Concept Algebra for Text-Controlled Vision Models

Zihao Wang[1], Lin Gui[1], Jeffrey Negrea[2], and Victor Veitch[1,2,3]

[1]*Department of Statistics, University of Chicago*
[2]*Data Science Institute, University of Chicago*
[3]*Google Research*

Abstract

This paper concerns the control of text-guided generative models, where a user provides a natural language prompt and the model generates samples based on this input. Prompting is intuitive, general, and flexible. However, there are significant limitations: prompting can fail in surprising ways, and it is often unclear how to find a prompt that will elicit some desired target behavior. A core difficulty for developing methods to overcome these issues is that failures are know-it-when-you-see-it—it's hard to fix bugs if you can't state precisely what the model should have done! In this paper, we introduce a formalization of "what the user intended" in terms of latent concepts implicit to the data generating process that the model was trained on. This formalization allows us to identify some fundamental limitations of prompting. We then use the formalism to develop *concept algebra* to overcome these limitations. Concept algebra is a way of directly manipulating the concepts expressed in the output through algebraic operations on a suitably defined representation of input prompts. We give examples using concept algebra to overcome limitations of prompting, including concept transfer through arithmetic, and concept nullification through projection. Code available at https://github.com/zihao12/concept-algebra.

1 Introduction

Large-scale text-controlled generative models are quickly becoming dominant in many parts of modern machine learning and artificial intelligence; e.g., they form the basis for state-of-the-art approaches in natural language understanding and vision [Bro+20; Rad+21; Bom+21; Koj+22]. The basic paradigm for such models is that a user provides a prompt in natural language (e.g., English) and the model generates samples based on this prompt—e.g., in large language models the sample is a natural language response, and in text-to-vision the sample is an image. This natural language interface is powerful, allowing users to make use of the models in a wide variety of applications. However, prompting also has significant limitations. For example, apparently small changes in prompts can have large effects on outputs, the precise effect of any given prompt is unclear, and there is no practical way to find a prompt that optimally elicits a desired response. The present work studies situations where prompting is inadequate for model control, and how to control models in these situations. Our focus is text-to-image models, though the core ideas are general.

Figure 1a shows an example where we may be unsatisfied with prompting. We wish to generate a picture of a frog playing a piano, so we try the prompt "A frog playing the piano, anthropomorphic".[1] What we observe—surprisingly—is that the outputs are mostly cartoons! Then, we respond to this by trying a refined prompt "a frog playing the piano, anthropomorphic, photorealistic"; see Figure 1b. This also doesn't work! The generated

[1] All examples are generated with Stable Diffusion [Rom+22].

(a) Prompting "a frog playing the piano, anthropomorphic" mainly generates cartoons.

(b) Prompting "a frog playing the piano, anthropomorphic, photorealistic" still doesn't generate the intended content!

(c) The prompt "a nurse sitting in a white room" almost always generates women

Figure 1: Examples of Prompting Failures

image is mainly either a cartoon of a frog playing the piano, or a photo of a frog sitting on a piano, not playing it.

As a second example, Figure 1c shows outputs sampled from the prompt "a nurse sitting in a white room." In this case, the model almost always generates images of women, even though the nurse's sex was not specified in the prompt. Moreover, it is not clear what (if any) prompt would have elicited a sex-balanced collection of nurse images.[2]

In both examples, the effects of a given prompt are not obvious *a priori* to the users, and it is not clear how to write a prompt that actually elicits the target the user has in mind.

Our aim here is to understand why such issues occur, and what can be done to address them. A key challenge is that it is not obvious what the generative model *should* have done— in what sense is outputting cartoon frogs or all female nurses actually a bug? One perspective on this issue is that there is a disconnect between what the user has in mind and what they actually manage to elicit. Accordingly, we need some way of specifying "what the user has in mind" that let us reason about the difference between this and what the model actually did. Informally, the view we take here is that the user imagines a set of *concepts* and attempts to control the model so that its outputs express these concepts. In Section 2 we will develop a formalization of this idea where the user's true intention is specified as a probability distribution over latent concepts. So, for example, if the user intended to generate a nurse with unspecified sex, then this would correspond to a point-mass distribution on a `nurse` value of the `profession` concept and a uniform distribution over {`male`, `female`} for the `sex` concept. In this view, a user first imagines a distribution over concepts they'd like to see expressed in the output, and then writes a prompt to try to elicit this distribution. This perspective will let us uncover some cases where it is difficult or impossible for a user to elicit their desired behavior by direct prompting—including the examples depicted in fig. 1.

This leads us to ask how a user could directly specify their intended concepts, without relying on direct prompting. In Section 3 we will develop a *concept algebra* as a way of manipulating the concepts expressed by the model through arithmetic operations on a suitable representation space. The inspiration here comes from an observed property of word embedding methods due to Mikolov et al. [MYZ13], who find that arithmetic on certain word embeddings seems to map on to changes in semantic meaning (specifically *analogies*)—

[2]For notational ease, we take sex to be a binary variable. This is not an ideological position. We acknowledge that sex and gender are distinct, and that intersex exists. We sometimes use gendered terms ("woman" and "man") in prompts to improve image generation results. The association of nursing with the "female" sex and the "woman" gender exists in many cultures, both in perception and in actual representation in the workforce [Mao+21].

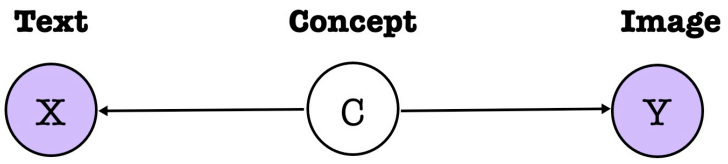


Figure 2: In the real-world data generating process, the text X and output Y are separated by latent concept variables C

e.g., $\gamma(\text{“king”}) - \gamma(\text{“man”}) + \gamma(\text{“woman”}) \approx \gamma(\text{“queen”})$. Our aim is to generalize this kind of algebraic concept manipulation to general text-controlled generative models. We'll see that the concept distribution view leads us naturally to a choice of representation and lets us establish conditions under which such algebraic operations correspond to semantic operations. In Section 4 we extend this reasoning to show how to "project out" unwanted concepts.

2 A Latent Concept View

Our goal is to understand how a model can be controlled to match the intention of its user. To make progress, we must first give a formalization of what the user's intention is—otherwise, it is impossible to say whether or not a given attempt at control succeeds. The view we take here is that the user imagines a set of concepts and attempts to control the model so that its outputs express these concepts. In this section, we formalize this idea, and use this formalization to understand the limitations of prompting.

Causal Structure Our aim is to formalize user intention in terms of abstract concepts. That begs the question: what is a concept, and why should the human user's imagination of a concept have anything to do with how the model operates?

We suppose that the real-world process that generated the training data has the following structure. First, images Y are generated according to some real-world, physical process. Then, some human looks at each image and writes a caption describing it. To write the caption, the human first maps the image to a set of high-level concepts summarizing the image's content. Then, the human uses these latent variables to generate the text X.

The causal structure of this process is depicted in fig. 2. Here, the random variable C captures all possible high-level concepts that are jointly relevant for image generation and caption writing. Variables in C[3] include attributes such as `has a woman` or `has a nurse`, or more abstract concepts such as `is a landscape` or `is a portrait`. These are random variables with law determined by the real-world process that generates the training data. Note these concept variables need not be independent—e.g., the variables `has a nurse` and `has a woman` may be correlated. We use P to denote the probability distribution on the sample space that the random elements (X, C, Y) are defined on, and we assume that X and C are discrete. We also assume all relevant marginal and conditional distributions for Y have differentiable densities.

Now, the full set of all possible concepts values is unwieldy. Usually, in any given instance, we only care about a small subset of concepts. Accordingly, we need language to break C into more manageable chunks.

Definition 2.1. A *concept variable* Z is a C-measurable random variable. The *concept* $\mathscr{Z}$ associated to Z is the sample space of Z. A set of concepts $\mathscr{Z}_1, \ldots, \mathscr{Z}_k$ is *complete for a caption* $x \in \mathscr{X}$ if the conditional density/probability mass function $p(c \mid x, Z_{1:k} = z_{1:k}) = p(c \mid Z_{1:k} = z_{1:k})$ for all $z_1, \ldots, z_k \in \mathscr{Z}_1 \times \cdots \times \mathscr{Z}_k$.

[3] formally, variables that are C-measurable.

In other words, we define specific concepts by coarsening the set of all possible concepts. The coarsening we choose will depend on the task at hand. For example, we may wish to coarsen to a `profession` concept and a `sex` concept. In general, each caption $x \in \mathscr{X}$ defines a distribution over all possible concept values. However, often there is a coarsening of the concepts that captures everything relevant for a given caption. For example, it seems reasonable that the concept of `profession` would be complete for the caption "A nurse". That is, this caption induces a distribution on other concepts (e.g., $\mathscr{W} = \{\texttt{male}, \texttt{female}\}$) only through the distribution it induces on the complete concept $\mathscr{Z} = \{\texttt{nurse}, \texttt{doctor}, \texttt{teacher}, \texttt{student}, \dots\}$.

Formalizing Model Control A text-controlled generative model takes in prompt text x and produces a random output Y. Implicitly, such models are maps from text strings to the space of probability distributions over Y ; i.e., $x \to f_{\lambda(x)}(\cdot)$, where $f_{\lambda(x)}(\cdot)$ is the (implicit) density that the model samples from. Here, $\lambda(x)$ is some representation of x—typically, the output of a neural network—that is passed into a fixed procedure for drawing Y. For instance, in diffusion vision models, $\lambda(x)$ might be the output of a language model, which is then passed into an iterative de-noising procedure applied a randomly sampled white noise starting image; in that case $f_{\lambda(x)}(\cdot)$ is the sampling distribution of this procedure. The objective used for training such models aims to learn a procedure such that the model's sampling distribution mimic's the real-world conditional distribution. In this paper, we will assume that this learning succeeds, and the model recovers the conditional distribution perfectly (i.e., there are no finite sample issues).

Intuitively, when controlling a generative model, the user has in mind certain concept values they want to appear in the sample—e.g., `nurse` or `photorealistic style`. When writing a prompt, the user's goal is to evoke these target concepts, then have the model generate Y consistent with the evoked concepts.

The following result connects concepts and prompts:

Proposition 2.2. *If the generative model succeeds in learning the data generating distribution ($f_{\lambda(x)}(y) = p(y \mid x)$) then for any prompt x such that concepts $\mathscr{Z}_1, \dots, \mathscr{Z}_k$ are complete for x, and $P(X = x) > 0$, we have:*

$$f_{\lambda(x)}(y) = \sum_{z_{1:k}} p(y \mid z_{1:k}) q_x(z_{1:k}), \tag{1}$$

where $q_x(z_{1:k}) = p(z_{1:k} \mid x)$ is the conditional probability mass function (PMF) of the concepts given prompt x.

There are two main points here. First, implicitly, each prompt x induces a distribution over concepts. This is done by inverting part of the real-world data generating process. In the training data, a human started with concept values z, w and then sampled captions from $p(x \mid z, w)$. At generation time, we write a prompt x and then sample concepts according to the posterior $p(z, w \mid x)$. The second point is that the sampling distribution of the model is ultimately determined by the real-world relationship between concepts and images: $p(y \mid z, w)$.

(We consider pairs of concepts because multiple concepts are useful for the subsequent development.)

Theorem 2.2 leads to a natural generalization of prompting by considering general distributions over latent concepts, including those not necessarily induced by any prompt.

Definition 2.3. A *concept distribution* Q is a probability distribution over concepts.

Then, each concept distribution (say, on $\mathscr{Z}_1 \times \cdots \times \mathscr{Z}_k$) defines a sampling distribution over Y according to:

$$p[Q](y) \coloneqq \sum_{z_{1:k}} p(y \mid Z_{1:k} = z_{1:k}) q(z_{1:k}), \tag{2}$$

where $p(y \mid z, w)$ is the density of the real-world conditional sampling distribution and q is the PMF corresponding to the probability distribution Q.

Now, we can formalize the generalized goal of model control as follows. First, the user specifies the concepts they want reflected in the output by specifying a target concept distribution Q. Then, they look for a setting of the model parameters, say λ_Q, such that the model's sampling distribution matches the sampling distribution induced by Q; i.e., $f_{\lambda_Q}(y) = p[Q](y)$. In the particular case of prompting, this second step is done by trying to find a prompt x such that $P(Z_{1:k} = z_{1:k} \mid X = x) = Q(z_{1:k})$, whence also $f_{\lambda(x)}(y) = p[Q](y)$.

This formalization may seem somewhat obscure. In particular, abstracting "target concept values the user wants to elicit" as a concept distribution is not obvious. Two examples help clarify why this is reasonable.

Examples First, consider the task of sampling from the distribution of pictures of male nurses. The concepts of interest are $\texttt{profession}$ $\mathscr{W}$ and $\texttt{sex}$ $\mathscr{Z}$. To fix the concepts to definite values we specify the density of the concept distribution as a product of delta functions; $q(z, w) = \delta_{\texttt{nurse}}(w)\delta_{\texttt{male}}(z)$. Then, we want samples from the model to mimic real pictures that are generated with these concept values—i.e., we want the model to mimic $p[Q](y) = p(y \mid W = \texttt{nurse}, Z = \texttt{male})$. This example reflects a common case where we want to set specific concept *values* (with no uncertainity) that will be expressed in the output.

Second, consider sampling from the distribution of pictures of nurses, with no sex bias in the outputs (i.e., a roughly equal proportion of male and female across the samples). Here, the target concepts setting is formalized as the concept distribution with the density $q(z, w) = \delta_{\texttt{nurse}}(w)p_Z(z)$, where p_Z is the PMF of the distribution of $\texttt{sex}$ in the real-world data (*not* conditioned on the $\texttt{profession}$ being $\texttt{nurse}$). In this case, formalizing the target behavior for the model requires a non-degenerate concept distribution.

Limitations of Prompting Each prompt x induces a concept distribution $Q_x(z_{1:k}) = P(Z_{1:k} = z_{1:k} \mid X = x)$. However, it's not clear in general how to find a prompt that induces a given concept distribution. For instance, it is not clear what—if any—prompt might induce the concept distribution with the density $q(z, w) = \delta_{\texttt{nurse}}(w)p_Z(z)$.

With the concept distribution formalization, we can identify two types of limitations of prompting. These are cases where we'd like to elicit a particular concept distribution Q but are unable to find a prompt x^* such that $Q = Q_{x^*}$.

i. Overlap Consider eliciting $q(z, w) = \delta_{z^*}(z)\delta_{w^*}(w)$; for e.g., $z^* = \texttt{photorealistic}$ and $w^* = \texttt{a frog playing the piano}$. If the PMF for the real-world data $p(z^*, w^*) = 0$, then there is no prompt x with $p(x) > 0$ that satisfies $p(z^*, w^* \mid x) > 0$. That is, the fact that there are no photorealistic images of a frog playing the piano means that there need not exist any prompt that evokes both the photorealistic and frog-playing-piano concepts simultaneously. Accordingly, evoking such concepts relies on the ability of the language model to extrapolate outside the support of the training data. Such extrapolation can happen; text-to-image models do often succeed at composing concepts [Rom+22; Ram+22; Sah+22]. However, extrapolation is not guaranteed and can fail in non-obvious ways, as with the frog playing the piano example.

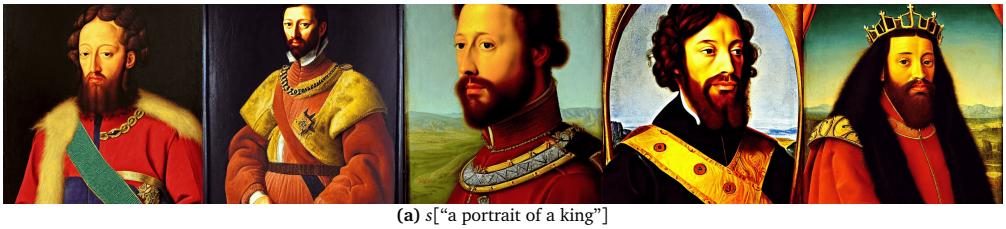
ii. Confounding Consider eliciting $q(z, w) = \delta_{w^*}(w)p_Z(z)$; for e.g., $w^* = \texttt{nurse}$, and p_Z uniform over $\texttt{male}$ or $\texttt{female}$. If Z and W are not independent in the training data, any prompt x such that $\mathscr{W}$ is complete for x will also change the distribution of Z, in the sense

that:

$$p(z \mid x) = \sum_w p(z \mid w, x) p(w \mid x) \neq p(z).$$

That is, the prompt $x =$ "A nurse" induces the distribution $p(z \mid x) = p(z \mid W = \texttt{nurse})$ on the gender concept. Merely omitting a concept from a prompt does not prevent the associated concept distribution from being altered.

3 Concept Algebra



(a) s["a portrait of a king"]

(b) s["a portrait of a king"]−s["a portrait of a man"]+s["a portrait of a woman"]

Figure 3: Concept transfer applied to transfer the sex concept.

In this section, we study a different approach to controlling generative models that can be effective even when prompting is not. The inspiration here comes from the vector space properties of analogies observed in word embeddings by Mikolov et al. [MYZ13]. A classical example of this is

$$\gamma(\text{"queen"}) \approx \gamma(\text{"king"}) - \gamma(\text{"man"}) + \gamma(\text{"woman"}),$$

where $\gamma(v) \in \mathbb{R}^d$ is the embedding of word v. In this manner, we can elicit the word embedding of "queen" by combining the embeddings of other words.

Our aim is to build an analogous *concept algebra* for general generative models. If we had such tools, we could elicit difficult-to-prompt distributions by combining easy-to-prompt distributions.

The question here is: what is the right notion of representation, analogous to word embedding, in the text-controlled generative model setting? Often, prompts are mapped to *sequences* of word embeddings by a language models [e.g., Rom+22]. Addition and subtraction are not even naturally defined on this space (see appendix E.2)! Even if we could collapse the prompt embeddings to a single representation vector, it's unclear why arithmetic on such vectors would correspond to manipulation of concepts for the generative model in the desired fashion. Indeed, it's not even clear what it means for arithmetic on representations to map to concept manipulation. Without precise statement, it's hard to say whether any given approach actually accomplishes the desired concept manipulation!

Concept Representations The first key idea is to move from representations of prompts to representations of concept distributions.

Definition 3.1. A *concept representation* Rep is a function that maps a concept distribution Q to a representation $\text{Rep}(Q) \in \mathscr{R}$, where $\mathscr{R}$ is a vector space.

Following Section 2, the role of a prompt x is to elicit a concept distribution $Q_x(z_{1:k}) := P(Z_{1:k} = z_{1:k} \mid X = x)$. Accordingly, Rep defines a representation map on prompt strings as (overloading notation) $\text{Rep}(x) = \text{Rep}(Q_x)$. With this generalized notion of representation, we will be able to reason about the relationship between representations of both promptable and unpromptable distributions.

Arithmetic Disentangleability Now, a key requirement for our representations is that it must be possible to manipulate *individual* concepts by arithmetic operations. That is, we should be able to change the `sex` concept without also changing the `profession` concept. The next step is to formalize what it means for a representation to admit this kind of "disentangled" manipulation.

Definition 3.2. A representation Rep is *arithmetically disentangled* with respect to concepts $\mathscr{Z}_1 \dots \mathscr{Z}_k$ if for all product distributions $Q_1 \dots Q_k$ on $\mathscr{Z}_1 \dots \mathscr{Z}_k$,

$$\text{Rep}(Q_1 \cdots Q_k) = \text{Rep}_0 + \sum_{i=1}^{k} \text{Rep}_i(Q_i) \tag{3}$$

for some $\text{Rep}_0 \in \mathscr{R}$, and functions Rep_i with range in $\mathscr{R}$.

Intuitively, for pairs of concepts that can be altered freely of each other, we can often express the model behavior in terms of a product distribution (e.g., the examples in Section 2). Then, it should be the case that changing only one part of the product distribution induces a change only in a subspace of the representation corresponding to that concept.

Formally, arithmetic disentaglability connects concept manipulation and algebraic operations as follows.

Proposition 3.3. *If* Rep *is arithmetically disentangled with respect to* $\mathscr{Z}, \mathscr{W}$*, then*

$$\begin{aligned}\text{Rep}(Q_W^* \times Q_Z^*) = \text{Rep}(Q_W^* \times Q_Z) \\ -\text{Rep}(Q_W \times Q_Z) + \text{Rep}(Q_W \times Q_Z^*)\end{aligned}$$

(The proof is immediate). The point here is that we can now elicit (the representation of) a target distribution Q^* by combining the three distributions on the right-hand side. So, for example, if we wanted to elicit the concept `queen` (and didn't know the word), we might have something like:

$$\begin{aligned}&\text{Rep}\left(\delta_{\text{monarch}}(w)\delta_{\text{woman}}(z)\right) \\ &= \text{Rep}(\text{“king”}) - \text{Rep}(\text{“man”}) + \text{Rep}(\text{“woman”}).\end{aligned}$$

(Though, as we will see shortly, there are some nuances for combining prompted distributions.)

The Score Representation Now, we know abstractly the kind of representation we need for algebraic manipulation of concepts to make sense. The next step is to find a specific representation function—learnable from data—that satisfies the requirement.

We will study the following choice.

Definition 3.4. The *score representation* s is defined by:

$$s[Q](y) := \nabla_y \log \sum_{z_{1:k}} p(y \mid z_{1:k}) q(z_{1:k})$$

Here, $s[Q]$ is itself a function of y and $\mathscr{R}$ is a vector space of functions equipped with the usual notion of addition of functions, and $p(y \mid z_{1:k})$ is the conditional density of Y given the concepts values $z_{1:k}$.[4]

The main motivation for studying the score representation is that, if concepts $\mathscr{Z}_1 \dots \mathscr{Z}_k$ are complete for prompt x, then the score representation of x is the gradient of the log of the data distribution $p(y \mid X = x)$,

$$s[x](y) := \nabla_y \log p(y \mid x) = s[Q_x](y).$$

The importance of this observation is that $\nabla_y \log p(y \mid x)$ is learnable from data. In fact, this score function is ultimately the basis of many controlled generation models [e.g., HJA20; Ram+22; Sah+22], because it characterizes the conditional while avoiding the need to compute the normalizing constant [HD05; SE19]. Accordingly, we can readily compute the score representation of prompts in many generative models, without any extra model training.

Causal Separability Now, it's not obvious that the score representation is arithmatically disentangled. In fact, in general, it isn't. The crux of issue is that concepts are reflected in the representation based on their effect on Y. If the way they effect Y depends fundamentally on some interaction between two concepts, the representation cannot hope to disentangle them. Thus, we must rule out this case.

Definition 3.5. We say that Y is *causally separable* with respect to $\mathscr{Z}_1, \dots, \mathscr{Z}_k$ if there exist Y-measurable variables $(Y_i)_{i \le k}$ and ξ such that

1. $Y = g(Y_{\mathscr{Z}_1}, \dots, Y_{\mathscr{Z}_k}, \xi)$ for some invertible and differentiable g, and
2. $p(y_{\mathscr{Z}_1}, \dots, y_{\mathscr{Z}_k}, \xi | z_{1:k}) = p(\xi) \prod_{i=1}^{k} p(y_{\mathscr{Z}_i} | z_i)$

Informally, the requirement is that we can separately generate $Y_{\mathscr{Z}_i}$ as the part of the output affected by $\mathscr{Z}_i$, for each $i \le k$ (and ξ as the part of the image unrelated to Z_is), then combine these parts to form the final image. That is, generating the visual features associated to a concept $\mathscr{W}$ can't require us to know the value of another concept $\mathscr{Z}$. As an example where causal separability fails, consider the concepts of species $\mathscr{W} = \{\texttt{deer}, \texttt{human}\}$ and sex $\mathscr{Z} = \{\texttt{male}, \texttt{female}\}$. It seems reasonable that there is a Y-measurable $Y_{\mathscr{W}}$ that is the species part of the image—e.g., the presence of fur vs skin, snouts vs noses, and so forth. However, there is no part of Y that corresponds to an sex concept in a manner that's free of species. The reason is that the visual characteristics of sex are fundamentally different across species—e.g., male deer have antlers, but humans usually do not.

It turns out it suffices to rule out this case:

Lemma 3.6. *If Y is causally separable with respect to $\mathscr{W}$ and $\mathscr{Z}$, then the score representation is arithmetically disentangled with respect to $\mathscr{W}$ and $\mathscr{Z}$.*

The proof is in the Appendix D.

Concept Transfer We now know that, given causal separability, the score representation allows us to arithmetically manipulate concepts. The final step is to translate this into a result about distributions elicited by prompting.

Theorem 3.7. *Suppose that Y is causally separable with respect to $\mathscr{W}$ and $\mathscr{Z}$, and that these concepts are complete for prompts x_1, x_2, x_3. Then, if $Q_{x_1} = Q^*_W \times Q_Z$, $Q_{x_2} = Q_W \times Q_Z$, and $Q_{x_3} = Q_W \times Q^*_Z$, we can construct the score representation of the target distribution $Q^*_Z \times Q^*_W$ as:*

$$s[Q^*_Z \times Q^*_W] = s[x_1] - s[x_2] + s[x_3]$$

[4] Formally $Y_\sharp P(\cdot \mid Z_{1:k} = z_{1:k})$.

(a) "a frog playing the piano, anthropomorphic, cartoon"

(b) s["a frog playing the piano, anthropomorphic, cartoon"] - s["cartoon"] + s["photorealistic"]

Figure 4: Score arithmetic successfully changes the style without changing the content.

Proof. With completeness, we can write $s[x_i] = s[Q_{x_i}]$. By Theorem 3.6 the score is arithmetically disentangled, and by Theorem 3.3 arithmetic disentanglement implies the required additive structure. □

In words: suppose we have a target concept distribution over two concepts that we don't know how to elicit with prompting. We can instead write prompts that elicit three concept distributions: one that matches the target on the $\mathscr{W}$ concept, one that matches on the $\mathscr{Z}$ concept, and one that "cancels out" the off-target concepts introduced by the first two prompts. This is essentially the same principle observed in word embedding analogies, but applied to score vectors.

3.1 Examples

We now demonstrate concept transfer using Stable Diffusion, a score-based text-to-image model [Rom+22]. Such de-noising models use a slightly different form of score function. The results hold with appropriate modifications to the assumptions; see appendix A. We give pseudocode for concept transfer in the diffusion model case in appendix B.

A frog playing the piano We return to the frog example. In Section 2 we saw that a target concept distribution $Q^* := Q^*_Z \times Q^*_W$ can be difficult to elicit with prompting when there is an overlap problem in the training data ($Q^*_Z = \delta_{z^*}$, $Q^*_W = \delta_{w^*}$ and $p(z^*, w^*) = 0$). E.g., this occurs when trying to create photorealistic images of an anthropomorphic frog playing the piano. However, it is often easy to elicit other distributions $Q_W \times Q^*_Z$ and $Q^*_W \times Q_Z$ that match the target on each marginal concept, as well as $Q_Z \times Q_W$. For example, the prompt "a frog playing the piano, anthropomorphic, cartoon" elicits the target content (the anthropomorphic frog) but off-target style (cartoon). By combining such partially on-target distributions, we can elicit the overall target. Precisely, Figure 4 show images generated according to the score given by s["a frog playing the piano, anthropomorphic, cartoon"]−s["cartoon"]+s["photorealistic"].

King is to queen as man is to woman Returning to the classic example, we consider transferring the sex concept in pictures of monarchs. Figure 3 shows samples generated by the representation

$$s\text{["a portrait of a king"]}-s\text{["a portrait of a man"]}+s\text{["a portrait of a woman"]}.$$

We use the same random seed for generating the original and modified samples. It's interesting to note that high-level features are often preserved across the paired samples—e.g., the green sash in the first image.

Necessity of assumptions Theorem 3.7 requires both causal seperability and concept completeness for the arithmetic on representations to be equivalent to manipulation of latent concepts. In appendix E.1 we show these conditions are necessary, demonstrating failure modes when the assumptions are violated.

4 Concept Projection

We now consider the task of "removing" a concept $\mathscr{Z}$. We specialize to a binary concept, $\mathscr{Z} = \{z_+, z_-\}$. For concreteness, consider the nurse example. Let $\mathscr{Z}$ denote the sex concept and $\mathscr{W}$ denote the profession concept. We'd like to modify the representation of the prompt $x_1 =$ "a nurse sitting in a white room" to remove the inadvertent activation of $\mathscr{Z}$. As described in Section 2, this prompt elicits the concept distribution $Q_{x_1} = \tilde{Q}_Z \times \delta_{w^*}$ where $\tilde{Q}_Z(z) = P(Z = z \mid W = w^*)$. That is, the prompt induces a skewed distribution over sex. So, we can formalize the problem of "removing" the sex concept as the problem of finding the representation of the target distribution $Q^* = \delta_{w^*} \times P_Z$, where P_Z is the marginal distribution over sex in the training data.

Concept Projection We propose an approach based on the following intuition. First, we want to identify a "direction" in representation space corresponding to the concept we want to remove. Second, we project out this direction. Finally, we add back in the representation of a prompt that induces the desired marginal distribution. In this last step, we only want to modify the target concept. So, we project the added-in part onto the direction of the concept. Morally, in the nurse example, we'd like:

$$(\mathbb{I} - \text{proj}_{\text{sex}})s[\text{"nurse"}] + \text{proj}_{\text{sex}} s[\text{"person"}],$$

where proj_{sex} is the projection onto the direction of the sex concept. So, we swap out the distribution over sex elicited by "nurse" for the one elicited by "person". The question is how to map this intuition into a concrete procedure.

The first problem is to make sense of the notion of a "direction" corresponding to a concept. To that end, suppose that we have access to representations for the extremal and marginal distributions over the concept.

Definition 4.1. Let $s^+ = s[Q_W \times \delta_{z_+}]$, $s^- = s[Q_W \times \delta_{z_-}]$, and $s^\perp = s[Q_W \times P_Z]$.

For example, we might have $s^+ = s[\text{"woman"}]$, $s^- = s[\text{"man"}]$, and $s^\perp = s[\text{"person"}]$. Recall that the score representation of a distribution, $s[Q]$, is itself a function over y. So, the notion of "direction" in representation space will also be a function over y. Then, we define the "direction" of the concept as:

Definition 4.2. The *$\mathscr{Z}$-direction at y* is $\text{dir}_{\mathscr{Z}}(y) := \text{span}(s^+(y) - s^-(y))$. The *$\mathscr{Z}$-projection at y* $\text{proj}_{\mathscr{Z}}(y)$, is the projection onto the $\text{dir}_{\mathscr{Z}}(y)$.

That is, for each input y, we define a y-specific direction for $\mathscr{Z}$ as a subspace of $\mathbb{R}^m$.[5]. And, we define the projection onto this direction at y to be the usual (Euclidean) orthogonal projection operator in $\mathbb{R}^m$.

We can now define our concept removal procedure:

Definition 4.3. The *$\mathscr{Z}$ concept projection procedure* is the operator on representations defined by,

$$\text{rm}_{\mathscr{Z}}[s](y) := \big(\mathbb{I} - \text{proj}_{\mathscr{Z}}(y)\big)s(y) + \text{proj}_{\mathscr{Z}}(y)s^\perp(y).$$

[5] Recall that $s(y) \in \mathbb{R}^m$ for all y, for all score representations.

(a) "A nurse sitting in a white room" generates mostly pictures of women.

(b) Removing the sex concept generates a diverse set of people.

(c) "A Portrait of a mathematician" generates mostly pictures of men.

(d) Removing the sex concept generates a diverse set of people.

Figure 5: Concept projection removes the sex skew.

In words: given the score representation s, we compute $s(y) \in \mathbb{R}^m$, project out the $\mathscr{Z}$-direction of this vector, and then add back in $\mathscr{Z}$-direction corresponding to the marginal distribution.

Formally, we take "removing" a concept from $s[Q_W^* \times \tilde{Q}_Z]$ to mean finding a representation of the target distribution, $s[Q_W^* \times P_Z]$. The following result shows that the concept projection procedure does indeed do this.

Theorem 4.4. *Suppose that:*

(i) prompt x elicits distribution $Q_W^* \times \tilde{Q}_Z$.
(ii) Y is causally separable with respect to $\mathscr{W}, \mathscr{Z}$, and these concepts are complete for prompt x.
(iii) orthogonality condition $(\frac{\partial y}{\partial y_{\mathscr{Z}}})(\frac{\partial y}{\partial y_{\mathscr{W}}})^T = 0$ holds

Then,

$$\mathrm{rm}_{\mathscr{Z}}[s[x]] = s[Q_W^* \times P_Z].$$

(Proof in Appendix D)

4.1 Examples

We give pseudocode for concept projection in the diffusion model case in appendix B.

Removing the Sex Concept In our running example, we want to remove the sex concept from the prompt "a nurse sitting in a white room". Following the discussion above, we need to define the sex direction by finding prompts that elicit extermal distributions, and a prompt

(a) "a baby labrador on the grass" generates mostly light-colored labradors.

(b) "a labrador" generates a diverse set of labradors.

(c) The projection procedure removes the labrador.color concept.

Figure 6: Concept projection removes the bias towards light-colored labradors in the example prompt.

that elicits the target marginal distribution. We'll use $s^+ = s[\text{"a woman"}]$, $s^- = s[\text{"a man"}]$, and $s^\perp = s[\text{"a person"}]$. Now, for any y, we calculate the concept projection operator using these three score representations. This requires only standard linear algebra on $\mathbb{R}^m$. We apply this concept removal operation to the prompt "a nurse sitting in a white room" in fig. 5b. We apply the *same* concept removal operation to the prompt "a portrait of a mathematician" in fig. 5d.

In both cases, we see that the operation succeeds in generating a set of people with diverse sexes. Interestingly, the same concept removal operation succeeds despite the fact that the initial outputs are very different—e.g., the nurses are mostly photos, the mathematicians mostly paintings. This suggests that, as predicted, there is indeed a sensible notion of sex direction, and representations can be modified on this direction in isolation.

Labradors Figure 6a shows the output of "a baby labrador on the grass". The majority (but not all) of the dogs are light-colored. Figure 6b shows the outputs of just "a labrador", and we see that here there is a roughly even mix of dark and light-colored dogs. We define an operator to project out the coloration. Here, we take $s^+ = s[\text{"a dark-colored labrador"}]$, $s^- = s[\text{"a light-colored labrador"}]$, and $s^\perp = s[\text{"a labrador"}]$. We then apply this operator to "a baby labrador on the grass" in fig. 6c. The coloration is now more evenly distributed.

5 Related Work

The work in this paper closely relates to work on detecting and explaining the vector space structure of word embedding methods and their relationship to concept analogies [Mik+13; MYZ13; PSM14; GL14; Aro+15; GAM17; AH19]. The development in this paper may be viewed as generalizing these ideas to text-controlled generative models. Reflecting the change of setting, the formalization and analysis we use here is entirely new. Also of note is that tying the algebraic structure to a downstream task (image generation) allows us to examine the effects of concept manipulation very directly (by looking at the pictures!). In particular, this lets us assess the construction of "embeddings" that have semantic meaning but that *could not be generated by prompting*. The projection idea has also

been studied empirically [Bol+16] and criticized [GG19] in the word embedding context. Such criticisms may apply here as well, though the development is dissimilar enough that it is not obvious.

There is also work on algebraic structure of energy-based and score-based models [e.g., Du+21; Liu+21; NBP22; Ano23]. Du et al. [DLM20] and Liu et al. [Liu+22] stand out as the most relevant. They arrive at a method that is closely related to concept transfer. The concept transfer operation applies to some tasks not covered by their methods, and concept projection is entirely new. Importantly, the development relies on distinct justifications; in this sense, the papers are complementary.

This work relies on recent developments in score-based generative modelling and denoising procedures [SD+15; HJA20; Son+20; DB+21; Ram+22; Rom+22; Sah+22]. Others have already observed diffusion models can fail to produce images faithful to some complex prompts [e.g., MDA22; Swi], motivating the present work.

6 Discussion

This paper was motivated by the need to understand the limitations of prompting for model control. The core challenge here is that, in the absence of a precise statement of what the user meant, it is difficult to systematically study how prompting fails, and how these failures can be overcome. Here, we have developed a particular formalization based on latent concepts in the data. We then demonstrated the utility of this formalism by finding some limitations of prompting and developing concept algebra to address these.

One obvious direction for future work is to understand how the latent concept view translates to large language models. These are also text-controlled generative models, albeit with a different output modality. However, it's less clear here what the analogue of the score representation should be. Some additional idea is needed to concretely connect language model representations to concept distributions.

Another exciting direction for future work is to use this formalism to better understand general approaches to model control. Not all problems can be naturally formalized in terms of latent concepts—e.g., it's unclear how to frame the requirement that language models be factual in terms of latent concepts. However, having a precise goal for at least some tasks may help us understand and develop control methods that can then be ported to other tasks.

Acknowledgements

We thank Ahmad Beirami for feedback on an earlier version, and Google Cloud for the compute credits used for the experiments. This work was partially supported by Open Philanthropy.

References

[AH19] C. Allen and T. Hospedales. "Analogies explained: towards understanding word embeddings". In: *International Conference on Machine Learning*. PMLR. 2019 (cit. on p. 12).

[Ano23] Anonymous. "Reduce, reuse, recycle: compositional generation with energy-based diffusion models and MCMC". In: *Submitted to The Eleventh International Conference on Learning Representations*. under review. 2023 (cit. on p. 13).

[Aro+15] S. Arora, Y. Li, Y. Liang, T. Ma, and A. Risteski. *A latent variable model approach to pmi-based word embeddings*. 2015. arXiv: 1502.03520 (cit. on p. 12).

[Bol+16] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai. "Man is to computer programmer as woman is to homemaker? debiasing word embeddings". *Advances in neural information processing systems* (2016) (cit. on p. 13).

[Bom+21] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, et al. *On the opportunities and risks of foundation models*. 2021. arXiv: 2108.07258 (cit. on p. 1).

[Bro+20] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. "Language models are few-shot learners". *Advances in neural information processing systems* (2020) (cit. on p. 1).

[DB+21] V. De Bortoli, J. Thornton, J. Heng, and A. Doucet. "Diffusion schrödinger bridge with applications to score-based generative modeling". *Advances in Neural Information Processing Systems* (2021) (cit. on p. 13).

[DLM20] Y. Du, S. Li, and I. Mordatch. "Compositional visual generation with energy based models". *Advances in Neural Information Processing Systems* (2020) (cit. on p. 13).

[Du+21] Y. Du, S. Li, Y. Sharma, J. Tenenbaum, and I. Mordatch. "Unsupervised learning of compositional energy concepts". *Advances in Neural Information Processing Systems* (2021) (cit. on p. 13).

[GAM17] A. Gittens, D. Achlioptas, and M. W. Mahoney. "Skip-gram- zipf+ uniform= vector additivity". In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2017 (cit. on p. 12).

[GL14] Y. Goldberg and O. Levy. *Word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method*. 2014. arXiv: 1402.3722 (cit. on p. 12).

[GG19] H. Gonen and Y. Goldberg. *Lipstick on a pig: debiasing methods cover up systematic gender biases in word embeddings but do not remove them*. 2019. arXiv: 1903.03862 (cit. on p. 13).

[HJA20] J. Ho, A. Jain, and P. Abbeel. "Denoising diffusion probabilistic models". *Advances in Neural Information Processing Systems* (2020) (cit. on pp. 8, 13, 16).

[HD05] A. Hyvärinen and P. Dayan. "Estimation of non-normalized statistical models by score matching." *Journal of Machine Learning Research* 4 (2005) (cit. on p. 8).

[Koj+22] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa. *Large language models are zero-shot reasoners*. 2022. arXiv: 2205.11916 (cit. on p. 1).

[Liu+21] N. Liu, S. Li, Y. Du, J. Tenenbaum, and A. Torralba. "Learning to compose visual relations". *Advances in Neural Information Processing Systems* (2021) (cit. on p. 13).

[Liu+22] N. Liu, S. Li, Y. Du, A. Torralba, and J. B. Tenenbaum. *Compositional visual generation with composable diffusion models*. 2022. arXiv: 2206.01714 (cit. on p. 13).

[Luo22] C. Luo. *Understanding diffusion models: a unified perspective*. 2022. arXiv: 2208.11970 (cit. on p. 16).

[Mao+21] A. Mao, P. L. Cheong, I. K. Van, and H. L. Tam. ""i am called girl, but that doesn't matter"-perspectives of male nurses regarding gender-related advantages and disadvantages in professional development". *BMC nursing* 1 (2021) (cit. on p. 2).

[MDA22] G. Marcus, E. Davis, and S. Aaronson. *A very preliminary analysis of dall-e 2*. 2022. arXiv: 2204.13807 (cit. on p. 13).

[Mik+13] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. "Distributed representations of words and phrases and their compositionality". *Advances in neural information processing systems* (2013) (cit. on p. 12).

[MYZ13] T. Mikolov, W.-t. Yih, and G. Zweig. "Linguistic regularities in continuous space word representations". In: *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*. 2013 (cit. on pp. 2, 6, 12).

[NBP22] N. G. Nair, W. G. C. Bandara, and V. M. Patel. *Unite and conquer: cross dataset multimodal synthesis using diffusion models*. 2022. arXiv: 2212.00793 (cit. on p. 13).

[PSM14] J. Pennington, R. Socher, and C. D. Manning. "Glove: global vectors for word representation". In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014 (cit. on p. 12).

[Rad+21] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. "Learning transferable visual models from natural language supervision". In: *International Conference on Machine Learning*. PMLR. 2021 (cit. on p. 1).

[Ram+22] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. *Hierarchical text-conditional image generation with clip latents*. 2022. arXiv: 2204.06125 (cit. on pp. 5, 8, 13).

[Rom+22] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. "High-resolution image synthesis with latent diffusion models". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022 (cit. on pp. 1, 5, 6, 9, 13).

[Sah+22] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes, et al. "Photorealistic text-to-image diffusion models with deep language understanding". *Advances in neural information processing systems* (2022) (cit. on pp. 5, 8, 13).

[SD+15] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. "Deep unsupervised learning using nonequilibrium thermodynamics". In: *International Conference on Machine Learning*. PMLR. 2015 (cit. on p. 13).

[SE19] Y. Song and S. Ermon. "Generative modeling by estimating gradients of the data distribution". *Advances in Neural Information Processing Systems* (2019) (cit. on p. 8).

[Son+20] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. "Score-based generative modeling through stochastic differential equations". In: *International Conference on Learning Representations*. 2020 (cit. on p. 13).

[Swi] Swimmer963. "What dall-e 2 can and cannot do". *LESSWRONG* () (cit. on p. 13).

A Concept Algebra in Diffusion Models

Text-to-image Diffusion Models use score representations in their generation. More specifically, suppose the target is to sample $Y = Y_0 \sim P^*$, with the corresponding score function denoted as s_0. The key ingredients for generation are the score function for Y_t (denoted as s_t), which is Y noised at different levels, (e.g. $Y_t = (1-\alpha_t)Y + \alpha_t \epsilon_t$ for standard independent Gaussian noise ϵ), for $t = 0, ..., T$. See [Luo22] for more details. To apply our results, we modify the assumptions a bit (requiring causal-separability holds for all $Y_t, t = 0, ..., T$):

Theorem A.1. *Suppose that Y_t is causally separable with respect to $\mathscr{W}$ and $\mathscr{Z}$ for each $t \in [T]$, and that these concepts are complete for prompts x_1, x_2, x_3. Then, if $\mathbb{P}(w, z \mid x) = Q_W^* \times Q_Z$, $Q_W \times Q_Z$, and $Q_W \times Q_Z^*$ for $x = x_1, x_2, x_3$ respectively, we have:*

$$s_t[Q_Z^* \times Q_W^*] = s_t[x_1] - s_t[x_2] + s_t[x_3]$$

(It follows immediately.)

Theorem A.2. *Suppose that the assumptions in Theorem 4.4 hold for all $t \in [T]$. Then, with $\mathrm{rm}^t_{\mathscr{Z}}[s_t[x]]$ defined the same way for all $t \in [T]$.*

$$\mathrm{rm}^t_{\mathscr{Z}}[s_t[x]] = s_t[Q_W^* \times P_Z].$$

(The proof is the same as that of Theorem 4.4.)

B Algorithm Pseudocode

Below is an implementation of concept transfer and projection based on DDPM [HJA20] (we can also implement different variants). Note there here we use residual $\epsilon_\theta(y_t, t \mid \gamma)$ instead of the score $s_\theta(y_t, t \mid \gamma)$ for generation, they are equivalent up to a time-varying constant.

Algorithm 1 Concept Algebra for Diffusion Models

Require Diffusion model $\epsilon_\theta(y_t, t|\gamma)$, guidance scale w, covariance matrix $\sigma_t^2 I$, prompt embedding for empty prompt γ_0,
prompt embeddings for algebra $\gamma_1, \gamma_2, \gamma_3$,
Initialize sample $y_T \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
for $t = T, \ldots, 1$ **do**
  $\epsilon_0 \leftarrow \epsilon_\theta(y_t, t|\gamma_0)$ //unconditional score
  $\epsilon_i \leftarrow \epsilon_\theta(y_t, t|\gamma_i)$ for $i = 1, 2, 3$ //conditional scores
  $\epsilon_{\text{cond}} \leftarrow \epsilon_1 - \epsilon_2 + \epsilon_3$ //apply concept algebra
  $\epsilon \leftarrow \epsilon_0 + w(\epsilon_{\text{cond}} - \epsilon_0)$
  $y_{t-1} \sim \mathcal{N}\left(y_t - \epsilon, \sigma_t^2 I\right)$ // sampling
end for

Algorithm 2 Concept Projection for Diffusion Models

Require Diffusion model $\epsilon_\theta(y_t, t|\gamma)$, guidance scale w, covariance matrix $\sigma_t^2 I$, prompt embedding for empty prompt γ_0,
prompt embeddings γ_1, γ_3,
prompt embeddings for projection direction γ_+, γ_-
Initialize sample $y_T \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
for $t = T, \ldots, 1$ **do**
  $\epsilon_0 \leftarrow \epsilon_\theta(y_t, t|\gamma_0)$ //unconditional score
  $\epsilon_i \leftarrow \epsilon_\theta(y_t, t|\gamma_i)$ for $i = 1, 3$ //conditional scores
  $u \leftarrow \epsilon_\theta(y_t, t|\gamma_+) - \epsilon_\theta(y_t, t|\gamma_-)$ //concept direction
  $u \leftarrow \frac{u}{||u||}$
  $\epsilon_{\text{cond}} \leftarrow \epsilon_1 - \langle(\epsilon_1 - \epsilon_3), u\rangle u$ //concept projection
  $\epsilon \leftarrow \epsilon_0 + w(\epsilon_{\text{cond}} - \epsilon_0)$
  $y_{t-1} \sim \mathcal{N}\left(y_t - \epsilon, \sigma_t^2 I\right)$ // sampling
end for

C Experiment Details

The actual experiments are done using components from `CompVis/stable-diffusion-v1-4`. See the attached jupyter notebook for detail.

D Proofs

We first specify some notation. The Z-part and W-part of the density $p[Q] = p[Q_Z \times Q_W]$ are defined as:

$$p_Z[Q_Z](y) := \int p(y_{\mathscr{Z}}|z) Q_Z(dz)$$
$$p_W[Q_W](y) := \int p(y_{\mathscr{W}}|w) Q_W(dw)$$

The Z-part and W-part score functions are defined as follows:

$$s_Z[Q_Z](y) := \nabla_y \log p_Z[Q_Z](y),$$
$$s_W[Q_W](y) := \nabla_y \log p_W[Q_W](y)$$

Also, as a reminder, we have

$$s^+ = s[Q_w \times \delta_{z_+}],\ s^- = s[Q_w \times \delta_{z_-}],\ s^\perp = s[Q_w \times P_Z].$$

We define corresponding Z and W part of above score functions as s_z^+, s_z^-, $s_z^\perp$ and s_w^+, s_w^-, $s_w^\perp$.

Before proving Theorem 4.4, we prove some preliminary lemmas.

Lemma D.1. *If Y is causally separable with respect to $\mathscr{W}$ and $\mathscr{Z}$, then the score representation is arithmetically disentangled with respect to $\mathscr{W}$ and $\mathscr{Z}$.*

Proof. By assumption in Theorem 3.5, we have

$$p(y|z,w) = p(y_{\mathscr{Z}}, y_{\mathscr{W}}, \xi(y)|z,w) \left|\det(\frac{\partial g}{\partial y})\right|$$
$$= p(y_{\mathscr{Z}}|z) p(y_{\mathscr{W}}|w) p(\xi(y)) \left|\det(\frac{\partial g}{\partial y})\right|$$

Therefore,

$$\begin{aligned} p[Q](y) =& p[Q_Z \times Q_W](y) \\ =& p_Z[Q_Z](y) p[Q_W](y) p(\xi(y)) \left| \det(\frac{\partial g}{\partial y}) \right|. \end{aligned}$$

Then, taking the log-derivative, we get its score function as follows:

$$s[Q_Z \times Q_W](y) = s_Z[Q_Z](y) + s_W[Q_W](y) + s_0(y) \tag{4}$$

where $s_0(y) := \nabla_y \log \left(p(\xi(y)) \left| \det(\frac{\partial g}{\partial y}) \right| \right)$. So the claim follows. □

Lemma D.2. *For a fixed y, the difference between the unbiased and biased Z-part score functions is in the Z direction. Mathematically,*

$$s_Z[P_Z](y) - s_Z[\tilde{Q}_Z](y) \in \mathrm{dir}_{\mathscr{Z}}(y) \tag{5}$$

Proof. We can write the biased $\tilde{Q}_Z$ as: $\tilde{Q}_Z(z) = \pi \delta_{z_+}(z) + (1-\pi)\delta_{z_-}(z)$ for some unknown $\pi \in [0,1]$, and the fair $P_Z(z) = \frac{1}{2}\delta_{z_+}(z) + \frac{1}{2}\delta_{z_-}(z)$. For a general linear combination of these two delta functions: $Q_Z(z) = \alpha \delta_{z_+}(z) + (1-\alpha)\delta_{z_-}(z)$, $\alpha \in [0,1]$, its score function can be rewritten as follows:

$$\begin{aligned} s_Z[Q_Z](y) &= \nabla_y \log \int p(y_{\mathscr{Z}}|z) Q_Z(dz) \\ &= \nabla_y \log (\alpha p(y_{\mathscr{Z}}|z_+) + (1-\alpha) p(y_{\mathscr{Z}}|z_-)) \\ &= \beta(y_Z) \big(\nabla_y \log p(y_{\mathscr{Z}}|z_+) \\ &+ (1-\beta(y_Z)) \nabla_y \log p(y_{\mathscr{Z}}|z_-) \big) \\ &= \beta(y_{\mathscr{Z}}) \big(\nabla_y \log p_Z[\delta_{z_+}](y) \\ &+ (1-\beta(y_{\mathscr{Z}})) \nabla_y \log p_Z[\delta_{z_-}](y) \big) \\ &= \beta(y_{\mathscr{Z}}) s_Z[\delta_{z_+}](y) + (1-\beta(y_{\mathscr{Z}})) s_Z[\delta_{z_-}](y) \end{aligned}$$

where $\beta(y_{\mathscr{Z}}) = \frac{\alpha p(y_{\mathscr{Z}}|z_+)}{\alpha p(y_{\mathscr{Z}}|z_+) + (1-\alpha) p(y_{\mathscr{Z}}|z_-)}$. Then, we can prove that the difference between this score function and $s_Z[\delta_{z_+}](y)$ is in $\mathrm{dir}_{\mathscr{Z}}(y)$ in the following way:

$$\begin{aligned} & s_Z[Q_Z](y) - s_Z[\delta_{z_+}](y) \\ =& -(1-\beta(y_{\mathscr{Z}})) \big(s_Z[\delta_{z_+}](y) - s_Z[\delta_{z_-}](y) \big) \in \mathrm{dir}_{\mathscr{Z}}(y). \end{aligned}$$

Thus, the difference between the unbiased and biased Z-part score functions is

$$\begin{aligned} & s_Z[P_Z](y) - s_Z[\tilde{Q}_Z](y) \\ =& \big(s_Z[P_Z](y) - s_Z[\delta_{z_+}](y) \big) - \big(s_Z[\tilde{Q}_Z](y) - s_Z[\delta_{z_+}](y) \big) \in \mathrm{dir}_{\mathscr{Z}}(y). \end{aligned}$$

□

Lemma D.3. *For a fixed y, if $(\frac{\partial y}{\partial y_{\mathscr{Z}}})(\frac{\partial y}{\partial y_{\mathscr{W}}})^T = 0$, then the difference between any two W-part score functions is in the complement of gender direction. Mathematically, for any $\mathscr{W}$-distribution Q_W^1 and Q_W^2,*

$$proj_{\mathscr{Z}}(y) \big(s_W[Q_W^1](y) - s_W[Q_W^0](y) \big) = 0 \tag{6}$$

Proof. It suffices to prove that for any $s_Z[Q_Z](y)$ and $s_W[Q_W](y)$, their dot product (defined by the Euclidean distance) is 0. Since we have $(\frac{\partial y}{\partial y_{\mathscr{Z}}})(\frac{\partial y}{\partial y_{\mathscr{W}}})^T = 0$, this proof is straightforward.

$$\begin{aligned}
&s_z[Q_Z](y)^T s_W[Q_W](y)\\
=&\nabla_y \log p_Z[Q_Z](y)^T \nabla_y \log p_W[Q_W](y)\\
=&\nabla_{y_{\mathscr{Z}}} \log p_Z[Q_Z](y)^T (\frac{\partial y}{\partial y_{\mathscr{Z}}})(\frac{\partial y}{\partial y_{\mathscr{W}}})^T \nabla_{y_{\mathscr{W}}} \log p_W[Q_W](y)\\
=&0
\end{aligned}$$

□

Theorem 4.4. *Suppose that:*

(i) prompt x elicits distribution $Q_W^* \times \tilde{Q}_Z$.
(ii) Y is causally separable with respect to $\mathscr{W}, \mathscr{Z}$, and these concepts are complete for prompt x.
(iii) orthogonality condition $(\frac{\partial y}{\partial y_{\mathscr{Z}}})(\frac{\partial y}{\partial y_{\mathscr{W}}})^T = 0$ holds

Then,

$$\mathrm{rm}_{\mathscr{Z}}[s[x]] = s[Q_W^* \times P_Z].$$

Proof. We will prove this theorem in three parts. The y is fixed throughout the proof.

Part I: Decompositions of the score functions

In this part, we try to decompose the score functions into sums of only-Z-related and only-W-related parts.

Since $\mathscr{W}$ and $\mathscr{Z}$ are complete for the prompt x (Theorem 4.4 (i)) and prompt x elicits the concept distribution $Q_W^* \times \tilde{Q}_Z$ (Theorem 4.4 (ii)) , we have

$$\begin{aligned}
s^\perp(y) =& s[\tilde{Q}_{\mathscr{W}} \times P_Z](y),\\
s[x](y) =& s[Q_W^* \times \tilde{Q}_Z](y).
\end{aligned} \tag{7}$$

Due to the causal seperability (Theorem 4.4 (i)) and Theorem 3.6, we can rewrite the difference $s^\perp(y)$ and $s[x](y)$ in the following way:

$$\begin{aligned}
&s^\perp(y) - s[x](y)\\
=&s[\tilde{Q}_W \times P_Z](y) - s[Q_W^* \times \tilde{Q}_Z](y)\\
=&s_W[\tilde{Q}_W](y) + s_Z[P_Z](y) + s_0(y)\\
&- s_W[Q_W^*](y) - s_Z[\tilde{Q}_Z](y) - s_0(y)
\end{aligned}$$

If we rearrange terms in the above equation, we get

$$\begin{aligned}
s^\perp(y) - s[x](y) =& s_Z[P_Z](y) - s_Z[\tilde{Q}_Z](y)\\
&+ s_W[\tilde{Q}_W](y) - s_W[Q_W^*](y)
\end{aligned} \tag{8}$$

With this equation, we decompose the difference of two score functions into a Z-part difference $s_Z[P_Z](y) - s_Z[\tilde{Q}_Z](y)$ and a W-part difference $s_W[\tilde{Q}_W](y) - s_W[Q_W^*](y)$.

Part II: Projections of the score functions

In this part, we consider projections on $\mathrm{dir}_{\mathscr{Z}}(y)$ for the Z-part difference and W-part difference that we obtained in the Part I, respectively.

For the Z-part, by Theorem D.2, we know that

$$s_Z[P_Z](y) - s_Z[\tilde{Q}_Z](y) \in \mathrm{dir}_{\mathscr{Z}}(y).$$

Accordingly, the projection of the Z-part difference (also the LHS of the above equation) on the $\mathrm{dir}_{\mathscr{Z}}(y)$ will be itself. That is,

$$\mathrm{proj}_{\mathscr{Z}}(y)\left(s_Z[P_Z](y) - s_Z[\tilde{Q}_Z](y)\right) = s_Z[P_Z](y) - s_Z[\tilde{Q}_Z](y) \tag{9}$$

For the W-part, by Theorem D.3, we can easily conclude that the projection of the W-part difference on the $\mathrm{dir}_{\mathscr{Z}}(y)$ should be 0, i.e.,

$$\mathrm{proj}_{\mathscr{Z}}(y)\left(s_W[\tilde{Q}_W](y) - s_W[Q^*_W](y)\right) = 0 \tag{10}$$

Part III: The final equation—concept removal achieves the fair score function

Now we can justify the formula for rm for $s[x]$ by the following calculation:

$$\begin{aligned}
\mathrm{rm}_{\mathscr{Z}}[s[x]] &= \mathrm{proj}_{\mathscr{Z}}(y)\left(s^{\perp}(y) - s[x](y)\right) + s[x](y) \\
&= \mathrm{proj}_{\mathscr{Z}}(y)\left(s[\tilde{Q}_{\mathscr{W}} \times P_Z](y) - s[Q^*_W \times \tilde{Q}_Z](y)\right) \\
&\quad + s[Q^*_W \times \tilde{Q}_{\mathscr{Z}}](y) \\
&= \mathrm{proj}_{\mathscr{Z}}(y)\left(s_W[\tilde{Q}_W](y) - s_W[Q^*_W](y)\right) \\
&\quad + \mathrm{proj}_{\mathscr{Z}}(y)\left(s_Z[P_Z](y) - s_Z[\tilde{Q}_Z](y)\right) \\
&\quad + s_W[Q^*_W](y) + s_Z[\tilde{Q}_Z](y) \\
&= s_Z[P_Z](y) - s_Z[\tilde{Q}_Z](y) + s_W[Q^*_W](y) + s_Z[\tilde{Q}_Z](y) \\
&= s_Z[P_Z](y) + s_W[Q^*_W](y) \\
&= s[Q^*_W \times P_Z](y).
\end{aligned}$$

The second and third equation refer to Equations (7) and (8). The fourth equation is based on Equations (9) and (10).

Therefore, we prove that the concept removal method is capable of achieving the target fair score function/probability distribution. □

E Additional experiments

In this section, we provide examples that demonstrate the necessity of the completeness and causal separability assumptions, Theorems 2.1 and 3.5 (Appendix E.1), as well as examples that compare image generation quality for manipulations of the score embedding with manipulations of a text embedding.

E.1 Necessity of Assumptions

We show that our methods would fail when necessary conditions—completeness (Theorem 2.1) and causal separability (Theorem 3.5)—become invalid. In this section, we show two concrete examples of failures and analyze which assumptions are violated.

Figure 7a shows that we are unable to transfer the gender of the nurse when we calculate the score function of a `male nurse` by s[“a female nurse”]$-s$[“a female deer”]$+s$[“a male deer”]. The target concept Z and W are `sex` $\in$ {`male`, `female`} and `species`$\in$ {`human`, `deer`}. It's obvious that the `sex` and `species` have an interaction effect on the image Y — different species induce different sexual characteristics.

Figure 8a displays a failure of style transfer. The style of a dog picture cannot be transferred from `cartoon` to `renaissance style` by changing the score function to s[“a dog, cartoon”]$-$

s["a man, cartoon"]+s["a man, renaissance"]. The target concept Z and W are style $\in$ {renaissance style/oil painting, cartoon} and subject $\in$ {man, dog}. We can understand it as a violation of completeness assumption:

$$P\big(\texttt{clothes} = \texttt{robe} \mid \texttt{man, oil painting},\ x = \text{“renaissance style”}\big) \approx 1,$$

but P (clothes = robe | man, oil painting) is definitely smaller (since all specific clothes kinds should appear almost uniformly).

In practice, when applying this score arithmetic-based method, we should check assumptions carefully to assure desired images.

E.2 Prompt-embedding arithmetic

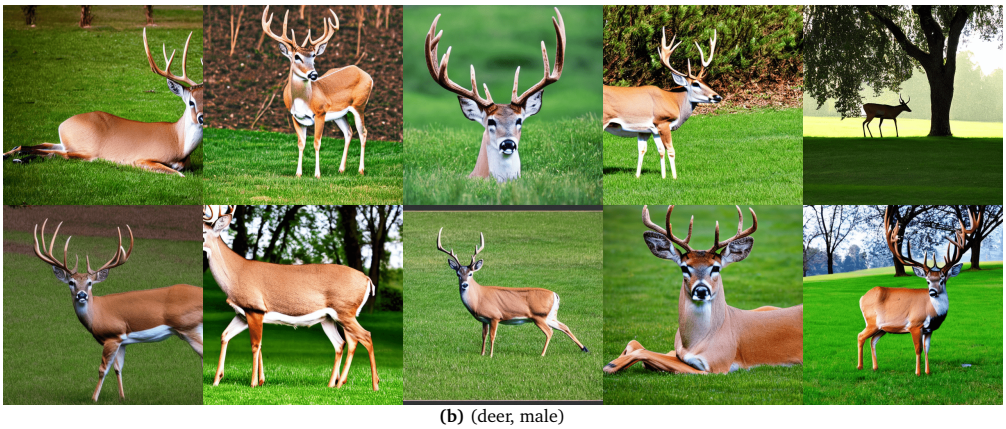
Although a natural idea is to apply arithmetic on prompt embedding $\gamma(x)$'s (and indeed it works for simple prompts fig. 9a), such operation is not well-defined for general language models. For example, in Stable Diffusion, $\gamma(x)$ is the concatenation of embeddings for each word in a sentence (padded with empty tokens to the same length). In this case, it's not clear what $\gamma(x_1)-\gamma(x_2)+\gamma(x_3)$ means when x_1, x_2 and x_3 are sentences of different lengths. It's not surprising the generated images have very poor qualities — e.g. in fig. 9c and fig. 9f.

For the arithmetic to make sense, we need to do some prompt engineering so that x_1, x_2 and x_3 share very similar sentence structure. For example, in order to generate a frog playing the piano, anthropomorphic, we would use

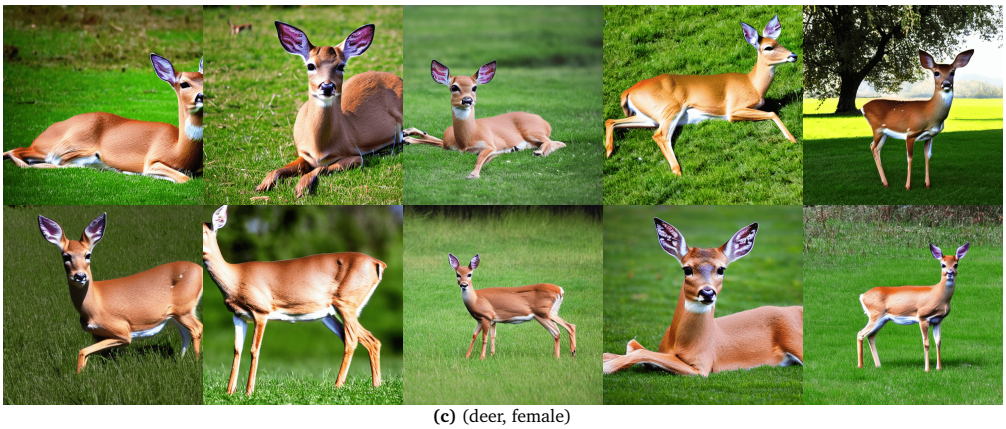
x_1 = “a frog playing the piano, anthropomorphic, cartoon”
x_2 = “a man playing the piano, anthropomorphic, cartoon”
x_3 = “a man playing the piano, anthropomorphic, photorealistic”

even though the sentences may not make sense (e.g. don't need “anthropomorphic” to describe a human behavior). It can generate images of ordinary quality, but has a lower success rate of transferring the style while retaining the content, as seen in the comparison between fig. 9d and fig. 9e. In the nurse example, we do similar engineering. However, the generated images are all man. Therefore it's not clear why prompt embedding arithmetic fails, and how to improve it.

(a) (nurse, female) - (deer, female) + (deer, male)



(b) (deer, male)



(c) (deer, female)

Figure 7: (nurse, female) - (deer, female) + (deer, male) fails to transfer gender

(a) (dog, cartoon) - (man, cartoon) + (man, renaissance)

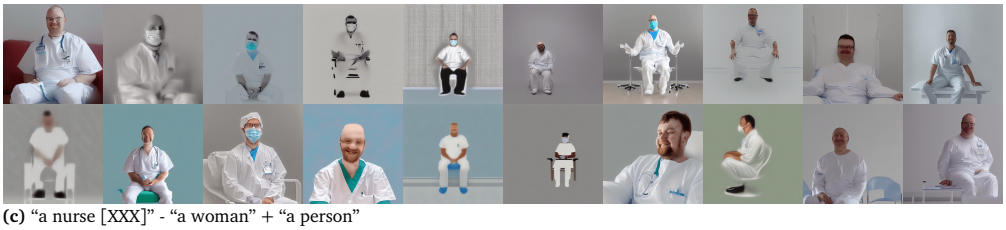
(b) (man, cartoon)

(c) (man, renaissance)

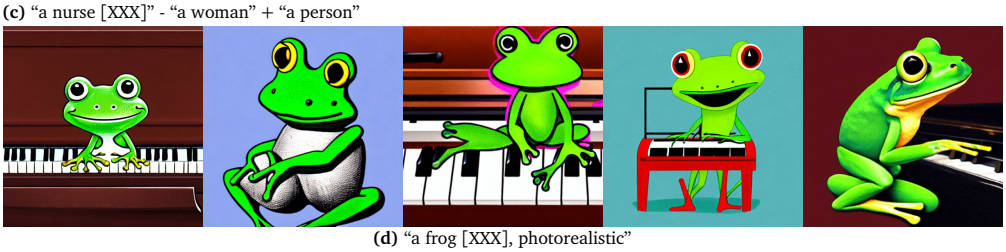
Figure 8: (dog, cartoon) - (man, cartoon) + (man, renaissance) fails to transfer style

(a) “a portrait of a king” - “a portrait of a woman” + “a portrait of a man”

(b) “a nurse [XXX]” - “a woman [XXX]” + “a person [XXX]”

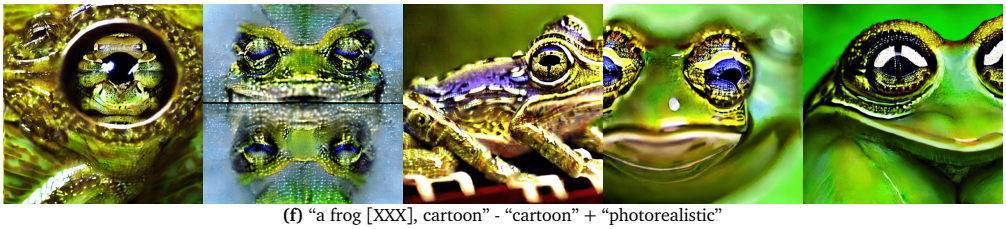


(c) “a nurse [XXX]” - “a woman” + “a person”



(d) “a frog [XXX], photorealistic”

(e) “a frog [XXX], cartoon” - “a man [XXX], cartoon” + “a man [XXX], photorealistic”



(f) “a frog [XXX], cartoon” - “cartoon” + “photorealistic”

Figure 9: Prompt embedding arithmetic only works when the prompts are simple and have identical structures. Even with prompt engineering so that the captions have the same sentence structure, it still fails to generate the intended concepts. The [XXX] above refer omitted parts that are shared across the prompts used.