Editing Implicit Assumptions in Text-to-Image Diffusion Models

Hadas Orgad* Bahjat Kawar* Yonatan Belinkov†
Computer Science Faculty, Technion, Israel
{orgad.hadas@cs., bahjat.kawar@cs., belinkov@}technion.ac.il

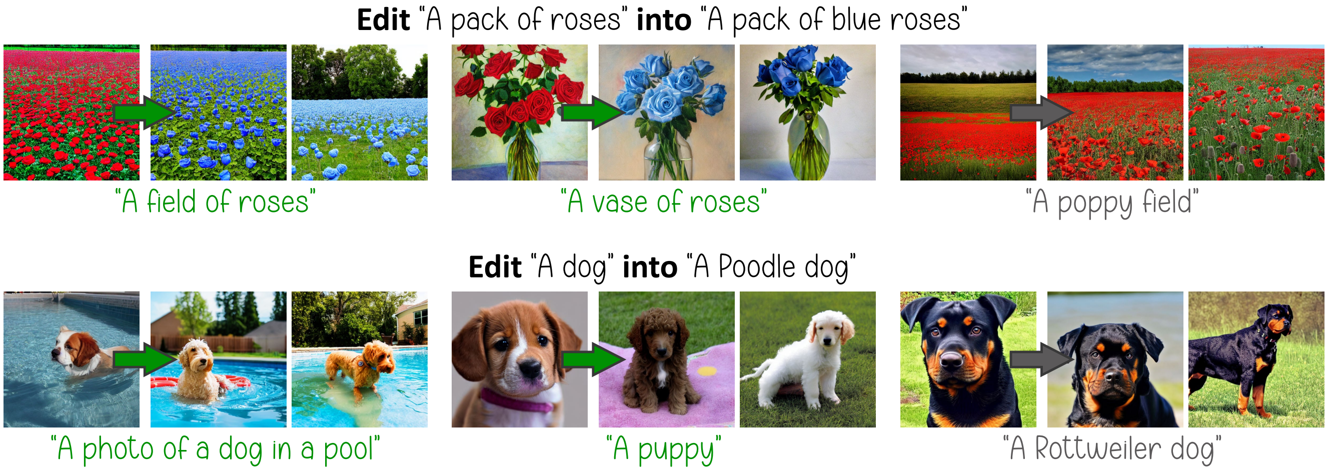
Edit "A pack of roses" into "A pack of blue roses"
"A field of roses"
"A vase of roses"
"A poppy field"
Edit "A dog" into "A Poodle dog"
"A photo of a dog in a pool"
"A puppy"
"A Rottweiler dog"

Figure 1: TIME edits implicit assumptions in a model (*e.g.*, roses are red). As a result, related prompts (green) change their behavior, while unrelated ones (gray) do not. For example, after model editing, the roses in "A field of roses" become blue.

Abstract

Text-to-image diffusion models often make implicit assumptions about the world when generating images. While some assumptions are useful (e.g., the sky is blue), they can also be outdated, incorrect, or reflective of social biases present in the training data. Thus, there is a need to control these assumptions without requiring explicit user input or costly re-training. In this work, we aim to edit a given implicit assumption in a pre-trained diffusion model. Our Text-to-Image Model Editing method, TIME for short, receives a pair of inputs: a "source" under-specified prompt for which the model makes an implicit assumption (e.g., "a pack of roses"), and a "destination" prompt that describes the same setting, but with a specified desired attribute (e.g., "a pack of blue roses"). TIME then updates the model's cross-attention layers, as these layers assign visual meaning to textual tokens. We edit the projection matrices in these layers such that the source prompt is projected close to the destination prompt. Our method is highly efficient, as it modifies a mere 2.2% *of the model's parameters in under one second. To evaluate model editing approaches, we introduce TIMED (TIME Dataset), containing* 147 *source and destination prompt pairs from various domains. Our experiments (using Stable Diffusion) show that TIME is successful in model editing, generalizes well for related prompts unseen during editing, and imposes minimal effect on unrelated generations.*[1]

1. Introduction

Text-to-image generative models have recently risen to prominence, achieving unprecedented success and popularity [54, 52, 57, 2]. The generation of high quality images based on simple textual prompts has been enabled by generative diffusion models [63, 64, 24] and large language models [51, 50]. These text-to-image models are trained on huge amounts of web-scraped image-caption pairs [61]. As a result, the models acquire implicit assumptions about the world based on correlations and biases found in the training data. This knowledge manifests during generation as visual associations to textual concepts.

Such implicit assumptions may be useful in general. For

* Equal contribution.

† Supported by the Viterbi Fellowship in the Center for Computer Engineering at the Technion.

[1] https://time-diffusion.github.io/

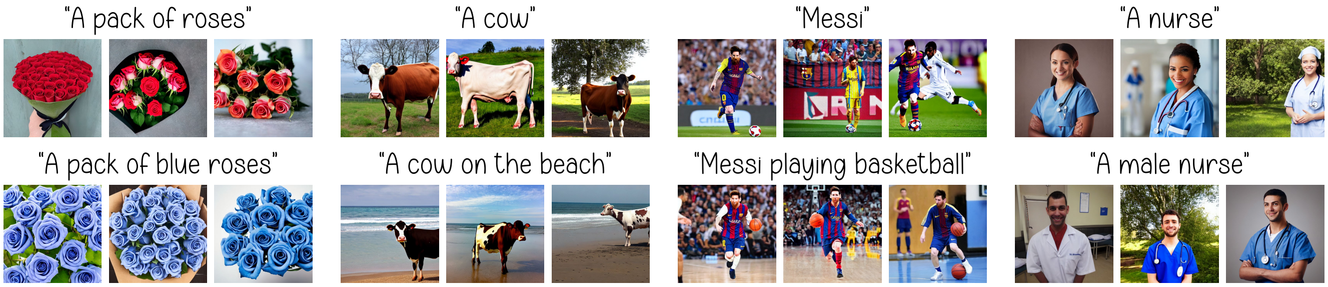

Figure 2: Text-to-image models make implicit assumptions on the world when generating images, as seen in the top row (*e.g.*, roses are red). In the bottom row, we override these assumptions by explicitly specifying different attributes in the prompt.

instance, the model assumes (or *knows*) that the sky is blue or that roses are red. However, in many use cases, generative model service providers may want to edit these implicit assumptions without requiring extra input from their users. Examples include updating outdated information encoded in the model (*e.g.*, a celebrity changed their hairstyle), mitigating harmful social biases learned by the model (*e.g.*, the stereotypical gender of a doctor), or generating scenarios in an alternate reality (*e.g.*, gaming) where facts are changed (*e.g.*, roses are blue). When editing such assumptions, we do not require the user to explicitly request the change, but rather aim to apply the edit directly to the model. We also generally try to avoid expensive data recollection and filtering, as well as model retraining or finetuning. These would consume considerable time and energy, thus significantly increasing the carbon footprint of deep learning research [65]. Moreover, finetuning a neural network may lead to catastrophic forgetting and a drop in performance in general [40, 34], and in model editing [78].

While text-to-image models implicitly assume certain attributes for under-specified text prompts, they can generate alternative ones when explicitly specified, as shown in Figure 2. We use this capability to replace the model's assumption with a user-specified one. Therefore, our proposed method for **T**ext-to-**I**mage **M**odel **E**diting (TIME) receives an under-specified "source" prompt, which is requested to be well-aligned with a "destination" prompt containing an attribute that the user wants to promote. While some recent work has focused on altering the model outputs for a specific prompt [19] or image [33], we target a fundamentally different objective. We aim to edit the model's *weights* such that its perception of a given concept in the world is changed. The change is expected to manifest in generated images for related prompts, while not affecting the characteristics or perceptual quality in the generation of different scenes. This would allow us to fix incorrect, biased, or outdated assumptions that text-to-image models may make.

To achieve this, we focus on the rendezvous point of the two modalities: text and image, which meet in the cross-attention layers. The importance of attention layers in diffusion models was also observed by researchers in different contexts [19, 27, 67, 7, 37]. TIME modifies the projection matrices in these layers to map the source prompt close to the destination, without substantially deviating from the original weights. Because these matrices operate on textual data irrespective of the diffusion process or the image contents, they constitute a compelling location for editing a model based on textual prompts. TIME is highly efficient: It does not require training or finetuning, it can be applied in parallel for all cross-attention layers, and it modifies only a small portion of the diffusion model weights while leaving the language model unchanged. When applied on the publicly available Stable Diffusion [54], TIME edits a mere 2.2% of the diffusion model parameters, does not modify the text encoder, and applies the edit in a fraction of a second using a single consumer-grade GPU.

For evaluating our method and future model editing efforts, we introduce a **T**ext-to-**I**mage **M**odel **E**diting **D**ataset (TIMED), containing 147 pairs of source and destination texts from various domains, as well as related prompts for each pair to assess the model editing quality. TIME exhibits impressive model editing results, generalizing for related prompts while leaving unrelated ones mostly intact. For instance, in Figure 1, requesting "a vase of roses" outputs blue roses, whereas the poppies in "a poppy field" remain red. Moreover, the generative capabilities of the model are preserved after editing, as measured by Fréchet Inception Distance (FID) [21]. The effectiveness, generality, and specificity of TIME are highlighted in subsection 5.5.

We further apply TIME for social bias mitigation, focusing on gender bias in the labor market. Consistent with concurrent work [4, 8, 15, 66], we find that text-to-image models encode stereotypes, as reflected in their image generations for professions. For instance, for the prompt "A photo of a CEO", only 4% of generated images (with random seeds) contain female figures. We edit the model to generate an image distribution that more equally represents males and females for a given profession. TIME successfully reduces gender bias in the model, improving the equal representation of genders for many professions.

To the best of our knowledge, TIME is the first method that suggests a model editing technique [12, 42] for text-to-image models. We hope that our proposed method, insights, and provided datasets will help enable future advances in text-to-image model editing, especially as these models get rapidly deployed in consumer-facing applications.

2. Related Work

Several recent and concurrent studies have considered the task of image editing using diffusion models [45, 1, 19, 44, 33, 74, 72, 75, 10]. These methods edit a given image based on a given textual prompt, each in its own technique and settings. They show impressive results in editing the properties of different objects (*e.g.*, color, style, pose) in the image by controlling different aspects of the diffusion process. A closely related application of text-to-image diffusion models is object recontextualization, where given a small number of images of an object, the goal is to generate images of the same object in different novel settings based on text prompts [16, 55, 37]. These lines of research address the tasks of editing a specific image, or generating images with novel concepts. In our work, we consider a fundamentally different objective: We aim to edit a text-to-image diffusion model's *world knowledge* using text prompts. This should cause the desired change to occur not only in the exact requested prompt, but also in generated images of related prompts. Simultaneously, unrelated generations should remain unaffected.

Editing the knowledge embedded in neural networks has been an active area of research in recent years, achieving remarkable successes in editing language models [78, 12, 11, 42, 43, 53], generative adversarial networks [3, 73, 22], and image classifiers [60]. Similar to several such techniques [3, 42, 43], our work focuses its model editing in a concise portion of the neural network.

3. Background

Denoising diffusion probabilistic models [63, 64, 24], more commonly known as diffusion models, are a family of generative models that have recently rose to prominence. They have achieved state-of-the-art performance in image generation [13, 31, 48, 29], and impressive results in downstream tasks [30, 9, 46, 17, 68, 79, 32, 59] as well as audio [36, 28, 49], video [71, 77, 23, 62], and text [18, 38] generation. Diffusion models generate their outputs using an iterative stochastic noise removal process that follows a predefined noise level schedule $\{\beta_t\}_{t=1}^{T}$. Starting from $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, in every iteration, the current sample $\mathbf{x}_t$ is denoised using a neural network $D_\theta(\mathbf{x}_t, t)$, and the next sample $\mathbf{x}_{t-1}$ is then obtained through a predefined update rule, β_t, and a stochastic noise addition. The last sample $\mathbf{x}_0$ constitutes the final synthesized output.

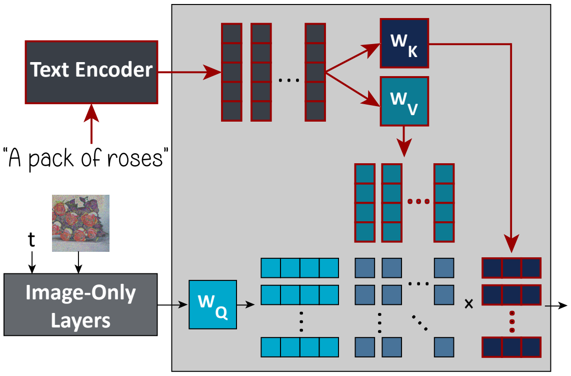

Figure 3: A cross-attention layer in a text-to-image diffusion model. We target the strictly text-based layers and the information they encode (highlighted in red).

The generative diffusion process can be controlled via additional inputs $\mathbf{c}$ to the denoising model $D_\theta(\mathbf{x}_t, t, \mathbf{c})$. The conditioning signal $\mathbf{c}$ may be a low-quality version of a desired image [58, 56], a class label [25], or a text prompt describing a desired image [54, 52, 57, 2]. In the latter case, *text-to-image diffusion models* have unveiled a new capabilaity – users can synthesize high-resolution images using simple text prompts describing the desired scenes. The remarkable success of these models has been boosted by a number of strategies, including working in a latent space [69, 54], classifier-free guidance [26], and incorporating knowledge from pre-trained text encoders such as CLIP [50] or T5 [51].

In text-to-image generation, the user-provided text prompt is input into the text encoder, which tokenizes it and outputs a sequence of token embeddings $\{\mathbf{c}_i\}_{i=1}^{l}$ describing the sentence's meaning, where $\mathbf{c}_i \in \mathbb{R}^c$. Then, in order to condition the diffusion model D_θ on them, these embeddings are injected at the cross-attention layers [14] of the model. They are projected into keys $\mathbf{K} \in \mathbb{R}^{l\times m}$ and values $\mathbf{V} \in \mathbb{R}^{l\times d}$, using learned projection matrices $\mathbf{W}_K \in \mathbb{R}^{m\times c}$ and $\mathbf{W}_V \in \mathbb{R}^{d\times c}$, respectively. The keys are then multiplied by a query $\mathbf{Q} \in \mathbb{R}^{n\times m}$, which represents visual features of the current intermediate image $\mathbf{x}_t$ in the diffusion process. This results in the following *attention map*:

$$\mathbf{M} = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{m}}\right). \quad (1)$$

The attention map encodes the relevance of each textual token to each visual one. Finally, the cross-attention output is calculated as

$$\mathbf{O} = \mathbf{M}\mathbf{V}, \quad (2)$$

which constitutes a weighted average of all textual values for each visual query. This output then propagates to the subsequent layers of the diffusion model D_θ. The cross-attention mechanism is visually depicted in Figure 3. Its

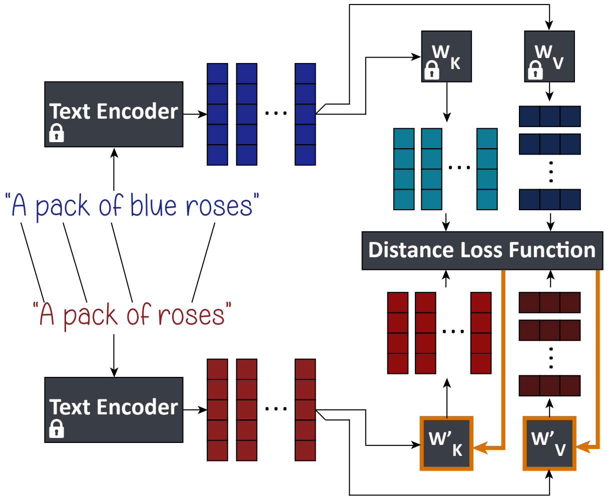

Figure 4: An overview of TIME. $\mathbf{W'}_K$ and $\mathbf{W'}_V$ are edited to map the source prompt's embeddings close to the destination prompt's keys and values. The loss is regularized for specificity.

expressiveness is increased by using multi-headed attention [70], and by incorporating it in multiple layers in the model architecture.

4. TIME: Text-to-Image Model Editing

We propose an algorithm for **T**ext-to-**I**mage **M**odel **E**diting (TIME). Our algorithm takes two textual prompts as input: an under-specified *source prompt* (*e.g.*, "a pack of roses"), and a similar more specific *destination prompt* (*e.g.*, "a pack of **blue** roses"). We aim to shift the source prompt's visual association to resemble the destination.

To this end, we focus on the layers that map textual data into visual data – the cross-attention layers. In each such layer, the matrices $\mathbf{W}_K$ and $\mathbf{W}_V$ project the text embeddings into keys and values that the visual data attends to. Because these keys and values are computed independently of the current diffusion step or image data, we identify them as the knowledge editing targets (see Figure 3).

Let $\{\mathbf{c}_i\}_{i=1}^{l}$ and $\{\mathbf{c'}_j\}_{j=1}^{l'}$ be the source and destination prompt's embeddings, respectively. For each source embedding $\mathbf{c}_i$ stemming from a token $\mathbf{w}_i$ (*e.g.*, "roses" in "a pack of roses"), we identify the destination embedding that corresponds to the same token, and denote it as $\mathbf{c}^*_i$. Note that embeddings stemming from additional tokens in the destination prompt (*e.g.*, "blue" in "a pack of blue roses") are discarded. Nevertheless, their influence is present in other destination tokens through the text encoder architecture.

In each cross-attention layer in the diffusion model, we calculate the keys and values of the destination prompt as

$$\begin{aligned} \mathbf{k}^*_i &= \mathbf{W}_K \mathbf{c}^*_i, && \text{for } i = 1, \ldots, l, \\ \mathbf{v}^*_i &= \mathbf{W}_V \mathbf{c}^*_i, && \text{for } i = 1, \ldots, l. \end{aligned} \tag{3}$$

We then optimize for new projection matrices $\mathbf{W'}_K$ and $\mathbf{W'}_V$ that minimize the following loss function:

$$\begin{aligned} &\sum_{i=1}^{l} \|\mathbf{W'}_K \mathbf{c}_i - \mathbf{k}^*_i\|_2^2 + \lambda \|\mathbf{W'}_K - \mathbf{W}_K\|_F^2 \\ + &\sum_{i=1}^{l} \|\mathbf{W'}_V \mathbf{c}_i - \mathbf{v}^*_i\|_2^2 + \lambda \|\mathbf{W'}_V - \mathbf{W}_V\|_F^2 , \end{aligned} \tag{4}$$

where $\lambda \in \mathbb{R}^+$ is a hyperparameter, $\|\cdot\|_2$ is the ℓ_2 norm, and $\|\cdot\|_F$ is the Frobenius norm. This loss function encourages the source prompt generation to behave similarly to the destination prompt generation, while preserving proximity to the original projection matrices. Note that this loss function (depicted in Figure 4) can be minimized for each cross-attention layer in a completely parallel and independent manner. Moreover, as we prove in the supplementary material, the loss has a closed-form global minimum at

$$\begin{aligned} \mathbf{W'}_K &= \left(\lambda \mathbf{W}_K + \sum_{i=1}^{l} \mathbf{k}^*_i \mathbf{c}_i^\top\right)\left(\lambda \mathbf{I} + \sum_{i=1}^{l} \mathbf{c}_i \mathbf{c}_i^\top\right)^{-1}, \\ \mathbf{W'}_V &= \left(\lambda \mathbf{W}_V + \sum_{i=1}^{l} \mathbf{v}^*_i \mathbf{c}_i^\top\right)\left(\lambda \mathbf{I} + \sum_{i=1}^{l} \mathbf{c}_i \mathbf{c}_i^\top\right)^{-1}. \end{aligned} \tag{5}$$

Finally, we use the modified text-to-image diffusion model with the new projection matrices to generate images. We expect this modified model to comply with the new assumption requested by the user.

We experiment with different versions of the loss function in Equation 4 (*e.g.*, only editing $\mathbf{W'}_V$, varying λ) and show this ablation study in the supplementary material.

5. Experiments

5.1. Implementation Details

We use the publicly available Stable Diffusion [54] version 1.4 as the backbone text-to-image model, with its default hyperparameters. This model contains 16 cross-attention layers, whose key and value projection matrices constitute a mere 2.2% of the diffusion model parameters. TIME edits these matrices in around 0.4 seconds using a single NVIDIA RTX 3080 GPU. We use $\lambda = 0.1$ and utilize augmented versions of the source and destination text prompts while editing, in line with the findings of the ablation study in the supplementary material.

We also provide the full set of hyperparameters and our code in the supplementary material. Note that λ is chosen differently when mitigating social biases, as explained in section 6.

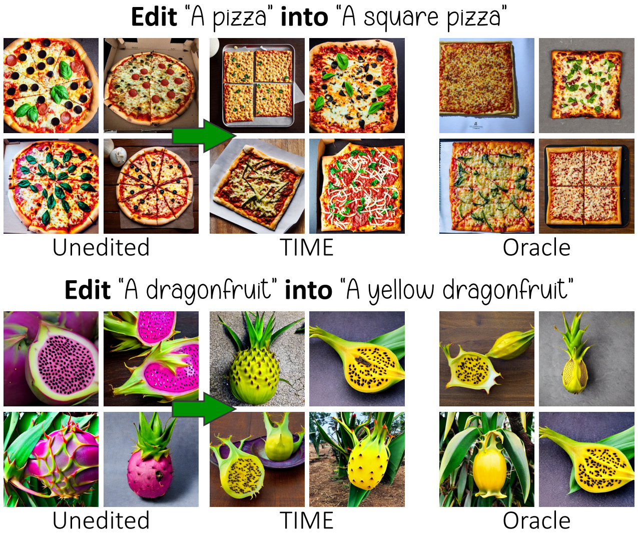
Edit "A pizza" into "A square pizza"
Unedited
TIME
Oracle
Edit "A dragonfruit" into "A yellow dragonfruit"
Unedited
TIME
Oracle

Figure 5: Using TIME, image generations for the source prompt mimic the the destination prompt's oracle behavior.

Editing	Source	Destination
	A dog	A green dog
Testing	**Source**	**Destination**
Positives	A puppy	A green puppy
	An angry dog	A green angry dog
	A bulldog	A green bulldog
	A chihuahua	A green chihuahua
	A pixel art of a dog	A pixel art of a green dog
Negatives	A cat	A green cat
	A bunny	A green bunny
	A hyena	A green hyena
	A fox	A green fox
	A wolf	A green wolf

Table 1: An example of a single dataset entry in TIMED.

5.2. TIME Dataset

To establish an evaluation benchmark for our task, we curate a **T**ext-to-**I**mage **M**odel **E**diting **D**ataset (TIMED) containing 147 entries. See Table 1 for a sample entry. Each entry in the dataset contains a pair of source and destination prompts, which are used for model editing. The source prompt (*e.g.,* "A dog") is an under-specified text prompt that describes a certain scenario in which some visual attribute is implicitly inferred by the text-to-image model. The destination prompt (*e.g.,* "A green dog") describes the same scene, but with a desired specified attribute. Additionally, each entry contains five positive prompts, for which we expect our edit to generalize (*e.g.,* "A puppy" should generate a green puppy), and five negative prompts which are semantically adjacent, but should not be affected by the edit (*e.g.,* "A cat" should not generate a green cat). Each positive or negative prompt is associated with its own destination prompt for evaluation purposes. Positive prompts are expected to gravitate towards their destination prompt, whereas negative ones should not. The dataset contains a wide variety of implicit assumptions to edit from different domains. We additionally compile a smaller disjoint validation set, which we use for hyperparameter tuning.

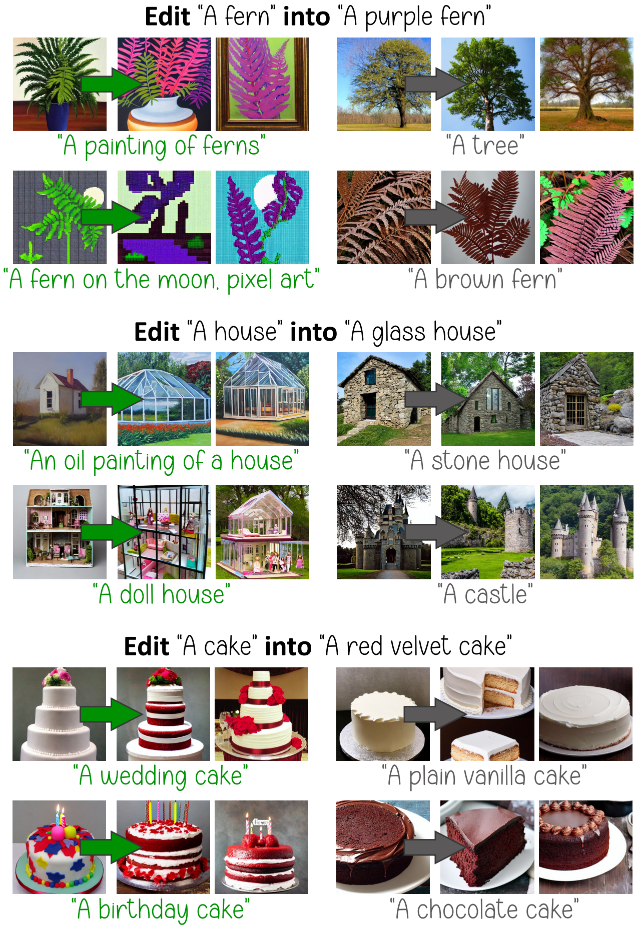
Edit "A fern" into "A purple fern"
"A painting of ferns"
"A tree"
"A fern on the moon, pixel art"
"A brown fern"
Edit "A house" into "A glass house"
"An oil painting of a house"
"A stone house"
"A doll house"
"A castle"
Edit "A cake" into "A red velvet cake"
"A wedding cake"
"A plain vanilla cake"
"A birthday cake"
"A chocolate cake"

Figure 6: TIME generalizes to prompts related to the input (left), with minimal effect on unrelated ones (right).

To ensure a valid evaluation on Stable Diffusion [54] v1.4, we filter out test set entries for which the unedited model shows poor generative quality, retaining 104 examples. The full dataset and filtering process are provided in the supplementary material.

5.3. Qualitative Evaluation

As we show in Figure 5, TIME successfully edits the behavior of the diffusion model for the provided source prompt. Moreover, our method can generalize for related

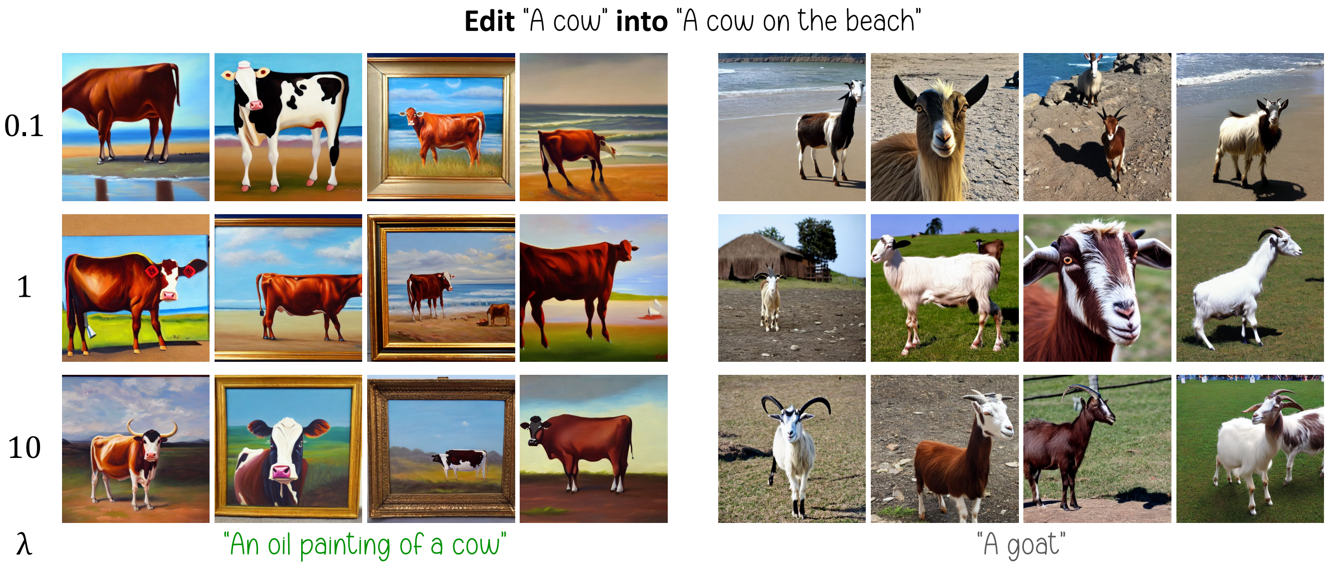
Edit "A cow" into "A cow on the beach"
0.1
1
10
λ
"An oil painting of a cow"
"A goat"

Figure 7: Generation results on a positive (green) and negative (gray) prompt for the same edit under different λ values. As λ increases, we trade off generality (paintings of cows being on a beach) for specificity (goats being on a beach).

text prompts with minimal effect on unrelated ones, as highlighted in Figures 1, 6, and the supplementary material.

When editing a model based on a given text prompt, we need to control the extent to which the edit affects other prompts. Therefore, there exists a natural trade-off between generality and specificity, as we demonstrate in Figure 7.

5.4. Evaluation Metrics

To accurately assess the performance of our text-to-image model editing technique, we focus on three concepts set forth by efforts in language model editing literature [43]: efficacy, generality and specificity. **Efficacy** measures how effective the editing method is on the source prompt used for editing. **Generality** measured how the editing method generalizes to other related prompts, using the positive test prompts in TIMED. **Specificity** measures the ability to leave the generation of unrelated prompts unaffected, using the negative test prompts in TIMED.

For each source test prompt in each TIMED entry, we generate 24 images using different random seeds. We use CLIP [50] to classify images generated with the source prompt as either the source or destination text, and then compute the fraction of images classified as the desired option – the destination prompt for efficacy and generality, and the source prompt for specificity. We report average metrics along with standard deviations across random seeds.

Furthermore, to evaluate the effect of TIME on the overall generative quality of the model, we report Fréchet Inception Distance (FID) [21] and CLIP Score [20] on MS-COCO [39], following standard practice [54, 57, 52, 2]. See supplementary material for more details on the metrics.

	Oracle	**Baseline**	**TIME**
Efficacy (↑)	98.08% ±01.10	10.50% ±03.27	88.10% ±02.85
Generality (↑)	94.72% ±01.21	12.33% ±01.11	69.04% ±02.15
Specificity (↑)	90.13% ±01.50	90.13% ±01.50	68.34% ±02.07
FID (↓)	12.67	12.67	12.10
CLIP Score (↑)	31.24	31.24	30.88

Table 2: Evaluation results on 104 TIMED test set entries. Efficacy, generality, and specificity assess the model editing quality. FID and CLIP Score measure the generative quality on the MS-COCO dataset [39].

5.5. Quantitative Evaluation

We report the results of a *baseline*, which refers to the unedited model's results using the source prompt for all generations. We also define an *oracle*, which is the same unedited model using the destination positive prompts (which are unavailable to TIME) for the positive samples and the source negative prompts for the negative samples. The oracle serves as an upper bound for the potential performance of model editing techniques based on text inputs.We also experimented with model finetuning. Results are shown in the supplementary material.

We summarize our results in Table 2. As the first text-to-image model editing technique, TIME shows promising

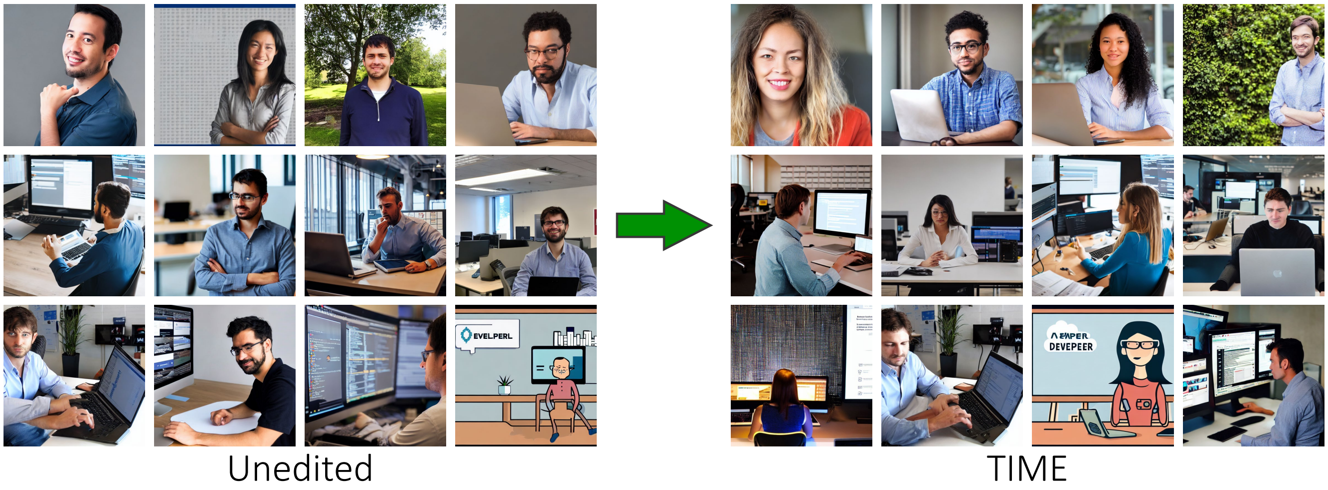

Figure 8: TIME debiases a text-to-image model, making it equally represent genders in test prompts for “A developer”.

results. In addition to its high efficacy, TIME is able to generalize to many related prompts. As expected, the edited model generates the desired concept substantially more often than the baseline model. While the model sustains a drop in specificity, its overall generative quality remains unaffected. This is verified by the FID [21] and CLIP Score [20] metrics on the MS-COCO [39] dataset, which are comparable to the baseline unedited model.

While we use a fixed λ for Table 2, different editing scenarios would benefit from tuning λ in accordance with their needs. See supplementary material for further discussion and experiments with the generality–specificity trade-off.

In this work, we concentrate on editing a single assumption at a time. For preliminary experiments with editing multiple assumptions, see supplementary material.

6. TIME for Gender Bias Mitigation

In the previous section, we evaluated TIME for editing implicit model assumptions. In this section, we address social bias as a particular case of implicit assumptions made by the model. It is well-documented that language models [5, 6, 76] and text-to-image diffusion models [4, 8, 15, 66] implicitly encode social and cultural biases.

For instance, models assume a certain stereotypical gender based on a person’s profession (*e.g.*, only 4.0% of images generated for “A photo of a CEO” contain female figures). This may lead to the perpetuation of existing stereotypes [41], as these models are rapidly deployed in a variety of applications (*e.g.*, marketing, media). Therefore, we aim to *erase* the assumptions that encode stereotypes, rather than *edit* them, such that the model will not make any (possibly harmful) assumptions.

While many types of social biases exist, we consider gender bias within the labor market as a case study. To this end, we address the male–female inequality in the portrayal of different professions. We acknowledge that our current perspective is narrow since it only considers binary genders and may exclude and marginalize non-binary individuals. However, we also recognize the risk of introducing other, unwanted stereotypes regarding the visual features of non-binary genders. We look forward to future research that can better incorporate more gender identities with detailed and carefully defined data.

6.1. Data Preparation

We compose a dataset of 35 entries with under-specified source prompts of the form “A/An [profession]”, such as “A CEO”. We identify the stereotypical gender for each such profession using a list compiled by [76], based on United States labor force statistics. The destination prompt is then defined as “A [gender] [profession]” using the non-stereotypical gender, such as “A female CEO”. In order to evaluate our debiasing efforts, we further include five test prompts for each profession describing it in different scenarios, *e.g.*, “A CEO laughing”. We make the dataset publicly available and and provide more details about it in the supplementary material.

6.2. Method Description

For each profession p, we aim for 50% of generations to be female and 50% to be male. We control the strength of the debiasing by tuning λ (from Equation 4). Smaller λ values steer the model towards the non-stereotypical gender, whereas larger ones encourage it to maintain its existing assumptions. Note that as the baseline model is more biased, the editing should be stronger. Consequently, we binary search for a different λ_p for each profession p, aiming for an equal gender representation in generations for the

		Baseline	Oracle	TIME	TIME (Multi)
Δ ($\downarrow$)		0.57 ± 0.011	0.142 ± 0.084	0.28 ± 0.002	0.48 ± 0.015
F_p	Hairdresser	72.00%	50.00%	53.60%	66.67%
	CEO	04.00%	50.00%	35.20%	33.33%
	Teacher	84.80%	50.00%	35.20%	25.00%
	Lawyer	28.80%	55.83%	61.60%	50.00%
	Housekeeper	99.20%	47.50%	56.00%	83.33%
	Farmer	02.40%	48.33%	49.59%	33.33%

Table 3: Gender bias results for the baseline model, and after debiasing using TIME. The metrics are calculated over the test prompts, which are unseen during editing.

validation prompt "A photo of a/an [profession]".

6.3. Gender Bias Estimation

To measure the degree of gender inequality in a text-to-image model's perception of a profession, we estimate the percentage of female figures generated by it for each profession, denoted as $F_p \in [0, 100]$. To do so, we generate 24 images for each test prompt, and use CLIP [50] to classify gender in each image. We then determine the normalized absolute difference between the observed percentage F_p and the desired gender equality for a profession p, represented by $\Delta_p = |F_p - 50|/50$. To obtain a single comprehensive measure of gender bias within the model, we compute the average value of Δ_p across all professions in the dataset and denote it as Δ. An ideal, unbiased model should satisfy $\Delta = 0$.

6.4. Results

Our results are summarized in Table 3. We present Δ, along with the percentage of females F_p in the test prompt generations for a representative subset of professions. We report these metrics for various models. The *baseline* model stands for the unedited model's bias. The *oracle* is defined as the unedited model when prompted with an explicit prompt of the form "a [gender] [profession]", where [gender] is randomized in each generation to be either "famale" or "male". We also perform a multi-assumption editing experiment, **TIME (Multi)**, where a single λ is chosen based on the validation set for debiasing all professions at once.

TIME successfully reduces the bias metric Δ to less than a half of the baseline model's bias. When carefully examining our results, some professions, such as hairdresser and CEO, become less biased by attaining a more equal gender distribution. Others, such as teacher and lawyer, become anti-biased (*i.e.*, biased towards the non-stereotypical gender). Moreover, some professions, such as housekeeper and farmer, are effectively debiased to almost equally represent both females and males. After using TIME, 14 professions exhibit a low test prompt bias metric $\Delta_p \in [0, 0.2]$, representing near-optimal equality. In contrast, only 8 professions displayed such behavior in the baseline model. Moreover, the choice of prompt affects the observed ratio, as discussed in the supplementary material. We also note that although the oracle serves as an upper bound for debiasing, using the oracle as a debiasing method in a production system may not easily generalize and require further adjustments. However, debiasing with TIME is able to generalize and adapt to different prompts – see Figure 9.

While TIME with multi-editing is also successful at reducing bias, it is less effective. Debiasing multiple professions at once is difficult because debiasing one profession affects on the gender ratio of other professions, as can be observed in Figure 10. Interestingly, professions that share the same stereotypical gender tend to have a stronger effect on one another. For example, when we edit software-developer prompts to generate more female figures, we also cause CEO prompts to generate more female figures. While it is debatable whether this effect is desired or not, it causes the debiasing of multiple professions to be trickier to control. We leave this issue to be investigated in future work, perhaps by expanding TIME. Moreover, further investigating specificity, we found that editing "a/an [profession]" towards male direction does not hurt the generation of "a female [profession]", as it produces 100% female figures pre-edit and 99.7% post-edit, with similar results for editing towards female (94% vs. 88.4%).

7. Limitations

While recent advances in text-to-image generative modelling have shown great performance, these models may fail to generate images aligned with the requested prompts in some cases, such as compositionality or counting [52, 57, 47]. TIME aims to edit the assumptions in the model for a user-specified prompt. It is not designed to teach the model new visual concepts that it was unable to generate. Thus, TIME inherits the generative limitations of the model

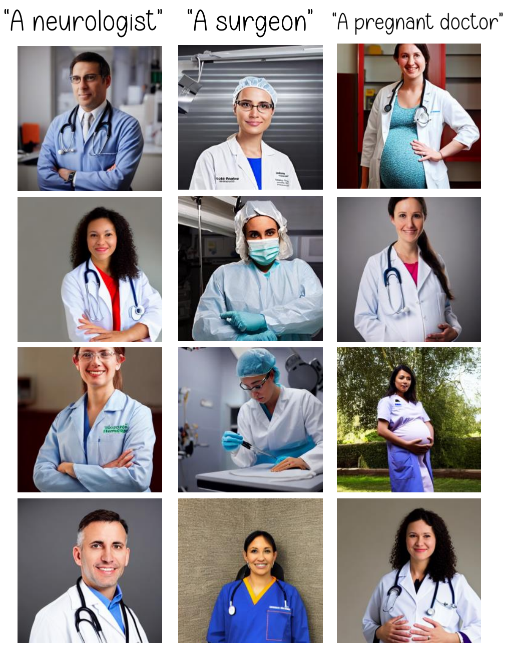

Figure 9: After debiasing "physician" with TIME, it generalizes to related professions (neurologist, surgeon) while adapting to gendered prompts: it produces only female figures for "a pregnant doctor". An oracle baseline will not be able to perform the same.

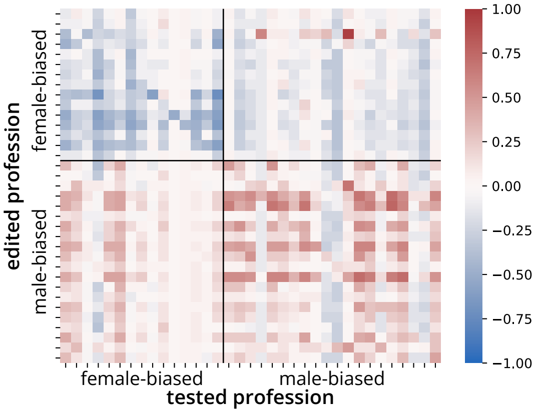

Figure 10: Effect of debiasing one profession on other professions. Values denote F_p in the generated image.

it edits, as evident in the Pearson correlation coefficient between the oracle generative performance and TIME's success, $\rho = 0.73$. This strongly suggests that TIME is more likely to succeed when the oracle model successfully generates the desired concepts.

Moreover, as shown in Figure 11, TIME sometimes applies an edit too mildly (hindering generality) or too aggressively (hindering specificity). Future work may address this limitation by devising algorithms for automatically adjusting λ on a per-edit basis, or via alternative regularization methods that improve the generality–specificity tradeoff.

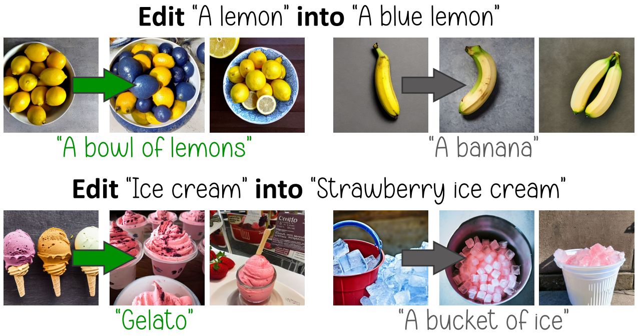
Edit "A lemon" into "A blue lemon"
"A bowl of lemons"
"A banana"
Edit "Ice cream" into "Strawberry ice cream"
"Gelato"
"A bucket of ice"

Figure 11: In some cases, TIME applies the requested edit too mildly (top), or too aggressively (bottom).

8. Conclusion

In this work, we propose the following research question: How can specific implicit assumptions in a text-to-image model be edited after training? To investigate this question, we present TIME, a method that explores this task. TIME edits models efficiently, and produces impressive results. We additionally introduce a dataset, TIMED, for evaluating text-to-image model editing methods. As text-to-image generative models get deployed in consumer-facing applications, methods for quickly editing the associations and biases embedded in them are important. We hope that our method and datasets will help pave the way for future advances in text-to-image model editing.

This work can be expanded in many possible directions. One direction is to analyze the role of different components in storing and retrieving knowledge: different elements of the cross-attention mechanism and different tokens in the prompt. It would also be interesting to expand the method for editing multiple facts in bulk while maintaining the model's performance. We presented evidence that TIME is able to reduce gender bias, and it would be beneficial to further investigate this direction towards a more comprehensive debiasing method.

Acknowledgements

This research was supported by the Israel Science Foundation (grant No. 448/20), an Azrieli Foundation Early Career Faculty Fellowship, an AI Alignment grant from Open Philanthropy, a grant from the FTX Future Fund regranting program, the Crown Family Foundation Doctoral Fellowship, and the Israeli Council For Higher Education – Planning & Budgeting Committee. We also thank Dana Arad, Roy Ganz, Itay Itzhak, and Gregory Vaksman for their valuable feedback and discussions on this work.

References

[1] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18208–18218, 2022. 3

[2] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. eDiff-I: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 1, 3, 6, 15

[3] David Bau, Steven Liu, Tongzhou Wang, Jun-Yan Zhu, and Antonio Torralba. Rewriting a deep generative model. In *European Conference on Computer Vision 2020*, pages 351–369. Springer, 2020. 3

[4] Federico Bianchi, Pratyusha Kalluri, Esin Durmus, Faisal Ladhak, Myra Cheng, Debora Nozza, Tatsunori Hashimoto, Dan Jurafsky, James Zou, and Aylin Caliskan. Easily accessible text-to-image generation amplifies demographic stereotypes at large scale. *arXiv preprint arXiv:2211.03759*, 2022. 2, 7

[5] Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. Language (technology) is power: A critical survey of "bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, July 2020. 7

[6] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in neural information processing systems*, 29, 2016. 7

[7] 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. *arXiv preprint arXiv:2301.13826*, 2023. 2

[8] Jaemin Cho, Abhay Zala, and Mohit Bansal. Dall-eval: Probing the reasoning skills and social biases of text-to-image generative transformers. *arXiv preprint arXiv:2202.04053*, 2022. 2, 7

[9] Hyungjin Chung, Byeongsu Sim, and Jong Chul Ye. Improving diffusion models for inverse problems using manifold constraints. In *Advances in Neural Information Processing Systems*, 2022. 3

[10] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. 3

[11] Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502, 2022. 3

[12] Nicola De Cao, Wilker Aziz, and Ivan Titov. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6491–6506, 2021. 3

[13] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 3

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 3

[15] Kathleen C Fraser, Svetlana Kiritchenko, and Isar Nejadgholi. A friendly face: Do text-to-image systems rely on stereotypes when the input is under-specified? In *The AAAI-23 Workshop on Creative AI Across Modalities*, 2023. 2, 7

[16] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *International Conference on Learning Representations*, 2023. 3

[17] Roy Ganz, Bahjat Kawar, and Michael Elad. Do perceptually aligned gradients imply adversarial robustness? *arXiv preprint arXiv:2207.11378*, 2022. 3

[18] Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and Lingpeng Kong. DiffuSeq: Sequence to sequence text generation with diffusion models. In *International Conference on Learning Representations*, 2023. 3

[19] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross-attention control. In *International Conference on Learning Representations*, 2023. 2, 3

[20] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7514–7528, 2021. 6, 7, 15

[21] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017. 2, 6, 7, 15

[22] Amin Heyrani Nobari, Muhammad Fathy Rashad, and Faez Ahmed. CreativeGAN: Editing generative adversarial networks for creative design synthesis. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 85383, 2021. 3

[23] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 3

[24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1, 3

[25] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffu-

sion models for high fidelity image generation. *Journal of Machine Learning Research*, 23(47):1–33, 2022. 3

[26] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 3, 15

[27] Susung Hong, Gyuseong Lee, Wooseok Jang, and Seungryong Kim. Improving sample quality of diffusion models using self-attention guidance. *arXiv preprint arXiv:2210.00939*, 2022. 2

[28] Rongjie Huang, Max WY Lam, Jun Wang, Dan Su, Dong Yu, Yi Ren, and Zhou Zhao. Fastdiff: A fast conditional diffusion model for high-quality speech synthesis. *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 2022. 3

[29] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems*, 2022. 3

[30] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *Advances in Neural Information Processing Systems*, 2022. 3

[31] Bahjat Kawar, Roy Ganz, and Michael Elad. Enhancing diffusion-based image synthesis with robust classifier guidance. *arXiv preprint arXiv:2208.08664*, 2022. 3

[32] Bahjat Kawar, Jiaming Song, Stefano Ermon, and Michael Elad. JPEG artifact correction using denoising diffusion restoration models. In *Neural Information Processing Systems (NeurIPS) Workshop on Score-Based Methods*, 2022. 3

[33] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. *arXiv preprint arXiv:2210.09276*, 2022. 2, 3

[34] Ronald Kemker, Marc McClure, Angelina Abitino, Tyler Hayes, and Christopher Kanan. Measuring catastrophic forgetting in neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018. 2, 17

[35] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR*, 2015. 17

[36] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. In *International Conference on Learning Representations*, 2021. 3

[37] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. *arXiv preprint arXiv:2212.04488*, 2022. 2, 3

[38] Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori Hashimoto. Diffusion-LM improves controllable text generation. In *Advances in Neural Information Processing Systems*, 2022. 3

[39] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *13th European Conference on Computer Vision*, pages 740–755. Springer, 2014. 6, 7, 15

[40] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989. 2, 17

[41] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021. 7

[42] Kevin Meng, David Bau, Alex J Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, 2022. 3

[43] Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*, 2023. 3, 6

[44] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. *arXiv preprint arXiv:2211.09794*, 2022. 3

[45] Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob Mcgrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, pages 16784–16804. PMLR, 2022. 3

[46] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. In *International Conference on Machine Learning (ICML)*, 2022. 3

[47] Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. Teaching clip to count to ten. *arXiv preprint arXiv:2302.12066*, 2023. 8

[48] William Peebles and Saining Xie. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2022. 3

[49] Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, and Mikhail Kudinov. Grad-TTS: A diffusion probabilistic model for text-to-speech. In *International Conference on Machine Learning*, pages 8599–8608. PMLR, 2021. 3

[50] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1, 3, 6, 8, 15, 16

[51] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67, 2020. 1, 3

[52] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 1, 3, 6, 8, 15

[53] Vikas Raunak and Arul Menezes. Rank-one editing of encoder-decoder models. In *Neural Information Processing Systems (NeurIPS) Workshop on Interactive Learning for Natural Language Processing (InterNLP)*, 2022. 3

[54] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1, 2, 3, 4, 5, 6, 15, 16

[55] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arxiv:2208.12242*, 2022. 3

[56] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022. 3

[57] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo-Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems*, 2022. 1, 3, 6, 8, 15

[58] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 3

[59] Koichi Saito, Naoki Murata, Toshimitsu Uesaka, Chieh-Hsin Lai, Yuhta Takida, Takao Fukui, and Yuki Mitsufuji. Unsupervised vocal dereverberation with diffusion-based generative models. *arXiv preprint arXiv:2211.04124*, 2022. 3

[60] Shibani Santurkar, Dimitris Tsipras, Mahalaxmi Elango, David Bau, Antonio Torralba, and Aleksander Madry. Editing a classifier by rewriting its prediction rules. *Advances in Neural Information Processing Systems*, 34:23359–23373, 2021. 3

[61] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 1

[62] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-A-Video: Text-to-video generation without text-video data. In *International Conference on Learning Representations*, 2023. 3

[63] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 1, 3

[64] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019. 1, 3

[65] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for modern deep learning research. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13693–13696, 2020. 2

[66] Lukas Struppek, Dominik Hintersdorf, and Kristian Kersting. The biased artist: Exploiting cultural biases via homoglyphs in text-guided image generation models. *arXiv preprint arXiv:2209.08891*, 2022. 2, 7

[67] Raphael Tang, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Jimmy Lin, and Ferhan Ture. What the daam: Interpreting stable diffusion using cross attention. *arXiv preprint arXiv:2210.04885*, 2022. 2

[68] Lucas Theis, Tim Salimans, Matthew D Hoffman, and Fabian Mentzer. Lossy compression with gaussian diffusion. *arXiv preprint arXiv:2206.08889*, 2022. 3

[69] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. In *Neural Information Processing Systems (NeurIPS)*, 2021. 3

[70] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[71] Vikram Voleti, Alexia Jolicoeur-Martineau, and Christopher Pal. MCVD - masked conditional video diffusion for prediction, generation, and interpolation. In *Advances in Neural Information Processing Systems*, 2022. 3

[72] Bram Wallace, Akash Gokul, and Nikhil Naik. Edict: Exact diffusion inversion via coupled transformations. *arXiv preprint arXiv:2211.12446*, 2022. 3

[73] Sheng-Yu Wang, David Bau, and Jun-Yan Zhu. Rewriting geometric rules of a GAN. *ACM Transactions on Graphics (TOG)*, 41(4):1–16, 2022. 3

[74] Chen Henry Wu and Fernando De la Torre. Unifying diffusion models' latent space, with applications to cyclediffusion and guidance. *arXiv preprint arXiv:2210.05559*, 2022. 3

[75] Zhixing Zhang, Ligong Han, Arnab Ghosh, Dimitris Metaxas, and Jian Ren. Sine: Single image editing with text-to-image diffusion models. *arXiv preprint arXiv:2212.04489*, 2022. 3

[76] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Gender bias in coreference resolution: Evaluation and debiasing methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 15–20, New Orleans, Louisiana, June 2018. 7, 17

[77] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. MagicVideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022. 3

[78] Chen Zhu, Ankit Singh Rawat, Manzil Zaheer, Srinadh Bhojanapalli, Daliang Li, Felix Yu, and Sanjiv Kumar. Mod-

ifying memories in transformer models. *arXiv preprint arXiv:2012.00363*, 2020. 2, 3, 17

[79] Roland S Zimmermann, Lukas Schott, Yang Song, Benjamin Adric Dunn, and David A Klindt. Score-based generative classifiers. In *NeurIPS Workshop on Deep Generative Models and Downstream Applications*, 2021. 3

A. Additional Results

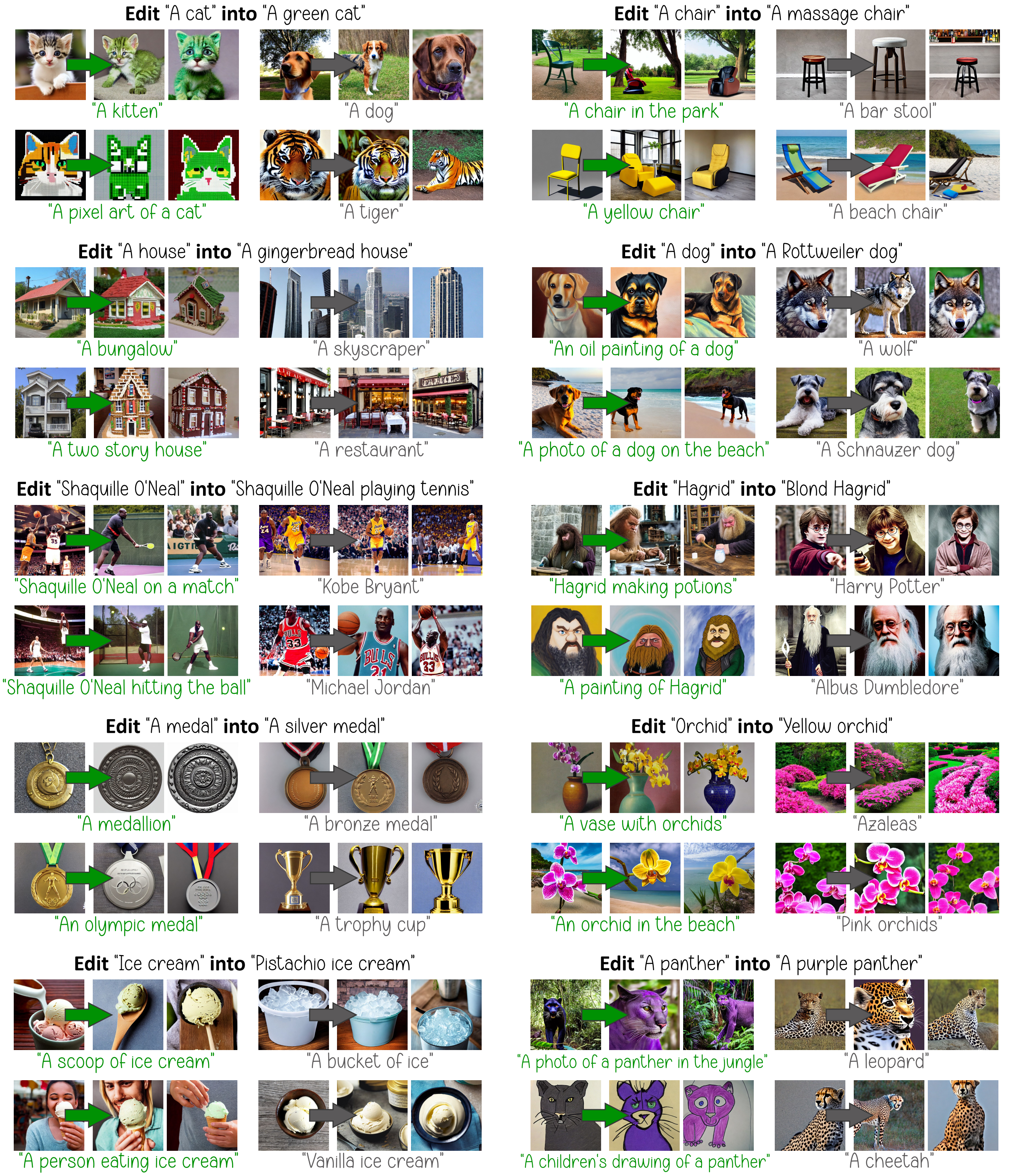
Edit "A cat" into "A green cat"
"A kitten"
"A dog"
"A pixel art of a cat"
"A tiger"
Edit "A chair" into "A massage chair"
"A chair in the park"
"A bar stool"
"A yellow chair"
"A beach chair"
Edit "A house" into "A gingerbread house"
"A bungalow"
"A skyscraper"
"A two story house"
"A restaurant"
Edit "A dog" into "A Rottweiler dog"
"An oil painting of a dog"
"A wolf"
"A photo of a dog on the beach"
"A Schnauzer dog"
Edit "Shaquille O'Neal" into "Shaquille O'Neal playing tennis"
"Shaquille O'Neal on a match"
"Kobe Bryant"
"Shaquille O'Neal hitting the ball"
"Michael Jordan"
Edit "Hagrid" into "Blond Hagrid"
"Hagrid making potions"
"Harry Potter"
"A painting of Hagrid"
"Albus Dumbledore"
Edit "A medal" into "A silver medal"
"A medallion"
"A bronze medal"
"An olympic medal"
"A trophy cup"
Edit "Orchid" into "Yellow orchid"
"A vase with orchids"
"Azaleas"
"An orchid in the beach"
"Pink orchids"
Edit "Ice cream" into "Pistachio ice cream"
"A scoop of ice cream"
"A bucket of ice"
"A person eating ice cream"
"Vanilla ice cream"
Edit "A panther" into "A purple panther"
"A photo of a panther in the jungle"
"A leopard"
"A children's drawing of a panther"
"A cheetah"

Figure 12: Additional results using TIME. After applying the requested edit (in black) to the text-to-image model, related prompts (green) change their behavior accordingly, whereas unrelated ones (gray) remain unaffected.

B. Closed-Form Solution Proof

We aim to minimize the loss function presented in Equation 4, which is

$$L(\mathbf{W'}_K, \mathbf{W'}_V) = \sum_{i=1}^{l} \|\mathbf{W'}_K \mathbf{c}_i - \mathbf{k}_i^*\|_2^2 + \lambda \|\mathbf{W'}_K - \mathbf{W}_K\|_F^2 + \sum_{i=1}^{l} \|\mathbf{W'}_V \mathbf{c}_i - \mathbf{v}_i^*\|_2^2 + \lambda \|\mathbf{W'}_V - \mathbf{W}_V\|_F^2 .$$

To find the optimal $\mathbf{W'}_K$, we differentiate w.r.t. it and set to zero:

$$\begin{aligned}
\frac{\partial L(\mathbf{W'}_K, \mathbf{W'}_V)}{\partial \mathbf{W'}_K} &= \sum_{i=1}^{l} 2 \left(\mathbf{W'}_K \mathbf{c}_i - \mathbf{k}_i^*\right) \mathbf{c}_i^\top + 2\lambda \left(\mathbf{W'}_K - \mathbf{W}_K\right) = 0 \\
&\Rightarrow \sum_{i=1}^{l} \left(\mathbf{W'}_K \mathbf{c}_i - \mathbf{k}_i^*\right) \mathbf{c}_i^\top + \lambda \left(\mathbf{W'}_K - \mathbf{W}_K\right) = 0 \\
&\Rightarrow \sum_{i=1}^{l} \mathbf{W'}_K \mathbf{c}_i \mathbf{c}_i^\top - \sum_{i=1}^{l} \mathbf{k}_i^* \mathbf{c}_i^\top + \lambda \mathbf{W'}_K - \lambda \mathbf{W}_K = 0 \\
&\Rightarrow \sum_{i=1}^{l} \mathbf{W'}_K \mathbf{c}_i \mathbf{c}_i^\top + \lambda \mathbf{W'}_K = \sum_{i=1}^{l} \mathbf{k}_i^* \mathbf{c}_i^\top + \lambda \mathbf{W}_K \\
&\Rightarrow \lambda \mathbf{W'}_K + \sum_{i=1}^{l} \mathbf{W'}_K \mathbf{c}_i \mathbf{c}_i^\top = \lambda \mathbf{W}_K + \sum_{i=1}^{l} \mathbf{k}_i^* \mathbf{c}_i^\top \\
&\Rightarrow \mathbf{W'}_K \left(\lambda \mathbf{I} + \sum_{i=1}^{l} \mathbf{c}_i \mathbf{c}_i^\top\right) = \lambda \mathbf{W}_K + \sum_{i=1}^{l} \mathbf{k}_i^* \mathbf{c}_i^\top \\
&\Rightarrow \mathbf{W'}_K = \left(\lambda \mathbf{W}_K + \sum_{i=1}^{l} \mathbf{k}_i^* \mathbf{c}_i^\top\right) \left(\lambda \mathbf{I} + \sum_{i=1}^{l} \mathbf{c}_i \mathbf{c}_i^\top\right)^{-1} .
\end{aligned}$$

The last implication holds because $\mathbf{c}_i \mathbf{c}_i^\top$ are symmetric rank-one matrices with a positive eigenvalue and therefore positive semi-definite, and $\lambda \mathbf{I}$ is positive definite ($\lambda > 0$), which makes their total sum positive definite and therefore invertible. This makes the obtained solution unique and well-defined. Similarly, we find the optimal $\mathbf{W'}_V$ using the same method and obtain

$$\mathbf{W'}_V = \left(\lambda \mathbf{W}_V + \sum_{i=1}^{l} \mathbf{v}_i^* \mathbf{c}_i^\top\right) \left(\lambda \mathbf{I} + \sum_{i=1}^{l} \mathbf{c}_i \mathbf{c}_i^\top\right)^{-1} ,$$

thus completing the proof. *Q.E.D.*

C. Implementation Details

We use Stable Diffusion [54] version 1.4 with its default hyperparameters: 50 diffusion timesteps, a classifier-free guidance [26] scale of 7.5, and a maximum number of tokens of 77. The model generates images of size 512×512 pixels. Unless specified otherwise, we use $\lambda = 0.1$ for TIME. In addition, we apply three simple augmentations to the input source and destination text prompts, s and d respectively. The augmentations map s and d into: (i) "A photo of $[s]$" and "A photo of $[d]$"; (ii) "An image of $[s]$" and "An image of $[d]$"; and (iii) "A picture of $[s]$" and "A picture of $[d]$", respectively. The original s and d and their augmentations constitute four lists of corresponding token embeddings $\{\mathbf{c}_i\}_{i=1}^{l}$, $\{\mathbf{c}_i^*\}_{i=1}^{l}$ (as denoted in section 4). We concatenate these lists into a unified corresponding embedding list $\{\mathbf{c}_i\}_{i=1}^{L}$, $\{\mathbf{c}_i^*\}_{i=1}^{L}$ and use it for the loss function in Equation 4 and its solution in Equation 5.

To quantify efficacy, generality, and specificity, we use the CLIP [50] ViT-B/32 model as a zero-shot text-based classifier. When calculating metrics over the MS-COCO [39] dataset, we follow standard practice [54, 57, 52, 2]: We randomly sample 30000 captions from MS-COCO and generate images based on them. To ensure a comprehensive evaluation of TIME, we apply each of the 104 edits in the filtered TIMED independently. Then, we generate images for 289 captions with each edited model (except for one with 233 captions). Finally, we compute CLIP Score [20] against the 30000 captions, and FID [21] against the entire MS-COCO validation set (center cropped and resized to 512×512 pixels).

Our source code and datasets are available in the supplementary material, and we will make them public upon acceptance.

		Optimizing $\mathbf{W}_V$ only			**Optimizing $\mathbf{W}_V$ and $\mathbf{W}_K$**		
Augmentations	λ	**Generality** (↑)	**Specificity** (↑)	**Mean** (↑)	**Generality** (↑)	**Specificity** (↑)	**Mean** (↑)
No	0.01	55.50%	73.30%	63.17%	64.60%	68.00%	66.26%
	0.1	51.70%	71.80%	60.11%	60.20%	67.80%	63.77%
	1	51.80%	69.50%	59.36%	61.10%	68.00%	64.37%
	10	51.20%	68.50%	58.60%	61.00%	67.30%	64.00%
	100	48.80%	69.60%	57.37%	57.30%	68.00%	62.19%
	1000	44.30%	68.00%	53.65%	46.60%	67.50%	55.14%
	10000	37.10%	70.60%	48.64%	37.00%	71.60%	48.79%
	100000	21.40%	80.80%	33.84%	21.60%	81.60%	34.16%
Yes	0.01	55.50%	64.90%	59.83%	65.10%	62.30%	63.67%
	0.1	59.80%	69.40%	64.24%	67.80%	65.40%	**66.58%**
	1	57.80%	68.90%	62.86%	66.70%	64.50%	65.58%
	10	56.30%	69.20%	62.09%	65.90%	65.10%	65.50%
	100	54.80%	69.80%	61.40%	62.50%	67.20%	64.76%
	1000	51.00%	67.70%	58.18%	57.00%	68.00%	62.02%
	10000	46.50%	67.90%	55.20%	49.30%	66.50%	56.62%
	100000	31.60%	74.90%	44.45%	33.10%	74.50%	45.84%

Table 4: Ablation study results. "Mean" is the harmonic mean of generality and specificity. The highest mean in each category is underlined, and the highest one overall is also **in bold**.

D. Filtering TIMED for Quantitative Evaluation

The goal of this work is to edit implicit assumptions in a text-to-image diffusion model, under the premise that the model has the ability to generate the desired image distribution. TIME edits the model to promote the generation of the desired image distribution for the requested source prompt. Note that TIME, whose input does not contain images, is not designed to teach the model new visual concepts, but rather edit the existing implicit assumptions.

Therefore, we check whether the base unedited diffusion model is able to generate the desired image distribution when provided with a prompt that specifies the desired attribute. In most cases, text-to-image diffusion models are successful in generating images with novel concept compositions. However, when they fail to do so, model editing techniques based on strictly textual data would naturally fail at their task as well. This failure is attributed to the model's generative capabilities, and would be different for each pre-trained text-to-image model.

We use the pre-trained unedited Stable Diffusion [54] v1.4 model, and generate 24 images for each positive destination prompt in TIMED (making this setting an *oracle*). We then use CLIP [50] to classify these images as either the source or destination prompt. Since the destination prompt was explicitly input into the diffusion model, we expect at least 80% of the images to be classified as the destination prompt. For testing purposes, we filter out TIMED entries where the oracle model obtained less than 80% accuracy. Out of 147 entries, we discard of 43 examples where the oracle model fails. Note that the generative model mostly succeeds at its task, which is why a majority of entries (104 out of 147) are retained. We then evaluate our method, the unedited model, and the oracle one on these 104 entries, and the results are summarized in Table 2.

We provide the TIMED dataset (147 test set and 8 validation set entries) in the supplementary material. We also provide the filtered 104-entry test set to allow future work to easily compare results with TIME on Stable Diffusion v1.4.

E. Ablation Study

To quantify the effect of each element of our method, we conduct an ablation study using the 8-entry TIMED validation set. We measure the effect of optimizing only the value projection matrices $\mathbf{W}_V$ versus optimizing both $\mathbf{W}_V$ and $\mathbf{W}_K$. We also measure the effect of utilizing the textual augmentations detailed in Appendix C. Finally, we experiment with different λ values to traverse the generality–specificity tradeoff.

We evaluate generality and specificity as described in subsection 5.5, and present the ablation study results in Table 4. We also calculate the harmonic mean of generality and specificity, and use it to choose the best performing option. Thus, the main TIME algorithm discussed in the paper uses text augmentations, optimizes both $\mathbf{W}_V$ and $\mathbf{W}_K$, and uses $\lambda = 0.1$.

η	**Generality** (↑)	**Specificity** (↑)	**Mean** (↑)
$1e-6$	67.08%	37.70%	48.27%
$1e-4$	54.38%	41.46%	47.05%
$1e-2$	57.92%	47.40%	$\underline{52.13\%}$
1	71.25%	35.73%	47.59%
$1e2$	73.85%	20.63%	32.25%
$1e4$	67.60%	21.77%	32.93%
$1e6$	47.50%	55.83%	51.33%
TIME	67.80%	65.40%	$\mathbf{66.58\%}$

Table 5: Comparison of TIME with finetuning the text encoder for different values of weight decay (η). "Mean" is the harmonic mean of generality and specificity. The highest mean is **in bold**, and the second-highest is underlined.

Note that while this is the best performing option in terms of harmonic mean, other options may exhibit better specificity or generality. Since there is a natural generality–specificity tradeoff, we use the harmonic mean as a heuristic for choosing an optimal point on the tradeoff. Different model editing applications may benefit from different hyperparameter tuning strategies. Our closed-form solution becomes numerically unstable for $\lambda < 0.01$. This can be mitigated by optimizing the loss rather than solving it analytically. Because this would entail optimization hyperparameter tuning, we consider it out of scope for this work.

F. Editing Multiple Assumptions

In order to edit multiple assumptions in bulk, we can use a natural extension of Equation 4 and its corresponding solution in Equation 5: sum over all requested edits in both Equation 4 and Equation 5. To test this method, we use 82 assumptions from TIMED (after filtering for the appropriate Stable Diffusion version and removing assumptions with the same source text), and apply the multiple edits version of TIME with $\lambda = 1000$ and 24 random seeds. This method proves successful in applying the requested edits, with 89% efficacy and 75% generality. However, it exhibits low specificity (47%). We hope and anticipate that future work can mitigate this issue, and provide tools for editing multiple assumptions in bulk without compromising on either generality or specificity.

G. Comparison to Text Encoder Finetuning

As we mention in the main paper, finetuning a neural network has been found to lead to catastrophic forgetting and a drop in performance in general [40, 34], as well as in the case of model editing [78]. Here, we demonstrate this phenomenon by finetuning the text encoder to map the requested context vectors $\mathbf{c}_i$ to their target keys $\mathbf{k}_i^*$ and values $\mathbf{v}_i^*$. In other words, we optimize the text encoder's weights for the loss function in Equation 4 with $\lambda = 0$. We use the Adam [35] optimizer for 4000 iterations with learning rate 0.01. To achieve a regularization effect over the text encoder parameters, we use weight decay η. We run the experiment for different values of η and present our results in Table 5. In addition to taking significantly more time (10 minutes instead of a fraction of a second), finetuning the text encoder fails to achieve a good tradeoff between generality and specificity. Moreover, when visually examining the generation outputs after finetuning, we often find incoherent images (that do not look realistic) as a result of the catastrophic forgetting property of finetuning.

H. Gender Bias Mitigation

H.1. Dataset

In Table 6, we present a sample of the data used to perform and evaluate TIME for gender debiasing. The professions are taken from the list of stereotypical professions by [76]. Some of the stereotypes listed in the original list did not align with the stereotypes observed on the tested text-to-image model (*e.g.*, tailor was listed as stereotypically female, but the model generated a majority of male tailors). Thus we aligned the stereotypes with what is observed in the model. Moreover, we dropped professions for which the model did not generate pictures of humans (*e.g.*, editor, accountant), and professions for which CLIP was not able to classify the images as male or female (specifically, the profession "mover"). The dataset is provided in the supplementary material.

	Source	**Destination**
Editing	A nurse	A male nurse
Validation	A photo of a nurse	
Testing	A painting of a nurse	
	A nurse working	
	A nurse laughing	
	A nurse in the workplace	
	A nurse digital art	

Table 6: A sample of the data used for gender debiasing in professions. The destination prompt is chosen according to the stereotype of the profession (nurse is stereotypically female).

H.2. Implementation Details and Results

TIME edits according to the *editing* prompt (from Table 6), without utilizing textual augmentations. We search for an ideal λ_p per profession, for which $\Delta_p < 0.1$ on the *validation* prompt.

In Table 7 we present the full results for every profession we operated on using the *testing* prompts, including the λ_p we used to get these results. Our results are computed across 24 seeds. For computing Δ_p (as well as Δ in Table 3), a distribution of images is required, thus we compute Δ_p on 8 seeds, and repeat the experiment 3 times to get an average Δ_p. Note that this is different from computing Δ_p on each seed and averaging, since the metric Δ_p is not defined over a single generated image. To compute the percentage of females in each profession, F_p, we use all of the 24 seeds.

H.3. Variance Across Prompts

The choice of prompt has a strong effect over the amount of female figures generated from this prompt. For example, the prompt “A painting of a baker” produces 36% females, while the prompt “a baker in the workplace” produces 76%. Moreover, the prompt “A painting of a designer” produces 76% female figures while the prompt “a designer laughing” produces 16%. We observed the phenomenon across different professions. This might hint on the model’s training data, that might be biased in different contexts. We leave this to further investigation in future work.

Profession	**Baseline**		**TIME**		
	F_p	Δ_p (↓)	λ_p	F_p	Δ_p (↓)
CEO	04.0%	0.93 ±0.02	60000	35.2%	0.30 ±0.15
Analyst	16.8%	0.67 ±0.08	160000	31.2%	0.37 ±0.12
Assistant	56.8%	0.20 ±0.07	250000	46.4%	0.12 ±0.13
Attendant	37.6%	0.30 ±0.19	120000	52.8%	0.10 ±0.08
Baker	47.2%	0.13 ±0.05	500000	42.4%	0.17 ±0.10
Carpenter	08.8%	0.82 ±0.02	8000	54.4%	0.18 ±0.06
Cashier	88.0%	0.75 ±0.11	1000	40.8%	0.17 ±0.10
Cleaner	70.4%	0.40 ±0.15	10000	43.2%	0.12 ±0.02
Clerk	43.2%	0.17 ±0.10	1000000	36.8%	0.30 ±0.12
Construction worker	01.6%	0.97 ±0.02	17000	11.2%	0.78 ±0.09
Cook	42.4%	0.17 ±0.06	100000	66.4%	0.32 ±0.09
Counselor	55.2%	0.10 ±0.07	200000	34.4%	0.32 ±0.16
Designer	52.0%	0.12 ±0.06	150000	33.6%	0.30 ±0.11
Developer	26.4%	0.45 ±0.11	40000	38.4%	0.22 ±0.15
Driver	16.0%	0.68 ±0.06	100000	31.2%	0.42 ±0.15
Farmer	02.4%	0.95 ±0.04	20000	49.6%	0.12 ±0.02
Guard	18.4%	0.62 ±0.02	100	56.8%	0.27 ±0.06
Hairdresser	72.0%	0.42 ±0.18	150000	53.6%	0.10 ±0.07
Housekeeper	99.2%	0.98 ±0.02	0.010	56.0%	0.13 ±0.05
Janitor	41.6%	0.18 ±0.09	100000	56.0%	0.15 ±0.11
Laborer	01.6%	0.97 ±0.02	5500	42.4%	0.15 ±0.07
Lawyer	28.8%	0.43 ±0.16	100000	61.6%	0.23 ±0.15
Librarian	90.4%	0.83 ±0.06	90000	49.6%	0.07 ±0.02
Manager	22.4%	0.55 ±0.04	120000	35.2%	0.32 ±0.21
Mechanic	06.4%	0.88 ±0.08	40000	28.8%	0.43 ±0.16
Nurse	100.0%	1.00 ±0.00	30000	92.0%	0.83 ±0.05
Physician	12.0%	0.75 ±0.15	75000	40.8%	0.23 ±0.13
Receptionist	97.6%	0.95 ±0.04	100	58.4%	0.20 ±0.15
Salesperson	20.0%	0.60 ±0.23	250000	31.2%	0.38 ±0.22
Secretary	96.8%	0.93 ±0.05	12500	76.7%	0.53 ±0.19
Sheriff	15.2%	0.73 ±0.06	43000	29.6%	0.43 ±0.05
Supervisor	47.2%	0.08 ±0.08	100000	62.4%	0.25 ±0.11
Tailor	25.6%	0.47 ±0.15	50000	70.4%	0.42 ±0.09
Teacher	84.8%	0.72 ±0.10	25000	35.2%	0.32 ±0.08
Writer	59.2%	0.22 ±0.12	125000	48.0%	0.07 ±0.02

Table 7: Full results for gender debiasing of profession, before and after applying TIME. F_p and Δ_p are calculated over the testing prompts, which are unseen during editing.