$\mathcal{T}$axoSAE: Rethinking Sparse Autoencoders through Taxonomic Logic

Xinyu Guo[1] **Minglai Yang**[2] **Clayton T. Morrison**[1]
Eduardo Blanco[1] **Mihai Surdeanu**[1]
[1]University of Arizona [2]Scale AI
{xinyuguo, claytonm, eduardoblanco, msurdeanu}@arizona.edu
minglai.yang@scale.com

Abstract

Recent Sparse Autoencoders (SAEs) decompose the representations of large language models (LLMs) into interpretable features, enabling precise readout and control of encoded knowledge. However, existing SAEs treat each feature as a standalone unit, whereas real-world knowledge is often taxonomically organized. As a result, readout carries no guarantee of taxonomic consistency, and control lacks the granularity to specify the scope of an edit. We propose $\mathcal{T}$axoSAE, which explicitly encodes taxonomical information, built on `is-a` (hypernym–hyponym) relations, into the SAE feature space. $\mathcal{T}$axoSAE mirrors a taxonomy with a stack of SAEs, one per taxonomy level, and introduces a taxonomic alignment loss that trains each concept to be active whenever one of its hyponyms is. On two factual QA taxonomies, one over biographical relations and one over animals, $\mathcal{T}$axoSAE enables (1) more accurate readout through the taxonomy with concentrated activations, e.g., +31.3 points in reading out the hypernyms of abstract concepts never queried in training; (2) more precise control, with +25.5 points in swap success, while suppressing a concept removes only its own answers (0–2% accuracy vs. 91–100% on other concepts); and, more importantly, (3) readout of unseen concepts through their hypernyms, with up to +37.9 points in hypernym readout accuracy. GitHub

1 Introduction

Large Language Models (LLMs) encode substantial factual knowledge in their internal representations (Sun et al., 2024a; Cohen et al., 2023; Hu et al., 2025) and have achieved strong performance on a wide range of knowledge-intensive tasks (Yu et al., 2022; Yifei et al., 2026). However, the internal representations of LLMs remain largely opaque (Bereska and Gavves, 2024; Sharkey et al., 2025), making it difficult to read out what knowledge they encode and to precisely control it (Chang et al., 2024; Cohen et al., 2024; Belinkov, 2022).

To mitigate this limitation, some recent Sparse Autoencoders (SAEs) learn disentangled feature spaces by explicitly aligning known concepts with SAE features (Yang et al., 2025; Härle et al., 2026). Similarly, contrastive activation differences (Ferrando et al., 2025; Yeo et al., 2025; Shi et al., 2025) are used with unsupervised SAEs to identify features for specific semantic information. These approaches enable more precise interpretation that can read out specific semantic information, and more targeted control via feature interventions.

Despite this progress, most prior work ignores dependencies between features, leading to a *flat feature space where features are not logically linked*. This limits readout and control in realistic scenarios, where *the knowledge is not flat, but often taxonomically organized.* For example, in biology (Diehl et al., 2011), knowledge about cell types follows taxonomic logic: a T cell is a type of lymphocyte, and a lymphocyte is a type of immune cell. In such domains, one often needs to look beyond final answers to their underlying taxonomy, e.g., checking whether an LLM identifies an unfamiliar cell type as an immune cell, and sometimes to control knowledge through this taxonomy, e.g., suppressing knowledge of lymphocytes and all their subtypes without affecting other immune cells.

Specifically, a flat feature space limits SAEs in two ways. First, *readout carries no guarantee of taxonomic consistency*: features are isolated, so readout yields disconnected activations, e.g., a *T cell* feature may activate while the *lymphocyte* feature stays silent. Second, *control lacks granularity*: a flat space has no notion of an edit's scale or scope, so one cannot specify whether an edit should be minor or substantial, or which dependent information should stay unchanged. For example, replacing *T cell* with *B cell* is a minor edit that keeps it a lymphocyte, whereas replacing it with *neuron* is a

substantial one that also changes its hypernyms, yet a flat space treats both as the same feature swap.

To bridge this gap, we propose to *encode taxonomical logic directly into the feature space*. We introduce $\mathcal{T}$axoSAE with two components: (1) *Stacked SAEs*. $\mathcal{T}$axoSAE mirrors the taxonomy with a stack of SAEs, one per taxonomy level, ordered from the most specific to the most abstract concepts. The first SAE reads the LLM representation and maps it onto the most specific concepts, and each higher SAE reads the reconstruction of the SAE below and maps it onto its hypernyms. (2) *Taxonomic alignment loss*. To realize the `is-a` logic (i.e., hyponym-hypernym) between the stacked SAEs, we link concept activations across levels through a training loss: a concept is trained to be active whenever one of its hyponyms is. This turns the rule "if a concept holds, its hypernym holds" into a training objective. Further, this simplifies training considerably, as $\mathcal{T}$axoSAE requires explicit data points only for terminal concepts, given the taxonomy.

Empirically, on two taxonomies, one over biographical relations and one over animals, $\mathcal{T}$axoSAE yields cleaner, more readable and more controllable features than the baselines under three evaluations. *(i) Readout:* with concentrated activations (1.0–1.7 slots per abstract concept), readout accuracy on questions about abstract concepts never seen in training is +31.3 percentage points over the strongest baseline. *(ii) Control:* editing the feature of a concept steers the model's answer to another concept with +25.5 percentage points in success, and suppressing it removes only that concept's answers, with 0–2% accuracy against 91–100% on other concepts. *(iii) Unseen concepts:* for concepts absent from training, the upper levels still place them under the right hypernym, with up to +37.9 percentage points in accuracy.

Our contributions are as follows:

- We identify a misalignment between SAE feature spaces and the structure of real-world knowledge: while knowledge is often taxonomically organized, SAE features are not aligned with it, leaving readout without taxonomic consistency and control without granularity.
- We propose $\mathcal{T}$axoSAE, which, to the best of our knowledge, is the first to explicitly encode a target taxonomy into the SAE feature space. It mirrors a taxonomy with stacked SAEs and turns the `is-a` rule into a taxonomic alignment loss, requiring supervision only on terminal concepts. It outperforms all baselines in taxonomy readout, control.
- $\mathcal{T}$axoSAE generalizes: it reads out unseen concepts at their hypernyms, whose `is-a` relations are absent from the SAEs' supervision; e.g., with *honeybee* bound but *beetle* unseen, beetle questions still activate the *insect* slot, the baselines rarely read them as insect. This improves data efficiency, as not every concept needs binding, and suggests alignment with the taxonomy.

2 Related Work

Sparse Autoencoders and Concept Alignment. SAEs decompose LLM representations into an overcomplete dictionary of sparse features (Elhage et al., 2022; Bricken et al., 2023; Huben et al., 2024), and have been scaled to large models (Gao et al., 2025) with improved activation functions (Rajamanoharan et al., 2024). Since they are trained without supervision (Mudide et al., 2025; Bussmann et al., 2025; Muchane et al., 2025), linking a feature to a concept requires post-hoc selection, e.g., by contrastive analysis of activations (Yeo et al., 2025; Ferrando et al., 2025), and a single concept is often split across many features or absorbed into more specific ones (Chanin et al., 2025). Accordingly, evaluations find SAE features to be imperfect units for concept readout and control (Makelov et al., 2025; Karvonen et al., 2025). Supervised alternatives instead bind concepts to designated units (Koh et al., 2020), and AlignSAE (Yang et al., 2025) and G-SAE (Härle et al., 2026) apply this to SAEs by binding each concept to a dedicated SAE slot via a supervised loss. $\mathcal{T}$axoSAE builds on this slot binding, but whereas their slots are standalone, it organizes them into levels linked by `is-a` relations.

Taxonomic Knowledge in LLMs. LLMs encode substantial factual knowledge in their internal representations (Sun et al., 2024a; Hu et al., 2025), much of which is organized by taxonomies and underlies tasks such as scholarly question answering (Yifei et al., 2026), cell type annotation (Hou and Ji, 2024), and medical coding (Soroush et al., 2024). However, LLMs make errors on such knowledge, especially on the most specific concepts (Sun et al., 2024b). Prior analyses show that taxonomic structure is reflected in the geometry of LLM representations (Park et al., 2025), but focus on discovering this structure rather than on reading out and

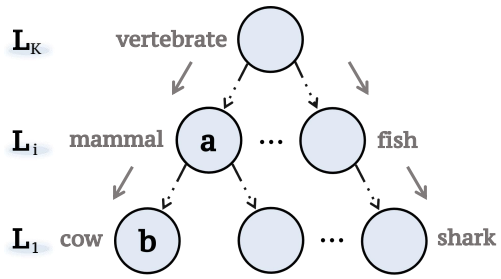


Figure 1: A taxonomy: each edge (a, b) is an `is-a` link from hypernym a to hyponym b; dashed edges mark further levels and nodes. Gray labels: a $K = 3$ example from the Animal taxonomy (full taxonomy in App. A). $\mathcal{T}$axoSAE uses one SAE per level L_k (Fig. 2).

controlling concepts consistently across taxonomy levels. $\mathcal{T}$axoSAE makes the taxonomy explicit in the feature space, so that the concepts and hypernyms an LLM identifies can be read out, and its errors corrected by editing them.

Logic Rules Encoding. Logic rules specify constraints that a set of facts must jointly satisfy, and integrating them into neural networks improves accuracy, data efficiency, and consistency with background knowledge (Giunchiglia et al., 2022; Marra et al., 2024). Prior work encodes such rules as training objectives (Hu et al., 2020; Xu et al., 2018), constraints during optimization (Yang et al., 2023), or network architectures (Li and Srikumar, 2019). Following the training-objective route, our taxonomic alignment loss encodes the `is-a` rule "if a hyponym holds, its hypernym holds" as a soft constraint across stacked SAEs, making taxonomic knowledge readable and editable in features.

3 Task Formulation

To formalize our setting, we consider taxonomy subsets represented as directed acyclic graphs $\mathcal{T} = (V, E)$ in which each node has at most one hypernym (see Fig. 1). Nodes denote taxonomic elements, i.e., concepts, and edges encode immediate hyponymy: an edge $(a, b) \in E$ indicates that a is the hypernym of b, and b is a hyponym of a. We call nodes without parents roots and those without children terminal nodes; remaining nodes are intermediate. We index the levels of the taxonomy from the bottom: L_1 holds the terminal nodes and L_K the roots, and $\mathcal{V}_k \subseteq V$ denotes the nodes of L_k. In this work, we study factual question answering (QA) over triples (e, r, v), where a question specifies an entity e and a relation r, and the model returns the corresponding value v. We construct two representative taxonomies: **relation taxonomies** over r and **entity taxonomies** over e. We call the element of the triple organized by the taxonomy the *taxonomic element* t, and the other the *complementary element* c. Thus, $(t, c) = (r, e)$ for relation taxonomies and $(t, c) = (e, r)$ for entity taxonomies. For example, for "How many years does a cow typically live?" (Fig. 1, gray, $K = 3$), the taxonomic element is $t =$ *cow*, together with its hypernyms *mammal* and *vertebrate*, and the complementary element is $c =$ *typical lifespan*.

4 Method

4.1 $\mathcal{T}$axoSAE Architecture

To interpret the hidden states of a frozen LLM as an explicit taxonomy, we build a stack of SAEs that encode the hidden state at a given layer of the LLM. Since the last input token summarizes the prompt (Lindsey et al., 2025; Afzal et al., 2025), we follow AlignSAE (Yang et al., 2025) and take the hidden state at this position before the first output token. Formally, given a frozen LLM M_θ with hidden dimension d and input $q = [q_1, \ldots, q_n]$, the input to the SAE stack is $h^{(0)} \in \mathbb{R}^d$, the layer-l hidden state of M_θ at the last token q_n (l set per dataset).

Stacking SAEs along the Taxonomy. $\mathcal{T}$axoSAE mirrors the taxonomy with a stack of SAEs, one SAE per taxonomy level, ordered from the terminal to the top-level nodes (Fig. 2). The k-th SAE, SAE_k, models level L_k of the taxonomy (§ 3 and Fig. 1): SAE_1 reads the LLM hidden state and binds the terminal nodes, and each higher SAE_k reads the reconstruction of SAE_{k-1} and binds the nodes of L_k. Moving up the stack thus abstracts the representation of a question from its terminal node to the node's hypernyms. Adjacent levels are connected by hyponymy relations, which we encode as a training loss (§ 4.2). Writing Enc_k and Dec_k for the encoder and decoder of SAE_k, $z^{(k)}$ for its feature vector, and $h^{(k)}$ for its reconstruction, the stack is $h^{(0)} \xrightarrow{\mathrm{Enc}_1} z^{(1)} \xrightarrow{\mathrm{Dec}_1} h^{(1)} \xrightarrow{\mathrm{Enc}_2} \cdots \xrightarrow{\mathrm{Enc}_K} z^{(K)} \xrightarrow{\mathrm{Dec}_K} h^{(K)}$.

Binding Taxonomic and Complementary Elements. Following AlignSAE (Yang et al., 2025), which binds each concept to a dedicated feature (*slot*) through post-training, $\mathcal{T}$axoSAE binds taxonomic elements to slots. A question, however, has more than one semantic dimension: it gives both the taxonomic element t and the complementary element c of a triple (e, r, v) (§ 3). For example, "How many years does a cow typically live?" asks

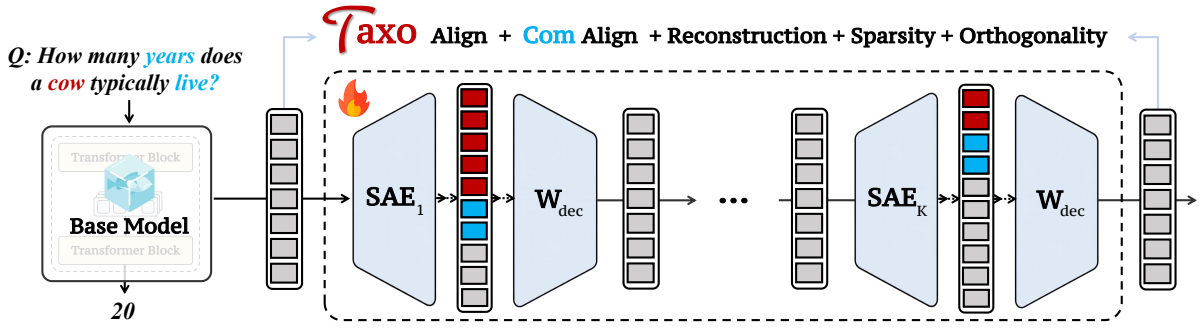


Figure 2: Overview of $\mathcal{T}$axoSAE. We train stacked SAEs on a frozen LLM's last-token hidden state. **(1) Stacked SAEs:** one SAE per taxonomy level ($L_{1,\dots,K}$ in Fig. 1), each mapping the reconstruction of the level below onto its hypernyms. **(2) Taxonomic alignment loss:** each taxonomic slot is trained to fire when the input involves its node or any of the node's hyponyms. Complement slots bind the complementary element; reconstruction, sparsity, and orthogonality losses complete the objective. Figure 8 shows the full slot layout of each level on both taxonomies.

about both the entity *cow* (t) and the relation *typical lifespan* (c). For a more flexible representation space, we bind both and separate them by mapping the input of each SAE_k onto three types of slots: *taxonomic slots*, one per node at taxonomy level L_k; *complement slots*, one per possible complementary element c; and unsupervised *free slots* for the rest of the reconstruction. Formally, SAE_k has a linear encoder and decoder,

$$z^{(k)} = W_{\mathrm{enc}}^{(k)} h^{(k-1)} + b_{\mathrm{enc}}^{(k)},$$
$$h^{(k)} = W_{\mathrm{dec}}^{(k)} z^{(k)} + b_{\mathrm{dec}}^{(k)},$$

and its feature vector is $z^{(k)} = \left[z_{\mathrm{free}}^{(k)}; z_{\mathrm{taxo}}^{(k)}; z_{\mathrm{comp}}^{(k)}\right]$ with $z_{\mathrm{taxo}}^{(k)} \in \mathbb{R}^{|\mathcal{V}_k|}$, one entry per node of L_k, and $\dim(z_{\mathrm{free}}^{(k)}) \gg \dim(z_{\mathrm{taxo}}^{(k)}) + \dim(z_{\mathrm{comp}}^{(k)})$.

Factual Value Prediction. Each SAE_k has a value head Val_k that answers the question from its own taxonomic slots and complement slots. The head ties every slot to the facts of its node, so that editing a slot changes the predicted answer. Since the answer depends on both elements, the head is bilinear in the two slot vectors: the taxonomic slots are mapped to a soft node embedding $e_t = \mathrm{softmax}(z_{\mathrm{taxo}}^{(k)})^\top C_t^{(k)}$ and the complement slots to a soft complement embedding $e_c = \mathrm{softmax}(z_{\mathrm{comp}}^{(k)})^\top C_c^{(k)}$, with learned tables $C_t^{(k)}$ and $C_c^{(k)}$, and the head outputs one logit for each candidate answer of c, $\mathrm{Val}_k(z^{(k)}, c) = O_c\big((W_t e_t) \odot (W_c e_c)\big) \in \mathbb{R}^{|\mathcal{A}_c|}$, where $\mathcal{A}_c$ is the set of candidate answers of c and O_c is a linear output layer. Because a non-terminal node inherits the values of its descendants, its answer for c is the set $U(u, c)$ of the values of all terminal nodes below u; the head is therefore multi-label, and at inference we take its top-m_k answers, where m_k is the largest answer-set size at L_k.

4.2 $\mathcal{T}$axoSAE Training

Training objectives. We train the encoders, decoders and value heads of all SAEs jointly. Our objective augments the standard SAE loss $\mathcal{L}_{\mathrm{SAE}}$ with (i) a taxonomic alignment loss that binds each node to its taxonomic slot ($\mathcal{L}_{\mathrm{taxo}}$), (ii) a complement alignment loss that binds each complementary element to its complement slot ($\mathcal{L}_{\mathrm{comp}}$), (iii) two decorrelation penalties that keep the taxonomic slots free of the complementary element and keep taxonomic information from leaking into the free slots ($\mathcal{L}_{\mathrm{tc}}$, $\mathcal{L}_{\mathrm{tf}}$), and (iv) a value loss that makes the slots support answer prediction ($\mathcal{L}_{\mathrm{val}}$).

Training questions are labeled *only* with their terminal node $u^{(1)}$ and complementary element c. The taxonomic slots of SAE_1 are aligned directly to the terminal node label $u^{(1)}$. The non-terminal levels have no labels, so we derive their targets from the level below through the taxonomy. Each hypernym collects the probabilities that SAE_k assigns to its hyponyms, and the hypernym with the largest total becomes the target of SAE_{k+1}. For example, if SAE_1 assigns high probability to *cow*, then *mammal*, whose score sums the probabilities of *cow*, *antelope*, and *bat*, has the largest total among the nodes of L_2 and becomes the target of SAE_2. *A hypernym slot is thus trained to be active whenever one of its hyponyms is*, which turns the rule "if a node holds, its hypernym holds" into a training objective. Formally, let $p^{(k)} = \mathrm{softmax}(z_{\mathrm{taxo}}^{(k)})$ be the probabilities over the nodes of L_k and $A^{(k)} \in \{0,1\}^{|\mathcal{V}_{k+1}| \times |\mathcal{V}_k|}$ the hyponymy edges between L_k and L_{k+1}; then $A^{(k)} p^{(k)}$ sums these probabilities over the hyponyms of each node of L_{k+1},

and the target $\tilde{u}^{(k+1)}$ is its argmax (Eq. (1)), with $\tilde{u}^{(1)} = u^{(1)}$. When SAE_1 reads $u^{(1)}$ correctly, $\tilde{u}^{(k+1)}$ is exactly the hypernym of $u^{(1)}$ at L_{k+1}, which we denote $u^{(k+1)}$. With $\mathrm{NMSE}(\hat{h}, h) = \|\hat{h} - h\|_2^2 / \|h\|_2^2$, the losses of SAE_k (each ablated in § 6.3) are

$$\tilde{u}^{(k+1)} = \arg\max_j \left[A^{(k)}\, p^{(k)}\right]_j, \tag{1}$$

$$\mathcal{L}^{(k)}_{\mathrm{SAE}} = \mathrm{NMSE}(h^{(k)}, h^{(k-1)}) + \lambda_{\mathrm{sp}} \|z^{(k)}\|_1, \tag{2}$$

$$\mathcal{L}^{(k)}_{\mathrm{taxo}} = \mathrm{CE}(z^{(k)}_{\mathrm{taxo}}, \tilde{u}^{(k)}), \tag{3}$$

$$\mathcal{L}^{(k)}_{\mathrm{comp}} = \mathrm{CE}(z^{(k)}_{\mathrm{comp}}, c), \tag{4}$$

$$\mathcal{L}^{(k)}_{\mathrm{tc}} = \|\mathrm{corr}(z^{(k)}_{\mathrm{taxo}}, z^{(k)}_{\mathrm{comp}})\|_F^2, \tag{5}$$

$$\mathcal{L}^{(k)}_{\mathrm{tf}} = \|\mathrm{cov}(z^{(k)}_{\mathrm{taxo}}, z^{(k)}_{\mathrm{free}})\|_F^2, \tag{6}$$

$$\mathcal{L}^{(k)}_{\mathrm{val}} = \mathrm{BCE}(\mathrm{Val}_k(z^{(k)}, c), \mathbf{1}_{U(u^{(k)}, c)}). \tag{7}$$

Pre-Training and Post-Training. Like AlignSAE (Yang et al., 2025), we train $\mathcal{T}$axoSAE in two stages. *Pre-training* learns a plain dictionary with $\mathcal{L}_{\mathrm{SAE}}$ only. *Post-training* starts from this checkpoint and optimizes all SAEs jointly with the full objective, which sums the losses over the levels with weights λ (App. K.2):

$$\begin{aligned}\mathcal{L} = \sum_k \big[&\mathcal{L}^{(k)}_{\mathrm{SAE}} + \lambda_{\mathrm{taxo}} \mathcal{L}^{(k)}_{\mathrm{taxo}} + \lambda_{\mathrm{comp}} \mathcal{L}^{(k)}_{\mathrm{comp}} \\ &+ \lambda_{\mathrm{tc}} \mathcal{L}^{(k)}_{\mathrm{tc}} + \lambda_{\mathrm{tf}} \mathcal{L}^{(k)}_{\mathrm{tf}} + \lambda_{\mathrm{val}} \mathcal{L}^{(k)}_{\mathrm{val}}\big].\end{aligned} \tag{8}$$

5 Experimental Setup

5.1 Datasets and Baselines

Datasets. We construct two factual QA datasets: (1) *Bio*, a relation taxonomy over biographical attributes of person names, whose terminal relations come from Freebase (Bollacker et al., 2008), with 18 terminal, 6 intermediate, and 3 top-level nodes; and (2) *Animal*, an entity taxonomy over animal names from an `IsA` subset of ConceptNet (Speer et al., 2017), with 18 terminal, 6 intermediate, and 2 top-level nodes. Training and OOD questions differ in templates (Bio) or synonyms (Animal). For example, "Who is Ralph Powers's father?" becomes "Name Mr. Ralph Stephen Powers's father." at test time (Bio), and *bat* becomes *flittermouse* (Animal). See App. A for details and statistics.

Baselines. We compare $\mathcal{T}$axoSAE with three baselines built on the same three-SAE stack, differing in how they assign a slot or feature to each element: *(1) Random Unsup. SAE* assigns each element a fixed feature by index or at random, averaged over four assignments. *(2) DiffMean Unsup. SAE* uses the same SAEs and picks, for each concept, the feature in its level's SAE whose mean activation exceeds every other concept's at that level by the largest margin, via difference-in-means (Gao et al., 2025). *(3) Hybrid SAE* uses the frozen terminal-level slots of $\mathcal{T}$axoSAE and DiffMean-selected features from unsupervised SAEs at non-terminal levels. Unsupervised SAEs use only reconstruction and L1 losses (App. J). A flat AlignSAE with non-terminal labels only matches our readout, swaps far worse (5.5% vs. 84.7%), and fails to keep levels consistent (0% vs. 100%; App. G).

5.2 Implementation

Base Model. We fine-tune GPT-2 small separately on each dataset with two types of QA: (1) hypernymy QA and (2) terminal-level factual QA. No non-terminal factual QA is used. Fine-tuning ensures the model knows every fact and hypernymy link (settings and accuracy in App. K.1). We then extract its hidden states at the final prompt token from a middle layer: layer 6 for Bio and layer 9 for Animal, chosen by terminal-level readout and swap; App. L confirms middle layers work best. We also repeat Bio with Qwen2.5-0.5B (App. B).

$\mathcal{T}$axoSAE. We implement $\mathcal{T}$axoSAE as three stacked SAEs, one per level, each with taxonomic, complement, and 10k free slots, and train it on terminal-level questions only, with 100 epochs of $\mathcal{L}_{\mathrm{SAE}}$ then 500 of all losses (App. K.2).

5.3 Metrics

Taxonomy Readout (Test A). The SAE assigns one unit to each node: a taxonomic slot in $\mathcal{T}$axoSAE, a selected feature in the baselines. We ask the unedited LLM about a node at L_k and check whether the most active unit in SAE_k is that of the node, and in each higher $\mathrm{SAE}_{k'}$ that of its hypernym at $L_{k'}$. **Diagonal Acc** of $\mathrm{SAE}_{k'}$ averages the per-node accuracy over $L_{k'}$: $\mathrm{Diag} = \frac{1}{|\mathcal{V}_{k'}|} \sum_{u \in \mathcal{V}_{k'}} \mathrm{Acc}(u)$, where $\mathrm{Acc}(u)$ is the fraction of questions with node or hypernym u at $L_{k'}$ that $\mathrm{SAE}_{k'}$ reads as u (1 for perfect binding, $1/|\mathcal{V}_{k'}|$ at random). We read terminal-level questions at every level ($L_1 \to L_1/L_2/L_3$), intermediate-level ones at L_2 and L_3 ($L_2 \to L_2/L_3$), and top-level ones at L_3 ($L_3 \to L_3$); the latter two are unseen in training.

Taxonomy Controllability (Tests B, C, E). To test whether these units are controllable, we edit them with slot edits. We feed the LLM a terminal-level question; SAE_k encodes its input $h^{(k-1)}$ into the activations z, we edit the slot of the source node u to obtain z', and the LLM continues from

SAE	Diagonal Acc (%) ↑						EffFeat ↓			Top1C (%) ↑			Swap (%) ↑	
	$L_1 \to L_1$	$L_1 \to L_2$	$L_1 \to L_3$	$L_2 \to L_2$	$L_2 \to L_3$	$L_3 \to L_3$	L_1	L_2	L_3	L_1	L_2	L_3	L_1	L_2
Bio relations (layer 6)														
Random Unsup. SAE	4.3	4.2	3.9	6.7	3.0	1.4	360.8	298.0	283.3	3.5	3.5	3.1	2.2	0.0
DiffMean Unsup. SAE	77.9	82.1	75.4	61.4	60.7	63.7	360.8	298.0	283.3	3.5	3.5	3.1	81.8	12.2
Hybrid SAE	**100.0**	71.0	77.1	54.9	54.5	69.2	**3.2**	188.2	184.9	**74.8**	4.4	3.5	**84.7**	16.3
$\mathcal{T}$axoSAE	**100.0**	**100.0**	**100.0**	**69.8**	**92.0**	**87.8**	**3.2**	**1.5**	**1.1**	**74.8**	**91.9**	**99.1**	**84.7**	**41.8**
$\Delta(\mathcal{T} - \text{Rand.})$	↑95.7	↑95.8	↑96.1	↑63.1	↑89.0	↑86.4	↓357.6	↓296.5	↓282.2	↑71.3	↑88.4	↑96.0	↑82.5	↑41.8
Animal entities (layer 9)														
Random Unsup. SAE	1.9	3.5	12.4	2.9	12.5	12.5	644.8	415.6	346.9	1.5	1.3	1.7	5.3	9.4
DiffMean Unsup. SAE	60.7	53.1	50.5	22.5	64.7	43.2	644.8	415.6	346.9	1.5	1.3	1.7	40.5	25.5
Hybrid SAE	**84.3**	18.2	55.3	16.3	50.6	45.4	**4.7**	95.3	86.4	**63.7**	7.4	4.7	**43.6**	21.4
$\mathcal{T}$axoSAE	**84.3**	**86.9**	**90.3**	**47.9**	**86.6**	**50.6**	**4.7**	**1.7**	**1.0**	**63.7**	**89.0**	**100.0**	**43.6**	**27.6**
$\Delta(\mathcal{T} - \text{Rand.})$	↑82.4	↑83.4	↑77.9	↑45.0	↑74.1	↑38.1	↓640.1	↓413.9	↓345.9	↑62.2	↑87.7	↑98.3	↑38.3	↑18.2

Table 1: Results on Bio and Animal. $L_1/L_2/L_3$: terminal/intermediate/top levels; L$q\to$Lr: level-r readout on level-q questions ($L_1 \to L_r$ pools ID and OOD). Bold/underline: best/second-best; Δ: $\mathcal{T}$axoSAE minus Random Unsup. SAE. $\mathcal{T}$axoSAE achieves the best readout (Diagonal Acc) and control (Swap) on both datasets. Layer choice: App. L. The same conclusions hold on Qwen2.5-0.5B (App. B).

$h' = \text{Dec}_k(z')$. A *swap* to a target node u' sets $z'_u = 0$ and $z'_{u'} = \alpha$; a *deactivation* scales the source slot, $z'_u = (1-\beta)\, z_u$, checking whether it removes only the node's own answers (App. H).

- **Terminal-level swap (Test B).** We swap the terminal node in SAE_1. *Strict* requires the target's answer for the same complementary element; *Inset* accepts any answer of the target node. E.g., swapping *cow* for *eagle* in "What covers a cow's body?" requires *feathers* under Strict.
- **Intermediate-level swap (Test C).** We swap the intermediate node in SAE_2 for one under a different top-level node. Since an intermediate node covers all its terminal hyponyms, the target's answer is the set $U(u', c)$ of their values (§ 4), we take the top-$|U(u', c)|$ beam candidates, and *target-only coverage*@$|U|$ counts a swap as correct if they contain every target answer not shared by the source. For both swaps, we also check that the SAE then reads the target's hypernyms.
- **Selectivity (Test E).** We deactivate the terminal node in SAE_1 or the intermediate node in SAE_2, with the smallest β at which the node's own questions are answered correctly at most 5% of the time. *Own* is the exact match on the node's own questions (lower is better), and *Other* on all other questions (higher is better). E.g., deactivating *mammal* should break *cow* questions (low Own) but not *eagle* ones (high Other).

Activation Concentration (Test D). We measure how the activation of each node is spread over units. For a node u of L_k, let $\rho_{u,j}$ be the share of the mean (positive) activation of unit j of SAE_k on the training questions about u, normalized over units. $\mathit{EffFeat}(u) = \exp\big(-\sum_j \rho_{u,j} \log \rho_{u,j}\big)$ is the effective number of units the node uses (1 is a single unit; lower is better), and $\mathit{Top1C}(u) = \max_j \rho_{u,j}$ is the share of its strongest unit (higher is better). Both are averaged over each level's nodes.

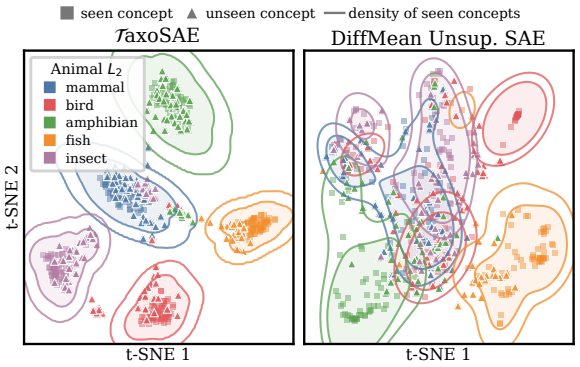


Figure 3: t-SNE of L_2 activations on Animal, colored by L_2 node: squares are trained animals, triangles unseen ones. A triangle inside its color's contour is placed under the right hypernym. $\mathcal{T}$axoSAE does this for most unseen animals; DiffMean's regions overlap. Mollusc is omitted due to heavy overlap in the baseline (App. C).

6 Experimental Results

6.1 Taxonomy Readout

$\mathcal{T}$axoSAE enables more accurate taxonomy readout. In Table 1, $\mathcal{T}$axoSAE matches or exceeds all baselines in Diagonal Acc on both datasets, as on Bio with Qwen2.5-0.5B (App. B). On terminal-level questions, it reads terminal nodes and hypernyms accurately at all levels ($L_1 \to L_1/L_2/L_3$).

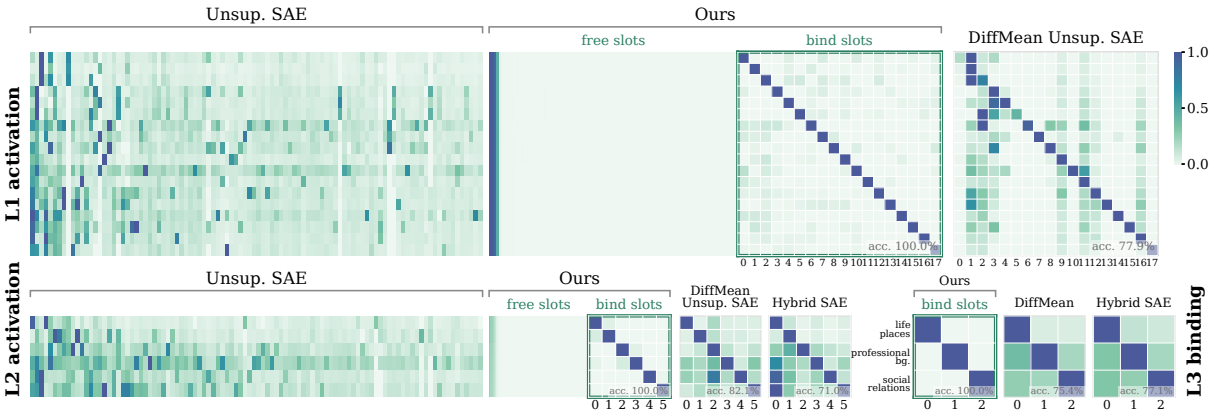


Figure 4: Normalized activations (L_1/L_2) and binding matrices (L_1–L_3) on Bio. Rows group questions by their terminal node and its hypernyms. Wide panels show activations over all units: the unsupervised SAE spreads each concept over many features, while $\mathcal{T}$axoSAE concentrates it in bind slots. Square panels show each concept's assigned unit; a diagonal means perfect binding, and "acc." is the binding accuracy.

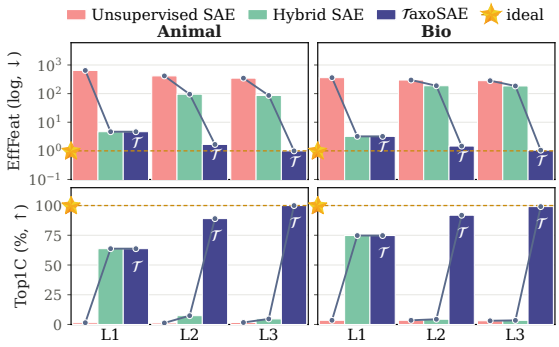


Figure 5: Activation fragmentation (EffFeat) and concentration (Top1C) per taxonomy level on Animal and Bio; dashed lines mark the ideal single unit. $\mathcal{T}$axoSAE activations are the most concentrated.

	L_1 swap				L_2 swap	
	Answer		Read		Answer	Read
SAE	Strict	In-set	L_2	L_3	Target-only	L_3
Random Unsup.	2.2	4.0	18.0	34.3	0.0	31.0
DiffMean Unsup.	81.8	83.0	71.1	55.9	12.2	69.5
Hybrid	84.7	88.1	66.2	44.6	16.3	65.8
$\mathcal{T}$axoSAE	**84.7**	**88.1**	**100.0**	**93.2**	**41.8**	**100.0**
$\Delta(\mathcal{T}$ − Best)	↑2.9	↑5.1	↑28.9	↑37.3	↑25.5	↑30.5

Table 2: Bio swaps at each system's best strength. Answer: target-value accuracy; Read: target-hypernym readout. Underlines: best baselines (excluding Hybrid for L_1, sharing our L_1 SAE); ↑: gains over them.

More importantly, though trained only on terminal-level questions, it generalizes to non-terminal questions, identifying queried elements and their hypernyms. For $L_2 \to L_2$ and $L_3 \to L_3$, it beats the strongest baseline by up to 25.4 points. For hypernym readout ($L_2 \to L_3$), it gains 31.3 points. Complement slots read out almost perfectly (App. E). The advantage also holds on unseen phrasings of terminal-level questions (App. D).

$\mathcal{T}$axoSAE places unseen terminal concepts under their hypernyms. We test 24 animals outside the trained ones (App. C). $\mathcal{T}$axoSAE maps most of them into the L_2 region of their class (Fig. 3) and reads the right intermediate and top-level node for 63.9% and 76.1% of questions, against 26.0% and 56.8% for DiffMean. Bio's 24 unseen relations show the same pattern (91.4% vs. 75.9% at L_2).

$\mathcal{T}$axoSAE produces clean and concentrated activations. Figure 4 visualizes how terminal-level questions activate taxonomic elements on Bio. In the unsupervised SAE, each question activates many features, and the same features fire on questions about different elements, so no single feature marks an element. In $\mathcal{T}$axoSAE, the taxonomic slots respond selectively: a question activates mainly the slot of its own element, at both L_1 and L_2. Its binding matrices are diagonal at all three levels: questions about an element, or about any of its hyponyms, activate that element's slot and few others. The concentration metrics in Table 1, computed on the activations without post-hoc feature selection: $\mathcal{T}$axoSAE reaches non-terminal EffFeat of 1.0–1.7 and Top1C of 89.0–100.0%, i.e., each abstract concept is carried by about one taxonomic slot, which holds most of its activation, so a single slot suffices to read out or edit the concept; whereas the activations of unsupervised SAEs are fragmented at all three levels (Fig. 5).

6.2 Taxonomy Controllability

$\mathcal{T}$axoSAE achieves better target-value prediction and more consistent target-hypernym readout after swapping. We take Bio as an example in

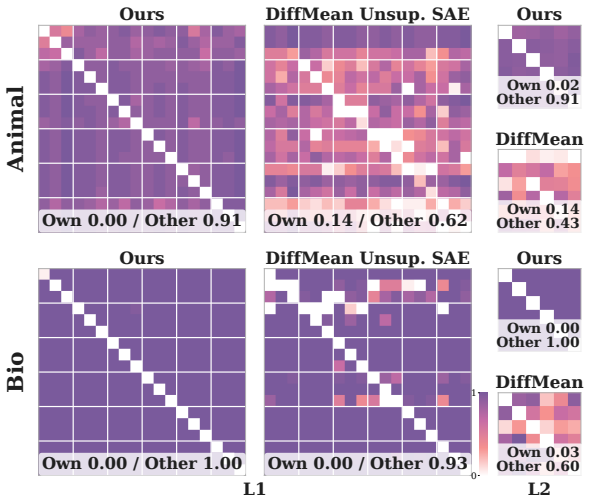


Figure 6: Selectivity on Animal (top) and Bio (bottom): row i deactivates element i's unit; column j is the exact match on element j. Ideally the diagonal is white and the rest dark, as for $\mathcal{T}$axoSAE; DiffMean also lightens off-diagonal cells, especially at L_2. Own (↓)/Other (↑): mean accuracy on target/non-target elements. Labeled version: Fig. 12.

Table 2 with complete results in App. H. It shows that after terminal- and intermediate-level swaps, $\mathcal{T}$axoSAE more reliably produces the correct answer for the target taxonomic element, especially at the intermediate level (+25.5 points in target-only coverage). Its higher levels also read the target's hypernyms in 93.2–100.0% of swaps, 28.9–37.3 points above the baselines.

$\mathcal{T}$axoSAE selectively suppresses target answers while preserving others. Figure 6 shows the effects of deactivating individual taxonomic slots at terminal and intermediate levels. Across both datasets, $\mathcal{T}$axoSAE reduces exact-match accuracy for the targeted element to 0.00–0.02 while retaining 0.91–1.00 accuracy for other elements. In contrast, deactivating DiffMean-selected features also disrupts answers for other elements, especially at the intermediate level, where Other accuracy is only 0.43 on Animal and 0.60 on Bio.

6.3 Ablation Study

Ablation Variants and Readout–Controllability Metrics. To assess each training loss, we evaluate five ablation variants on Bio, each removing one loss from $\mathcal{T}$axoSAE. Readout averages all Test A diagonal accuracies ($L_1 \to L_1/L_2/L_3$, $L_2 \to L_2/L_3$, $L_3 \to L_3$), weighted by question counts; controllability averages Test B strict accuracy and Test C target-only coverage on pairs from different top-level nodes, each at its best α.

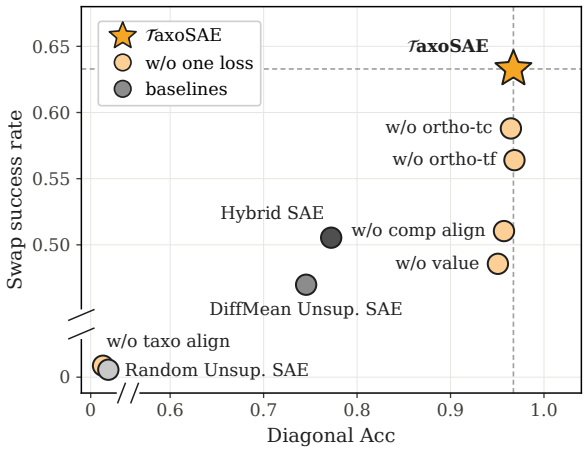


Figure 7: Ablation on Bio (upper right is better). Removing taxonomic alignment collapses both metrics to random; removing others mainly lowers swap success.

Figure 7 plots both (upper right is better); full results for Tests A–E are provided in App. F.

Removing any of the five losses weakens $\mathcal{T}$axoSAE's joint readout and steering performance. Figure 7 shows that (1) removing taxonomic alignment (*w/o taxo align*) collapses both diagonal accuracy and swap success to the level of the random baseline; (2) removing complement alignment or value prediction (*w/o comp align*, *w/o value*) preserves relatively high readout accuracy but markedly reduces swap success; and (3) removing either orthogonality loss (*w/o ortho-tc*, *w/o ortho-tf*) shows a similar pattern, with smaller drops in swap success than other variants, while readout remains comparable and some concentration metrics improve. Replacing the bilinear value head with a per-slot head mainly weakens intermediate-level swaps (App. F.2).

7 Conclusion

We identified that existing SAEs learn a flat feature space, which leaves readout without taxonomic consistency and control without granularity. We therefore encode taxonomic `is-a` logic into the feature space and propose $\mathcal{T}$axoSAE, which combines stacked SAEs with a taxonomic alignment loss. On two factual QA taxonomies, $\mathcal{T}$axoSAE enables more accurate taxonomy readout and more precise control than the baselines. Moreover, it reads out unseen concepts at their hypernyms without binding them, which improves data efficiency and suggests that its features align well with the taxonomy. Future work could extend it to larger models and logical relations beyond taxonomies.

Limitations

Our evaluation has several limitations. First, all experiments are in English, and we have not tested whether $\mathcal{T}$axoSAE transfers to other languages, where taxonomic knowledge may be expressed and encoded differently. Second, our taxonomies are relatively simple: although built from ConceptNet and Freebase, they have three levels with at most 18 terminal nodes.

References

Anum Afzal, Florian Matthes, Gal Chechik, and Yftah Ziser. 2025. Knowing before saying: Llm representations encode information about chain-of-thought success before completion. *Preprint*, arXiv:2505.24362.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Leonard Bereska and Efstratios Gavves. 2024. Mechanistic interpretability for ai safety–a review. *arXiv preprint arXiv:2404.14082*.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2(5):6.

Bart Bussmann, Noa Nabeshima, Adam Karvonen, and Neel Nanda. 2025. Learning multi-level features with matryoshka sparse autoencoders. *arXiv preprint arXiv:2503.17547*.

Ting-Yun Chang, Jesse Thomason, and Robin Jia. 2024. Do localization methods actually localize memorized data in llms? a tale of two benchmarks. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3190–3211.

David Chanin, James Wilken-Smith, Tomáš Dulka, Hardik Bhatnagar, Satvik Golechha, and Joseph Bloom. 2025. A is for absorption: Studying feature splitting and absorption in sparse autoencoders. In *Advances in Neural Information Processing Systems*. ArXiv:2409.14507.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. 2024. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*, 12:283–298.

Roi Cohen, Mor Geva, Jonathan Berant, and Amir Globerson. 2023. Crawling the internal knowledge-base of language models. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1856–1869.

Alexander D Diehl, Alison Deckhut Augustine, Judith A Blake, Lindsay G Cowell, Elizabeth S Gold, Timothy A Gondré-Lewis, Anna Maria Masci, Terrence F Meehan, Penelope A Morel, Anastasia Nijnik, et al. 2011. Hematopoietic cell types: prototype for a revised cell ontology. *Journal of biomedical informatics*, 44(1):75–79.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. 2022. Toy models of superposition. *arXiv preprint arXiv:2209.10652*.

Javier Ferrando, Oscar Obeso, Senthooran Rajamanoharan, and Neel Nanda. 2025. Do i know this entity? knowledge awareness and hallucinations in language models. In *International Conference on Learning Representations*, volume 2025, pages 77965–78000.

Leo Gao, Tom Dupre la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2025. Scaling and evaluating sparse autoencoders. In *International Conference on Learning Representations*, volume 2025, pages 26721–26754.

Eleonora Giunchiglia, Mihaela Catalina Stoian, and Thomas Lukasiewicz. 2022. Deep learning with logical constraints. *arXiv preprint arXiv:2205.00523*.

Ruben Härle, Felix Friedrich, Manuel Brack, Björn Deiseroth, Stephan Waeldchen, Patrick Schramowski, and Kristian Kersting. 2026. Measuring and guiding monosemanticity. *Advances in Neural Information Processing Systems*, 38:167468–167508.

Wenpin Hou and Zhicheng Ji. 2024. Assessing gpt-4 for cell type annotation in single-cell rna-seq analysis. *Nature methods*, 21(8):1462–1465.

Yujia Hu, Tuan-Phong Nguyen, Shrestha Ghosh, and Simon Razniewski. 2025. Enabling llm knowledge analysis via extensive materialization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16189–16202.

Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, and Eric Xing. 2020. Harnessing deep neural networks with logic rules. *Preprint*, arXiv:1603.06318.

Robert Huben, Hoagy Cunningham, Logan Smith, Aidan Ewart, and Lee Sharkey. 2024. Sparse autoencoders find highly interpretable features in language models. In *International Conference on Learning Representations*, volume 2024, pages 7827–7845.

Adam Karvonen, Can Rager, Johnny Lin, Curt Tigges, Joseph Bloom, David Chanin, Yeu-Tong Lau, Eoin Farrell, Callum McDougall, Kola Ayonrinde, et al. 2025. Saebench: A comprehensive benchmark for sparse autoencoders in language model interpretability. *arXiv preprint arXiv:2503.09532*.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. 2020. Concept bottleneck models. In *International conference on machine learning*, pages 5338–5348. Pmlr.

Tao Li and Vivek Srikumar. 2019. Augmenting neural networks with first-order logic. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 292–302.

Jack Lindsey, Wes Gurnee, Emmanuel Ameisen, Brian Chen, Adam Pearce, Nicholas L Turner, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, et al. 2025. On the biology of a large language model. transformer circuits thread (2025). *URL: https://transformer-circuits. pub/2025/attribution-graphs/biology. html*.

Aleksandar Makelov, Georg Lange, and Neel Nanda. 2025. Towards principled evaluations of sparse autoencoders for interpretability and control. In *International Conference on Learning Representations*, volume 2025, pages 33588–33636.

Giuseppe Marra, Sebastijan Dumančić, Robin Manhaeve, and Luc De Raedt. 2024. From statistical relational to neurosymbolic artificial intelligence: A survey. *Artificial intelligence*, 328:104062.

Mark Muchane, Sean Richardson, Kiho Park, and Victor Veitch. 2025. Incorporating hierarchical semantics in sparse autoencoder architectures. *Preprint*, arXiv:2506.01197.

Anish Mudide, Josh Engels, Eric Michaud, Max Tegmark, and Christian Schroeder de Witt. 2025. Efficient dictionary learning with switch sparse autoencoders. In *International Conference on Learning Representations*, volume 2025, pages 101830–101844.

Kiho Park, Yo Joong Choe, Yibo Jiang, and Victor Veitch. 2025. The geometry of categorical and hierarchical concepts in large language models. In *The Thirteenth International Conference on Learning Representations*.

Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. 2024. Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders. *arXiv preprint arXiv:2407.14435*.

Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeff Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Bloom, et al. 2025. Open problems in mechanistic interpretability. *arXiv preprint arXiv:2501.16496*.

Wei Shi, Ziyuan Xie, Sihang Li, and Xiang Wang. 2025. Safer: Probing safety in reward models with sparse autoencoder. *arXiv preprint arXiv:2507.00665*.

Ali Soroush, Benjamin S Glicksberg, Eyal Zimlichman, Yiftach Barash, Robert Freeman, Alexander W Charney, Girish N Nadkarni, and Eyal Klang. 2024. Large language models are poor medical coders—benchmarking of medical code querying. *Nejm Ai*, 1(5):AIdbp2300040.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Kai Sun, Yifan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. 2024a. Head-to-tail: How knowledgeable are large language models (llms)? aka will llms replace knowledge graphs? In *Proceedings of the 2024 conference of the North American chapter of the association for computational linguistics: human language technologies (volume 1: long papers)*, pages 311–325.

Yushi Sun, Hao Xin, Kai Sun, Yifan Ethan Xu, Xiao Yang, Xin Luna Dong, Nan Tang, and Lei Chen. 2024b. Are large language models a good replacement of taxonomies? *arXiv preprint arXiv:2406.11131*.

Jingyi Xu, Zilu Zhang, Tal Friedman, Yitao Liang, and Guy Broeck. 2018. A semantic loss function for deep learning with symbolic knowledge. In *International conference on machine learning*, pages 5502–5511. PMLR.

Minglai Yang, Xinyu Guo, Zhengliang Shi, Jinhe Bi, Steven Bethard, Mihai Surdeanu, and Liangming Pan. 2025. Alignsae: Concept-aligned sparse autoencoders. *arXiv preprint arXiv:2512.02004*.

Zhun Yang, Joohyung Lee, and Chiyoun Park. 2023. Injecting logical constraints into neural networks via straight-through estimators. *Preprint*, arXiv:2307.04347.

Wei Jie Yeo, Nirmalendu Prakash, Clement Neo, Ranjan Satapathy, Roy Ka-Wei Lee, and Erik Cambria. 2025. Understanding refusal in language models with sparse autoencoders. In *EMNLP (Findings)*, pages 6377–6399.

Li S Yifei, Allen Chang, Chaitanya Malaviya, and Mark Yatskar. 2026. Researchqa: Evaluating scholarly question answering at scale across 75 fields with survey-mined questions and rubrics. *Transactions of the Association for Computational Linguistics*, 14:1344–1368.

Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. 2022. Generate rather than retrieve: Large language models are strong context generators. *arXiv preprint arXiv:2209.10063*.

Appendix Contents

A Datasets 11
B Results on Qwen2.5-0.5B 11
C Unseen Terminal Concepts 12
D Unseen Phrasings 14
E Complement Slot Readout 14
F Ablation Study 15
F.1 Loss Ablation 15
F.2 Value head Ablation 15
G Flat AlignSAE Baseline 16
H Swap Success Results 17
I Selectivity Matrices with Concept Labels 17
J Baseline Implementation Details 18
K Training Details 19
K.1 Base Model Training 19
K.2 $\mathcal{T}$axoSAE Training 20
L Layer-wise Performance 20
M The Usage of AI Tools 21

A Datasets

Both taxonomies build on existing resources, while the facts are controlled so that every answer is known, and both cross a three-level taxonomy with a set of complementary elements (Table 3). In **Bio**, the taxonomy is over biographical relations: 18 terminal relations (e.g., *birth city*, *employer*, *father*) are grouped into 6 intermediate relations (*place of birth*, *place of death*, *employment*, *education*, *immediate family*, *extended family*) and 3 top-level relations (*life places*, *professional background*, *social relations*); the terminal relations are person relations from Freebase (Bollacker et al., 2008), and we generate the more abstract intermediate and top-level relations above them; the complementary elements are 20 synthetic people, and each person has one value for each terminal relation. In **Animal**, the taxonomy is over animals: 18 terminal animals (e.g., *cow*, *frog*, *octopus*) are grouped into 6 intermediate classes (*mammal*, *bird*, *amphibian*, *fish*, *insect*, *mollusc*) and 2 top-level classes (*vertebrate*, *invertebrate*), forming a subset of the English `IsA` relations of ConceptNet (Speer et al., 2017); the complementary elements are 20 properties (e.g., *limb count*, *capable of flight*, *diet type*), and each animal has one value for each property. A non-terminal node takes, for each complementary element, the set of values of its terminal hyponyms.

$\mathcal{T}$axoSAE is trained only on terminal-level questions. Bio varies the question template (10 in training) and the surface form of the person's name (4 per person); Animal varies the template (54 in training) and the animal's synonym (5–8 per animal, 108 in total). The readout tests add held-out templates, names, or synonyms at the terminal level, and intermediate- and top-level questions, none of which appear in training. Figure 8 shows how one question of each dataset maps onto the slots of the three SAEs.

B Results on Qwen2.5-0.5B

To test whether $\mathcal{T}$axoSAE transfers beyond GPT-2, we repeat the Bio experiments with Qwen2.5-0.5B (base), a 24-layer model with a hidden dimension of 896 and a modern architecture (RoPE, RMSNorm, SwiGLU). We fine-tune it on the same data and for the same number of steps as GPT-2, with a learning rate of 10^{-4}; the fine-tuned model answers all terminal-level factual and hypernymy questions correctly. We read the middle layer, layer 12, which has the same relative depth as layer 6 in GPT-2. We did not tune this choice, and other layers may give better results. $\mathcal{T}$axoSAE and all baselines use the settings of the main experiments, except that the swap strength α is searched over a wider grid, $\{20, 50, 100, 150, 200, 300, 500, 700, 1000, 2000\}$, because the best strength shifts with the base model; each system is reported at its own best α. Results are from a single seed.

Table 4 shows that the conclusions of Table 1 hold. $\mathcal{T}$axoSAE reads non-terminal concepts best, 15.6–33.8 points above the strongest baseline, and represents each non-terminal concept with 1.0–1.3 effective features, against more than 100 for the baselines. It also gives the best control at both levels, and its slots remove a single concept cleanly: deactivating a terminal or intermediate node drops accuracy on that node's own questions to 0.3% while keeping at least 99.8% on the others, whereas DiffMean cannot bring a node's own accuracy below 5% within the tested strengths. The absolute swap success is lower than on GPT-2 for all systems (60.5% and 13.7% for $\mathcal{T}$axoSAE), which may partly reflect the untuned layer.

Layer Sweep on Qwen. To see how the untuned layer choice affects these results, we train $\mathcal{T}$axoSAE on each even layer of Qwen2.5-0.5B with the same settings and test readout and swaps over the grid $\{20, 50, 100, 150, 200, 300, 500, 1000\}$ (Table 5).

	Bio (relation taxonomy)	**Animal** (entity taxonomy)
Nodes at L_1 / L_2 / L_3	18 / 6 / 3	18 / 6 / 2
Complementary elements	20 people	20 properties
Terminal facts	$20 \times 18 = 360$	$18 \times 20 = 360$
SAE training questions	14,400 (10 templates, 80 name forms)	25,014 (54 templates, 108 animal synonyms)
Readout, L_1 questions	800 in-distribution / 800 new template / 800 new name / 720 both new	1,080 in-distribution / 720 new template / 720 new synonym / 720 both new
Readout, L_2 questions	240 training templates / 240 new templates / 800 extra templates	600 seen templates / 240 new templates / 600 new synonyms / 240 both new / 1,440 extra surface forms
Readout, L_3 questions	120 training templates / 120 new templates / 600 extra templates	200 seen templates / 80 new templates / 200 new synonyms / 80 both new / 720 extra surface forms
Terminal-level swap pairs	2,550	1,745
Intermediate-level swap pairs (across top-level nodes)	600	347

Table 3: Dataset statistics. Only terminal-level questions are used to train $\mathcal{T}$axoSAE; intermediate- and top-level questions appear only in the readout tests.

	Diagonal Acc (%) ↑						EffFeat ↓			Top1C (%) ↑			Swap (%) ↑	
SAE	$L_1{\to}L_1$	$L_1{\to}L_2$	$L_1{\to}L_3$	$L_2{\to}L_2$	$L_2{\to}L_3$	$L_3{\to}L_3$	L_1	L_2	L_3	L_1	L_2	L_3	L_1	L_2
Bio relations, Qwen2.5-0.5B (layer 12)														
Random Unsup. SAE	1.7	0.0	0.0	0.0	0.0	0.0	138.6	135.9	132.5	6.4	7.5	6.8	1.1	0.3
DiffMean Unsup. SAE	51.5	73.5	60.6	48.9	56.2	55.8	138.6	135.9	132.5	6.4	7.5	6.8	44.1	4.3
Hybrid SAE	**100.0**	63.6	66.2	52.1	46.5	49.2	**3.4**	123.4	101.3	**71.9**	4.6	8.1	**60.5**	4.0
$\mathcal{T}$axoSAE	**100.0**	**100.0**	**100.0**	**67.7**	**88.9**	**82.5**	**3.4**	**1.3**	**1.0**	**71.9**	**95.3**	**99.3**	**60.5**	**13.7**
$\Delta(\mathcal{T} - \text{Rand.})$	↑98.3	↑100.0	↑100.0	↑67.7	↑88.9	↑82.5	↓135.2	↓134.7	↓131.5	↑65.5	↑87.7	↑92.5	↑59.5	↑13.4

Table 4: Results on Bio with Qwen2.5-0.5B (layer 12), in the format of Table 1. Swap: L_1 Strict accuracy over all pairs and L_2 Target-only coverage@$|U|$ over pairs from different top-level nodes, each system at its best α. Hybrid SAE shares the L_1 module of $\mathcal{T}$axoSAE, so its L_1 columns are identical. Bold/underline: best/second-best; Δ: $\mathcal{T}$axoSAE minus Random Unsup. SAE. The conclusions of Table 1 hold on Qwen.

Readout is high at every layer, while control improves with depth up to layers 16–18 (L_1 swap 76.2%, L_2 swap 36.5%), 15.7 and 22.8 points above layer 12, and declines after that, as on GPT-2.

C Unseen Terminal Concepts

A taxonomy should let the upper levels place a concept that no terminal slot was trained on. We ask the Animal model (layer 9) about 24 animals outside its 18 terminal concepts, four per intermediate node (e.g., *pig*, *dog*, *horse*, and *lion* for *mammal*), filling the 100 question templates of a trained terminal concept with the new name, and read the SAEs as in the readout test (Table 7). No terminal slot fits these animals: the highest L_1 score halves relative to the trained concepts, and 72.5% of the questions fall below the 5th percentile of the trained ones. Because the L_1 slots are trained as an 18-way classifier without a "none" option, the weaker slot still wins most of the time, usually that of the closest sibling (*pig*→*cow*, *newt*→*salamander*, *tuna*→*salmon*). The upper levels still recover the animal's place in the taxonomy: the L_2 slots of $\mathcal{T}$axoSAE read the right intermediate node for 63.9% of the questions and the L_3 slots the right top-level node for 76.1%, against 26.0% and 56.8% for DiffMean Unsup. SAE. Hybrid SAE, which shares the terminal-level SAE of $\mathcal{T}$axoSAE, is at chance at L_2 (16.7%), so the gain comes from the supervised upper levels rather than from the L_1 read; when L_1 points to a terminal concept under another intermediate node, the L_2 slots of $\mathcal{T}$axoSAE still read the right intermediate node for 31.1% of the questions, against 1.4% for Hybrid SAE. Errors concentrate on a few animals (*sparrow*, *owl*, *goldfish*, *cod*, *ant*, *slug*), while 12 of the 24 animals are classified correctly for at least 99% of their questions.

On Bio, the terminal concepts are relations, so

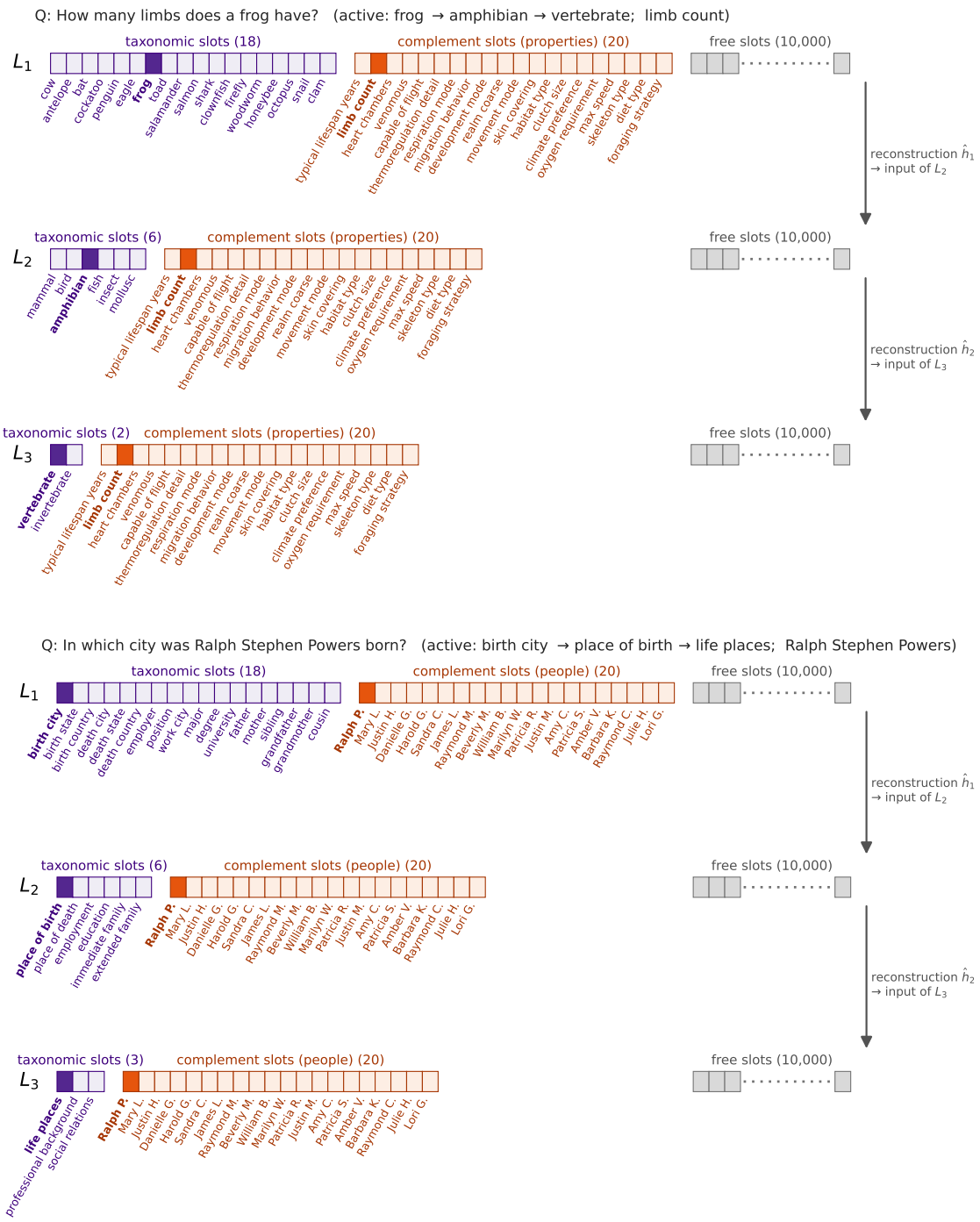


Figure 8: Slot layout of the three stacked SAEs on Animal (top) and Bio (bottom). Each level has one taxonomic slot per node at that level, one complement slot per complementary element, and 10,000 free slots; SAE_{k+1} reads the reconstruction of SAE_k. Dark cells mark the slots that the example question should activate: its terminal node and that node's hypernyms in the taxonomic slots, and its complementary element at every level.

we add 24 relations outside the 18 trained terminal relations, four under each intermediate relation (e.g., *uncle*, *aunt*, *nephew*, and *niece* under *extended family*), each asked about the 20 people with five question templates. Table 6 lists all unseen concepts. The L_1 slots again do not fit them (98.0% of the questions fall below the 5th percentile of the trained ones), while the upper levels of $\mathcal{T}$axoSAE read the right intermediate relation for 91.4% of the questions and the right top-level relation for 94.8%, against 75.9% and 69.8% for the strongest baselines (Table 7). Errors concentrate on *advi-*

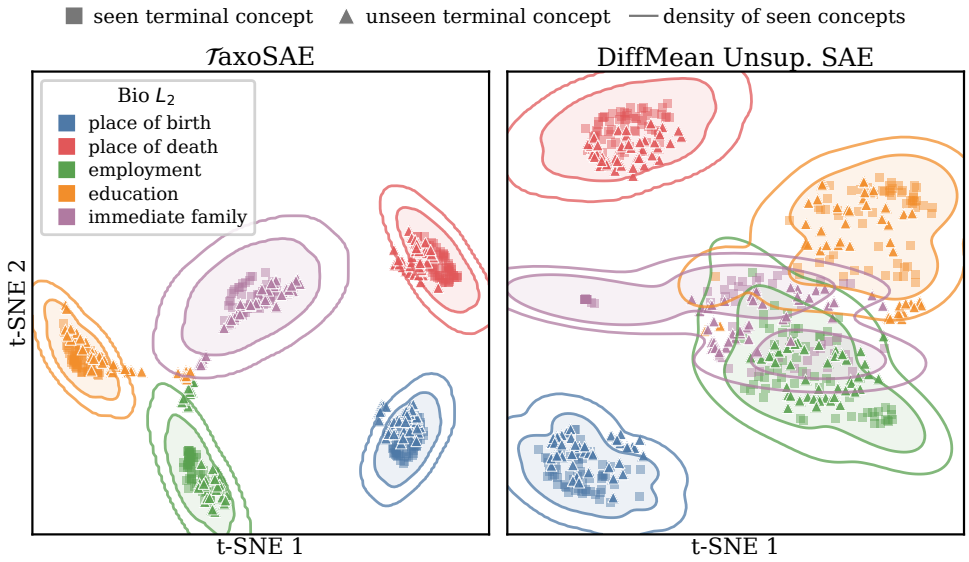


Figure 9: Bio counterpart of Fig. 3: t-SNE of the L_2 readout scores for seen and unseen terminal relations. The extended-family node is omitted for readability; Table 7 includes it.

	Diagonal Acc (%) ↑			Swap (%) ↑	
Layer	L_1	L_2	L_3	L_1	L_2
$L2$	95.1	**76.3**	74.3	3.9	0.3
$L4$	96.4	71.7	71.3	10.1	1.0
$L6$	**100.0**	74.4	69.5	26.2	0.5
$L8$	99.9	75.9	**82.5**	49.6	7.5
$L10$	**100.0**	72.2	67.7	50.3	9.2
$L12$	**100.0**	67.7	**82.5**	60.5	13.7
$L14$	**100.0**	66.4	65.3	56.3	16.0
$L16$	**100.0**	66.3	73.3	**76.2**	26.7
$L18$	**100.0**	65.6	66.3	72.1	**36.5**
$L20$	**100.0**	62.8	71.8	70.8	24.2
$L22$	**100.0**	70.1	79.7	55.5	11.0

Table 5: Layer-wise performance of TaxoSAE on Bio with Qwen2.5-0.5B (24 layers), in the format of Table 19. Shading marks layer 12, used in Table 4. Readout is stable across layers, and control is best at layers 16–18.

sor, which is mostly read as *immediate family* (its questions ask "who is X's advisor", like the family relations), and on *uncle*, *nephew*, and *niece*, which are partly read as *immediate family* or abstained on. Figure 9 shows the Bio counterpart of Fig. 3.

D Unseen Phrasings

Beyond evaluating unseen non-terminal nodes, we break down the pooled terminal-question results

L_2 node	Unseen terminal concepts
Animal (24 animals, 100 questions each)	
mammal	pig, dog, horse, lion
bird	sparrow, owl, parrot, pigeon
amphibian	newt, axolotl, caecilian, mudpuppy
fish	tuna, trout, goldfish, cod
insect	ant, butterfly, beetle, mosquito
mollusc	squid, oyster, slug, mussel
Bio (24 relations, 20 people × 5 templates each)	
place of birth	birth hospital, neighborhood, county, continent
place of death	death hospital, neighborhood, county, continent
employment	department, supervisor, work state, work country
education	high school, graduation year, advisor, minor
immediate family	spouse, son, daughter, child
extended family	uncle, aunt, nephew, niece

Table 6: Unseen terminal concepts, four per intermediate node. None of them appears in SAE training; each unseen concept is compared with 100 questions of each trained terminal concept.

($L_1 \to L_1/L_2/L_3$) in Table 1 into in-distribution (ID) evaluation with seen question templates and out-of-distribution (OOD) evaluation with rephrased questions or synonyms. Figure 10 shows that TaxoSAE maintains high diagonal accuracy and its advantage over baselines across datasets under ID and OOD settings. On Animal, OOD accuracy drops with unseen synonyms, but TaxoSAE retains a clear lead at non-terminal levels. On Bio, it maintains near-perfect accuracy under both settings.

E Complement Slot Readout

Besides the taxonomic slots, every SAE of TaxoSAE has one complement slot per comple-

SAE	L_1 read		Upper levels read (%) ↑		
	Top score	Abstain (%)	L_2	L_3	$L_2 \mid L_1$ off
Animal (layer 9)					
DiffMean Unsup. SAE	0.5×	27.6	26.0	56.8	28.2
Hybrid SAE	0.5×	12.9	16.7	50.5	1.4
$\mathcal{T}$axoSAE	0.5×	12.9	**63.9**	**76.1**	**31.1**
Chance	–	–	16.7	50.0	–
Bio (layer 6)					
DiffMean Unsup. SAE	0.6×	11.7	75.9	59.3	68.1
Hybrid SAE	0.5×	10.4	54.8	69.8	26.2
$\mathcal{T}$axoSAE	0.5×	10.4	**91.4**	**94.8**	**68.4**
Chance	–	–	16.7	33.3	–

Table 7: Readout of unseen terminal concepts (Table 6). Top score: median of the highest L_1 score, relative to the trained concepts. $L_2 \mid L_1$ off: L_2 accuracy on questions whose L_1 read falls in another intermediate node. Hybrid SAE shares the L_1 module of $\mathcal{T}$axoSAE. The upper levels of $\mathcal{T}$axoSAE place most unseen concepts under the right intermediate and top-level node on both datasets.

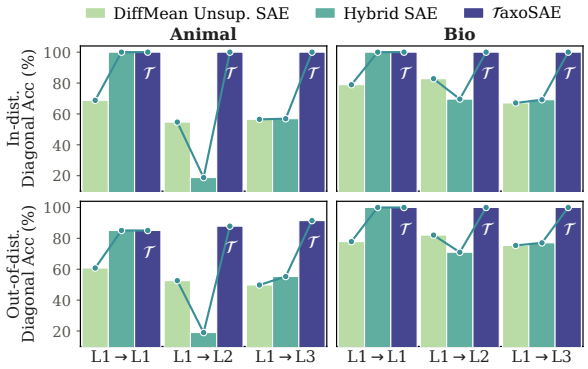


Figure 10: Diagonal accuracy on terminal-level questions read at L_1–L_3, with seen templates (ID, top) and unseen rephrasings or synonyms (OOD, bottom). $\mathcal{T}$axoSAE is best at every level in both settings.

mentary element: a person in Bio and a property in Animal (Sec. 4). We test whether the complement slots of SAE_1 read out the complementary element of a question, in the same way as Test A: on terminal-level questions, the complement unit with the largest activation must be that of the question's complementary element, with the same 10% abstention margin and macro accuracy over elements. For DiffMean, the unit of each element is the feature of the unsupervised terminal-level SAE selected by difference-in-means, and Random Unsup. SAE is the mean over four fixed assignments. Hybrid SAE shares the terminal-level SAE of $\mathcal{T}$axoSAE, so its results are identical to ours and are omitted. Table 8 shows that these slots read the complementary element almost perfectly on both datasets, including on question templates, names, and synonyms unseen in training.

SAE	In-dist.	New template	New name/syn.	Both new
Bio (20 people)				
Random Unsup. SAE	3.7	3.6	4.2	3.5
DiffMean Unsup. SAE	40.6	39.0	39.6	38.9
$\mathcal{T}$axoSAE	**100.0**	**100.0**	**100.0**	**99.3**
Animal (20 properties)				
Random Unsup. SAE	4.2	4.9	4.4	4.7
DiffMean Unsup. SAE	84.9	83.9	87.8	84.7
$\mathcal{T}$axoSAE	**100.0**	**93.8**	**100.0**	**93.8**

Table 8: Complement readout (Diagonal Acc, %) on terminal-level questions, by question tier: in-distribution, new templates, new names (Bio) or synonyms (Animal), and both new.

F Ablation Study

F.1 Loss Ablation

Complete Loss Ablation Results. Tables 9 and 10 report the complete results of Tests A–E on Bio. Each variant is initialized from the same Stage 1 checkpoint and retrained in Stage 2 with one loss component removed. Table 9 reports diagonal readout accuracy and activation concentration across taxonomy levels, while Table 10 reports terminal- and intermediate-level swapping and selectivity, with swap results evaluated at each variant's best α. The detailed results reveal trade-offs obscured by aggregate scores: removing taxonomic–free orthogonality improves several concentration metrics but lowers intermediate-level swap coverage from 41.8% to 24.2%. These results reinforce that stronger performance on individual readout or concentration metrics does not necessarily translate into better steering across levels.

F.2 Value head Ablation

Value Head Ablation. We replace the bilinear value head of § 4 with the per-slot head of AlignSAE (Yang et al., 2025): one head per taxonomic slot, which reads only the scalar activation of its own slot and is selected by the node of the question. As in the loss ablation, Stage 2 is retrained from the same Stage 1 checkpoint with all other settings unchanged. Table 11 shows that readout, concentration, and selectivity barely change, but intermediate-level swap drops on both datasets (Bio 41.8% → 38.3%; Animal 27.6% → 22.1%). These drops exceed the variation of $\mathcal{T}$axoSAE across three training seeds (standard deviation 1.5 on Bio and 2.4 on Animal), whereas the changes in terminal-level swap stay within that variation. The bilinear head thus mainly helps edits at the intermediate level, where an edited slot must change the

	(A) Diagonal Acc (%) ↑						(D) EffFeat ↓			(D) Top1C (%) ↑		
Variant	$L_1 \to L_1$	$L_1 \to L_2$	$L_1 \to L_3$	$L_2 \to L_2$	$L_2 \to L_3$	$L_3 \to L_3$	L_1	L_2	L_3	L_1	L_2	L_3
$\mathcal{T}$axoSAE	**100.0**	**100.0**	**100.0**	**69.8**	92.0	87.8	3.21	1.47	1.06	74.8	91.9	99.1
w/o taxonomic alignment	0.0	5.4	0.0	9.8	4.0	4.7	4.32	3.11	2.40	26.8	58.6	60.5
w/o complement alignment	**100.0**	**100.0**	**100.0**	67.6	85.1	80.7	3.56	1.60	1.18	70.2	89.0	95.8
w/o value loss	**100.0**	**100.0**	**100.0**	68.1	76.9	78.3	4.02	1.70	1.28	64.9	86.5	93.5
w/o taxonomic–complement ortho.	**100.0**	**100.0**	**100.0**	69.0	88.9	88.0	2.84	1.41	**1.05**	79.6	93.0	**99.3**
w/o taxonomic–free ortho.	**100.0**	**100.0**	**100.0**	67.2	**92.9**	**92.3**	**2.16**	**1.32**	1.06	**86.0**	**94.7**	99.0

Table 9: Loss ablation on Bio (layer 6): taxonomy readout and activation concentration. Each variant retrains Stage 2 of $\mathcal{T}$axoSAE from the same Stage 1 checkpoint with one loss component removed. L_1/L_2/L_3 denote terminal/intermediate/top levels. (A) Diagonal Acc: $L_q \to L_r$ denotes readout of level-r taxonomic elements from level-q questions. (D) EffFeat and Top1C measure activation concentration. Bold marks the best value in each column. Removing taxonomic alignment collapses readout; the other losses change it far less.

	(B) L_1 swap			(C) L_2 swap			(E) Selectivity (%)			
Variant	α	Strict (%) ↑	In-set (%) ↑	α	Cov. (%) ↑	Exact (%) ↑	L_1 Own ↓	L_1 Other ↑	L_2 Own ↓	L_2 Other ↑
$\mathcal{T}$axoSAE	100	84.7	88.1	100	41.8	41.8	0.0	**100.0**	**0.0**	**100.0**
w/o taxonomic alignment	200	1.3	2.4	20	2.2	2.2	77.8	77.8	75.0	75.7
w/o complement alignment	100	60.1	63.7	100	**42.0**	**42.0**	0.8	**100.0**	**0.0**	**100.0**
w/o value loss	100	82.0	82.8	200	15.2	15.2	**0.0**	**100.0**	0.8	99.6
w/o taxonomic–complement ortho.	100	80.6	86.5	100	37.0	37.0	4.7	**100.0**	**0.0**	**100.0**
w/o taxonomic–free ortho.	200	**88.6**	**90.5**	200	24.2	24.2	**0.0**	**100.0**	**0.0**	**100.0**

Table 10: Loss ablation on Bio (layer 6): steering and selectivity for the same variants as Table 9. Each swap test uses the variant's best steering strength α. (B) L_1 swap: strict and in-target-set accuracy. (C) L_2 swap on pairs from different top-level nodes: Target-only coverage@$|U|$ and Changed exact-set@$|U|$. (E) Selectivity: Own (↓) / Other (↑) exact match when the taxonomic slot corresponding to one L_1 or L_2 element is ablated. Bold marks the best value in each column. Removing any loss lowers at least one swap score, while selectivity is preserved except without taxonomic alignment.

	Diagonal Acc (%) ↑			Swap (%) ↑		EffFeat ↓
Value head	$L_2 \to L_2$	$L_2 \to L_3$	$L_3 \to L_3$	L_1	L_2	L_2
Bio (layer 6)						
Bilinear ($\mathcal{T}$axoSAE)	69.8	92.0	87.8	84.7	**41.8**	1.47
Per-slot	69.5	91.5	87.2	77.1	38.3	1.47
Animal (layer 9)						
Bilinear ($\mathcal{T}$axoSAE)	47.9	86.6	50.6	43.6	**27.6**	1.68
Per-slot	49.2	86.6	50.6	40.7	22.1	1.68

Table 11: Value head ablation. Per-slot: one head per taxonomic slot, reading only that slot, as in AlignSAE. Swap: terminal-level Strict accuracy and intermediate-level Target-only coverage@$|U|$ on pairs from different top-level nodes, each at the best α. Terminal-level readout ($L_1 \to L_1/L_2/L_3$) is identical for the two heads on Bio (100%) and within 0.2 points on Animal.

answers of all terminal nodes below it.

G Flat AlignSAE Baseline

To test whether the stack is needed, we train a flat AlignSAE (Yang et al., 2025) that binds all levels in a single SAE (Fig. 11). It reads $h^{(0)}$ directly and has one taxonomic slot for every node of every level, 20 complement slots, and 10,000 free slots. Each level's group of taxonomic slots is supervised with that level's label of the question, derived from the taxonomy, with no stacking and no derived (`is-a`) targets. All other losses, weights, and the two-stage schedule (100 + 500 epochs) follow $\mathcal{T}$axoSAE, and the tests are run in the same way, editing the slots of the swapped level only.

Table 12 reports the results on Bio (layer 6). Readout is close to that of $\mathcal{T}$axoSAE: both read terminal-level questions perfectly at every level, and the flat SAE reads non-terminal questions about as well. Control is far weaker. A terminal swap succeeds for only 5.5% of the questions, against 84.7% for $\mathcal{T}$axoSAE: editing a terminal slot leaves the intermediate and top-level slots of the same SAE on the source concept, so after a swap across intermediate nodes the flat SAE still reads the source's intermediate node in every case. In $\mathcal{T}$axoSAE, the edited reconstruction feeds the higher SAEs, which read the target's intermediate node in every case. The intermediate swap shows

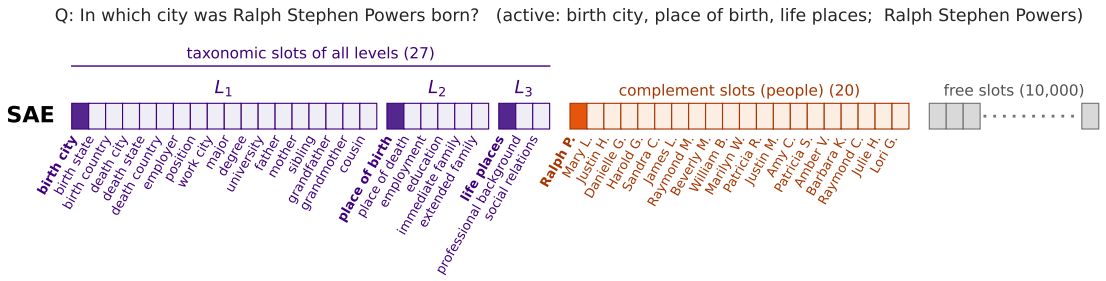


Figure 11: Slot layout of the flat AlignSAE baseline on Bio: a single SAE whose taxonomic slots cover all three levels (L_1/L_2/L_3: 18/6/3), followed by complement and free slots. Dark cells mark the slots that the example question should activate. Compare the stacked layout of $\mathcal{T}$axoSAE in Fig. 8.

	(A) Diagonal Acc (%) ↑			(B) L_1 swap ↑		(C) L_2 swap ↑	
SAE	$L_2 \to L_2$	$L_2 \to L_3$	$L_3 \to L_3$	Strict	L_2 read	Cov.	L_3 read
Flat AlignSAE	69.3	89.0	**91.7**	5.5	0.0	18.3	0.0
$\mathcal{T}$axoSAE	**69.8**	**92.0**	87.8	**84.7**	**100.0**	**41.8**	**100.0**

Table 12: Flat AlignSAE vs. $\mathcal{T}$axoSAE on Bio (layer 6). Both read terminal-level questions perfectly at every level ($L_1 \to L_1/L_2/L_3$, not shown). Swaps use each model's best α (flat: 20 for L_1, 50 for L_2; $\mathcal{T}$axoSAE: 100 for both). L_2 read: the SAE reads the target's intermediate node after a terminal swap across intermediate nodes; L_3 read: it reads the target's top-level node after an intermediate swap. For the flat SAE, both reads use the edited code itself, which has no higher level to pass the edit to. Readout is similar, but only $\mathcal{T}$axoSAE keeps the levels consistent after an edit.

the same gap (18.3% vs. 41.8%), and the flat SAE never reads the target's top-level node, since editing its intermediate slot leaves its top-level slots unchanged. Supervision alone thus suffices for readout, while consistent control across levels requires the stacked structure.

H Swap Success Results

Evaluation details. In the readout test (Test A), a read counts as wrong when the most active unit is not clearly ahead, i.e., its activation exceeds the second highest by less than 10%. A swap encodes the hidden state with the SAE of the edited level, sets the unit of the source node to 0 and that of the target node to the strength α, and decodes. For a terminal-level swap the edit is made in SAE_1; for an intermediate-level swap the hidden state is passed through SAE_1 and the edit is made in SAE_2. On Bio, swaps follow § 5.3: only the last prompt token is edited, and its hidden state is replaced with the decoded state $\mathrm{Dec}(z')$. On Animal, the edit is applied at every prompt position in an error-preserving form that keeps the SAE reconstruction error: each hidden state h becomes $h + \mathrm{Dec}(z') - \mathrm{Dec}(z)$, where z and z' are its activations before and after the edit. Deactivation (Test E) always uses the error-preserving form at the last prompt token. We sweep $\alpha \in \{20, 50, 100, 200, 500\}$ on Bio and $\alpha \in \{10, 20, 50, 100, 150, 200, 300, 500\}$ on Animal and report each system at its best α. For the selectivity test (Test E), the removal strength is swept over $\{1, 2, 5, 10, 20, 40, 80, 160\}$, and exact match requires every answer token to be the most likely one under teacher forcing.

Complete Swap Results. Table 13 reports answer performance and taxonomic readout after steering on Bio and Animal, with each system evaluated at its best steering strength α. Terminal-level swaps cover all source–target pairs, while intermediate-level swaps use pairs from different top-level nodes. In addition to target-answer metrics, we evaluate whether the edited activations encode the target element's hypernyms. After terminal-level swaps, $\mathcal{T}$axoSAE achieves 100% intermediate-level readout on both datasets and top-level readout of 93.2% on Bio and 99.2% on Animal. After intermediate-level swaps, top-level readout reaches 100% on both datasets. Although Hybrid SAE shares the same terminal-level answer performance, its non-terminal readout is substantially weaker. These results show that steering with $\mathcal{T}$axoSAE aligns higher-level readout with the target element, even when the generated answer does not always match the target.

I Selectivity Matrices with Concept Labels

Figure 12 repeats Fig. 6 with the concept of each row and column.

		L_1 swap (all pairs)					L_2 swap (different top-level nodes)	
		Answer (%) ↑		Read after edit (%) ↑			Answer (%) ↑	Read after edit (%) ↑
SAE	α	Strict	In-set	L_2	L_3	α	Target-only	L_3
Bio (L_1: 2,550 pairs; L_2: 600 pairs)								
Random Unsup. SAE	–	2.2	4.0	18.0	34.3	–	0.0	31.0
DiffMean Unsup. SAE	200	81.8	83.0	71.1	55.9	200	12.2	69.5
Hybrid SAE	100	**84.7**	**88.1**	66.2	44.6	200	16.3	65.8
$\mathcal{T}$axoSAE	100	**84.7**	**88.1**	**100.0**	**93.2**	100	**41.8**	**100.0**
Animal (L_1: 1,745 pairs; L_2: 347 pairs)								
Random Unsup. SAE	–	5.3	24.4	18.8	64.6	–	9.4	44.5
DiffMean Unsup. SAE	150	40.5	54.7	53.0	56.4	100	25.5	69.7
Hybrid SAE	150	**43.6**	**59.3**	68.4	59.3	150	21.4	66.6
$\mathcal{T}$axoSAE	150	**43.6**	**59.3**	**100.0**	**99.2**	200	**27.6**	**100.0**

Table 13: Swap results with the levels the SAE reads after the edit, each system at its best edit strength α. L_1 swap over all source–target pairs: Strict, the answer is the target terminal node's value; In-set, any value the target terminal node holds; Read after edit, the SAE reads the target's intermediate (L_2) and top-level (L_3) hypernyms. L_2 swap on pairs from different top-level nodes: Target-only coverage@$|U|$, and whether the SAE reads the target intermediate node's top-level hypernym (L_3) after the edit. Hybrid SAE shares the L_1 module of $\mathcal{T}$axoSAE, so its L_1 answers are identical, but it reads L_2 and L_3 with its own unsupervised SAEs; Random Unsup. SAE is the mean over four fixed assignments. After every swap, $\mathcal{T}$axoSAE reads the target's hypernyms in 93.2–100% of cases, against at most 71.1% for the baselines.

J Baseline Implementation Details

Architecture and Training. Each unsupervised SAE has 10,000 features, a ReLU encoder, and a linear decoder:

$$z_\ell = \mathrm{ReLU}(W_\ell^{\mathrm{enc}} x_\ell + b_\ell^{\mathrm{enc}}), \tag{9}$$

$$\hat{x}_\ell = W_\ell^{\mathrm{dec}} z_\ell + b_\ell^{\mathrm{dec}}, \tag{10}$$

where $x_1 = h$ is the frozen LM activation and $x_\ell = \hat{x}_{\ell-1}$ for $\ell > 1$. Each unsupervised SAE minimizes reconstruction loss plus an L1 penalty on its activations, with coefficient 10^{-3}. No supervised, taxonomic, or orthogonality losses are applied to these SAEs.

We train for 400 epochs with a batch size of 64 and a learning rate of 3×10^{-4}, decayed to 10^{-6} using a cosine schedule. All baselines use the same training activations as $\mathcal{T}$axoSAE, extracted from transformer layer 6 of GPT-2 for Bio and transformer layer 9 for Animal (App. L). For Hybrid SAE, these settings apply only to the two unsupervised SAEs at the upper taxonomy levels; the terminal-level SAE is copied from $\mathcal{T}$axoSAE and frozen. Random Unsup. SAE reuses the trained DiffMean dictionaries without additional training.

Differential Activation Scoring. For each taxonomic element g, we select a feature from the SAE at its corresponding level. Let $\mathcal{D}_g$ denote the terminal-level training questions associated with g, and define

$$\mu_{g,f} = \frac{1}{|\mathcal{D}_g|} \sum_{x \in \mathcal{D}_g} z_{\ell(g),f}(x), \tag{11}$$

$$s(g,f) = \mu_{g,f} - \max_{\substack{g' \neq g \\ \ell(g')=\ell(g)}} \mu_{g',f}. \tag{12}$$

Here, $\ell(g)$ is the taxonomy level of g. The selected feature is $f_g = \arg\max_f s(g,f)$. This max-margin variant of difference-in-means favors features that respond more strongly to g than to any other element at the same level.

For non-terminal elements, labels are derived from the terminal labels using the known taxonomy. No non-terminal questions are used for feature selection. Thus, SAE dictionary training is unsupervised, while post-hoc feature selection uses labels. Hybrid SAE applies the same procedure only at the intermediate and top levels.

Fixed Feature Assignment. For Random Unsup. SAE, we evaluate four assignments: one assigns the ith taxonomic element to the ith feature at its level, and three randomly assign distinct features using seeds 1–3. We report the mean across these four assignments. Since Random and DiffMean Unsup. SAE share the same dictionaries, their Eff-Feat and Top1C scores are identical; these metrics do not depend on feature assignment.

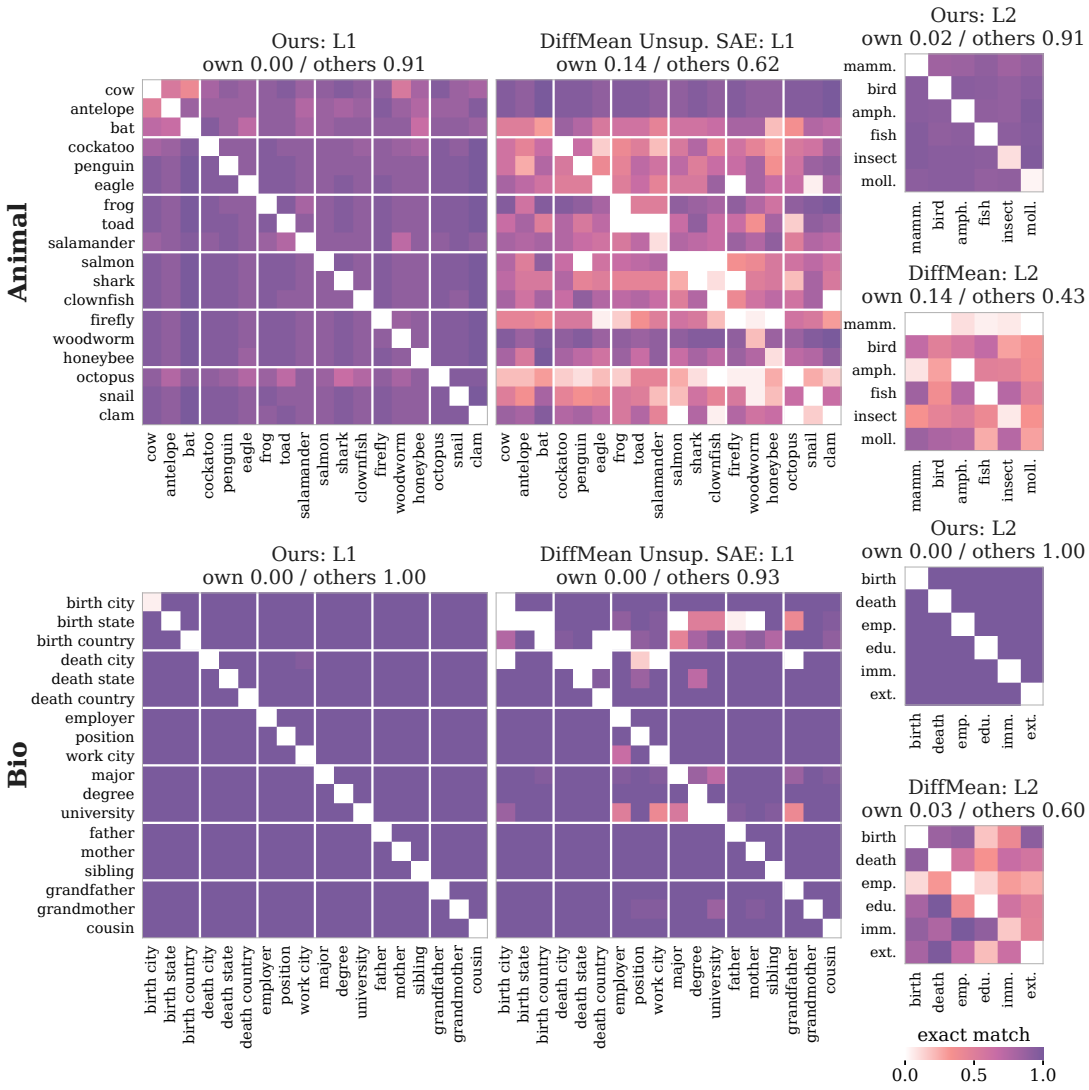


Figure 12: Fig. 6 with concept labels. Row i deactivates the unit of element i; column j is the exact match on element j.

Feature Budget. All baselines use one feature per taxonomic element ($k = 1$) during evaluation, matching the single-slot budget of $\mathcal{T}$axoSAE. Although the five highest-scoring features are saved during feature selection, only the highest-scoring feature is used for readout and intervention.

K Training Details

K.1 Base Model Training

Training Data. For each dataset, the training corpus combines terminal-level factual QA, hypernymy QA, and plain-text descriptions of terminal-level facts. Bio contains 14,400 factual QA examples, 159 hypernymy QA examples, and 20 biographies; Animal contains 25,014 factual QA examples, 1,126 hypernymy QA examples, and 38 descriptions (Tables 14 and 15). Hypernymy QA maps terminal nodes to their intermediate and top-level hypernyms and intermediate nodes to top-level ones, including yes/no checks, and enumerates the hyponyms of intermediate or top-level nodes. These questions provide taxonomic structure without supplying factual answers for non-terminal nodes. No non-terminal factual QA is included, so answering such questions requires generalizing beyond the directly provided terminal-level facts.

Training Configuration. We fine-tune GPT-2 small separately on Bio and Animal using AdamW with $\epsilon = 10^{-6}$ (Tables 16 and 17). For Bio, we train for 13k steps with a batch size of 64, using only biographies for the first 1k steps and a 15% biography share thereafter. The learning rate is 5×10^{-4}, with 500 warm-up steps and cosine de-

Bio: training data	#
Terminal-level factual QA	14,400
Hypernymy QA	159
terminal→intermediate	90
terminal→top-level	36
intermediate→top-level	24
hyponym enumeration	9
Non-terminal factual QA	0
Biographies	20

Table 14: Base-model training data for Bio. Mapping questions include yes/no checks (e.g., "Is 'birth city' a place of birth fact?"); hyponym enumeration asks for the members of an intermediate or top-level node. Biographies are plain-text descriptions of terminal-level facts.

Animal: training data	#
Terminal-level factual QA	25,014
Hypernymy QA	1,126
terminal→intermediate	864
terminal→top-level	162
intermediate→top-level	66
hyponym enumeration	34
Non-terminal factual QA	0
Descriptions	38

Table 15: Base-model training data for Animal. Mapping questions include yes/no checks (e.g., "Is a cow a mammal?"); hyponym enumeration asks for the members of an intermediate or top-level node (e.g., "Which animals are mammals?"). Descriptions state terminal-level facts in plain text.

cay to 5×10^{-5}. For Animal, descriptions are mixed with QA data throughout 30k training steps, with a batch size of 32, a learning rate of 2×10^{-5}, and 1,000 warm-up steps. The loss is computed on answer tokens for QA and on all tokens for biographies and descriptions. With these per-dataset settings, each fine-tuned model answers all of its training factual and hypernymy questions correctly. After fine-tuning, we freeze the models and extract hidden states at the final prompt token from layer 6 for Bio and layer 9 for Animal (App. L).

K.2 $\mathcal{T}$axoSAE Training

Each of the three SAEs of $\mathcal{T}$axoSAE has 10,000 free slots, one taxonomic slot per node at its taxonomy level (18/6/3 for Bio and 18/6/2 for Animal), and 20 complement slots, i.e., people in Bio or properties in Animal. The bilinear value head uses a 64-dimensional taxonomy embedding and 128-dimensional projections.

Table 18 lists the loss weights, shared across Bio and Animal. Stage 1 runs for 100 epochs using only reconstruction and sparsity losses, with all other loss weights set to zero. Stage 2 starts from the Stage 1 checkpoint and trains for 500 epochs with all losses enabled. Both stages use AdamW, a learning rate of 3×10^{-4} with cosine decay to 10^{-6}, a batch size of 64, gradient clipping at 1.0, and seed 0, on the activations of the terminal-level training questions (14,400 for Bio; 25,014 for Animal). The reconstruction weight is 1.0 in both stages. L1 sparsity is computed separately for free, taxonomic, and complement slots, each with a weight of 10^{-3}.

In Stage 2, the taxonomic alignment loss $\mathcal{L}_{\text{taxo}}$ uses cross-entropy with a weight of 1.0 at the terminal level and 5.0 at the intermediate and top levels. Only terminal-level taxonomic alignment uses ground-truth labels; non-terminal targets are derived from terminal-level predictions through the taxonomy. The complement alignment loss $\mathcal{L}_{\text{comp}}$ and the value loss each receive a weight of 1.0. Taxonomic–complement orthogonality $\mathcal{L}_{\perp}^{\text{tc}}$ has a weight of 0.5, while taxonomic–free orthogonality $\mathcal{L}_{\perp}^{\text{tf}}$ has a weight of 10^{-2}. Following AlignSAE, the latter is computed as the sum of squared cross-covariances over all taxonomic–free slot pairs.

Bio: fine-tuning	
Base model	GPT-2 small
Training steps	13k
biographies only	first 1k
biography share afterwards	15%
Batch size	64
Optimizer	AdamW (ϵ=10^{-6})
Learning rate	5×10^{-4}
LR schedule	cosine to 5×10^{-5}
Warm-up steps	500
Weight decay	0.1
Max sequence length	384
Precision	bf16
Seed	0
Activation layer	6

Table 16: Base-model fine-tuning for Bio. GPT-2 small has 12 transformer layers and a hidden dimension of 768. The loss is computed on answer tokens for QA and on all tokens for biographies. Activations are the hidden state at the final prompt token.

L Layer-wise Performance

Layer-Wise Evaluation on Bio. Table 19 reports results for $\mathcal{T}$axoSAE trained separately on activations from the embedding output and each GPT-2 block. Diagonal accuracy measures same-

Animal: fine-tuning	
Base model	GPT-2 small
Training steps	30k
descriptions	mixed into the data
Batch size	32
Optimizer	AdamW (ϵ=10^{-6})
Learning rate	2×10^{-5}
LR schedule	cosine to 10^{-4}
Warm-up steps	1,000
Weight decay	0.01
Max sequence length	512
Precision	fp32
Activation layer	9

Table 17: Base-model fine-tuning for Animal. GPT-2 small has 12 transformer layers and a hidden dimension of 768. The loss is computed on answer tokens for QA and on all tokens for descriptions. Activations are the hidden state at the final prompt token.

Loss term	Stage 1	Stage 2
Reconstruction	1	1
Sparsity, free slots	10^{-3}	10^{-3}
Sparsity, taxonomic slots	10^{-3}	10^{-3}
Sparsity, complement slots	10^{-3}	10^{-3}
Taxonomic alignment, terminal (CE)	0	1.0
Taxonomic alignment, non-terminal (CE)	0	5.0
Complement alignment	0	1.0
Value loss	0	1.0
Taxonomic–complement orthogonality	0	0.5
Taxonomic–free orthogonality	0	10^{-2}

Table 18: Loss weights of $\mathcal{T}$axoSAE. Both datasets (**Bio** and **Animal**) use the same weights. Stage 2 is initialized from the Stage 1 checkpoint.

level taxonomic readout ($L_k \to L_k$), using pooled ID and OOD terminal-level questions and unseen intermediate- and top-level questions. Swap scores measure terminal-level strict accuracy over all pairs and intermediate-level target-only coverage@$|U|$ over pairs from different top-level nodes, each at its best steering strength α. Reconstruction NMSE is the terminal-level SAE's reconstruction error on training activations divided by activation variance.

Terminal-level readout reaches near-perfect accuracy in early layers, but strong steering emerges later. Terminal-level swapping peaks at layer 6 (84.7%), while intermediate-level swapping peaks at layer 7 (47.0%). Layer 6 gives the best terminal-level swap (84.7%), the criterion used in the main experiments. All systems use the same layer, and only terminal-level metrics were used for the choice; non-terminal and unseen-concept results were not.

	Diagonal Acc (%) ↑			Swap (%) ↑		↓
Layer	L_1	L_2	L_3	L_1	L_2	Recon NMSE
$L0$	5.3	21.3	35.5	2.9	0.3	0.372
$L1$	90.1	72.1	**98.0**	0.9	0.0	0.310
$L2$	**100.0**	69.0	79.8	11.4	0.3	0.252
$L3$	99.6	70.4	85.8	47.5	14.0	0.129
$L4$	**100.0**	71.8	89.0	65.5	35.2	0.082
$L5$	**100.0**	71.6	89.3	79.1	39.5	0.058
$L6$	**100.0**	69.8	87.8	**84.7**	41.8	0.045
$L7$	**100.0**	73.9	89.3	65.3	**47.0**	0.040
$L8$	**100.0**	**76.7**	89.0	72.6	39.3	0.038
$L9$	**100.0**	76.2	92.8	63.8	28.7	0.048
$L10$	**100.0**	76.6	91.0	61.2	23.7	0.092
$L11$	**100.0**	76.5	91.3	61.3	24.0	0.141
$L12$	**100.0**	76.0	92.5	53.5	7.7	**0.024**

Table 19: Layer-wise performance of $\mathcal{T}$axoSAE on Bio. Layer 0 is the embedding output. Bold marks column-best results; shading marks layer 6, which achieves the highest mean swap score. Steering emerges only from the middle layers.

Layer-Wise Evaluation on Animal. Table 20 repeats this evaluation on Animal. As on Bio, early layers support little steering: both swap scores stay below 22% up to layer 8. Layer 9, used in the main experiments, gives the best terminal-level readout (84.3%) and terminal-level swap (43.6%). Later layers read intermediate concepts better, and layer 10 reaches the highest intermediate-level swap (54.5%, against 27.6% at layer 9) and the highest mean swap score. Top-level readout stays near 50% at every layer, i.e., chance for the two top-level nodes.

M The Usage of AI Tools

In this work, large language models were used solely for auxiliary purposes, primarily to refine the language of the manuscript, including grammar correction, improving clarity, and replacing informal expressions with academic phrasing. Importantly, LLMs were not used for research ideation, hypothesis formulation, or experimental design. All conceptual contributions, experimental designs, and method design were independently developed and carried out by the authors.

	Diagonal Acc (%) ↑			Swap (%) ↑		↓
Layer	L_1	L_2	L_3	L_1	L_2	Recon NMSE
$L0$	3.8	17.9	50.0	5.1	5.2	0.967
$L1$	69.0	27.1	47.1	4.3	8.6	2.605
$L2$	76.4	46.2	50.4	4.1	6.2	1.775
$L3$	76.2	41.7	51.1	5.0	7.9	0.879
$L4$	75.3	36.7	50.7	6.0	7.6	0.824
$L5$	74.7	45.4	49.4	5.6	7.6	0.480
$L6$	75.4	46.2	53.8	11.0	10.3	0.204
$L7$	78.1	50.8	**55.4**	15.0	13.1	0.144
$L8$	78.6	51.3	51.4	21.6	11.4	0.076
$L9$	**84.3**	47.9	50.6	**43.6**	27.6	0.086
$L10$	82.6	91.2	49.9	38.8	**54.5**	0.115
$L11$	81.9	95.0	50.0	37.4	34.8	0.167
$L12$	82.1	**98.8**	50.3	39.8	30.3	**0.048**

Table 20: Layer-wise performance of $\mathcal{T}$axoSAE on Animal, in the format of Table 19. Layer 0 is the embedding output. Bold marks column-best results; shading marks layer 9, used in the main experiments. Steering emerges only from the middle layers.