
Beyond the Black Box: Identifiable Interpretation and Control in Generative Models via Causal Minimality

Anonymous Author(s)

Affiliation

Address

email

Abstract

1 Deep generative models, while revolutionizing fields like image and text generation,
2 largely operate as opaque “black boxes”, hindering human understanding, control,
3 and alignment. Current empirical interpretability tools often lack theoretical guaran-
4 tees, risking subjective or unreliable insights. In this work, we tackle this challenge
5 by establishing a principled foundation for interpretable and controllable generative
6 models. We demonstrate that the principle of causal minimality – favoring the sim-
7 plest causal explanation – can endow the latent representations of diffusion vision
8 and autoregressive language models with clear causal interpretation and robust,
9 component-wise identifiable control. We introduce a novel theoretical framework
10 for hierarchical selection models, where higher-level concepts emerge from the
11 constrained composition of lower-level variables, better capturing the complex
12 dependencies in data generation. Under theoretically derived minimality conditions
13 (manifesting as sparsity or compression constraints), we show that learned represen-
14 tations can be equivalent to the true latent variables of the data-generating process.
15 Empirically, applying these constraints to leading generative models allows us to
16 extract their innate hierarchical concept graphs, offering fresh insights into their
17 internal knowledge organization. Furthermore, these causally grounded concepts
18 serve as effective levers for fine-grained steering of model outputs, paving the way
19 for more transparent, reliable systems.

20 1 Introduction

21 The transformative power of deep generative models, including diffusion models [1–6] and language
22 models [7–10], is reshaping numerous domains. However, their escalating complexity and scale
23 frequently cast them as opaque “black boxes” [11, 12]. This opacity presents a formidable barrier
24 to genuine human understanding, severely curtails our ability to exert precise control over their
25 behavior [13–16], and complicates the crucial task of aligning them with human values and intentions.

26 Although recent empirical tools, such as sparse autoencoders (SAEs) for Large Language Models
27 (LLMs) [17–19] and diffusion models [20–24], offer avenues for probing these models, a fundamental
28 gap persists. Without rigorous theoretical underpinnings, interpretations derived from these methods
29 risk being subjective or susceptible to human biases, rendering them potentially untrustworthy for
30 risk-sensitive applications [25–27]. In this work, we directly tackle this critical challenge, seeking to
31 establish a principled foundation for interpretable and controllable generative models.

32 Our investigation centers on two questions: Under what *theoretical conditions* can we reliably identify
33 meaningful, interpretable latent concepts within the intricate architectures of modern generative
34 models? And, crucially, what *actionable, theoretically-grounded insights* can empower us to advance
35 both the interpretability and the controllability of these powerful systems?

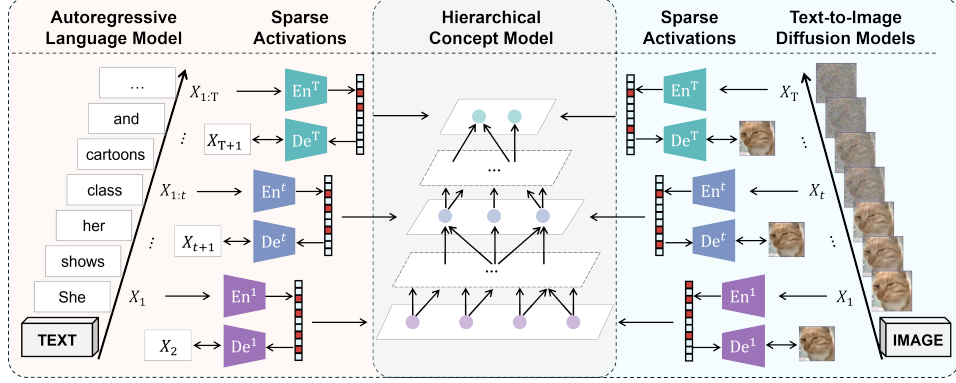


Figure 1: **Extracting hierarchical concept graphs.** We view diffusion and language models as series of “autoencoders” parameterized by t and enforce sparsity constraints on the “encoder” outputs (i.e., models’ intermediate features). For diffusion models (right), higher-level concepts correspond to features at timesteps closer to the noise input. For language models (left), the hierarchical level depends on the token position in the sequence – features generated at later token positions represent higher-level semantics.

We posit and demonstrate that the principle of *causal minimality* [28–30] offers a powerful and unifying lens for this endeavor. This principle, advocating for the simplest causal model consistent with observations, allows for the identification of latent hierarchical concept structures. In our context, minimality translates to either sparsity in the concept graphs or the most parsimonious set of active discrete concept states. We specifically explore its application to autoregressive language models (e.g., [7–9]) and text-to-image diffusion models (e.g., [31, 32, 3]). Our findings indicate that imposing sparsity constraints on internal representations (akin to SAEs) is instrumental for identifying continuous visual concepts and discrete textual concepts.

A cornerstone of our contribution is establishing the first identifiability results for *selection*-based [33–40] hierarchical models. In such models, higher-level variables emerge as effects of compositions of lower-level variables, where higher-level variables control and select the configuration of lower-level ones. This fundamentally diverges from traditional hierarchical causal models [41–43], in which causal influence typically propagates from higher to lower levels. The selection model structure is particularly adept at capturing the intricate conditional dependencies among low-level features for forming coherent high-level concepts – it explains how specific arrangements of wheels, doors, and a roof constitute a recognizable “car”, rather than a disjointed collection of parts. Traditional hierarchical models often neglect such intra-level dependencies by assuming no within-layer causal edges, as explicitly modeling them would yield overly dense graphs. The selection mechanism, in contrast, offers a more parsimonious approach to this essential coordination. Its adherence to the minimality principle strongly favors it as a more accurate representation of the true model.

Despite the appeal, their identifiability has been underexplored. Prior research has largely centered on traditional hierarchical structures. Moreover, their techniques often rest on simplifying assumptions (e.g., linearity [44–47] or achieve only subspace-level identifiability [48]). Such methods are generally inapplicable to the hierarchical selection models. Our framework is the first to establish *component-wise* identifiability for both *continuous and discrete hierarchical selection* models. Specifically, we demonstrate that under well-defined minimality conditions (Conditions 3.2-iv and 3.4-iii), the learned representations are equivalent to the true latent variables of the underlying hierarchical process. This disentanglement of individual, atomic concepts is what affords significantly more nuanced interpretability and precise control in the resulting generative models.

We substantiate our theory empirically. By applying the derived sparsity constraints to state-of-the-art diffusion and language models, we successfully extract their innate hierarchical concept graphs (Figure 1). This not only illuminates their internal knowledge organization but also shows that these causally-grounded concepts serve as highly effective levers for steering generative outputs.

Please refer to Appendix A for discussion with related work.

2 Deep Generative Models as a Hierarchical Latent Variable Model

Notations. We denote random variables with upper-case characters (e.g., X) and values with lower-case characters (e.g., x). We distinguish multidimensional objects with bold fonts (e.g., \mathbf{X}) and refer

to their dimensionality as $n(\cdot)$. With a slight abuse of notation, we view multidimensional variables as sets when appropriate. Parents $\text{Pa}(\cdot)$ and children $\text{Ch}(\cdot)$ relations are defined based on the selection graph Figure 2. If X has only one child Y , we refer to X as a pure parent of Y , i.e., $X \in \text{PPa}(Y)$; if X has other children than Y , we refer to X as a hybrid parent of Y , i.e., $X \in \text{HPa}(Y)$. We denote the set of natural numbers $\{1, \dots, M\}$ as $[M]$. More background information is in Appendix C.

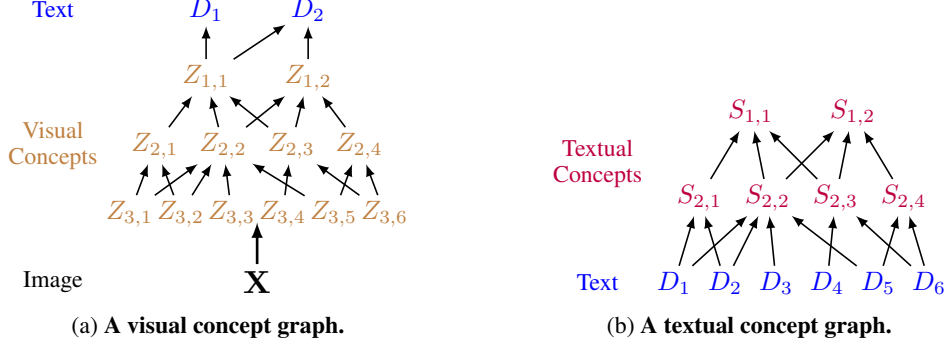


Figure 2: **Hierarchical selection models.** We denote text as \mathbf{D} , discrete textual concepts as \mathbf{S} , continuous visual concepts as \mathbf{Z} , and the image as \mathbf{X} . High-level concepts function as selection variables of low-level variables, capturing the conditional dependence of low-level concepts given high-level concepts in natural data.

We denote the image as the continuous variable $\mathbf{X} \in \mathbb{R}^{n(\mathbf{X})}$ and text as the discrete variable $\mathbf{D} \in \mathbb{N}^{n(\mathbf{D})}$. Visual concepts are $\mathbf{Z} := [\mathbf{Z}_1, \dots, \mathbf{Z}_{L_V}]$, where L_V is the number of visual hierarchical levels and $\mathbf{Z}_l \in \mathbb{R}^{n(\mathbf{Z}_l)}$ are concepts at level l . Textual concepts are $\mathbf{S} := [\mathbf{S}_1, \dots, \mathbf{S}_{L_T}]$, where L_T is the number of textual hierarchical levels and $\mathbf{S}_l \in \Omega_l \subset \mathbb{N}^{n(\mathbf{S}_l)}$ are concepts at level l (Figure 2).

Hierarchical processes and selection mechanisms. Our framework conceptualizes high-level concepts as emerging from or being effects of lower-level concepts. This is captured by a *selection mechanism* [33–40], where variables \mathbf{V}_l at a higher level of abstraction (smaller l) is determined by its constituent, more detailed components \mathbf{V}_{l+1} (i.e., its “parents”). The selection function $g_{\mathbf{V}_l}$ maps these lower-level constituents to the higher-level concept:

$$\mathbf{V}_l := g_{\mathbf{V}_l}(\mathbf{V}_{l+1}). \quad (1)$$

This can be viewed as: \mathbf{V}_l is a selection variable over \mathbf{V}_{l+1} . In many natural data distributions of interest, we can only observe the data points for which the selection criterion is met, i.e., \mathbf{V}_l only takes on a strict subset of its range Ω . Therefore, the distribution of \mathbf{V}_{l+1} is always the conditional distribution $\mathbb{P}(\mathbf{V}_{l+1}|\mathbf{V}_l)$. This conditioning on \mathbf{V}_l can induce dependencies among components in \mathbf{V}_{l+1} . For instance, if $\mathbf{V}_{l+1,i} \rightarrow \mathbf{V}_l \leftarrow \mathbf{V}_{l+1,j}$, conditioning on \mathbf{V}_l makes $\mathbf{V}_{l+1,i}$ and $\mathbf{V}_{l+1,j}$ dependent.

Under this formulation, one can leverage the inverse process of (1) to sample observable data (images, text), proceeding from higher-level abstract concepts to lower-level concrete details:

$$\begin{aligned} \mathbf{Z}_0 &\sim \mathbb{P}(\mathbf{Z}_0), \quad \mathbf{Z}_l \sim \mathbb{P}(\mathbf{Z}_l|\mathbf{Z}_{l-1}), \quad l \in \{1, \dots, L_V + 1\}; \\ \mathbf{S}_1 &\sim \mathbb{P}(\mathbf{S}_1), \quad \mathbf{S}_l \sim \mathbb{P}(\mathbf{S}_l|\mathbf{S}_{l-1}), \quad l \in \{2, \dots, L_T + 1\}, \end{aligned} \quad (2)$$

where we denote $\mathbf{Z}_0 := \mathbf{D}$, $\mathbf{Z}_{L_V+1} := \mathbf{X}$, and $\mathbf{S}_{L_T+1} := \mathbf{D}$. While (2) defines the generative pathway, the underlying structure is shaped by the selection principle of (1): the conditional distributions in (2) are implicitly learned if one has learned selection mechanisms in (1) and vice versa.

Why is this “selection” formulation? The “selection” perspective is critical for modeling how abstract concepts enforce coherence among their more concrete constituents. Consider generating an image of a “bicycle” (a high-level concept \mathbf{Z}_l). Its components – wheels, frame, handlebars (lower-level concepts \mathbf{Z}_{l+1}) – must not only be present but also be arranged in a specific, structurally sound configuration. Traditional hierarchical models [42, 49, 43] assume independent low-level concepts $\mathbf{Z}_{l+1,i}$ given high-level concepts \mathbf{Z}_l and stochastically sample these components, which could potentially lead to unrealistic arrangements (e.g., wheels detached from the frame if the learned conditional is not perfect). Therefore, these models must additionally incorporate causal edges within each hierarchical level to capture this conditional dependency, resulting in highly dense causal graphs. In contrast, the selection model, by positing that \mathbf{Z}_l is an effect of a specific configuration of \mathbf{Z}_{l+1} ,

emphasizes that the “bicycle” concept arises from a coherent selection and composition of its parts. This structured dependency, induced by the selection mechanism, yields a much simpler graphical model to describe the natural data distribution, thus preferred by the *minimality* principle.

Connections to text-to-image diffusion models. The iterative denoising process in diffusion aligns with our hierarchical data construction. These models involve a sequence of transformations $\{f_t\}_{t=1}^T$, parameterized by timestep t , that progressively restore a less noisy image \mathbf{X}_t from a more corrupted version \mathbf{X}_{t+1} . As interpreted by Kong et al. [50], each f_{t+1} can be viewed as an autoencoder: it extracts a representation $\mathbf{Z}_{S(t+1)}$ ($S(t+1)$ indexes U-Net features associated with timestep $t+1$) from the noisy input \mathbf{X}_{t+1} , and uses this representation to produce the less noisy \mathbf{X}_t . In this view, representations $\mathbf{Z}_{S(t+1)}$ from higher noise levels (larger t , where \mathbf{X}_{t+1} is closer to pure noise) correspond to higher-level, more abstract concepts in our hierarchy (e.g., \mathbf{Z}_l with smaller l), as fine-grained details are obscured by noise. Conversely, representations from lower noise levels (smaller t) capture more concrete, lower-level details (e.g., \mathbf{Z}_l with larger l). The diffusion model’s step-wise refinement thus mirrors our hierarchical generation $\mathbb{P}(\mathbf{Z}_{l+1}|\mathbf{Z}_l)$, with the initial text prompt \mathbf{D} typically guiding the most abstract visual concepts (e.g., $\mathbf{Z}_1 \sim \mathbb{P}(\mathbf{Z}_1|\mathbf{D})$, see Figure 1).

Connections to autoregressive language models. An autoregressive language model can be seen as learning an “encoder” that maps a sequence of input tokens ($D_{1:t}$) to an internal state $\hat{\mathbf{S}}_l$ (e.g., activations within transformer layers). This internal state $\hat{\mathbf{S}}_l$ then informs the “decoder” to predict the subsequent token. For optimal prediction, this learned representation $\hat{\mathbf{S}}_l$ should ideally capture the information of the true concept \mathbf{S}_l that *d-separates* the input tokens $D_{1:t}$ from the next token D_{t+1} . To achieve this d-separation, \mathbf{S}_l should belong to a higher concept level for a larger span of text $D_{1:t}$ (i.e., larger $t \rightarrow$ smaller l , see Figure 1). Consequently, broad thematic or narrative structures spanning larger text segments can be compressed into higher-level concepts in our hierarchy (e.g., \mathbf{S}_1), while more localized syntactic or lexical choices correspond to lower-level concepts (e.g., \mathbf{S}_{L_T}).

Identifiability and interpretability. In light of the connection, a crucial question remains: are the internal representations learned by these models (e.g., U-Net features, transformer activations) truly reflective of the ground-truth concepts of the data, or are they merely effective for the generation task without being inherently interpretable and controllable? This motivates the need for *identifiability* guarantees that affirm the equivalence between the two worlds, which we present in Section 3.

3 Identifiable Representations under the Causal Minimality Principle

In this section, we establish the theoretical underpinnings for achieving interpretable and controllable generative models. Our central idea is that by adhering to the *causal minimality principle* [28–30], which favors the simplest causal explanation consistent with observations, we can endow the latent representations of complex generative models with component-wise identifiability. This identifiability, in turn, is crucial for meaningful human understanding and precise intervention.

We realize the causal minimality principle differently for continuous visual concepts and discrete textual concepts. For visual concepts, minimality manifests as a preference for *sparse graphical dependencies* within the latent hierarchy. This implies that concepts are formed through a limited set of direct causal influences, making the underlying structure easier to discern. For textual concepts, minimality translates to seeking the *most compressed representation*, which corresponds to achieving minimal support sizes for these discrete concepts while preserving full information.

A key challenge we address is the identifiability of *hierarchical selection models*. In these models, higher-level concepts are effects of lower-level concepts. This contrasts with traditional hierarchical models where causality often flows from abstract to concrete, and where latent variables typically act as confounders [49, 43, 42, 51, 50, 44–46, 48, 47]. In our selection framework, latent variables act as colliders, rendering many existing identifiability results inapplicable. This distinction necessitates the novel theoretical development presented herein. Our goal is to achieve *component-wise identifiability*:

Definition 3.1 (Component-wise Identifiability). Let \mathbf{Z} and $\hat{\mathbf{Z}}$ be variables under two model specifications. We say that \mathbf{Z} and $\hat{\mathbf{Z}}$ are *identified component-wise* if there exists a permutation π such that for each $i \in [n(\mathbf{Z})]$, $\hat{Z}_i = h_i(Z_{\pi(i)})$ where h_i is an invertible function.

This strong form of identifiability ensures that each learned latent component \hat{Z}_i corresponds to a single true latent component $Z_{\pi(i)}$. This is vital for unambiguous interpretation and targeted control.

We assume the standard faithfulness condition [52–54, 44], meaning the graphical model accurately reflects all conditional independence relations in the data.

3.1 Learning Visual Concepts via Graph Sparsification

We first consider the identification of continuous latent visual concepts \mathbf{Z} .

Condition 3.2 (Visual Concept Identification Conditions).

- i **Informativeness:** *There exists a diffeomorphism $g_l : (\mathbf{Z}_l, \epsilon_l) \mapsto \mathbf{X}$ for $l \in [0, L]$, where ϵ_l denotes independent exogenous variables.*
- ii **Smooth Density:** *The probability density function $p(\mathbf{z}_{l+1}|\mathbf{z}_l)$ is smooth for any $l \in [L_V]$.*
- iii **Sufficient Variability:** *For each Z and its parents $\tilde{\mathbf{Z}} := \text{Pa}(Z)$, at any value $\tilde{\mathbf{z}}$ of $\tilde{\mathbf{Z}}$, there exist $n(\tilde{\mathbf{Z}}) + 1$ distinct values of Z , denoted as $\{z^{(n)}\}_{n=0}^{n(\tilde{\mathbf{Z}})}$, such that the vectors $\mathbf{w}(\tilde{\mathbf{z}}, z^{(n)}) - \mathbf{w}(\tilde{\mathbf{z}}, z^{(0)})$ are linearly independent where $\mathbf{w}(\tilde{\mathbf{z}}, z) = \left(\frac{\partial \log p(\tilde{\mathbf{z}}|z)}{\partial \tilde{z}_1}, \dots, \frac{\partial \log p(\tilde{\mathbf{z}}|z)}{\partial \tilde{z}_{n(\tilde{\mathbf{Z}})}} \right)$.*
- iv **Sparse Connectivity (Minimality):** *For each parent concept \tilde{Z} , there exists a subset of its children $\mathbf{Z} \subseteq \text{Ch}(\tilde{Z})$ such that their only common parent is \tilde{Z} , i.e., $\bigcap_{Z \in \mathbf{Z}} \text{Pa}(Z) = \{\tilde{Z}\}$.*

Interpreting Condition 3.2. Condition 3.2-i ensures that the observed data \mathbf{X} (e.g., an image) fully captures the information about the latent concepts \mathbf{Z}_l . This is a natural assumption as high-dimensional observations contain rich information. Condition 3.2-ii is a standard regularity assumption for analysis. Both are common in nonlinear ICA literature [55–59, 48]. Condition 3.2-iii formalizes the idea that distinct lower-level concepts (e.g., "wheel," "door") respond in sufficiently distinct ways to changes in a shared higher-level concept (e.g., "car"), thus facilitating the identification of these lower-level concepts. Condition 3.2-iv is an instantiation of causal minimality for visual concepts. It posits that the causal graph of concepts is sparse – each concept has a somewhat unique "fingerprint" in terms of its connections. This sparsity is crucial for disentanglement [60–65] and is a less restrictive assumption than, for example, purely observed children for each latent variable [66–68]. This condition formalizes a core principle: concepts are learned through *comparison*. A concept is identifiable only if the data is rich enough to distinguish it from alternatives. For instance, if "Knight" and "Horse" always co-occur, they are learned as a single, fused concept; learning them separately requires data that breaks this correlation.

Theorem 3.3 (Visual Concept Identification). *Assume the hierarchical process for visual concepts in (2). If a model specification θ_V satisfies Condition 3.2, and an alternative specification $\hat{\theta}_V$ satisfies Conditions 3.2-i and 3.2-ii, along with a sparsity constraint such that for corresponding \hat{Z} and Z :*

$$n(\text{Pa}(\hat{Z})) \leq n(\text{Pa}(Z)), \quad (3)$$

then, if both models θ_V and $\hat{\theta}_V$ generate the same observed data distribution $\mathbb{P}(\mathbf{X})$, the latent visual concepts \mathbf{Z}_l are component-wise identifiable for every level $l \in [L_V]$.

Proof sketch for Theorem 3.3. The proof proceeds by identifying the hierarchical model level by level, from the top (most abstract concepts) \mathbf{Z}_1 downwards to \mathbf{Z}_{L_V} . 1) The paired text data \mathbf{D} acts as an auxiliary variable, providing diverse "influences" on the top-level \mathbf{Z}_1 . Condition 3.2-iii ensures these interventions have distinguishable effects. Analogous to techniques in nonlinear ICA [55, 56, 69], each component D allows the identification of the subspace of \mathbf{Z}_1 variables it influences. 2) With these subspaces identified, one can identify the intersection of these subspaces [59, 70, 71]. Therefore, if the graphical structure is sufficiently sparse, as specified in Condition 3.4-iv, one can identify the top-level latent variable \mathbf{Z}_1 component-wise. 3) Once \mathbf{Z}_1 is identified, its components can serve as the auxiliary variables for identifying the next level, \mathbf{Z}_2 . This process is repeated iteratively down the hierarchy, identifying \mathbf{Z}_l using the already identified \mathbf{Z}_{l-1} , until all L_V levels are identified.

Implications for text-to-image diffusion models. Theorem 3.3 underscores that the sparsity constraint (3) is pivotal for identifying true visual concepts. In practice, this constraint can be encouraged by techniques like SAEs applied to the internal representations (e.g., U-Net features at various timesteps) of diffusion models. By sparsifying activations, SAEs promote sparser conceptual graphs required by our theory. Our experiments in Section 4.1 demonstrate that this sparsity constraint allows for constructing the hierarchical visual concept model from frontier diffusion models and direct controls over the generating process.

3.2 Learning Textual Concepts via State Compression

We now turn to the identification of discrete textual concepts \mathbf{S} .

Condition 3.4 (Textual Concept Identification Conditions).

- i **Natural Selection:** Each selection variable S_l has a support $\text{supp } S_l$ that is a proper subset of its potential range if its constituent parts (lower-level variables) were combined randomly. That is, $\text{supp } S_l \subsetneq f_{\mathbf{D} \rightarrow S_l}(\Omega^n(\text{Pa}(S_l)))$, where $f_{\mathbf{D} \rightarrow S_l}$ is the function from \mathbf{D} to S_l .
- ii **Bottlenecks:** The support size of any concept S_l is strictly larger than the joint support size of its parents $\text{Pa}(S_l)$ in the selection graph.
- iii **Minimal Supports:** For any S , the condition distribution $\mathbb{P}(\mathbf{D} \setminus \text{Pa}(S) | S = s, \text{HPa}(S) = \tilde{s})$ is a one-to-one function w.r.t. the argument s .
- iv **No-Twins:** Distinct latent variables must have distinct sets of adjacent (parent/child) variables.
- v **Maximality:** The identified latent structure is maximal in the sense that splitting any latent concept variable would violate either the Markov conditions or the No-Twins condition.

Interpreting Condition 3.4. Condition 3.4-i posits that meaningful text (or textual concepts) occupies a small, structured subset of the vast space of all possible token combinations. We rarely encounter truly random sequences of words in natural language. Conditions 3.4-ii and 3.4-iii are direct manifestations of causal minimality for discrete concepts. ii implies an information compression moving up the hierarchy—abstract concepts are more succinct. iii demands that each state of a concept s offers unique information about the rest of the text, given its context. This forces the model to learn the most compressed (minimal number of states) representation so that each state contains unique information. Conditions 3.4-iv and 3.4-v are standard necessary conditions in discrete latent variable model identification [53, 72], precluding redundant or fragmented latent structures.

Theorem 3.5 (Textual Concept Identification). *Assume the hierarchical process as per (2). Let the true underlying parameters be $\theta_{\mathbf{T}}$. If $\theta_{\mathbf{T}}$ satisfies Condition 3.4, and an alternative learned model $\hat{\theta}_{\mathbf{T}}$ satisfies Condition 3.4-iii, then if both models produce the same observed distribution $\mathbb{P}(\mathbf{D})$, the latent textual concepts \mathbf{S}_l are component-wise identifiable for every level $l \in [L_{\mathbf{T}}]$.*

Proof sketch for Theorem 3.5. The identification for textual concepts proceeds from the bottom level (tokens, $\mathbf{S}_{L_{\mathbf{T}}}$) upwards to the most abstract concepts (\mathbf{S}_1). (1) At each level $l + 1$, we make use of the conditional independence relations that the high-level variable $S_{l,i}$ and its hybrid parents $\text{HPa}(S_{l,i})$ d-separate its pure parents $\text{PPa}(S_{l,i})$ from the other variables $\mathbf{S}_l \setminus \{\text{Pa}(S_{l,i})\}$ on level l . This relation allows us to identify subsets of \mathbf{S}_{l+1} that share children on level l [73, 50] and thus reveals the connectivity between variables in \mathbf{S}_l and \mathbf{S}_{l+1} . (2) Once the graphical connections are known, we recover the function $\text{Pa}(S_{l,i}) \mapsto S_{l,i}$ (i.e., how lower-level concepts combine to form $S_{l,i}$). This is done by merging states of $\text{Pa}(S_{l,i})$ that are predictively equivalent. The “Minimal Supports” (Condition 3.4-iii) principle dictates that we choose the function that results in the largest equivalence classes over the parent states (i.e., the most compressed representation for $S_{l,i}$). This ensures that the learned concept $\hat{S}_{l,i}$ has the minimum number of necessary states. (3) This process of structure learning and function recovery is repeated from $\mathbf{S}_{L_{\mathbf{T}}}$ (initially using observed tokens \mathbf{D} as $\mathbf{S}_{L_{\mathbf{T}}+1}$) up to \mathbf{S}_1 , thereby identifying the entire hierarchy.

Implications for autoregressive language models. Theorem 3.5 suggests that by enforcing a minimality regularization for the most compressed representation (Condition 3.4-iii), the learned internal states $\hat{\mathbf{S}}$ of a language model can become equivalent to the underlying textual concepts \mathbf{S} . SAEs, when applied to transformer activations, can be seen as a practical way to approximate this minimality. By forcing most latent units to be inactive, SAEs force the model to encode information with the minimal active units, which aligns with our theoretical condition for state compression. This result provides a principled justification for the observed interpretability of SAE-derived features and guides our empirical approach in Section 4.2 to extract hierarchical textual concept graphs.

4 Experiments

In this section, we present experimental results on both text-to-image diffusion models and autoregressive language models. Our model learns concepts at different levels of abstraction. For a complex

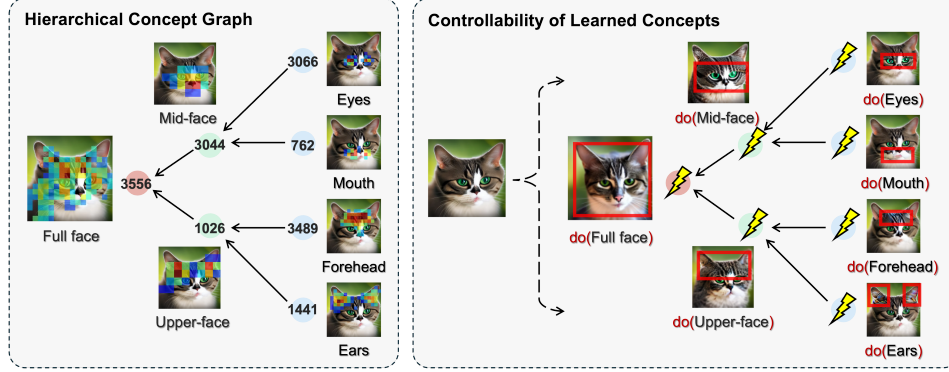


Figure 3: **Examples of hierarchical concept graphs for text-to-image models.** Our method successfully recovers meaningful hierarchical structures, where each node encodes distinct semantic concepts. On the right, we demonstrate feature steering, where manipulating individual nodes leads to changes in the output that align with their position in the hierarchy. Intervening on a high-level concept in the learned graph (“Full face”) alters the cat’s entire facial structure and fur pattern. In contrast, intervening on a learned lower-level concept (e.g., “Eye”) produces a much more localized edit, changing only the shape and color of the eyes while leaving the rest of the face intact. More examples in Appendix E.

Method	I2P ↓	RING-A-BELL ↓				P4D ↓	UATK ↓	COCO	
		K77	K38	K16	AVG			FID ↓	CLIP ↑
SD 1.4 [3]	17.8	85.26	87.37	93.68	88.10	98.70	69.70	16.71	31.3
ESD [75]	2.87	20.00	29.47	35.79	28.42	15.49	2.87	18.18	30.2
SA [76]	2.81	63.15	56.84	56.84	58.94	12.68	2.81	25.80	29.7
CA [77]	1.04	86.32	91.69	94.26	90.76	5.63	1.04	24.12	30.1
MACE [78]	1.51	2.10	0.00	0.00	0.70	2.82	1.51	16.80	28.7
UCE [79]	0.87	10.52	9.47	12.61	10.87	9.86	0.87	17.99	30.2
RECE [80]	0.72	5.26	4.21	5.26	4.91	5.63	0.72	17.74	30.2
SDID [81]	3.77	94.74	95.79	90.53	93.68	69.54	30.99	22.16	31.1
SLD-MAX [82]	1.74	23.16	32.63	42.11	32.63	9.14	2.44	28.75	28.4
SLD-STRONG [82]	2.28	56.84	64.21	61.05	60.70	33.10	3.10	24.40	29.1
SLD-MEDIUM [82]	3.95	92.63	88.42	91.05	90.70	24.00	1.98	21.17	29.8
SD1.4-NegPrompt [3]	0.74	17.89	40.42	34.74	31.68	10.00	1.46	18.33	30.1
SAFREE [83]	1.45	35.78	47.36	55.78	46.31	10.56	1.45	19.32	30.1
TRASCE [84]	0.45	1.05	2.10	2.10	1.75	3.97	0.70	17.41	29.9
ConceptSteer [22]	0.36	3.16	8.42	9.47	7.02	1.99	2.11	18.67	30.8
Ours	0.25	1.05	0.00	2.11	1.05	0.66	2.11	17.02	31.3

Table 1: **Model unlearning comparisons.** Our method delivers state-of-the-art results on unlearning tasks without compromising standard text-to-image generation. Detailed comparison in Appendix C.

object like a “textured tree stump,” we can independently steer the “stump” (a mid-level concept) and its “texture” (a low-level concept). In contrast, a non-hierarchical SAE may learn entangled features for every combination (e.g., one feature for “textured stump,” another for “mossy stump”), making fine-grained, compositional editing difficult. By identifying a concept’s hierarchical level, our model can intervene at diffusion timesteps corresponding to suitable concept levels. In contrast, a standard SAE applies steering across all timesteps, potentially introducing unwanted artifacts.

4.1 Text-to-image Diffusion Models

Implementation. We analyze the feature representations in stable diffusion (SD) 1.4. We use prompts from the Laion-COCO dataset [74]. Following Surkov et al. [20], we extract residual features from three key blocks within the UNet architecture of SD 1.4—namely, `down.2.1`, `mid.0`, and `up.1.0`. To enforce the sparsity constraint, we plug in SAEs into the pretrained model. In particular, we collect features at timesteps 899, 500, and 100. We then train K-sparse autoencoders on the features from each timestep. We consider the causal relationships among features captured at different timesteps, as they encode varying levels of semantic information. To this end, we apply a causal discovery algorithm (i.e., PC [34]) over these learned features. More details in Appendix D.

Metric	SD 1.4	SD1.4 (SAE w/o hier.)	SD1.4 (Ours)
Add tabby pattern – CLIP-I ↓	0.91 ± 0.05	0.83 ± 0.07	0.93 ± 0.04
Add tabby pattern – CLIP-T ↑	0.27 ± 0.00	0.28 ± 0.02	0.28 ± 0.01
Add mountains – CLIP-I ↓	0.84 ± 0.06	0.83 ± 0.04	0.91 ± 0.03
Add mountains – CLIP-T ↑	0.33 ± 0.01	0.32 ± 0.01	0.33 ± 0.01
Replace rock w/ stump – CLIP-I ↓	0.93 ± 0.02	0.95 ± 0.02	0.96 ± 0.02
Replace rock w/ stump – CLIP-T ↑	0.31 ± 0.01	0.29 ± 0.01	0.31 ± 0.01

Table 2: **Controllable image generation results.** Our method achieves the best CLIP-I metric, demonstrating greater fidelity to the input images, while reliably executing the target edits.

Visualization. Figure 3 illustrates a hierarchical graph learned through our approach (more in Appendix E). On the left, we display activation maps of different SAE features. The brown nodes indicate features from the SAE trained on representations extracted at timestep 899, green nodes correspond to features from timestep 500, and blue nodes represent features from timestep 100. We observe that the brown nodes capture high-level, global information about the generated image; for instance, node 3556 captures the entire face of a cat. In contrast, green nodes reflect intermediate-level features, such as node 3044, capturing the central part of the face. Blue nodes correspond to fine-grained details—node 3066 activates on the eyes, while node 762 corresponds to the mouth. This demonstrates a clear progression from coarse to fine-grained concepts across timesteps.

Qualitative results. We conduct concept steering using our discovered features, as shown on the right side of Fig. 3 (more in Appendix E). Given a model intermediate feature x , the SAE encoder E and decoder D are trained to reconstruct x . To steer a specific concept, we obtain the latent representation $z = E(x)$, and extract the steering vector v corresponding to the desired feature. We then modify the original feature to create a steered version $x' = x + \lambda D(v)$, where λ modulates the strength. By feeding the steered x' back into the diffusion process at the same timestep, we generate images that reflect the influence of the selected concept. For example, steering node 3556 – associated with the entire face of a cat – results in a significantly altered cat face. Steering the green node 1026 modifies only the upper part of the face, illustrating that it encodes localized information specific to that region.

Quantitative results. Thanks to our proposed theoretical framework, we can perform both controllable image generation and model unlearning tasks naturally. We provide quantitative results of model unlearning on four benchmark datasets: IP2P [82], three splits of RING-A-BELL [85], P4D [86], and UnlearnDiffATK [87]. These benchmarks focus on removing nudity-related concepts, and we report the accuracy of a pretrained nudity detector. Our method achieves the best results across all benchmarks. In addition, to assess whether our method preserves general text-to-image capability, we apply feature steering on normal prompts from MSCOCO [88]. The 10K results, reflected in low FID and high CLIP scores, demonstrate that our method successfully identifies and removes nudity concepts without affecting unrelated concepts. We also provide results on style removal in the appendix (Table 3) and we achieve superior performance across different metrics and tasks.

We also evaluate controllable image generation on three editing tasks: adding tabby patterns to cat faces, adding mountains to landscape images, and replacing rocks with textured tree stumps. Our method achieves superior results compared to both the standard text-guided model and SAE without hierarchical modeling. In particular, for the rock-to-stump task (as shown in Fig.4), we first apply feature steering on the mid-level node 3933 to replace the rock with a stump, and then refine the result by modifying the low-level node 4374 to add texture. This highlights the importance of hierarchical modeling for fine-grained image generation.

Ablation. As established in the theoretical framework, sparsity is crucial for identifiability. To empirically validate this, we visualize the resulting causal graphs under varying levels of sparsity, as shown in Fig. 5 (more in Appendix E). When sparsity is not enforced, the resulting graph becomes overly dense, making it difficult to interpret and diminishing its semantic clarity. Conversely, imposing excessive sparsity leads to an overly pruned graph that lacks sufficient structure to meaningfully explain the generation process, such as in the case of the cat image. These observations highlight the importance of balancing sparsity to preserve interpretability while maintaining explanatory power.



Figure 4: **Examples of controllable image generation.**

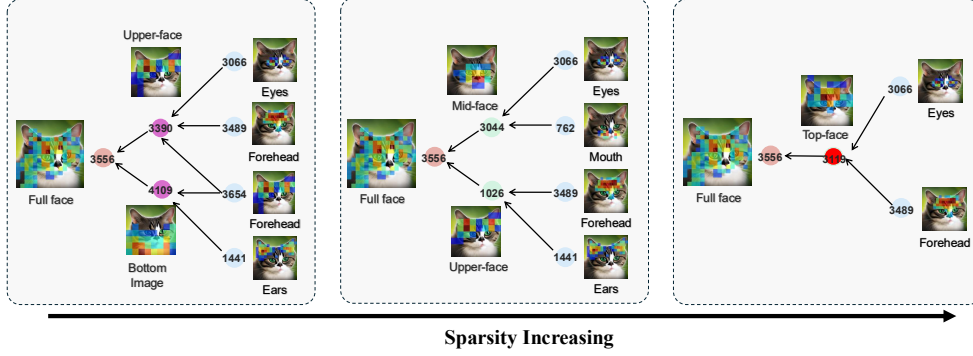


Figure 5: **Ablation studies on the sparsity constraint.** We control feature sparsity at timestep 500. Without enforcing sparsity, the resulting concepts tend to be dense, and the features are less interpretable. Conversely, increasing sparsity leads to a more interpretable, sparser graph. However, when sparsity becomes too high, the resulting graph may become overly sparse and fail to adequately capture or explain the generation of the cat face.

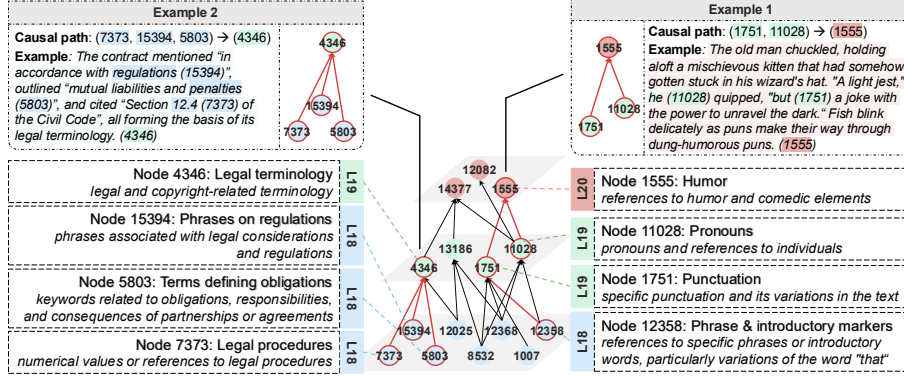


Figure 6: **The learned hierarchical concept graph for autoregressive language models.** By modeling the hierarchy of concepts based on the token sequence order, we recover a meaningful hierarchical graph. The brown nodes (corresponding to later tokens) capture global, high-level information, while the green nodes (from intermediate tokens) represent more localized, lower-level concepts.

4.2 Autoregressive Language Models

Implementation. In this section, we present our implementation for analyzing autoregressive language models. We utilize pretrained SAEs [89] for Gemma-2-2b-it [90]. We partition tokens into three parts based on their positions in the input sequence. This segmentation reflects the expectation that tokens convey increasingly abstract or high-level information as the sequence progresses. Finally, we apply causal discovery algorithms to uncover the relationships among features across the different SAEs. More details in Appendix D.

Results. Figure 6 shows a learned hierarchical graph (more in Appendix E). Nodes 1555 and 12082 correspond to features from SAEs trained on the final tokens of the sequence, and thus capture high-level semantics. Specifically, node 1555 is associated with the humorous tone, while node 12082 represents the role of the dog. Interestingly, node 11028, derived from intermediate tokens, emerges as a causal factor for both 1555 and 12082. This node encodes pronouns and references to individuals, which play a critical role in shaping both the humor and the characterization of the dog.

5 Conclusion

In this work, we present a theoretical framework using causal minimality for identifying latent concepts in hierarchical selection models. We prove that generative model representations can map to true latent variables. Empirically, applying these constraints enables extracting meaningful hierarchical concept graphs from leading models, enhancing interpretability and grounded control.

References

- [1] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the International Conference on Machine Learning (ICML 2015)*, 2015.
- [2] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, 2020.
- [3] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, 2022.
- [4] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR 2022)*, 2022.
- [5] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. In *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*, 2021.
- [6] Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *Proceedings of the International Conference on Machine Learning (ICML 2021)*, 2021.
- [7] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. *OpenAI blog*, 2018.
- [8] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [9] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.
- [10] 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(140):1–67, 2020.
- [11] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*, 2017.
- [12] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020. doi: 10.23915/distill.00024.001. <https://distill.pub/2020/circuits/zoom-in>.
- [13] Ali Jahanian, Lucy Chai, and Phillip Isola. On the "steerability" of generative adversarial networks. In *International Conference on Learning Representations*, 2019.
- [14] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable gan controls. *Advances in neural information processing systems*, 33:9841–9850, 2020.
- [15] Yujun Shen, Jinjin Gu, Xiaou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9243–9252, 2020.
- [16] Zongze Wu, Dani Lischinski, and Eli Shechtman. Stylespace analysis: Disentangled controls for stylegan image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12863–12872, 2021.
- [17] Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*, 2023.

- [18] Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*, 2023.
- [19] Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*, 2024.
- [20] Viacheslav Surkov, Chris Wendler, Mikhail Terekhov, Justin Deschenaux, Robert West, and Caglar Gulcehre. Unpacking sdxl turbo: Interpreting text-to-image models with sparse autoencoders. *arXiv preprint arXiv:2410.22366*, 2024.
- [21] Dahye Kim, Xavier Thomas, and Deepti Ghadiyaram. Revelio: Interpreting and leveraging semantic information in diffusion models. *arXiv preprint arXiv:2411.16725*, 2024.
- [22] Dahye Kim and Deepti Ghadiyaram. Concept steerers: Leveraging k-sparse autoencoders for controllable generations. *arXiv preprint arXiv:2501.19066*, 2025.
- [23] Bartosz Cywiński and Kamil Deja. Saeuron: Interpretable concept unlearning in diffusion models with sparse autoencoders. *arXiv preprint arXiv:2501.18052*, 2025.
- [24] Victor Shea-Jay Huang, Le Zhuo, Yi Xin, Zhaokai Wang, Peng Gao, and Hongsheng Li. Tide: Temporal-aware sparse autoencoders for interpretable diffusion transformers in image generation. *arXiv preprint arXiv:2503.07050*, 2025.
- [25] Jean Kaddour, Aengus Lynch, Qi Liu, Matt J Kusner, and Ricardo Silva. Causal machine learning: A survey and open problems. *arXiv preprint arXiv:2206.15475*, 2022.
- [26] Gemma E. Moran and Bryon Aragam. Towards interpretable deep generative models via causal representation learning, 2025.
- [27] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.
- [28] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.
- [29] Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, Prediction, and Search*. MIT press, 2000.
- [30] Christopher Hitchcock. Probabilistic Causation. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Spring 2021 edition, 2021.
- [31] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.
- [32] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2022)*, 2022.
- [33] Yujia Zheng, Zeyu Tang, Yiwen Qiu, Bernhard Schölkopf, and Kun Zhang. Detecting and identifying selection structure in sequential data. In *International Conference on Machine Learning*, pages 61498–61525. PMLR, 2024.
- [34] Peter Spirtes, Christopher Meek, and Thomas Richardson. Causal inference in the presence of latent variables and selection bias. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pages 499–506, 1995.
- [35] Miguel A Hernán, Sonia Hernández-Díaz, and James M Robins. A structural approach to selection bias. *Epidemiology*, 15(5):615–625, 2004.

- 432 [36] Jiji Zhang. On the completeness of orientation rules for causal discovery in the presence of
433 latent confounders and selection bias. *Artificial Intelligence*, 172(16-17):1873–1896, 2008.
- 434 [37] Elias Bareinboim, Jin Tian, and Judea Pearl. Recovering from selection bias in causal and
435 statistical inference. *Probabilistic and causal inference: The works of Judea Pearl*, pages
436 433–450, 2022.
- 437 [38] Patrick Forré and Joris M Mooij. Causal calculus in the presence of cycles, latent confounders
438 and selection bias. In *Uncertainty in Artificial Intelligence*, pages 71–80. PMLR, 2020.
- 439 [39] Juan D Correa, Jin Tian, and Elias Bareinboim. Identification of causal effects in the presence
440 of selection bias. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33,
441 pages 2744–2751, 2019.
- 442 [40] Leihao Chen, Onno Zoeter, and Joris M Mooij. Modeling latent selection with structural
443 causal models. *arXiv preprint arXiv:2401.06925*, 2024.
- 444 [41] Judea Pearl. *Causality*. Cambridge university press, 2009.
- 445 [42] Myung Jin Choi, Vincent YF Tan, Animashree Anandkumar, and Alan S Willsky. Learning
446 latent tree graphical models. *Journal of Machine Learning Research*, 12:1771–1812, 2011.
- 447 [43] Nevin L Zhang. Hierarchical latent class models for cluster analysis. *The Journal of Machine
448 Learning Research*, 5:697–723, 2004.
- 449 [44] Feng Xie, Biwei Huang, Zhengming Chen, Yangbo He, Zhi Geng, and Kun Zhang. Identifi-
450 cation of linear non-gaussian latent hierarchical structure. In *International Conference on
451 Machine Learning*, pages 24370–24387. PMLR, 2022.
- 452 [45] Biwei Huang, Charles Jia Han Low, Feng Xie, Clark Glymour, and Kun Zhang. Latent
453 hierarchical causal structure discovery with rank constraints. *Advances in Neural Information
454 Processing Systems*, 35:5549–5561, 2022.
- 455 [46] Xinshuai Dong, Biwei Huang, Ignavier Ng, Xiangchen Song, Yujia Zheng, Songyao Jin,
456 Roberto Legaspi, Peter Spirtes, and Kun Zhang. A versatile causal discovery framework to
457 allow causally-related hidden variables. In *The Twelfth International Conference on Learning
458 Representations*, 2023.
- 459 [47] Animashree Anandkumar, Daniel Hsu, Adel Javanmard, and Sham Kakade. Learning linear
460 bayesian networks with latent variables. In *International Conference on Machine Learning*,
461 pages 249–257. PMLR, 2013.
- 462 [48] Lingjing Kong, Biwei Huang, Feng Xie, Eric Xing, Yuejie Chi, and Kun Zhang. Identification
463 of nonlinear latent hierarchical models. *Advances in Neural Information Processing Systems*,
464 36, 2023.
- 465 [49] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*.
466 Morgan Kaufmann, 1988.
- 467 [50] Lingjing Kong, Guangyi Chen, Biwei Huang, Eric P. Xing, Yuejie Chi, and Kun Zhang. Learn-
468 ing discrete concepts in latent hierarchical models. In *The Thirty-eighth Annual Conference
469 on Neural Information Processing Systems*, 2024. URL [https://openreview.net/forum?
470 id=b05bUxvH6m](https://openreview.net/forum?id=b05bUxvH6m).
- 471 [51] Yuqi Gu and David B. Dunson. Bayesian pyramids: Identifiable multilayer discrete latent
472 structure models for discrete data. In *Journal of the Royal Statistical Society Series B:
473 Statistical Methodology*, 2023.
- 474 [52] Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT
475 press, 2001.
- 476 [53] Bohdan Kivva, Goutham Rajendran, Pradeep Ravikumar, and Bryon Aragam. Learning latent
477 causal graphs via mixture oracles. *Advances in Neural Information Processing Systems*, 34:
478 18087–18101, 2021.

- [54] Feng Xie, Ruichu Cai, Biwei Huang, Clark Glymour, Zhifeng Hao, and Kun Zhang. Generalized independent noise condition for estimating latent variable causal graphs. *Advances in Neural Information Processing Systems*, 33:14891–14902, 2020.
- [55] Aapo Hyvarinen and Hiroshi Morioka. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. *Advances in neural information processing systems*, 29, 2016.
- [56] Aapo Hyvarinen, Hiroaki Sasaki, and Richard Turner. Nonlinear ica using auxiliary variables and generalized contrastive learning. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 859–868. PMLR, 2019.
- [57] Ilyes Khemakhem, Ricardo Monti, Diederik Kingma, and Aapo Hyvarinen. Ice-beem: Identifiable conditional energy-based deep models based on nonlinear ica. *Advances in Neural Information Processing Systems*, 33:12768–12778, 2020.
- [58] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pages 2207–2217. PMLR, 2020.
- [59] Julius Von Kügelgen, Yash Sharma, Luigi Gresele, Wieland Brendel, Bernhard Schölkopf, Michel Besserve, and Francesco Locatello. Self-supervised learning with data augmentations provably isolates content from style. *Advances in neural information processing systems*, 34:16451–16467, 2021.
- [60] Yujia Zheng, Ignavier Ng, and Kun Zhang. On the identifiability of nonlinear ica: Sparsity and beyond. *arXiv preprint arXiv:2206.07751*, 2022.
- [61] Yujia Zheng and Kun Zhang. Generalizing nonlinear ica beyond structural sparsity. *Advances in Neural Information Processing Systems*, 36:13326–13355, 2023.
- [62] Sébastien Lachapelle, Pau Rodríguez López, Yash Sharma, Katie Everett, Rémi Le Priol, Alexandre Lacoste, and Simon Lacoste-Julien. Nonparametric partial disentanglement via mechanism sparsity: Sparse actions, interventions and sparse temporal dependencies. *arXiv preprint arXiv:2401.04890*, 2024.
- [63] Danru Xu, Dingling Yao, Sébastien Lachapelle, Perouz Taslakian, Julius Von Kügelgen, Francesco Locatello, and Sara Magliacane. A sparsity principle for partially observable causal representation learning. *arXiv preprint arXiv:2403.08335*, 2024.
- [64] Sébastien Lachapelle, Pau Rodriguez, Yash Sharma, Katie E Everett, Rémi Le Priol, Alexandre Lacoste, and Simon Lacoste-Julien. Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ica. In *Conference on Causal Learning and Reasoning*, pages 428–484. PMLR, 2022.
- [65] Sébastien Lachapelle, Tristan Deleu, Divyat Mahajan, Ioannis Mitliagkas, Yoshua Bengio, Simon Lacoste-Julien, and Quentin Bertrand. Synergies between disentanglement and sparsity: Generalization and identifiability in multi-task learning. *arXiv preprint arXiv:2211.14666*, 2022.
- [66] Sanjeev Arora, Rong Ge, and Ankur Moitra. Learning topic models—going beyond svd. In *2012 IEEE 53rd annual symposium on foundations of computer science*, pages 1–10. IEEE, 2012.
- [67] Sanjeev Arora, Rong Ge, Yonatan Halpern, David Mimno, Ankur Moitra, David Sontag, Yichen Wu, and Michael Zhu. A practical algorithm for topic modeling with provable guarantees. In *International conference on machine learning*, pages 280–288. PMLR, 2013.
- [68] Gemma E Moran, Dhanya Sridhar, Yixin Wang, and David M Blei. Identifiable variational autoencoders via sparse decoding. *arXiv preprint arXiv:2110.10804*, 2021.
- [69] Lingjing Kong, Shaoan Xie, Weiran Yao, Yujia Zheng, Guangyi Chen, Petar Stojanov, Victor Akinwande, and Kun Zhang. Partial disentanglement for domain adaptation. In *International Conference on Machine Learning*, pages 11455–11472. PMLR, 2022.

- [70] Dingling Yao, Danru Xu, Sebastien Lachapelle, Sara Magliacane, Perouz Taslakian, Georg Martius, Julius von Kügelgen, and Francesco Locatello. Multi-view causal representation learning with partial observability. In *The Twelfth International Conference on Learning Representations*, 2023.
- [71] Lingjing Kong, Martin Q. Ma, Guangyi Chen, Eric P. Xing, Yuejie Chi, Louis-Philippe Morency, and Kun Zhang. Understanding masked autoencoders via hierarchical latent variable models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7918–7928, June 2023.
- [72] Bohdan Kivva, Goutham Rajendran, Pradeep Ravikumar, and Bryon Aragam. Identifiability of deep generative models without auxiliary information. *Advances in Neural Information Processing Systems*, 35:15687–15701, 2022.
- [73] Joel E Cohen and Uriel G Rothblum. Nonnegative ranks, decompositions, and factorizations of nonnegative matrices. *Linear Algebra and its Applications*, 190:149–168, 1993.
- [74] Christoph Schuhmann, Andreas Köpf, Theo Coombes, Richard Vencu, Romain Beaumont, and Benjamin Trom. Laion-coco: 600m synthetic captions from laion2b-en. LAION.ai blog, September 2022. URL <https://laion.ai/blog/laion-coco/>.
- [75] Rohit Gandikota, Joanna Materzynska, Jaden Fiotto-Kaufman, and David Bau. Erasing concepts from diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2426–2436, 2023.
- [76] Alvin Heng and Harold Soh. Selective amnesia: A continual learning approach to forgetting in deep generative models. *Advances in Neural Information Processing Systems*, 36:17170–17194, 2023.
- [77] Nupur Kumari, Bingliang Zhang, Sheng-Yu Wang, Eli Shechtman, Richard Zhang, and Jun-Yan Zhu. Ablating concepts in text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22691–22702, 2023.
- [78] Shilin Lu, Zilan Wang, Leyang Li, Yanzhu Liu, and Adams Wai-Kin Kong. Mace: Mass concept erasure in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6430–6440, 2024.
- [79] Rohit Gandikota, Hadas Orgad, Yonatan Belinkov, Joanna Materzyńska, and David Bau. Unified concept editing in diffusion models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5111–5120, 2024.
- [80] Chao Gong, Kai Chen, Zhipeng Wei, Jingjing Chen, and Yu-Gang Jiang. Reliable and efficient concept erasure of text-to-image diffusion models. In *European Conference on Computer Vision*, pages 73–88. Springer, 2024.
- [81] Hang Li, Chengzhi Shen, Philip Torr, Volker Tresp, and Jindong Gu. Self-discovering interpretable diffusion latent directions for responsible text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12006–12016, 2024.
- [82] Patrick Schramowski, Manuel Brack, Björn Deiseroth, and Kristian Kersting. Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22522–22531, 2023.
- [83] Jaehong Yoon, Shoubin Yu, Vaidehi Patil, Huaxiu Yao, and Mohit Bansal. Safree: Training-free and adaptive guard for safe text-to-image and video generation. *arXiv preprint arXiv:2410.12761*, 2024.
- [84] Anubhav Jain, Yuya Kobayashi, Takashi Shibuya, Yuhta Takida, Nasir Memon, Julian Togelius, and Yuki Mitsufuji. Trasce: Trajectory steering for concept erasure. *arXiv preprint arXiv:2412.07658*, 2024.

- [85] Yu-Lin Tsai, Chia-Yi Hsu, Chulin Xie, Chih-Hsun Lin, Jia-You Chen, Bo Li, Pin-Yu Chen, Chia-Mu Yu, and Chun-Ying Huang. Ring-a-bell! how reliable are concept removal methods for diffusion models? *arXiv preprint arXiv:2310.10012*, 2023.
- [86] Zhi-Yi Chin, Chieh-Ming Jiang, Ching-Chun Huang, Pin-Yu Chen, and Wei-Chen Chiu. Prompting4debugging: Red-teaming text-to-image diffusion models by finding problematic prompts. *arXiv preprint arXiv:2309.06135*, 2023.
- [87] Yimeng Zhang, Jinghan Jia, Xin Chen, Aochuan Chen, Yihua Zhang, Jiancheng Liu, Ke Ding, and Sijia Liu. To generate or not? safety-driven unlearned diffusion models are still easy to generate unsafe images... for now. In *European Conference on Computer Vision*, pages 385–403. Springer, 2024.
- [88] 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 *European conference on computer vision*, pages 740–755. Springer, 2014.
- [89] Joseph Bloom, Curt Tigges, Anthony Duong, and David Chanin. Saelens. <https://github.com/jbloomAus/SAELens>, 2024.
- [90] Gemma Team. Gemma. *Kaggle*, 2024. doi: 10.34740/KAGGLE/M/3301. URL <https://www.kaggle.com/m/3301>.
- [91] Kiho Park, Yo Joong Choe, Yibo Jiang, and Victor Veitch. The geometry of categorical and hierarchical concepts in large language models. *arXiv preprint arXiv:2406.01506*, 2024.
- [92] Kun Zhang, Shaoan Xie, Ignavier Ng, and Yujia Zheng. Causal representation learning from multiple distributions: A general setting. In *Forty-first International Conference on Machine Learning*, 2024. URL <https://openreview.net/forum?id=Pte6iiXvpf>.
- [93] Patrik Reizinger, Alice Bizeul, Attila Juhos, Julia E Vogt, Randall Balestriero, Wieland Brendel, and David Klindt. Cross-entropy is all you need to invert the data generating process. *arXiv preprint arXiv:2410.21869*, 2024.
- [94] Yuhang Liu, Dong Gong, Erdun Gao, Zhen Zhang, Biwei Huang, Mingming Gong, Anton van den Hengel, and Javen Qinfeng Shi. I predict therefore i am: Is next token prediction enough to learn human-interpretable concepts from data? *arXiv preprint arXiv:2503.08980*, 2025.
- [95] Emanuele Marconato, Sébastien Lachapelle, Sebastian Weichwald, and Luigi Gresele. All or none: Identifiable linear properties of next-token predictors in language modeling. *arXiv preprint arXiv:2410.23501*, 2024.
- [96] Goutham Rajendran, Simon Buchholz, Bryon Aragam, Bernhard Schölkopf, and Pradeep Kumar Ravikumar. From causal to concept-based representation learning. In *Causality and Large Models @NeurIPS 2024*, 2024. URL <https://openreview.net/forum?id=FcVnIBYbkW>.
- [97] Yibo Jiang, Goutham Rajendran, Pradeep Kumar Ravikumar, Bryon Aragam, and Victor Veitch. On the origins of linear representations in large language models. In *Forty-first International Conference on Machine Learning*, 2024. URL <https://openreview.net/forum?id=otuTw4Mghk>.
- [98] Jack Brady, Roland S Zimmermann, Yash Sharma, Bernhard Schölkopf, Julius Von Kügelgen, and Wieland Brendel. Provably learning object-centric representations. In *International Conference on Machine Learning*, pages 3038–3062. PMLR, 2023.
- [99] Sébastien Lachapelle, Divyat Mahajan, Ioannis Mitliagkas, and Simon Lacoste-Julien. Additive decoders for latent variables identification and cartesian-product extrapolation. *Advances in Neural Information Processing Systems*, 36, 2024.
- [100] Shruti Joshi, Andrea Dittadi, Sébastien Lachapelle, and Dhanya Sridhar. Identifiable steering via sparse autoencoding of multi-concept shifts. *arXiv preprint arXiv:2502.12179*, 2025.

- 622 [101] Andrey Voynov and Artem Babenko. Unsupervised discovery of interpretable directions in the
623 gan latent space. In *International conference on machine learning*, pages 9786–9796. PMLR,
624 2020.
- 625 [102] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. Interfacegan: Interpreting the
626 disentangled face representation learned by gans. *IEEE transactions on pattern analysis and*
627 *machine intelligence*, 44(4):2004–2018, 2020.
- 628 [103] Anish Mudide, Joshua Engels, Eric J Michaud, Max Tegmark, and Christian Schroeder de Witt.
629 Efficient dictionary learning with switch sparse autoencoders. In *The Thirteenth International*
630 *Conference on Learning Representations*, 2025. URL [https://openreview.net/forum?](https://openreview.net/forum?id=k2ZVAzVeMP)
631 [id=k2ZVAzVeMP](https://openreview.net/forum?id=k2ZVAzVeMP).
- 632 [104] Wei Shi, Sihang Li, Tao Liang, Mingyang Wan, Gojun Ma, Xiang Wang, and Xiangnan He.
633 Route sparse autoencoder to interpret large language models. *CoRR*, abs/2503.08200, March
634 2025. URL <https://doi.org/10.48550/arXiv.2503.08200>.
- 635 [105] Claude E Shannon. A mathematical theory of communication. *The Bell system technical*
636 *journal*, 27(3):379–423, 1948.
- 637 [106] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.
- 638 [107] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-
639 solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint*
640 *arXiv:2211.01095*, 2022.
- 641 [108] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason
642 Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse
643 text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Appendix for “Beyond the Black Box: Identifiable Interpretation and Control in Generative Models via Causal Minimality”

A Related Work

Hierarchical models. Complex real-world data distributions frequently exhibit inherent hierarchical structures among their underlying latent variables, a characteristic that has motivated extensive research. Initial explorations primarily focus on continuous latent variables with linear interactions [44–47]. Other lines of work have centered on discrete latent variables; however, these approaches are often constrained in their applicability to continuous data modalities like images [49, 43, 42, 51, 50]. Furthermore, prevalent latent tree models, which connect variables via a single undirected path [49, 43, 42], risk oversimplifying the multifaceted relationships present in complex systems. More recently, while Park et al. [91] make progress in capturing geometric properties of language model representations using hierarchical models, their work does not address the critical issue of latent variable identification. Kong et al. [48] tackle nonlinear, continuous latent hierarchical models, but their framework, operating under rather opaque functional conditions, falls short of component-wise identifiability, thereby leaving room for concept entanglement. Our work distinctively investigates *selection* hierarchical models, contending that their structural properties yield a more faithful representation of latent concepts in natural data distributions. In these models, latent variables function as colliders, a significant departure from their role as confounders in the aforementioned prior art. This critical distinction renders existing identification techniques largely inapplicable. We address this gap by leveraging the minimality principle alongside other transparent conditions. To the best of our knowledge, we are the first to provide *component-wise* identifiability for *both continuous and discrete hierarchical selection models*.

Latent variable identification. Identifying latent variables is a cornerstone of representation learning. A significant body of work establishes identifiability for single-level latent variable models, often assuming the availability of auxiliary information like domain or class labels [58, 57, 55, 56, 92]. Recently, research into language models has explored the linear representation hypothesis, yielding linear-subspace identifiability for latent variables [93–97]. Another research direction [98, 99, 62, 63, 65, 64, 60, 100] leverages sparsity for identification but overlooks the causal relationships among latent variables. Distinct from these approaches, our work formulates the concept space using *hierarchical* models that allow for the explicit modeling of intricate, multi-level conceptual interactions.

Interpretability for generative models. Despite the remarkable advancements of generative models, their internal mechanisms often remain opaque. This presents a significant challenge to understanding and control. Considerable research has focused on obtaining interpretable features to enable more controllable generation. Early efforts center on analyzing the latent space of generative adversarial networks, e.g., [14, 101, 102]. Recently, sparse autoencoders (SAEs) have gained prominence for interpreting hidden representations, particularly in language models. These studies show that SAEs trained on transformer residual-stream activations can identify latent units corresponding to linguistically meaningful features [17–19, 103, 104]. These interpretability techniques have also been successfully extended to diffusion models. Surkov et al. [20] reveal interpretable features and specialization across diffusion model blocks. Other work includes training SAEs with lightweight classifiers on diffusion model features [21] and steering generation away from undesirable visual attributes [24]. Distinctly, our work employs a unifying *minimality principle* to enhance the interpretability of both vision and language generative models. Our theory provides a theoretical basis for latent concept identification, complementing sparsity-based techniques, and allows for extracting hierarchical concept graphs, offering new insights into the representational structures of black-box generative models.

B Proofs

B.1 Proof for Theorem 3.3

Lemma B.1 (Base Case Visual Concept Identification). *Assume the following data-generating process:*

$$\mathbf{C} \sim \mathbb{P}(\mathbf{C}|\mathbf{U}), \mathbf{V} \sim \mathbb{P}(\mathbf{C}), \mathbf{X} := g(\mathbf{C}, \mathbf{V}). \quad (4)$$

We have the following conditions.

i **Informativeness**: The function $g(\cdot)$ is a diffeomorphism.

ii **Smooth Density**: The probability density function $p(\mathbf{c}, \mathbf{v}|\mathbf{u})$ is smooth.

iii **Sufficient Variability**: At any value \mathbf{c} of \mathbf{C} , there exist $n(\mathbf{C}) + 1$ distinct values of \mathbf{U} , denoted as $\{\mathbf{u}^{(n)}\}_{n=0}^{n(\mathbf{C})}$, such that the vectors $\mathbf{w}(\mathbf{c}, \mathbf{u}^n) - \mathbf{w}(\mathbf{c}, \mathbf{u}^0)$ are linearly independent where $\mathbf{w}(\mathbf{c}, \mathbf{u}) = \left(\frac{\partial \log p(\mathbf{c}|\mathbf{u})}{\partial c_1}, \dots, \frac{\partial \log p(\mathbf{c}|\mathbf{u})}{\partial c_{n(\mathbf{c})}} \right)$.

If a specification θ satisfies i,ii, and iii, another specification $\hat{\theta}$ satisfies i,ii, and they generate matching distribution $\mathbb{P}(\mathbf{X})$, then we can verify that \mathbf{C} and $\hat{\mathbf{C}}$ can be identified component-wise.

Proof. Since we have matched distribution, it follows that:

$$p(\mathbf{x}|\mathbf{u}) = \hat{p}(\mathbf{x}|\mathbf{u}). \quad (5)$$

As the generating function g has a smooth inverse (i), we can derive:

$$\begin{aligned} p(g(\mathbf{c}, \mathbf{v})|\mathbf{u}) &= p(\hat{g}(\hat{\mathbf{c}}, \hat{\mathbf{v}})|\mathbf{u}) \implies \\ p(\mathbf{c}, \mathbf{v}|\mathbf{u}) | \mathbf{J}_{g^{-1}} | &= \hat{p}(g^{-1} \circ \hat{g}(\hat{\mathbf{c}}, \hat{\mathbf{v}})|\mathbf{u}) | \mathbf{J}_{g^{-1}} |. \end{aligned}$$

Notice that the Jacobian determinant $|\mathbf{J}_{g^{-1}}| > 0$ because of $g(\cdot)$'s invertibility and let $h := g^{-1} \circ \hat{g} : (\hat{\mathbf{c}}, \hat{\mathbf{v}}) \mapsto (\mathbf{c}, \mathbf{v})$ which is smooth and has a smooth inverse thanks to those properties of g and \hat{g} . It follows that

$$\begin{aligned} p(\mathbf{c}, \mathbf{v}|\mathbf{u}) &= \hat{p}(h(\hat{\mathbf{c}}, \hat{\mathbf{v}})|\mathbf{u}) \implies \\ p(\mathbf{c}, \mathbf{v}|\mathbf{u}) &= \hat{p}(\hat{\mathbf{c}}, \hat{\mathbf{v}}|\mathbf{u}) | \mathbf{J}_{h^{-1}} |. \end{aligned}$$

The independence relation in the generating process implies that

$$\log p(\mathbf{c}|\mathbf{u}) + \sum_{i \in [n(\mathbf{v})]} \log p(V_i) = \log \hat{p}(\hat{\mathbf{c}}|\mathbf{u}) + \sum_{i \in [n(\hat{\mathbf{v}})]} \log \hat{p}(\hat{V}_i) + \log |\mathbf{J}_{h^{-1}}|. \quad (6)$$

For any realization \mathbf{u}^0 , we subtract (6) at any $\mathbf{u} \neq \mathbf{u}^0$ with that at \mathbf{u}^0 :

$$\log p(\mathbf{c}|\mathbf{u}) - \log p(\mathbf{c}|\mathbf{u}^0) = \log \hat{p}(\hat{\mathbf{c}}|\mathbf{u}) - \log \hat{p}(\hat{\mathbf{c}}|\mathbf{u}^0). \quad (7)$$

Taking derivative w.r.t. \hat{v}_j for $j \in [n(\hat{\mathbf{v}})]$ yields:

$$\sum_{i \in [n(\mathbf{c})]} \frac{\partial}{\partial c_i} (\log p(\mathbf{c}|\mathbf{u}) - \log p(\mathbf{c}|\mathbf{u}^0)) \cdot \frac{\partial c_i}{\partial \hat{v}_j} = 0. \quad (8)$$

The left-hand side zeros out because $\hat{\mathbf{c}}$ is not a function of $\hat{\mathbf{v}}$.

Condition iii ensures the existence of at least $n(\mathbf{c})$ such equations with $\mathbf{u}^1, \dots, \mathbf{u}^{n(\mathbf{c})}$ that are linearly independent, constituting a full-rank linear system. Since the choice of $j \in [n(\hat{\mathbf{v}})]$ is arbitrary. It follows that

$$\frac{\partial c_i}{\partial \hat{v}_j} = 0, \forall i \in [n(\mathbf{c})], j \in [n(\hat{\mathbf{v}})]. \quad (9)$$

Therefore, the Jacobian matrix \mathbf{J}_h is of the following structure:

$$\mathbf{J}_h = \begin{bmatrix} \frac{\partial \mathbf{v}}{\partial \hat{\mathbf{v}}} & \frac{\partial \mathbf{v}}{\partial \hat{\mathbf{c}}} \\ \frac{\partial \mathbf{c}}{\partial \hat{\mathbf{v}}} & \frac{\partial \mathbf{c}}{\partial \hat{\mathbf{c}}} \end{bmatrix} \quad (10)$$

(9) suggests that the block $\frac{\partial \mathbf{c}}{\partial \hat{\mathbf{v}}} = 0$. Since \mathbf{J}_h is full-rank, we can deduce that $\frac{\partial \mathbf{c}}{\partial \hat{\mathbf{c}}}$ must have full row-rank and $n(\mathbf{c}) \leq n(\hat{\mathbf{c}})$. The sparsity constraint in (3) further implies that $n(\mathbf{c}) = n(\hat{\mathbf{c}})$. That is, we can correctly identify the dimensionality of the changing subspace \mathbf{c} . Moreover, since \mathbf{J}_h is full-rank and the block $\frac{\partial \mathbf{c}}{\partial \hat{\mathbf{v}}}$ is zero, we can derive that the corresponding block $\frac{\partial \hat{\mathbf{c}}}{\partial \hat{\mathbf{v}}}$ in its inverse matrix \mathbf{J}_{h-1} is also zero. Therefore, there exists an invertible map $\hat{\mathbf{c}} \mapsto \mathbf{c}$, which concludes the proof. \square

Lemma B.2 (Determining Intersection Cardinality from Union Cardinalities). *Let $\mathcal{A} = \{A_1, A_2, \dots, A_n\}$ be a finite collection of finite sets. If for any non-empty subset of indices $K \subseteq \{1, 2, \dots, n\}$, the cardinality of the union $|\bigcup_{k \in K} A_k|$ is known, then for any non-empty subset of indices $S \subseteq \{1, 2, \dots, n\}$, the cardinality of the intersection $|\bigcap_{s \in S} A_s|$ can be determined.*

Proof. We proceed by induction on the size of the set of indices S , denoted by $|S|$, for which we want to determine the intersection cardinality.

Base Case: $|S| = 1$. Let $S = \{i\}$ for some $i \in \{1, 2, \dots, n\}$. We aim to determine the cardinality $|\bigcap_{s \in S} A_s| = |A_i|$. The union of a single set A_i is simply A_i itself. That is, $A_i = \bigcup_{k \in \{i\}} A_k$. By the premise of the theorem, the cardinality $|\bigcup_{k \in \{i\}} A_k|$ is known. Therefore, $|A_i|$ is known. The base case holds.

Inductive Hypothesis: Assume that for some integer $m \geq 1$, the cardinality of any intersection of j sets, $|\bigcap_{j \in J} A_j|$, can be determined from the known union cardinalities for all non-empty index sets J such that $1 \leq |J| \leq m$.

Inductive Step: We want to show that the cardinality of any intersection of $m + 1$ sets can be determined. Let S_{m+1} be an arbitrary non-empty subset of indices from $\{1, 2, \dots, n\}$ such that $|S_{m+1}| = m + 1$. Our goal is to determine $|\bigcap_{s \in S_{m+1}} A_s|$.

Consider the Principle of Inclusion-Exclusion (PIE) applied to the union of the sets whose indices are in S_{m+1} :

$$\left| \bigcup_{s \in S_{m+1}} A_s \right| = \sum_{\emptyset \neq K \subseteq S_{m+1}} (-1)^{|K|-1} \left| \bigcap_{k \in K} A_k \right|$$

This sum runs over all non-empty subsets K of S_{m+1} . We can separate the term where $K = S_{m+1}$ (which corresponds to the intersection of all $m + 1$ sets) from the other terms in the sum:

$$\left| \bigcup_{s \in S_{m+1}} A_s \right| = \left(\sum_{\emptyset \neq K \subset S_{m+1}} (-1)^{|K|-1} \left| \bigcap_{k \in K} A_k \right| \right) + (-1)^{|S_{m+1}|-1} \left| \bigcap_{s \in S_{m+1}} A_s \right|$$

Here, the sum is now over all non-empty *proper* subsets K of S_{m+1} . We can rearrange this equation to solve for the term $|\bigcap_{s \in S_{m+1}} A_s|$:

$$(-1)^{|S_{m+1}|-1} \left| \bigcap_{s \in S_{m+1}} A_s \right| = \left| \bigcup_{s \in S_{m+1}} A_s \right| - \sum_{\emptyset \neq K \subset S_{m+1}} (-1)^{|K|-1} \left| \bigcap_{k \in K} A_k \right|$$

Multiplying both sides by $(-1)^{|S_{m+1}|-1}$ (noting that $((-1)^{|S_{m+1}|-1})^2 = 1$):

$$\left| \bigcap_{s \in S_{m+1}} A_s \right| = (-1)^{|S_{m+1}|-1} \left(\left| \bigcup_{s \in S_{m+1}} A_s \right| - \sum_{\emptyset \neq K \subset S_{m+1}} (-1)^{|K|-1} \left| \bigcap_{k \in K} A_k \right| \right)$$

Let us analyze the terms on the right-hand side of this equation:

1. The factor $(-1)^{|S_{m+1}|-1}$ is a known sign, since $|S_{m+1}| = m + 1$.
2. The term $|\bigcup_{s \in S_{m+1}} A_s|$ is the cardinality of a union of $m + 1$ sets. Since S_{m+1} is a non-empty subset of indices, this value is known by the premise of the theorem.

741 3. Consider the sum $\sum_{\emptyset \neq K \subseteq S_{m+1}} (-1)^{|K|-1} |\bigcap_{k \in K} A_k|$. Each K in this summation is a
742 non-empty proper subset of S_{m+1} . Therefore, the size of each such K satisfies $1 \leq$
743 $|K| \leq m$. By the Inductive Hypothesis, for any such K (i.e., for any intersection of j
744 sets where $1 \leq j \leq m$), the cardinality $|\bigcap_{k \in K} A_k|$ can be determined from the known
745 union cardinalities. Consequently, every term in this summation, including its sign factor
746 $(-1)^{|K|-1}$, is determinable.

747 Since all components on the right-hand side of the equation are known or can be determined based on
748 the theorem's premise and the inductive hypothesis, the value of $|\bigcap_{s \in S_{m+1}} A_s|$ can be determined.

749 In conclusion, by the principle of mathematical induction, for any non-empty subset of indices
750 $S \subseteq \{1, 2, \dots, n\}$, the cardinality of the intersection $|\bigcap_{s \in S} A_s|$ can be determined if the cardinality
751 of any union $|\bigcup_{k \in K} A_k|$ (for any non-empty $K \subseteq \{1, 2, \dots, n\}$) is known. \square

752 **Lemma B.3** (Intersection Block Identification [71]). *We assume the following data-generating*
753 *process:*

$$[\mathbf{v}_1, \mathbf{v}_2] = g(\mathbf{c}, \mathbf{s}_1, \mathbf{s}_2), \quad (11)$$

$$\mathbf{v}_1 = g_1(\mathbf{c}, \mathbf{s}_1), \quad (12)$$

$$\mathbf{v}_2 = g_2(\mathbf{c}, \mathbf{s}_2), \quad (13)$$

754 where $\mathbf{c} \in \mathcal{C} \subset \mathbb{R}^{d_c}$, $\mathbf{s}_1 \in \mathcal{S} \subset \mathbb{R}^{d_{s_1}}$, and $\mathbf{s}_2 \in \mathcal{S}_2 \subset \mathbb{R}^{d_{s_2}}$. Both g_1 and g_2 are smooth and have non-
755 singular Jacobian matrices almost anywhere, and g is invertible. If $\hat{g}_1 : \mathcal{Z} \rightarrow \mathcal{V}_1$ and $\hat{g}_2 : \mathcal{Z} \rightarrow \mathcal{V}_2$
756 assume the generating process of the true model (g_1, g_2) and match the joint distribution $p_{\mathbf{v}_1, \mathbf{v}_2}$, then
757 there is a one-to-one mapping between the estimate $\hat{\mathbf{c}}$ and the ground truth \mathbf{c} over $\mathcal{C} \times \mathcal{S} \times \mathcal{S}$, that is,
758 \mathbf{c} is block-identifiable.

759 **Lemma B.4** (One-level Visual Concept Identification). *Assume the process for visual concepts in (2)*
760 *with $L_V = 1$. If a model specification θ_V satisfies Condition 3.2, and an alternative specification $\hat{\theta}_V$*
761 *satisfies Conditions 3.2-i and 3.2-ii, along with a sparsity constraint such that for corresponding \hat{Z}*
762 *and Z :*

$$n(\text{Pa}(\hat{Z})) \leq n(\text{Pa}(Z)), \quad (14)$$

763 then, if both models θ_V and $\hat{\theta}_V$ generate the same observed data distribution $\mathbb{P}(\mathbf{X})$, the latent visual
764 concepts \mathbf{Z}_1 are component-wise identifiable for every level.

765 *Proof.* For notational convenience, we denote \mathbf{Z}_1 as \mathbf{S} and \mathbf{D} as \mathbf{U} in this proof. This proof consists
766 of two steps. In step one, we identify the connectivity between U and S variables. In step two, we
767 further show the identifiability of the blocks resulting from intersecting the parent sets $\text{Pa}(U)$ of
768 multiple U variables.

769 **Step 1: connectivity identification.** Since we have access to the joint distribution $\mathbb{P}(\mathbf{S}, \mathbf{U})$, we can
770 derive conditional distributions $\mathbb{P}(\mathbf{S} | \{U_i\}_{i \in \mathcal{H}})$ for any index subset $\mathcal{H} \subseteq [n(\mathbf{U})]$. By Lemma B.1,
771 we can identify the dimensionality of the set of variables \mathbf{S} that are connected to *any* variable in
772 $\{U_i\}_{i \in \mathcal{H}}$ for any $\mathcal{H} \subseteq [n(\mathbf{U})]$. Lemma B.2 implies that we can identify the dimensionality of the set
773 of variables \mathbf{S} that are connected to *all* variables in $\{U_i\}_{i \in \mathcal{H}}$ for any $\mathcal{H} \subseteq [n(\mathbf{U})]$. This information
774 gives rise to a partition of S components, in which each part is connected to the same set of U
775 variables. Therefore, we have identified the bipartite graph between \mathbf{S} and \mathbf{U} up to a permutation.

776 **Step 2: intersection block identification.** Denote the indices of S variables that are connected to U_i
777 as $\mathcal{I}(i) \subseteq [n(\mathbf{S})]$. We denote the block of S components connected to *all* variables in $\{U_i\}_{i \in \mathcal{H}}$ as
778 $\mathbf{S}_{\cap_{i \in \mathcal{H}} \mathcal{I}(i)}$ for any $\mathcal{H} \subseteq [n(\mathbf{U})]$. Thanks to Lemma B.1, we can identify the block $\mathbf{S}_{\mathcal{I}(i)}$ connected
779 to the variable U_i for any $i \in [n(\mathbf{U})]$. Lemma B.3 allows us to identify the intersection of any two
780 blocks $\mathbf{S}_{\mathcal{I}(i) \cap \mathcal{I}(j)}$ for $i \neq j$. Therefore, repeated applications of Lemma B.3 leads to the identification
781 of the intersection block $\mathbf{S}_{\cap_{i \in \mathcal{H}} \mathcal{I}(i)}$ for any $\mathcal{H} \subseteq [n(\mathbf{U})]$. This concludes the proof. \square

782 **Condition 3.2** (Visual Concept Identification Conditions).

783 *i Informativeness:* There exists a diffeomorphism $g_l : (\mathbf{Z}_l, \epsilon_l) \mapsto \mathbf{X}$ for $l \in [0, L]$, where ϵ_l
784 denotes independent exogenous variables.

785 **ii Smooth Density:** The probability density function $p(\mathbf{z}_{l+1}|\mathbf{z}_l)$ is smooth for any $l \in [L_V]$.
786 **iii Sufficient Variability:** For each Z and its parents $\tilde{\mathbf{Z}} := \text{Pa}(Z)$, at any value $\tilde{\mathbf{z}}$ of $\tilde{\mathbf{Z}}$, there
787 exist $n(\tilde{\mathbf{Z}}) + 1$ distinct values of Z , denoted as $\{z^{(n)}\}_{n=0}^{n(\tilde{\mathbf{Z}})}$, such that the vectors $\mathbf{w}(\tilde{\mathbf{z}}, z^{(n)}) -$
788 $\mathbf{w}(\tilde{\mathbf{z}}, z^{(0)})$ are linearly independent where $\mathbf{w}(\tilde{\mathbf{z}}, z) = \left(\frac{\partial \log p(\tilde{\mathbf{z}}|z)}{\partial \tilde{z}_1}, \dots, \frac{\partial \log p(\tilde{\mathbf{z}}|z)}{\partial \tilde{z}_{n(\tilde{\mathbf{Z}})}} \right)$.
789 **iv Sparse Connectivity (Minimality):** For each parent concept \tilde{Z} , there exists a subset of its
790 children $\mathbf{Z} \subseteq \text{Ch}(\tilde{Z})$ such that their only common parent is \tilde{Z} , i.e., $\bigcap_{Z \in \mathbf{Z}} \text{Pa}(Z) = \{\tilde{Z}\}$.
791 **Theorem 3.3** (Visual Concept Identification). Assume the hierarchical process for visual concepts
792 in (2). If a model specification θ_V satisfies Condition 3.2, and an alternative specification $\hat{\theta}_V$ satisfies
793 Conditions 3.2-i and 3.2-ii, along with a sparsity constraint such that for corresponding \hat{Z} and Z :

$$n(\text{Pa}(\hat{Z})) \leq n(\text{Pa}(Z)), \quad (3)$$

794 then, if both models θ_V and $\hat{\theta}_V$ generate the same observed data distribution $\mathbb{P}(\mathbf{X})$, the latent visual
795 concepts \mathbf{Z}_l are component-wise identifiable for every level $l \in [L_V]$.

796 *Proof.* By Lemma B.4, we can identify the set of variables \mathbf{Z}_1 that are directly connected to the text
797 variables \mathbf{D} and their causal graph. Treating the identified \mathbf{Z}_1 as the \mathbf{U} in Lemma B.4, we can further
798 identify \mathbf{Z}_2 . Repeating this procedure yields the identifiability of the entire model. \square

799 B.2 Proof for Theorem 3.5

800 **Definition B.5** (Non-negative Rank). The non-negative rank of a non-negative matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$
801 is equal to the smallest number p such that there exists a non-negative $m \times p$ -matrix \mathbf{B} and a
802 non-negative $p \times n$ -matrix \mathbf{C} such that $\mathbf{A} = \mathbf{BC}$.

803 **Lemma B.6** (Conditional Independence and Nonnegative Rank [73]). Let $\mathbf{P} \in \mathbb{R}^{m \times n}$ be a bi-variate
804 probability matrix. Then its non-negative rank $\text{rank}_+(\mathbf{P})$ is the smallest non-negative integer p such
805 that \mathbf{P} can be expressed as a convex combination of p rank-one bi-variate probability matrices.

806 **Lemma B.7** (One-level Textual Concept Identification). Assume the hierarchical process as per
807 (2) with $L_T = 1$. Let the true underlying parameters be θ_T . If θ_T satisfies Condition 3.4, and
808 an alternative learned model $\hat{\theta}_T$ satisfies Condition 3.4-iii, then if both models produce the same
809 observed distribution $\mathbb{P}(\mathbf{D})$, the latent textual concepts \mathbf{S}_1 are component-wise identifiable.

810 *Proof.* For each observed variable D , we search for the minimal set of variables $\mathbf{C} \subseteq (\mathbf{D} \setminus D)$ such
811 that the following conditional independence holds:

$$D \perp\!\!\!\perp \underbrace{\mathbf{D} \setminus (\{D\} \cup \mathbf{C})}_{\mathbf{R}} \mid (\mathbf{C}, \text{Ch}(D)). \quad (15)$$

812 Note that all D , \mathbf{C} , and \mathbf{R} belong to observed variables, and $\text{Ch}(D)$ is latent. Thanks to Condition 3.4-
813 ii and Lemma B.6, we can select \mathbf{C} with which the nonnegative rank of the probability table
814 $\mathbf{T}_{D, \mathbf{D} \setminus (\{D\} \cup \mathbf{C}) \mid \mathbf{C}}$ is strictly smaller than the support size of D .

815 We argue that such \mathbf{C} is the group of variables adjacent to the same variable S at the next level as D .
816 In other words, they are the co-parents of D , $\text{CoPa}(D)$.

817 This is because such \mathbf{C} makes 15 hold and thus $\text{CoPa}(D) \subseteq \mathbf{C}$. Otherwise, there would be open paths
818 passing S that induce dependence between D and $\text{CoPa}(D)$, violating the conditional independence
819 relation in (15). Therefore, the minimality constraint would enforce that $\mathbf{C} = \text{CoPa}(D)$. Repeating
820 this procedure to all $D \in \mathbf{D}$, we can construct \mathbf{S} variables at the next level and the adjacency relations
821 between \mathbf{S} and \mathbf{D} .

822 We proceed to identify the function $\mathbf{D} \mapsto \mathbf{S}$. We refer to D as a pure parent if D is adjacent to only
823 one variable S in the discovered graph. For each S , we denote its pure parents as \mathbf{D}^S and non-pure
824 parents as $\tilde{\mathbf{D}}^S$. We employ the conditional independence relation $\mathbf{D}^S \perp\!\!\!\perp \mathbf{D} \setminus \text{Pa}(\tilde{\mathbf{D}}^S) \mid (S, \tilde{\mathbf{D}}^S)$ and
825 Condition 3.4-iii to identify the value of S , i.e., the function $f_S := (\mathbf{d}^S, \tilde{\mathbf{d}}^S) \mapsto s$.

826 We first make use of the conditional independence

$$\mathbf{D}^S \perp\!\!\!\perp \mathbf{D} \setminus \text{Pa}(S) | (S, \tilde{\mathbf{D}}^S) \quad (16)$$

827 to merge the states of pure parents \mathbf{D}^S conditioned on the non-pure parents $\tilde{\mathbf{D}}^S$. Specifically, we con-
 828 dition on non-pure parents $\tilde{\mathbf{D}}^S = \tilde{\mathbf{d}}^S$ for any $\tilde{\mathbf{d}}^S$ present in the support. We define an equivalence re-
 829 lation \sim over values of $(\mathbf{D}^S, \tilde{\mathbf{D}}^S)$ where $(\mathbf{d}_1^S, \tilde{\mathbf{d}}^S) \sim (\mathbf{d}_2^S, \tilde{\mathbf{d}}^S)$ iff they give rise to an identical condi-
 830 tional distribution $\mathbb{P}(\mathbf{D} \setminus \text{Pa}(S) | \mathbf{D}^S = \mathbf{d}_1^S, \tilde{\mathbf{D}}^S = \tilde{\mathbf{d}}^S) = \mathbb{P}(\mathbf{D} \setminus \text{Pa}(S) | \mathbf{D}^S = \mathbf{d}_2^S, \tilde{\mathbf{D}}^S = \tilde{\mathbf{d}}^S)$.

831 We further resort a more global conditional independence by considering $(\mathbf{D}^S, \tilde{\mathbf{D}}^S)$ as a meta-variable
 832 and all the children $\text{Ch}(\tilde{\mathbf{D}}^S)$ associated to this meta-variable:

$$(\mathbf{D}^S, \tilde{\mathbf{D}}^S) \perp\!\!\!\perp \mathbf{D} \setminus \text{Pa}(\text{Ch}(\tilde{\mathbf{D}}^S)) | (\text{Ch}(\tilde{\mathbf{D}}^S), \underbrace{\text{Pa}(\text{Ch}(\tilde{\mathbf{D}}^S)) \setminus \{\mathbf{D}^S, \tilde{\mathbf{D}}^S\}}_{:=\tilde{\mathbf{D}}^S}), \quad (17)$$

833 where $(\mathbf{D}^S, \tilde{\mathbf{D}}^S)$ has become a pure parent of the latent variable $\text{Ch}(\tilde{\mathbf{D}}^S)$.
 834 We further group values $([\mathbf{d}^S], \tilde{\mathbf{d}}^S)$ following the rule that $([\mathbf{d}^S]_1, \tilde{\mathbf{d}}_1^S) \sim$
 835 $([\mathbf{d}^S]_2, \tilde{\mathbf{d}}_2^S)$ iff $\mathbb{P}(\mathbf{D} \setminus \text{Pa}(\text{Ch}(\tilde{\mathbf{D}}^S)) | ([\mathbf{d}^S], \tilde{\mathbf{D}}^S) = ([\mathbf{d}^S]_1, \tilde{\mathbf{d}}_1^S), \tilde{\mathbf{D}}^S = \tilde{\mathbf{d}}^S) =$
 836 $\mathbb{P}(\mathbf{D} \setminus \text{Pa}(\text{Ch}(\tilde{\mathbf{D}}^S)) | ([\mathbf{d}^S], \tilde{\mathbf{D}}^S) = ([\mathbf{d}^S]_2, \tilde{\mathbf{d}}_2^S), \tilde{\mathbf{D}}^S = \tilde{\mathbf{d}}^S)$ for each $\tilde{\mathbf{d}}^S$ on the support. That is,
 837 conditioning on any $\tilde{\mathbf{d}}^S$ on the support, $([\mathbf{d}^S]_1, \tilde{\mathbf{d}}_1^S)$ and $([\mathbf{d}^S]_2, \tilde{\mathbf{d}}_2^S)$ cannot be distinguished. Thus,
 838 we group them into an equivalence class $[(\mathbf{d}^S, \tilde{\mathbf{d}}^S)]$.

839 Finally, for each equivalent class $[(\mathbf{d}^S, \tilde{\mathbf{d}}^S)]$, we assign a distinct value \hat{s} . This constitutes a function
 840 $\hat{f}_S := (\mathbf{d}^S, \tilde{\mathbf{d}}^S) \mapsto \hat{s}$. Due to the deterministic relation from latent variables and their children in (1),
 841 \hat{f}_S is well-defined. We denote the random variable $\hat{S} := \hat{f}_S(\mathbf{D}^S, \tilde{\mathbf{D}}^S)$.

842 In the following, we show that \hat{S} and S are equivalent up to a bijection. We show this by con-
 843 tradiction. Suppose that there existed (s_0, \hat{s}_0) on their respective support, such that their pre-
 844 images partially overlapped $(\mathbf{d}_0^S, \tilde{\mathbf{d}}_0^S) \in \hat{f}_S^{-1}(\hat{s}_0) \cap f_S^{-1}(s_0)$ and $\hat{f}_S^{-1}(\hat{s}_0) \neq f_S^{-1}(s_0)$, where
 845 $f_S : (\mathbf{d}^S, \tilde{\mathbf{d}}^S) \mapsto s$ represents the true model. Suppose that $f_S^{-1}(s_0)$ missed some elements in
 846 $\hat{f}_S^{-1}(\hat{s}_0)$, i.e., $\exists (\mathbf{d}_1^S, \tilde{\mathbf{d}}_1^S) \in \hat{f}_S^{-1}(\hat{s}_0) \setminus f_S^{-1}(s_0)$. In this case, $(\mathbf{d}_0^S, \tilde{\mathbf{d}}_0^S)$ and $(\mathbf{d}_1^S, \tilde{\mathbf{d}}_1^S)$ would lead
 847 to distinct values s_0 and s_1 under model f_S^{-1} . By the construction of \hat{f}_S^{-1} , this would indicate
 848 $\mathbb{P}(\mathbf{D} \setminus \text{Pa}(S) | S = s_0) = \mathbb{P}(\mathbf{D} \setminus \text{Pa}(S) | S = s_1)$ and $\mathbb{P}(\mathbf{D} \setminus \text{Pa}(\text{Ch}(\tilde{\mathbf{D}}^S)) | S = s_0, \tilde{\mathbf{D}}^S = \tilde{\mathbf{d}}^S) =$
 849 $\mathbb{P}(\mathbf{D} \setminus \text{Pa}(\text{Ch}(\tilde{\mathbf{D}}^S)) | S = s_1, \tilde{\mathbf{D}}^S = \tilde{\mathbf{d}}^S)$ for each $\tilde{\mathbf{d}}^S$ on the support. Since $s_0 \neq s_1$, this violates
 850 Condition 3.4-iii, giving rise to a contradiction.

851 Suppose that $\hat{f}_S^{-1}(\hat{s}_0)$ contains additional elements, i.e., $\exists (\mathbf{d}_2^S, \tilde{\mathbf{d}}_2^S) \in \hat{f}_S^{-1}(\hat{s}_0) \setminus \hat{f}_S^{-1}(\hat{s}_0)$. In this case,
 852 $(\mathbf{d}_0^S, \tilde{\mathbf{d}}_0^S)$ and $(\mathbf{d}_2^S, \tilde{\mathbf{d}}_2^S)$ would lead to one value s_0 under model f_S^{-1} . By the construction of \hat{f}_S^{-1} , this
 853 would indicate either $\mathbb{P}(\mathbf{D} \setminus \text{Pa}(S) | \mathbf{D}^S = \mathbf{d}_0^S, \tilde{\mathbf{D}}^S = \tilde{\mathbf{d}}_0^S) \neq \mathbb{P}(\mathbf{D} \setminus \text{Pa}(S) | \mathbf{D}^S = \mathbf{d}_2^S, \tilde{\mathbf{D}}^S = \tilde{\mathbf{d}}_2^S)$
 854 or $\mathbb{P}(\mathbf{D} \setminus \text{Pa}(\text{Ch}(\tilde{\mathbf{D}}^S)) | ([\mathbf{D}^S], \tilde{\mathbf{D}}^S) = ([\mathbf{d}^S]_0, \tilde{\mathbf{d}}_0^S), \tilde{\mathbf{D}}^S = \tilde{\mathbf{d}}^S) \neq$
 855 $\mathbb{P}(\mathbf{D} \setminus \text{Pa}(\text{Ch}(\tilde{\mathbf{D}}^S)) | ([\mathbf{D}^S], \tilde{\mathbf{D}}^S) = ([\mathbf{d}^S]_2, \tilde{\mathbf{d}}_2^S), \tilde{\mathbf{D}}^S = \tilde{\mathbf{d}}^S)$ for some $\tilde{\mathbf{d}}^S$ on the support.
 856 By construction of \hat{f}_S , this would violate conditional independence (16) or (17) which the graphical
 857 structure implies, which leads to a contradiction.

858 Therefore, we have shown that for each pair (s, \hat{s}) on their respective support, their pre-images should
 859 be identical as long as they intersect: $\hat{f}_S^{-1}(\hat{s}) \cap f_S^{-1}(s) \neq \emptyset \implies \hat{f}_S^{-1}(\hat{s}) = f_S^{-1}(s)$, which is
 860 equivalent to that \hat{S} and S are equivalent up to a bijection. \square

861 **Condition 3.4** (Textual Concept Identification Conditions).

862 *i Natural Selection:* Each selection variable S_l has a support $\text{supp } S_l$ that is a proper subset of
 863 its potential range if its constituent parts (lower-level variables) were combined randomly. That
 864 is, $\text{supp } S_l \subsetneq f_{\mathbf{D} \rightarrow S_l}(\Omega^{n(\text{Pa}(S_l))})$, where $f_{\mathbf{D} \rightarrow S_l}$ is the function from \mathbf{D} to S_l .

- 865 ii **Bottlenecks:** The support size of any concept S_l is strictly larger than the joint support size of
866 its parents $\text{Pa}(S_l)$ in the selection graph.
 - 867 iii **Minimal Supports:** For any S , the condition distribution $\mathbb{P}(\mathbf{D} \setminus \text{Pa}(S) | S = s, \text{HPa}(S) = \tilde{s})$
868 is a one-to-one function w.r.t. the argument s .
 - 869 iv **No-Twins:** Distinct latent variables must have distinct sets of adjacent (parent/child) variables.
 - 870 v **Maximality:** The identified latent structure is maximal in the sense that splitting any latent
871 concept variable would violate either the Markov conditions or the No-Twins condition.
- 872 **Theorem 3.5** (Textual Concept Identification). Assume the hierarchical process as per (2). Let the
873 true underlying parameters be θ_T . If θ_T satisfies Condition 3.4, and an alternative learned model
874 $\hat{\theta}_T$ satisfies Condition 3.4-iii, then if both models produce the same observed distribution $\mathbb{P}(\mathbf{D})$, the
875 latent textual concepts \mathbf{S}_l are component-wise identifiable for every level $l \in [L_T]$.

876 *Proof.* By Lemma B.7, we can identify the set of variables \mathbf{S}_1 adjacent to \mathbf{D} and the bipartite causal
877 graph between these two sets of variables. We then employ the identified \mathbf{S}_1 to serve as \mathbf{D} in the first
878 step to identify \mathbf{S}_2 . Repeating this procedure yields the identifiability of the entire model. \square

879 C Key Concept Discussions

880 **The roles and purposes of “Selection-based hierarchy and causality minimality”.** The selection-
881 based hierarchy and causal minimality are constraints on the natural data distribution (images or
882 text), which is a standard modeling practice in causal representation learning [27]. Specifically, the
883 selection-based hierarchy considers concepts as effects of their constituent parts [33], while causal
884 minimality assumes this underlying causal graph is sparse in a specific way (e.g., Condition 3.2-iv).

885 **“Innate” hierarchical concept graphs.** “Innate” refers to the causal structure inherent in the natural
886 data-generating process itself. Latent concepts in the real world interact (e.g., ‘eyes’ and ‘nose’ are
887 components of a ‘face’), forming a pre-existing causal structure which we refer to as the “innate
888 concept graph.”

889 **True latent variables and their verifications.** “True latent variables” follow the standard notion
890 in causal representation learning [27]: they are the disentangled, interpretable, semantic factors
891 of the real-world data-generating process (e.g., age, object pose). This is in contrast to a deep
892 learning model’s learned features, which are often an entangled, uninterpretable mixture optimized
893 for a specific training objective. Aligning learned features with true latent variables (referred to as
894 “identification”) is the central goal, as it enables reliable interpretation (e.g., “this feature is age”) and
895 precise control (e.g., “increase this feature to make the face older”). This is a fundamental question
896 that our work addresses through both theoretical guarantees and empirical validation. Our work
897 provides the guarantee that if the data-generating process fulfills the property of causal minimality
898 and our learning objective enforces this (e.g., via sparsity), the model’s learned features are provably
899 equivalent to the true latent variables. We then validate this empirically via intervention, a standard
900 practice in causal research [27]. Our experiments (Figure 3 and Figure 6) show that manipulating
901 the theoretically identified features provides semantic control over the generated output, providing
902 evidence that these features are the meaningful causal levers of the generative process.

903 **Validity of the conditions.** While assumptions on the unobserved data-generating process may
904 not be validated directly, we have reasoned for the plausibility of our conditions by reflecting on
905 natural properties of real-world data. Beyond standard regularity assumptions like smoothness and
906 variability [58, 57, 55, 56, 92], our key minimality conditions—Sparse Connectivity (Condition 3.2-
907 iv) for vision and Minimal Supports (Condition 3.4-ii,iii) for text—are motivated by the observation
908 that concepts typically arise from a sparse set of causes [62–64, 60, 68] and that language is inherently
909 structured and compressible [105, 106]. Perhaps a more convincing validation is the empirical results.
910 Our experiments provide strong indicative support for these assumptions: by actively enforcing
911 sparsity/compression via SAEs, we successfully extract meaningful concept hierarchies in both vision
912 (Figure 3) and text (Figure 6) that are otherwise dense and not easily interpretable. This success
913 provides support for the usefulness of our overall approach and the validity of our assumptions. We
914 acknowledge that these assumptions, like any in this field, may not hold universally. Fortunately, our
915 strong empirical results suggest they seem effective and plausible for the complex, real-world data
916 we study.

Concept variable interpretation. Our theory proves the existence of a clean, one-to-one mapping between a learned feature and a true latent variable. This guarantee is what makes a principled interpretation possible in the first place. The subsequent step—assigning a human-understandable description to this now-identified concept—is intrinsically a task that requires human validation. This is a fundamental aspect of all interpretability research (perhaps modern vision-language models have the potential to automate this process).

Comparison with recent work [23]. On the technique side, Cywiński and Deja [23] feature an elegant concept location technique by utilizing the score function, which could significantly benefit our algorithm. For example, we could employ SAeUron Cywiński and Deja [23] to confirm whether our features at various timesteps match the concept location it identifies. Our causal learning algorithm explicitly learns the inter-connectivity among concepts across hierarchical levels. Thus, to modify a part of a high-level concept, we could focus our scope on only the variables connected to this specific high-level concept, which lowers the search complexity. In our experiment example, to implement two changes, “replacing the rock with tree stump” and “adding texture to tree stump”, SAeUron may need to perform two independent searches across all timesteps and node indices. Our method can help reduce the search space to only the low-level nodes connected to “tree stump”. In addition, pinpointing specific diffusion timesteps to intervene on potentially aids in managing undesirable artifacts. Moreover, our explicit concept graph could also give an interpretable, intuitive characterization of the model’s knowledge. On the message side, Cywiński and Deja [23] propose a novel score function to select the timestep and node index for accurate concept unlearning. Our work’s focus is to provide concise and informative theoretical conditions to understand concept learning in both vision and language modalities, with potential applications like concept easing or controllable generation. With this work, we hope the theoretical insights will facilitate the development of refined and dedicated methods in the community.

Comparison with recent work Kim et al. [21]. Revelio [21] relies on training a classifier on a specific classification dataset. Revelio trains SAEs and a classifier on a specific dataset (e.g., Caltech-101) to evaluate which features and timesteps are most correlated with class labels. Our work, in contrast, does not involve class labels. Our primary contribution is a hierarchical, causal framework designed to interpret the generative process itself. We apply causal discovery algorithms to discover the causal relationships across different levels of concepts without any class labels. We are able to understand how semantic concepts causally relate to one another across different levels of abstraction to form a coherent output (e.g., how “ear” and “mouth” features causally contribute to a “cat face”). Moreover, Kim et al. [21] do not perform interventions or analyze the compositional structure of generation, which are the central themes of our paper.

D Implementation Details

Computing resources. We use one L40 GPU for training the SAEs and a standard MacBook Pro with an M1 chip for causal discovery. Training one SAE takes around 8 hours.

Vision experiments. For the diffusion sampling process, we utilize the `sde-dpmsolver++` [107] sampler, which adds stochasticity between successive steps. We train the K-sparse autoencoder using a latent dimension of 5120, a batch size of 4096, and the Adam optimizer with a learning rate of 0.0001, setting $K = 10$. A 10k subset of prompts is selected from LAION-COCO [74]. We then extract SAE features from layers `down.2.1`, `mid.0`, and `up.1.0` at timesteps 899,500,100 respectively. The PC algorithm subsequently uses all resulting feature indices for causal discovery. For the sparsity ablation study, we control the top-K value used in the SAE. Specifically, we train additional SAEs with $K=4$ and $K=100$ at timestep 500. To evaluate the effect of sparsity, we then perform causal discovery by replacing the SAE features with $K=10$ with those from the $K=4$ or $K=100$ models.

LLM experiments. We utilize the pretrained SAEs for `gemma-2-2b-it` available from GemmaScope [90]. To collect features, we use the `pile-10k` corpus [108]. For each sample, we first exclude padding tokens and divide the remaining meaningful tokens into three sequential segments. The first segment is processed through the SAE at layer 18 to obtain feature indices representing lower-level information. The second segment is passed through the SAE at layer 19 to capture intermediate-level features. The final segment is input to the SAE at layer 20 to extract higher-level features. We then apply the PC algorithm for causal discovery using the feature indices from these three representational levels.

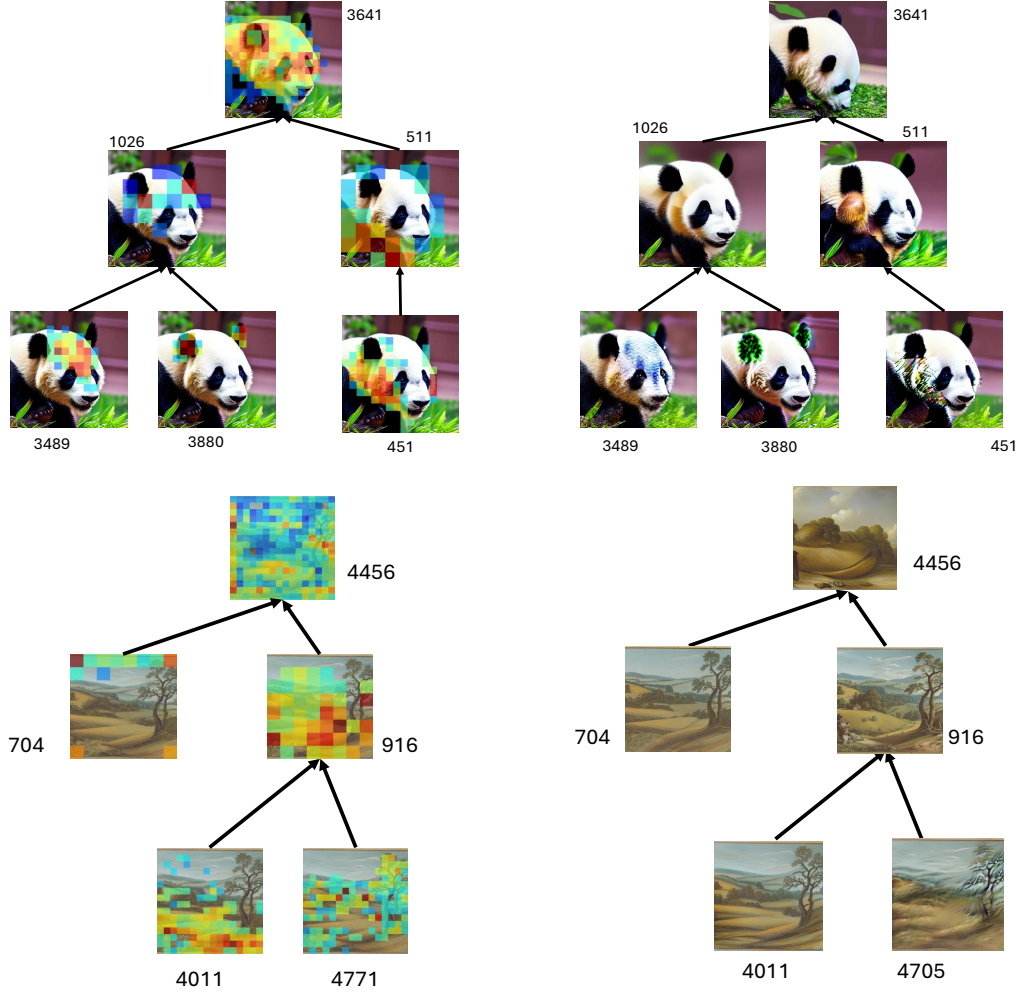


Figure 7: **Discovered hierarchical concept graphs and feature steering visualization for text-to-image generation.** We can observe that features on the hierarchical model represent a part-whole relation, and steering a feature yields corresponding visual variation (e.g., the panda’s ears).

E Additional Empirical Results

More examples for Figure 3. Figure 7 and Figure 8 contain more examples of Figure 3. For example, node 3641 in the SAE at timestep 899 contains comprehensive information about the panda, as illustrated by the heatmap. When feature steering is applied, it results in the generation of a new panda. Meanwhile, nodes 1026 and 511 in the SAE at timestep 500 represent different components of the panda. At a finer level of detail, nodes 3489, 3880, and 451 in the SAE at timestep 100 capture specific image features. These hierarchical concept graphs effectively illustrate how the panda is generated.

More results for model unlearning In addition to the four benchmark datasets in the main paper, we report results on another commonly used benchmark dataset with two tasks: *Remove Van Gogh* and *Remove Kelly McKernan* in Table.3. We evaluate performance using four metrics: LPIPS_e (similarity for prompts with the target style), LPIPS_u (similarity for prompts without the style), Acc_e (how well the target style was removed), and Acc_u (how well other styles were preserved), with accuracy ratings assessed using GPT-4o. Our method achieves competitive performance across all metrics and tasks.

Understanding the sparsity constraint. Figure 9 and Table 4 contain the ablation study for the sparsity constraint. We can observe that a proper sparsity strength can indeed give rise to desirable

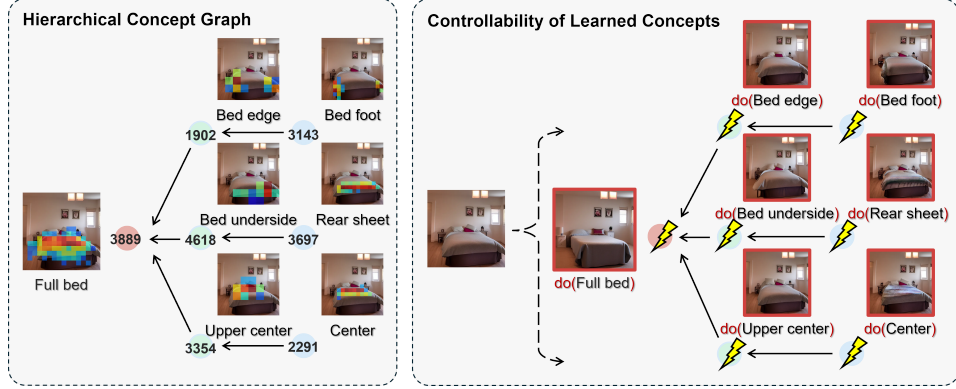


Figure 8: **More examples of the learned hierarchical concept graphs for text-to-image models.** Under appropriate sparsity and noise conditions, our method successfully recovers meaningful hierarchical structures, where each node encodes distinct semantic concepts. On the right, we demonstrate feature steering, where manipulating individual nodes leads to changes in the output that align with their position in the hierarchy – higher-level nodes produce broader semantic shifts, while lower-level nodes control more fine-grained aspects.

Method	LPIPS _e ↑	LPIPS _u ↓	Acce ↓	Accu ↑
Task: Remove “Van Gogh”				
SD-v1.4	–	–	0.95	0.95
CA [77]	0.30	0.13	0.65	0.90
RECE [80]	0.31	0.08	0.80	0.93
UCE [79]	0.25	0.05	0.95	0.98
SLD-Medium [82]	0.21	0.10	0.95	0.91
SAFREE [83]	0.42	0.31	0.35	0.85
Ours	0.53	0.26	0.30	0.88
Task: Remove “Kelly McKernan”				
SD-v1.4	–	–	0.80	0.83
CA [77]	0.22	0.17	0.50	0.76
RECE [80]	0.29	0.04	0.55	0.76
UCE [79]	0.25	0.03	0.80	0.81
SLD-Medium [82]	0.22	0.18	0.50	0.79
SAFREE [83]	0.40	0.39	0.40	0.78
Ours	0.48	0.20	0.35	0.81

Table 3: **Results on style removal.** We apply negative feature steering to the node to suppress the styles in the image.

interpretability results, while too small and too large sparsity constraints may be harmful in practice. As shown in Table 4, a low sparsity penalty results in visualized maps with significant overlap. On the other hand, applying a strong sparsity penalty leads to low node coverage, indicating that the nodes alone are insufficient to fully explain the generation of the entire image.

More examples for Figure 6. Figure 10 contains more examples for Figure 6. As discussed in the main paper, we divide the tokens into three segments based on their sequence order, with later tokens expected to encode higher-level information—consistent with the behavior of autoregressive language models. At the highest level, node 11859 represents the “yell mode,” characterized by capitalized words conveying a strong tone. The green node 1033, located at an intermediate sequence position, emphasizes importance or intensity—typically a component of the yell mode. At the lowest level, nodes 304, 2009, and 2818 capture various aspects and meanings related to the concept of importance.

	Overlap ↓	Coverage ↑
K=4	0.108 ± 0.128	26.37 ± 17.24
K=10	0.089 ± 0.079	47.90 ± 12.50
K=100	0.235 ± 0.132	37.46 ± 17.31

Table 4: **Quantitative ablation results.** We generate 100 panda images using different random seeds and visualize the feature heatmaps at timestep 500. We adjust the top-K value in the SAE at timestep 500 to control the level of sparsity. To evaluate, we compute the intersection-over-union (IoU) of intermediate heatmaps to measure concept disentanglement, and the union of all features to assess coverage. IoU reflects how distinctly the intermediate concepts are represented, while coverage in percentage indicates the extent to which the intermediate nodes collectively account for the image generation.

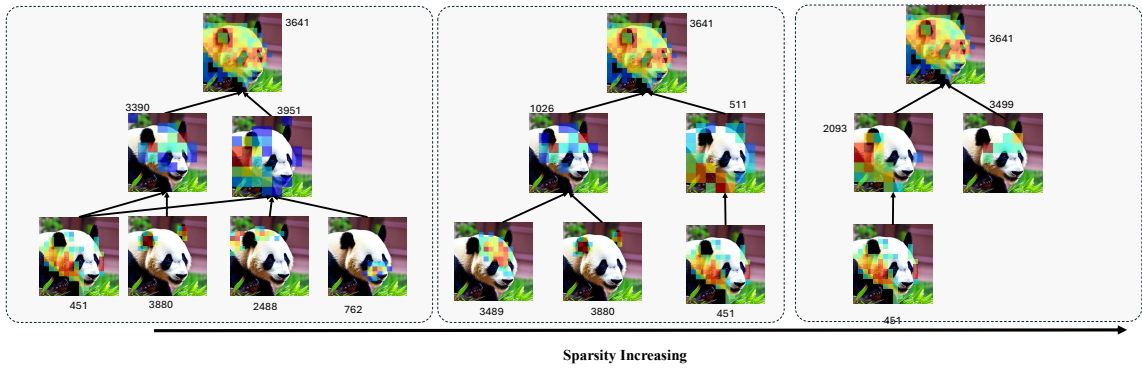


Figure 9: **Understanding the sparsity constraint.** We adjust the top-K value in the SAE at timestep 500 to control the level of sparsity, effectively modifying the sparsity strength of the SAE at this middle layer. As sparsity decreases, the resulting graph becomes denser, introducing many redundant and semantically irrelevant edges. This reduces the overall interpretability of the concept graph. Conversely, increasing sparsity yields a cleaner, more concise graph. However, if sparsity is too high, it may hinder the formation of a complete and interpretable concept graph necessary for image generation.

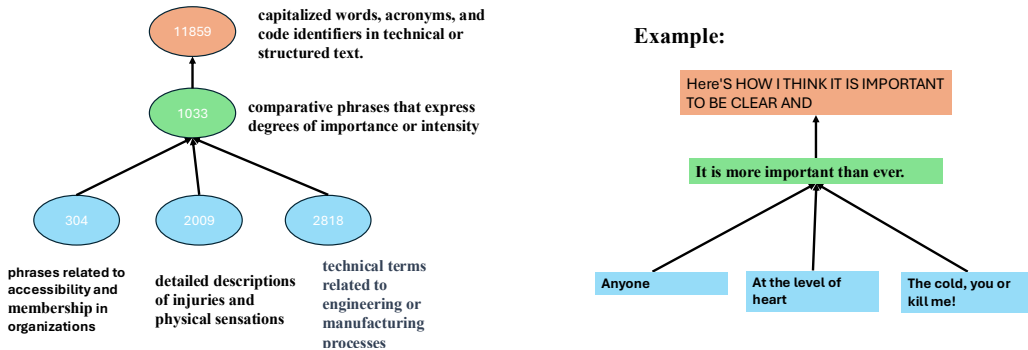


Figure 10: **An example of a discovered hierarchical concept graph for autoregressive language modeling.** Node 11859 represents a "yell mode," characterized by capitalized words that convey a strong tone. The green node 1033 captures the concept of emphasizing importance or intensity. Blue nodes correspond to lower-level information—for instance, node 304 represents entities mentioned throughout the text.