression [18], [14, p.63], we reduce the dimensionality of network outputs x_i via PCA [19], and then normalize their z-scores (*i.e.*, we set the mean of each feature channel of $\{x_i\}$ to 0, and the standard deviation of each feature channel of $\{x_i\}$ to 1 [13]).

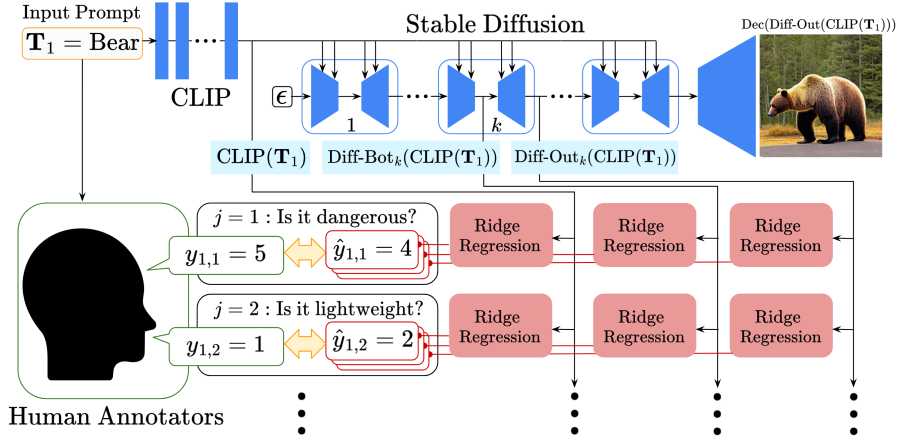


Fig. 1: An overview of our probing method, focused on iteration k of the latent generation. The stimulus text prompt $\mathbf{T}_1$ (in this case, “Bear”) is passed to Stable Diffusion. The intermediate object representation $\text{CLIP}(\mathbf{T}_1)$, $\text{Diff-Bot}_k(\text{CLIP}(\mathbf{T}_1))$, and $\text{Diff-Out}_k(\text{CLIP}(\mathbf{T}_1))$ are being extracted from the model during the generation process of the image of a bear. For every attribute j , we would like to decode, each intermediate representation is passed to a unique ridge regression model that is trained to predict this attribute value. These predictions $\hat{y}_{i,j}$ are compared against the human annotator responses $y_{i,j}$, which are judgments about the attribute intensities for the object “Bear”. Note that each ridge regression produces a unique set of predictions $\{\hat{y}_{i,j}\}$. In the diagram, the predictions for ridge regressions on $\text{CLIP}(\mathbf{T}_1)$ are shown on the front-most red boxes. Not all extracted intermediate representations have been shown: each $\text{CLIP}_l(\mathbf{T}_i)$, $\text{Diff-Bot}_k(\text{CLIP}(\mathbf{T}_i))$, and $\text{Diff-Out}_k(\text{CLIP}(\mathbf{T}_i))$ for all l and k are extracted, and have their own ridge regressions. The model is tested on stimuli $\mathbf{T}_i$ that have been withheld during training.

Note that error measures by itself do not prove statistical significance and ridge regression is also not the only possible regression strategy [14]. To address both of these concerns, we perform a permutation test on each ridge regression

and report the P-values [10], which measure the likelihood that the regression’s performance is only due to chance. This also decouples our alignment result from the particular choice of ridge regression. Following Ojala *et al.*’s approach [31], we conduct our permutation test by computing the RMSE, but with permuted human responses:

$$\pi_p(Y_j) = (y_{\pi_p(1),j}, y_{\pi_p(2),j}, \dots, y_{\pi_p(n),j}) \quad (9)$$

and calculate the P-value from a collection of permutations $\Pi = \{\pi_p\}$ as:

$$p = \frac{|\{\pi_p \in \Pi \mid \text{RMSE}(\pi_p(Y_j), \hat{Y}_j) < \text{RMSE}(Y_j, \hat{Y}_j)\}| + 1}{|\Pi| + 1}. \quad (10)$$

A sufficiently low p then implies that the probe’s success is statistically significant, *i.e.*, the network has a meaningful understanding of attribute j .

For our probing of Stable Diffusion, our stimuli $\{s_i\}$ consist of single-word text prompts of concrete nouns $\{\mathbf{T}_i\}$, all of which are common objects. The response $y_{i,j}$ is a human rating of the noun $\mathbf{T}_i$ for an attribute j , and is an integer value between 1 and 5. For example, if $\mathbf{T}_i$ is “Bear”, and attribute j is “is it dangerous?”, then the human rating $y_{i,j}$ could be 5, as bears are generally considered dangerous. Conversely, if attribute j' is “is it lightweight?”, $y_{i,j'}$ could be 1, as bears are heavy. As indicated in Figure 1, to apply probing to Stable Diffusion, we now use separate probes for each CLIP $_\ell$, Diff-Bot $_k$, and Diff-Out $_k$.

3.3 Measuring Entanglement

In addition to measuring the model-human perception alignment on each attribute individually, we also want to measure whether the complete collection of attributes are related to each other in the same way in both the model and human perception domains. We call this relationship *entanglement*, and consider two attributes to be entangled in a domain if their representations are significantly similar, and disentangled if they are significantly dissimilar. It is now interesting to investigate the difference in entanglement in the model and human perception domain.

The regression weights β_j and $\beta_{j'}$ act as the representation for attributes j and j' in the model domain, as they will be high if certain features correlate with certain attributes consistently. The human annotation responses Y_j and $Y_{j'}$ act as the attribute representation in the human perception domain. In both domains, we measure the attribute pair similarity by computing the cosine similarity for the model weights and for the human responses. After normalizing the channel-wise z-score of all regression weights $\{\beta_j\}$, or all attribute representations $\{Y_j\}$, we can compute the model and human perception similarities as follows:

$$\text{M-SIM}(\beta_j, \beta_{j'}) = \frac{\beta_j^T \beta_{j'}}{\|\beta_j\| \cdot \|\beta_{j'}\|}, \text{H-SIM}(Y_j, Y_{j'}) = \frac{Y_j^T Y_{j'}}{\|Y_j\| \cdot \|Y_{j'}\|}. \quad (11)$$

As in Sec. 3.2, we can quantify the significance of an entanglement via a permutation test, following the methodology presented by Ojala *et al.* [31]. We

will show how to carry this test out for the human perception similarity, although the implementation is isomorphic in the model case. We reuse the permutation notation defined in Eq. (9) to calculate the P-value as:

$$p = \frac{|\{\pi_n \in \Pi | (\text{H-SIM}(\pi_n(Y_j), Y_{j'})) < \text{H-SIM}(Y_j, Y_{j'})\}| + 1}{|\Pi| + 1}. \quad (12)$$

With P-values we can quantify entanglement. If, in a given domain, the attribute pair’s P-value is sufficiently high, we say the attribute pair is *positively* entangled, and we expect the attributes to be semantically similar. If the P-value is sufficiently low, we say that the attributes are *negatively* entangled, and we expect the attributes to be semantically opposite. Otherwise, we say that the attributes are *disentangled*, and expect them to be semantically unrelated. We conclude our numerical analyses by looking at the changes in entanglement between domains in Sec. 4.4.

4 Experiments

4.1 Datasets

To train and test our probes, we use a dataset of human annotations collected from the Mechanical Turk crowd-sourcing platform, which we refer to as the MTurk dataset [49]. It consists of 1,000 concrete nouns, 229 attributes, and ground-truth ratings ranging from 1 to 5 for every object/attribute pair. Ratings in MTurk are the median rating from at least three human annotators. In contrast to [52], this dataset provides significantly more nouns and attributes and in contrast to [27], it provides direct integer ratings.

Due to the large size of the training data for Stable Diffusion, which was trained on the LAION 400-M dataset with 400 million image/text pairs [44], and for CLIP, which was trained on the WebImageText dataset that also contains the same amount of image/text pairs [35], we do not expect a distribution shift from the MTurk data.

4.2 Implementation Details

When we run Stable Diffusion, we use DDIM sampling [47] rather than DDPM sampling, as it is more computationally efficient. We sample Stable Diffusion 50 times for each text prompt $\mathbf{T}_i$, and treat each collected latent feature map $\text{Diff-Bot}_k(\text{CLIP}(\mathbf{T}_i))$ and $\text{Diff-Out}_k(\text{CLIP}(\mathbf{T}_i))$ as a unique intermediate representation for probing. For robust probe results, we implement nested cross validation [39] and report results taken across all outer folds, see the supplement for details. We conduct a grid search on two hyperparameters: the regularization α_j and the number of principal components of our regression inputs x_i . See the supplement for details. The Ridge regression parameters are calculated using Cholesky decomposition [12] for a precise closed form solution. Throughout this

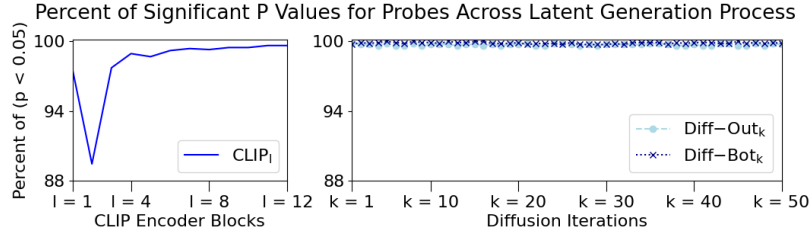


Fig. 2: The percentage of significant predicted attributes with $p < 0.05$ across all folds. On the left, we visualize the percentages of CLIP_l probes. On the right, we visualize the percentages, both for probes of Diff-Out_k and Diff-Bot_k . We observe that most P-values are significant for probes across Stable Diffusion, with only a few non-significant ones across the hundreds of attributes that we probe. We provide a further analysis of the non-significant ones in the supplemental material.

work, we use $|II| = 2500$ permutations for permutation tests. Following convention, we use $p < 0.05$ as the significance threshold for the RMSE P-value. Likewise, when discussing entanglement, we say that an attribute pair is positively entangled if $p > 0.95$, and a pair is negatively entangled if $p < 0.05$.

4.3 Alignment Between Stable Diffusion and Humans

Visualizing P-Values. In Fig. 2, we visualize the P-values for alignment that we obtain from our probes. We observe that 99.74% of the probes on the final output latent feature maps $\text{Diff-Out}(\text{CLIP}(\mathbf{T}_i))$ have a significant P-value. This is a striking result, as it shows that even the latent feature map that is decoded into an image has a general semantic representation of the objects it contains. Furthermore, the performance of the ridge regression probes is overall significantly above chance, and we conclude that there is an alignment between Stable Diffusion’s latent feature maps and the human perception of objects across a wide range of object attributes during the latent generation process.

Even the output of CLIP exhibits a significant alignment for most attributes. As the CLIP features are inserted into each repetition and at various layers of the U-Net, it may also indicate that the alignment is actually induced by the CLIP encoder and not the diffusion model which is verified by the analysis in the following section. In the subsequent analyses, we only evaluate probes which achieved $p < 0.05$, as we regard the outputs of the remaining probes as not meaningful.

Average RMSE and Standard Error. To visualize how close the predicted attribute ratings match the human annotations, we plot the RMSE across all attributes and outer folds over different stages of the model in Fig. 3. We observe the following: 1) The average RMSE is always lower for probes of Diff-Bot_k than for probes of Diff-Out_k at every diffusion step k and furthermore nearly constant. We conjecture that this is due to the CLIP features that are inserted at every iteration before the bottleneck. 2) The average RMSE for Diff-Out_k increases for

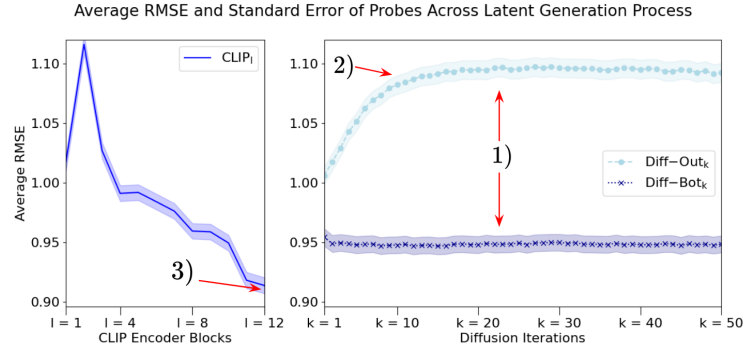


Fig. 3: The average RMSE of the probes visualized with the standard error. Left, we show the RMSE of CLIP_l probes as a baseline. Right, we visualize the RMSE for probes of Diff-Out_k and Diff-Bot_k. Observations 1), 2), and 3) are elaborated on in the main text.

probes at early iterations and then plateaus. This is expected, as the diffusion process converges from initial semantic concepts to pixel-level visual details. 3) The average RMSE is lowest after the final layer of CLIP. This finding indicates that the semantically most meaningful representation does not actually come from the diffusion model, but instead from the pretrained CLIP model. The diffusion model on the other hand only serves as a "visual decoding" of the representation provided by CLIP. We verify that the RMSE minimum at CLIP's output is significantly lower than the RMSE across Diff-Out_k and Diff-Bot_k using paired samples t-tests [53]. We find that the differences are significant ($p < 0.05$) across all Diff-Out_k and Diff-Bot_k. We conclude that the semantic representation of an object in Stable Diffusion is in fact most human-like at the output of CLIP. Critically, the reverse diffusion process degrades this alignment between representations.

Spatial and Non-Spatial Attribute Analysis. Having demonstrated the effectiveness of our probing method in general, we want to explore whether certain groupings of attributes are more or less predictable. Of special interest to us is the difference between *spatial* and *non-spatial* attributes. Here, spatial attributes describe anything related to physicality or appearance of an object, for example: "does it have corners?", and non-spatial attributes are the remaining, for example: "do you love it?". For a full list of these attributes, see the supplement. We care about this distinction because we expect a model which generates images to have a better understanding of spatial attributes. Comparisons between spatial and non-spatial attributes are shown in Fig. 4. On average, the non-spatial attributes are more decodable than the spatial attributes across all CLIP_l, Diff-Out_k, and Diff-Bot_k. This is unexpected, as Stable Diffusion is trained to produce an image latent feature map, so we expect spatial attributes to be more accurately decoded by probing. Additionally, in the supplement, we examine finer-grained subgroups for further insights.

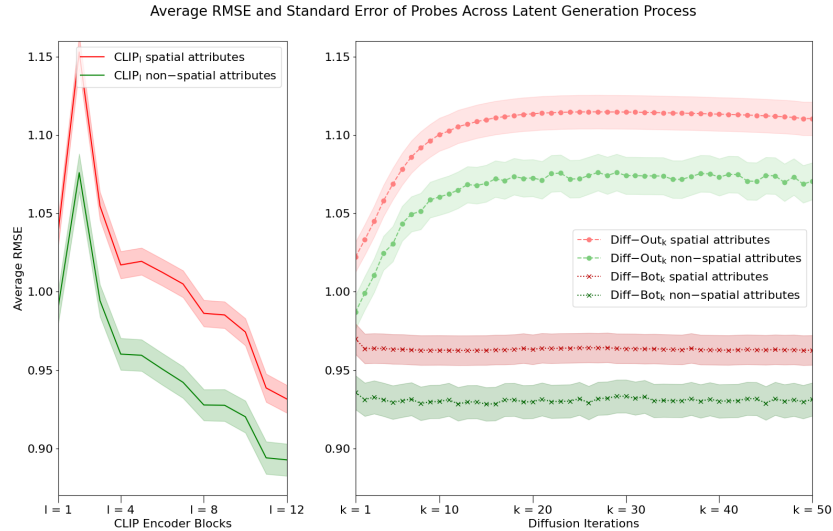


Fig. 4: We visualize the average RMSE and standard error of all spatial (red) and non-spatial (green) attributes. Spatial attributes have a higher average RMSE across all CLIP_l, Diff-Out_k, and Diff-Bot_k.

4.4 Weight Entanglements

We analyze entanglement of CLIP_l, Diff-Out_k, and Diff-Bot_k, with the aim of understanding how entanglement changes between the model and human perception domains. To do this, we look at two quantities in particular: the first being the amount of attribute pairs which are entangled by the probes, but are disentangled by humans. The second being the amount of attribute pairs which are entangled by humans, but disentangled by the probes. See Tab. 1 for our numerical results.

From these results, we conclude that CLIP can effectively disentangle attributes which humans entangle. The opposite case occurs across Diff-Out_k. Here, it is more common for attribute pairs which are disentangled by humans to become entangled in our models. We argue that the image latent feature maps do not effectively disambiguate related attributes. Given the relatively high RMSE for the Diff-Out_k, this suggests that attributes are generally less interpretable in the latent feature maps than in other regions of the model. Across Diff-Bot_k, there are more attribute pairs which are entangled by humans that become disentangled in our models than attribute pairs which are disentangled by humans and entangled in our models, although the difference is not as pronounced as in CLIP. So, in the bottleneck of the U-Net, attribute pairs are effectively disambiguated, although not as strongly as in CLIP. This finding supports our comments in Sec. 4.3, that the U-Net bottleneck is more semantically interpretable than the output latent feature map.

Generally, as an object representation passes from the text prompt to the image latent feature map, attributes become more entangled in the probe models. As the model generates an image from text, the object semantics may become less pertinent to the model as the representation shifts to the visual pixel space.

	Humans Disentangle more than Probes	Probes Disentangle more than Humans	Agreement between Humans and Probes
CLIP _l	3.7 %	31.5%	64.8 %
Diff-Bot _k	10.1 %	19.0 %	70.9 %
Diff-Out _k	20.2 %	4.0 %	75.8 %

Table 1: We compare how attribute pairs are disentangled between the human ratings and probes across CLIP_l, Diff-Out_k, and Diff-Bot_k for the outer folds of nested cross validation. We compare the total percentage of attribute pairs that are disentangled by humans and entangled in the probe weights (marked in **red**) with the total percentage of attribute pairs that are entangled by humans and disentangled in the probe weights (marked in **blue**). In the final column, we list the percentage of attribute pairs that agree in both domains, that is, the attributes that are entangled by both, humans and the probes, or the attributes that are disentangled by both, humans and the probes. The final column is the remainder from the previous columns. For CLIP_l, the **blue** percentage is much higher than the **red** percentage. This difference is smaller for Diff-Bot_k. For Diff-Out_k, the drastically more attribute pairs become entangled from the human to the model than vice versa. The overall observation from the last column is that all models agree with humans to a large degree in entangle- and disentanglement.

4.5 Discussion and Future Direction

Our investigation shows that Stable Diffusion’s reverse diffusion process does overall not improve semantic understanding much more than what can be obtained from CLIP. This is an interesting finding, as it indicates that the diffusion process does not learn semantics, but instead serves only as a visual decoder of the representation already available. Notably, diffusion models for image generation without language conditioning also exist and are capable of generating high-quality images. A future step in our investigation could be to apply our technique to those models next, to see if they exhibit any alignment with humans. However, to accomplish this requires a dataset of images and attribute labels that is not available today. We plan to create such a dataset in the future. This will also allow to not only compare attributes that are associated to general prompts, but to actually generated images.

An additional finding from Sec. 4.3 is that spatial attributes are less-well represented than non-spatial ones. This indicates that although Stable Diffusion is trained to output 2D images of the scenes, its training process does not learn good representations for spatial relationships and motivates future research on designing models that bring in such spatial relationship explicitly either in 2D or 3D.

On the other hand, our results indicate that CLIP is able to disentangle attributes better than humans in many cases, which indicates that unsupervised training of large vision-language models is a promising approach to learn semantics that do not involve, or involve only limited spatial understanding. Future research should also investigate how well large language-only and vision-language models can align with human perception, respectively.

As a next step, it will also be interesting to apply our technique to a variety of models in general, including more text-to-image diffusion models as well as GANs, to understand their respective differences in human alignment and reveal any favorable architecture biases.

5 Conclusion

In this work, we have explored the alignment between Stable Diffusion’s latent representations of objects and human perceptions. We found that most human attribute ratings can be predicted from the model representations with an accuracy significantly below chance, and that CLIP is primarily responsible for generating these decodable representations. Not only are CLIP’s object representations more decodable, but they are also more disentangled than those later in the generation process. In general, non-spatial attributes are in average more accurately decoded than spatial ones. The most salient insight from this analysis is that despite being a model trained to generate images of physical objects, conceptually high-level semantic attribute probes are more accurate than attributes related to physicality. In the future, we aim to create an image dataset labeled with attribute ratings and hope to generalize our results by probing a wide range of generative models. Our work represents a step towards documenting an alignment between generative models and human perception, that in the long term we hope will enable us to design AI models that are more in line with the human understanding of the world than today.

6 Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – GRK 2853/1 “Neuroexplicit Models of Language, Vision, and Action” - project number 471607914. We would like to thank Tom Fischer and Raza Yunus for their feedback on this work. The authors gratefully acknowledge support from MPI for Software Systems.

A Supplementary Material

In Sec. A.1, we provide details of the nested cross validation procedure promised in Sec. 4.2. In Sec. A.2, we provide details of the grid search promised in Sec. 4.2. In Sec. A.3, we provide the analysis of smaller subgroups promised in Fig. 2. In Sec. A.4, we provide an analysis of attributes whose probes had non-significant P-values at the output of CLIP. In Sec. A.5, we address the methods used to acquire the MTurk dataset. Finally, in Sec. A.6, we list the attributes used in subgroups, both in Sec 4.3, and Sec. A.3.

A.1 Nested Cross Validation Details

Nested Cross Validation [9, 48] is a robust method for assessing models, which we use in our method. We divide our human annotation data into 5 outer folds, each consisting of 200 objects and all of their attribute annotations from the MTurk data. Each outer fold has ridge regressions trained on a regression regularization parameter α_j , and a number of principal components, that have been optimized via cross validation for RMSE on the other 4 outer folds.

For this cross validation, the 4 outer folds are treated as inner folds. We have regressions trained on each subset of 3 inner folds, and validated on the remaining inner fold. This means that, for every α_j and principal component count, we have RMSE evaluated on 4 models. We select the α_j and principal component configuration with the lowest average RMSE across these 4 regression models through a grid search (see details in Sec. A.2), and which is used as the configuration for the ridge regression for the outer fold.

The ridge regression model is trained on the 4 outer folds used during the cross validation, and is evaluated on the remaining outer fold. RMSE values which we report in our results section are the average across the evaluations of all the outer folds. Percentages of P-values are also calculated across all outer folds.

A.2 Grid Search Details

To find optimal hyperparameters for the inner fold cross validation, we conduct a grid search across the ridge regression regularizer α , and the number of principal components, optimizing for the cumulative root mean squared error (RMSE) of the ridge regression models when assessed against the validation data.

Due to the high computation effort of a unique grid search across each regression model β_j , c_j , we search for hyperparameters which worked well across all CLIP_{*l*}, all Diff-Out_{*k*}, and all Diff-Bot_{*k*}, respectively. We argue that each of these components have a relatively similar representation space across *l* or *k*, and therefore results will be close to the fine-grained grid searches. Furthermore, optimizing over the average RMSE for all attributes, rather than optimizing over each attribute individually makes our analysis of the alignment of Stable Diffusion more robust, as we are not over optimizing each attribute.

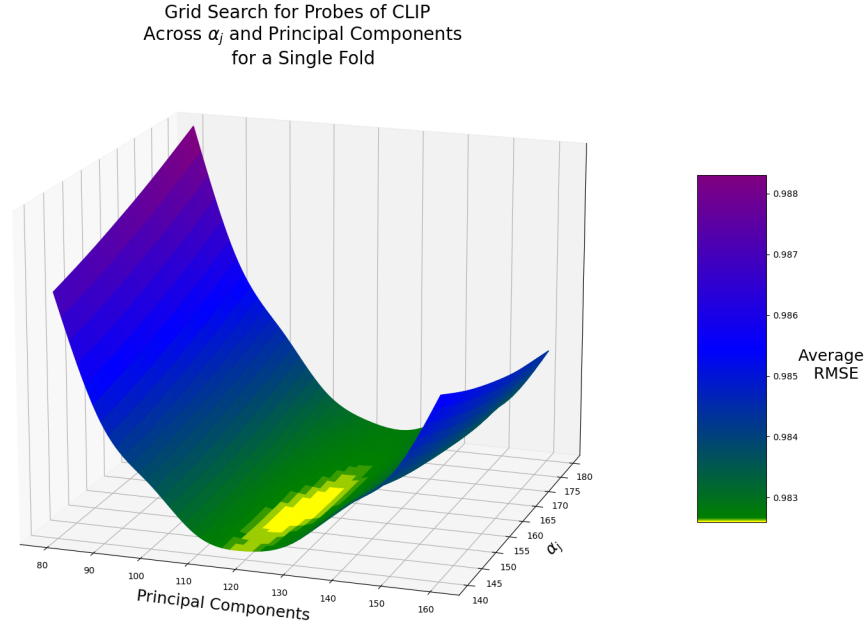


Fig. 5: We visualize the grid search for the ridge regression hyperparameters for probes of CLIP for a single fold. The average RMSE has a saddle point near 120 principal components, with $\alpha_j = 150$. Therefore, we use these hyperparameters for the probes that are evaluated in our work.

For example, on the grid search for CLIP_l , we compute the average RMSE for a hyperparameter configuration across all attributes, and all indices l . In the case of Diff-Out_k and Diff-Bot_k , we evaluate only across every 10th layer for computational efficiency.

For CLIP_l , our grid searches are run for α_j values between 110 and 180, and a number of principal components between 80 and 160. For Diff-Out_k , our grid searches are run for α_j values between 8,000 and 14,000, and a number of principal components between 1,550 and 1,950. For Diff-Bot_k , our grid searches are run for α_j values between 5,000 and 7,000, and a number of principal components between 600 and 1,100. We provide a visualization of this hyperparameter grid search for all CLIP_l in one fold of in Fig. 5

A.3 Further Subgroup Analyses

We extend our analysis in Sec. 4.3, to understand probe performance over more focused subgroups of attributes —*i.e.* attributes describing animacy, attributes describing perceptual features, and attributes describing size. Full lists of the attributes in these groups are in Sec. A.6. For each of these subgroups, we compare the average RMSE of our probes across the latent generation process for

attributes within a subgroup, against all remaining attributes. We show and discuss the results for each of these subgroups below.

Animacy. Animacy attributes are rated highly for living things. See Fig. 6 for the results. Attributes relating to the animacy of an object have lower RMSE across all components than the average. This suggests that Stable Diffusion maintains a strong understanding of animacy across the entire generation. The attributes describing perceptual features have higher RMSE than the average across all viewed components.

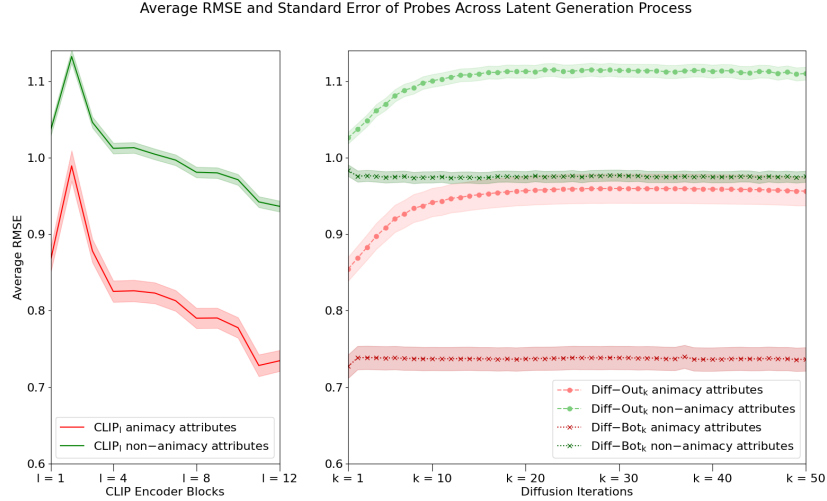


Fig. 6: We visualize the average RMSE and standard error of all animacy attributes (red), and the remaining attributes (green). Animacy attributes have a lower average RMSE across all CLIP_l, Diff-Out_k, and Diff-Bot_k.

Perceptual. Perceptual attributes describe low level visual features. See Fig. 7 for the results. Perceptual attributes describe global spatial properties, which may not be semantically meaningful, and hence not present in the U-Net bottleneck or in CLIP. Such a global spatial property may also not be easily interpreted by a linear predictor.

Size. Size attributes describe the physical size of the object. See Fig. 8 for the results. We observe that size-related attributes have lower error than average for the later layers of the CLIP encoder and in the bottleneck of the U-Net, but have higher than average RMSE for the U-Net output. We conjecture that because size is a global object property rather than a local one, it is easily decodable in the compact representation of the U-Net. However, it becomes challenging for the linear model to decode this property in the U-Net output, as it is not expressed anywhere locally in the representation.

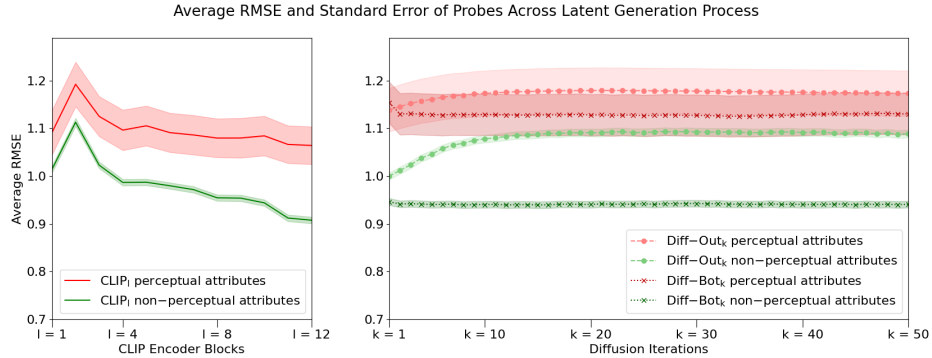


Fig. 7: We visualize the average RMSE and standard error of all perceptual attributes (red), and the remaining attributes (green). Perceptual attributes have a higher average RMSE across all CLIP_l, Diff-Out_k, and Diff-Bot_k.

A.4 Attributes with Non-Significant P-Values

We examine the probes on the output of CLIP, CLIP(**T**) which fail the permutation test. The attribute probes which we found to not have a significant P-value for at least one fold of evaluation were: “Is it a person?”, “is it dense?”, “is it a specific gender?”, and “does it have feathers?”.

The distribution of ratings for the attributes “Is it a person?”, “is it a specific gender?”, and “does it have feathers?” are extremely unimodal, that is, they have one value which occurs much more often than all others. This type of distribution can make regression models ineffective, as they are rewarded for setting β_j nearly to 0 and c_j to the mode of the distribution. This can make the regression more susceptible to outliers, and prone to a high P-value in the permutation test.

The one attribute with a more even rating distribution whose regressions are still failing the permutation test is “is it dense?”. We think this attribute is difficult to predict from CLIP because it depends on the ratio between the weight and size of an object. While the size of an object may be clear from an image, the weight may be difficult to understand without a deeper understanding of the world’s physics. CLIP may have an approximate understanding of weight by equating it to size, but this could actually make its understanding of density (the ratio between weight and size), less accurate. Additionally, density may not be mentioned in image captions too frequently, meaning that neither modality that CLIP is trained on would have a strong notion of density.

In Diff-Bot and Diff-Out, the only additional attribute which fails the permutation test at some inverse diffusion iterations is “is it an insect?”. In this case, the distribution is also extremely unimodal, which can help explain why the permutation test failed.

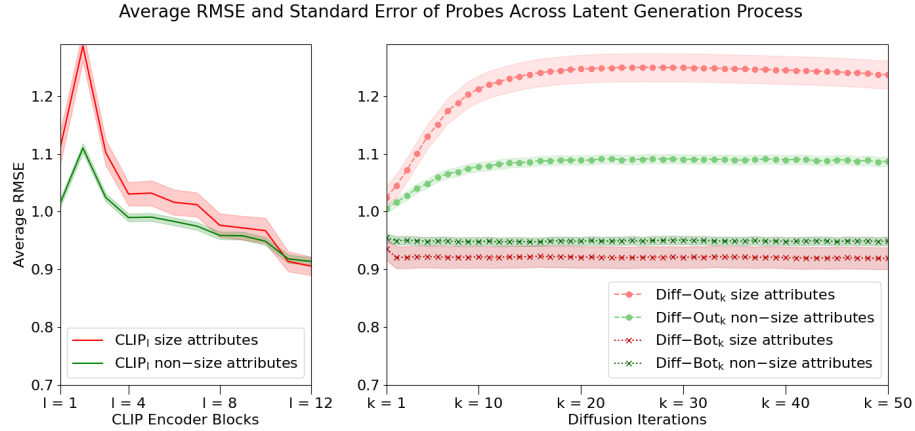


Fig. 8: We visualize the average RMSE and standard error of all size attributes (red), and the remaining attributes (green). Size attributes have a lower average RMSE across all Diff-Bot_k, but a higher average across all Diff-Out_k and most CLIP_l.

A.5 Crowd Sourcing Details

The MTurk dataset [49] was crowdsourced. It does not contain any information which may identify participants. For a wider conversation on the ethics of MTurk, we refer the reader to [28].

A.6 Attribute Subgroups

In this subsection, we enumerate the attributes present in the MTurk dataset, and the subgroups which we created for our analyses in Sec. A.3. We informally looked through the attributes, and found subgroups pertaining to animacy (Tab. 2), size (Tab. 3), and perceptual features (Tab. 4).

Table 2: Animacy attributes

Does it have a tail?
Does it have legs?
Does it have four legs?
Does it have feet?
Does it have paws?
Does it have feathers?
Does it have some sort of nose?
Does it have a hard nose/beak?
Can it run?
Is it fast?

Can it fly?
Can it jump?
Can it float?
Can it swim?
Can it dig?
Can it climb trees?
Can it cause you pain?
Can it bite or sting?
Does it stand on two legs?
Is it wild?
Is it a herbivore?
Is it a predator?
Is it warm blooded?
Is it conscious?
Does it have feelings?
Is it smart?

Table 3: Size attributes

Is it smaller than a golfball?
Is it bigger than a loaf of bread?
Is it bigger than a microwave oven?
Is it bigger than a bed?
Is it bigger than a car?
Is it bigger than a house?
Is it taller than a person?

Table 4: Perceptual attributes

Internal details
Verticality
Horizontalness
Left-diagonalness
Right-diagonalness
Aspect-ratio: skinny->fat
Prickiliness
Line curviness
3d curviness

We also provide our split of attributes into *spatial* (Tab. 5) and *non-spatial* (Tab. 6) attributes, which is used in Sec. 4.3. We define a spatial attribute to be

anything relating to size, shape, color, material, subcomponents, being part of a larger entity, or anything else related to direct physicality. Non-spatial attributes were all remaining components. These typically involved higher level semantics.

Table 5: Spatial attributes

Is it made of metal?
Is it made of plastic?
Is part of it made of glass?
Is it made of wood?
Is it shiny?
Can you see through it?
Is it colorful?
Is one more than one colored?
Is it always the same color(s)?
Is it white?
Is it red?
Is it orange?
Is it flesh-colored?
Is it yellow?
Is it green?
Is it blue?
Is it silver?
Is it brown?
Is it black?
Is it curved?
Is it straight?
Is it flat?
Does it have a front and a back?
Does it have a flat / straight top?
Does it have flat / straight sides?
Is taller than it is wide/long?
Is it long?
Is it pointed / sharp?
Is it tapered?
Is it round?
Does it have corners?
Is it symmetrical?
Is it hairy?
Is it fuzzy?
Is it clear?
Is it smooth?
List continued on the next page

Is it soft?
Is it heavy?
Is it lightweight?
Is it dense?
Is it slippery?
Can it bend?
Can it stretch?
Can it break?
Is it fragile?
Does it have parts?
Does it have moving parts?
Does it come in pairs?
Does it come in a bunch/pack?
Does it live in groups?
Is it part of something larger?
Does it contain something else?
Does it have internal structure?
Does it open?
Is it hollow?
Does it have a hard outer shell?
Does it have at least one hole?
Is it manufactured?
Does it come in different sizes?
Is it smaller than a golfball?
Is it bigger than a loaf of bread?
Is it bigger than a microwave oven?
Is it bigger than a bed?
Is it bigger than a car?
Is it bigger than a house?
Is it taller than a person?
Does it have a tail?
Does it have legs?
Does it have four legs?
Does it have feet?
Does it have paws?
Does it have claws?
Does it have horns / thorns / spikes?
Does it have hooves?
Does it have a face?
Does it have a backbone?
Does it have wings?
Does it have ears?
List continued on the next page

Does it have roots?
Does it have seeds?
Does it have leaves?
Does it have feathers?
Does it have some sort of nose?
Does it have a hard nose/beak?
Does it contain liquid?
Does it have wires or a cord?
Does it have writing on it?
Does it have wheels?
Does it roll?
Does it stand on two legs?
Is it mechanical?
Is it electronic?
Does it cast a shadow?
Can you hold it?
Can you hold it in one hand?
Can you pick it up?
Can you sit on it?
Can you ride on/in it?
Could you fit inside it?
Would you find it on a farm?
Would you find it in a school?
Would you find it in a zoo?
Would you find it in an office?
Would you find it in a restaurant?
Would you find in the bathroom?
Would you find it in a house?
Would you find it near a road?
Would you find it in a dump/landfill?
Would you find it in the forest?
Would you find it in a garden?
Would you find it in the sky?
Do you find it in space?
Does it live above ground?
Does it live in water?
Internal details
Verticality
Horizontalness
Left-diagonalness
Right-diagonalness
Aspect-ratio: skinny->fat
List continued on the next page

Prickiliness
Line curviness
3d curviness

Table 6: Non-Spatial attributes

Is it an animal?
Is it a body part?
Is it a building?
Is it a building part?
Is it clothing?
Is it furniture?
Is it an insect?
Is it a kitchen item?
Is it manmade?
Is it a tool?
Can you eat it?
Is it a vehicle?
Is it a person?
Is it a vegetable / plant?
Is it a fruit?
Does it change color?
Can it change shape?
Does it have a hard inside?
Is it alive?
Was it ever alive?
Is it a specific gender?
Was it invented?
Was it around 100 years ago?
Are there many varieties of it?
Does it grow?
Does it come from a plant?
Does it make a sound?
Does it make a nice sound?
Does it make sound continuously when active?
Is its job to make sounds?
Can it run?
Is it fast?
Can it fly?

List continued on the next page

Can it jump?
Can it float?
Can it swim?
Can it dig?
Can it climb trees?
Can it cause you pain?
Can it bite or sting?
Is it wild?
Is it a herbivore?
Is it a predator?
Is it warm blooded?
Is it a mammal?
Is it nocturnal?
Does it lay eggs?
Is it conscious?
Does it have feelings?
Is it smart?
Does it use electricity?
Can it keep you dry?
Does it provide protection?
Does it provide shade?
Do you see it daily?
Is it helpful?
Do you interact with it?
Can you touch it?
Would you avoid touching it?
Do you hold it to use it?
Can you play it?
Can you play with it?
Can you pet it?
Can you use it?
Do you use it daily?
Can you use it up?
Do you use it when cooking?
Is it used to carry things?
Can you control it?
Is it used for transportation?
Is it used in sports?
Do you wear it?
Can it be washed?
Is it cold?
Is it cool?
List continued on the next page

Is it warm?
Is it hot?
Is it unhealthy?
Is it hard to catch?
Can you peel it?
Can you walk on it?
Can you switch it on and off?
Can it be easily moved?
Do you drink from it?
Does it go in your mouth?
Is it tasty?
Is it used during meals?
Does it have a strong smell?
Does it smell good?
Does it smell bad?
Is it usually inside?
Is it usually outside?
Does it get wet?
Can it live out of water?
Do you take care of it?
Does it make you happy?
Do you love it?
Would you miss it if it were gone?
Is it scary?
Is it dangerous?
Is it friendly?
Is it rare?
Can you buy it?
Is it valuable?

References

1. Abdou, M., Gonzalez, A.V., Toneva, M., Hershcovich, D., Søgaard, A.: Does injecting linguistic structure into language models lead to better alignment with brain recordings? (2021)
2. Alain, G., Bengio, Y.: Understanding intermediate layers using linear classifier probes (2018)
3. Arkhipkin, V., Filatov, A., Vasilev, V., Maltseva, A., Azizov, S., Pavlov, I., Agafonova, J., Kuznetsov, A., Dimitrov, D.: Kandinsky 3.0 technical report (2023)
4. Aw, K.L., Toneva, M.: Training language models to summarize narratives improves brain alignment (2023)

5. Balaji, Y., Nah, S., Huang, X., Vahdat, A., Song, J., Zhang, Q., Kreis, K., Aittala, M., Aila, T., Laine, S., Catanzaro, B., Karras, T., Liu, M.Y.: ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers (2023)
6. Belinkov, Y.: Probing classifiers: Promises, shortcomings, and advances (2021)
7. Bie, F., Yang, Y., Zhou, Z., Ghanem, A., Zhang, M., Yao, Z., Wu, X., Holmes, C., Golnari, P., Clifton, D.A., He, Y., Tao, D., Song, S.L.: Renaissance: A survey into ai text-to-image generation in the era of large model (2023)
8. Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: Learning to follow image editing instructions (2023)
9. Cawley, G., Talbot, N.: On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research* **11**, 2079–2107 (07 2010)
10. Good, P.: *Permutation Tests*. Springer New York (2000). <https://doi.org/10.1007/978-1-4757-3235-1>, <http://dx.doi.org/10.1007/978-1-4757-3235-1>
11. Gupta, A., Boleda, G., Baroni, M., Padó, S.: Distributional vectors encode referential attributes. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (09 2015). <https://doi.org/10.18653/v1/D15-1002>
12. Haddad, C.N.: Cholesky Factorization, pp. 374–377. Springer US, Boston, MA (2009). https://doi.org/10.1007/978-0-387-74759-0_67, https://doi.org/10.1007/978-0-387-74759-0_67
13. Han, J., Kamber, M., Pei, J.: *Data mining concepts and techniques*, third edition (2012), http://www.amazon.de/Data-Mining-Concepts-Techniques-Management/dp/0123814790/ref=tmm_hrd_title_0?ie=UTF8&qid=1366039033&sr=1-1
14. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning*. Springer Series in Statistics, Springer New York Inc., New York, NY, USA (2001)
15. Hentschel, S., Kobs, K., Hotho, A.: CLIP knows image aesthetics. *Frontiers in Artificial Intelligence* **5** (Nov 2022). <https://doi.org/10.3389/frai.2022.976235>, <https://doi.org/10.3389/frai.2022.976235>
16. Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., Cohen-Or, D.: Prompt-to-prompt image editing with cross attention control (2022)
17. Hong, S., Lee, G., Jang, W., Kim, S.: Improving sample quality of diffusion models using self-attention guidance (2023)
18. Jeffers, J.N.R.: Two case studies in the application of principal component analysis. *Applied Statistics* **16**(3), 225 (1967). <https://doi.org/10.2307/2985919>, <http://dx.doi.org/10.2307/2985919>
19. Jolliffe, I.T., Cadima, J.: Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **374**(2065), 20150202 (Apr 2016). <https://doi.org/10.1098/rsta.2015.0202>, <http://dx.doi.org/10.1098/rsta.2015.0202>
20. Kang, M., Zhu, J.Y., Zhang, R., Park, J., Shechtman, E., Paris, S., Park, T.: Scaling up gans for text-to-image synthesis (2023)
21. Kim, Y., Lee, J., Kim, J.H., Ha, J.W., Zhu, J.Y.: Dense text-to-image generation with attention modulation (2023)
22. Kwon, M., Jeong, J., Uh, Y.: Diffusion models already have a semantic latent space (2023)
23. Köhn, A.: What’s in an embedding? analyzing word embeddings through multi-lingual evaluation. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. pp. 2067–2073 (01 2015). <https://doi.org/10.18653/v1/D15-1246>

24. Lewis, M., Nayak, N.V., Yu, P., Yu, Q., Merullo, J., Bach, S.H., Pavlick, E.: Does clip bind concepts? probing compositionality in large image models (2023)
25. Li, Y., Liu, H., Wu, Q., Mu, F., Yang, J., Gao, J., Li, C., Lee, Y.J.: Gligen: Open-set grounded text-to-image generation (2023)
26. Luo, C.: Understanding diffusion models: A unified perspective (2022)
27. McRae, K., Cree, G.S., Seidenberg, M.S., McNorgan, C.: Semantic feature production norms for a large set of living and nonliving things. *Behavior research methods* **37**(4), 547–559 (2005)
28. Moss, A.J., Rosenzweig, C., Robinson, J., Jaffe, S.N., Litman, L.: Is it ethical to use mechanical turk for behavioral research? relevant data from a representative survey of mturk participants and wages. *Behavior Research Methods* **55**(8), 4048–4067 (May 2023). <https://doi.org/10.3758/s13428-022-02005-0>, <http://dx.doi.org/10.3758/s13428-022-02005-0>
29. Muttenthaler, L., Dippel, J., Linhardt, L., Vandermeulen, R.A., Kornblith, S.: Human alignment of neural network representations (2023)
30. Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., Chen, M.: Glide: Towards photorealistic image generation and editing with text-guided diffusion models (2022)
31. Ojala, M., Garriga, G.C.: Permutation tests for studying classifier performance. In: 2009 Ninth IEEE International Conference on Data Mining. pp. 908–913 (2009). <https://doi.org/10.1109/ICDM.2009.108>
32. Park, Y.H., Kwon, M., Choi, J., Jo, J., Uh, Y.: Understanding the latent space of diffusion models through the lens of riemannian geometry (2023)
33. Peebles, W., Xie, S.: Scalable diffusion models with transformers (2023)
34. Po, R., Yifan, W., Golyanik, V., Aberman, K., Barron, J.T., Bermano, A.H., Chan, E.R., Dekel, T., Holynski, A., Kanazawa, A., Liu, C.K., Liu, L., Mildenhall, B., Nießner, M., Ommer, B., Theobalt, C., Wonka, P., Wetzstein, G.: State of the art on diffusion models for visual computing (2023)
35. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision (2021)
36. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents (2022)
37. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation (2021)
38. Ray, A., Radenovic, F., Dubey, A., Plummer, B.A., Krishna, R., Saenko, K.: Cola: A benchmark for compositional text-to-image retrieval (2023)
39. Refaellizadeh, P., Tang, L., Liu, H.: Cross-Validation, pp. 532–538. Springer US, Boston, MA (2009). https://doi.org/10.1007/978-0-387-39940-9_565, https://doi.org/10.1007/978-0-387-39940-9_565
40. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models (2022)
41. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation (2015)
42. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S.K.S., Ayan, B.K., Mahdavi, S.S., Lopes, R.G., Salimans, T., Ho, J., Fleet, D.J., Norouzi, M.: Photorealistic text-to-image diffusion models with deep language understanding (2022)
43. Schiappa, M.C., Cogswell, M., Divakaran, A., Rawat, Y.S.: Probing conceptual understanding of large visual-language models (2023)

44. Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., Komatsuzaki, A.: Laion-400m: Open dataset of clip-filtered 400 million image-text pairs (2021)
45. Shi, X., Padhi, I., Knight, K.: Does string-based neural mt learn source syntax? In: Conference on Empirical Methods in Natural Language Processing. pp. 1526–1534 (01 2016). <https://doi.org/10.18653/v1/D16-1159>
46. Sohl-Dickstein, J., Weiss, E.A., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics (2015)
47. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models (2022)
48. Stone, M.: Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society: Series B (Methodological)* **36**(2), 111–133 (Jan 1974). <https://doi.org/10.1111/j.2517-6161.1974.tb00994.x>, <http://dx.doi.org/10.1111/j.2517-6161.1974.tb00994.x>
49. Sudre, G., Pomerleau, D., Palatucci, M., Wehbe, L., Fyshe, A., Salmelin, R., Mitchell, T.: Tracking neural coding of perceptual and semantic features of concrete nouns. *NeuroImage* **62**(1), 451–463 (Aug 2012). <https://doi.org/10.1016/j.neuroimage.2012.04.048>, <https://doi.org/10.1016/j.neuroimage.2012.04.048>
50. Tang, R., Liu, L., Pandey, A., Jiang, Z., Yang, G., Kumar, K., Stenertorp, P., Lin, J., Ture, F.: What the daam: Interpreting stable diffusion using cross attention (2022), <https://arxiv.org/abs/2210.04885>
51. Thrush, T., Jiang, R., Bartolo, M., Singh, A., Williams, A., Kiela, D., Ross, C.: Winoground: Probing vision and language models for visio-linguistic compositionality (2022)
52. Wang, S., Zhang, Y., Zhang, X., Sun, J., Lin, N., Zhang, J., Zong, C.: An fmri dataset for concept representation with semantic feature annotations. *Scientific Data* **9**(1), 721 (2022)
53. Xu, M., Fralick, D., Zheng, J.Z., Wang, B., Tu, X.M., Feng, C.: The differences and similarities between two-sample t-test and paired t-test. *Shanghai Arch. Psychiatry* **29**(3), 184–188 (Jun 2017)
54. Yun, T., Bhalla, U., Pavlick, E., Sun, C.: Do vision-language pretrained models learn composable primitive concepts? (2023)
55. Zhang, C., Zhang, C., Zhang, M., Kweon, I.S.: Text-to-image diffusion models in generative ai: A survey (2023)
56. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models (2023), <https://arxiv.org/abs/2302.05543>