Visual Exploration of Feature Relationships in Sparse Autoencoders with Curated Concepts

Xinyuan Yan
University of Utah
xinyuan.yan@utah.edu

Shusen Liu
Lawrence Livermore National Laboratory
liu42@llnl.gov

Kowshik Thopalli
Lawrence Livermore National Laboratory
thopalli1@llnl.gov

Bei Wang
University of Utah
beiwang@sci.utah.edu

Abstract

Sparse autoencoders (SAEs) have emerged as a powerful tool for uncovering interpretable features in large language models by learning sparse directions in their activation spaces. However, the sheer number of extracted directions renders comprehensive exploration intractable. Conventional embedding methods such as UMAP can reveal global organization but often introduce high-dimensional compression artifacts, overplotting, and misleading neighborhood distortions. In this work, we introduce a focused exploration framework that prioritizes curated concepts and their associated SAE features over exhaustive visualization of all features. We present an interactive visualization system that integrates topology-based visual encodings with dimensionality reduction to preserve both local and global relationships among selected features. This hybrid approach enables targeted, interpretable analysis of SAE behavior, supporting deeper insight into how concepts are represented across layers of large language models.

1 Introduction

Sparse autoencoders (SAEs) have emerged as a powerful technique for extracting interpretable features from large language models (LLMs), decomposing superposed neural representations into disentangled components [1, 2, 3, 4]. Recent work has demonstrated remarkable scalability, extracting millions of interpretable features from state-of-the-art models [5, 6, 7]. Yet this success introduces a paradox: the sheer number of learned sparse directions, often hundreds of thousands to millions, renders comprehensive exploration both computationally and cognitively intractable. Moreover, recent studies [8, 9] reveal that many SAE features are *polysemantic* (encoding multiple, unrelated concepts) or low-quality, raising fundamental questions about whether visualizing all features simultaneously is even meaningful.

In this work, we advocate a *focused exploration* paradigm that emphasizes carefully curated concept sets and their corresponding SAE features, rather than attempting to visualize all learned features at once [10, 11, 12]. By concentrating on well-defined concepts, researchers can avoid the confusion introduced by the vast number of noisy or potentially low-quality features and instead conduct targeted, hypothesis-driven investigations into concept representation and feature relationships.

We pursue three interconnected analytical objectives that are central to understanding the conceptual organization of SAE features. First, we examine how cosine similarity among features corresponds to semantic similarity, providing a quantitative validation of whether SAEs capture meaningful semantic structure [13]. Second, we evaluate how human-curated concept hierarchies are reflected in the

Preprint.

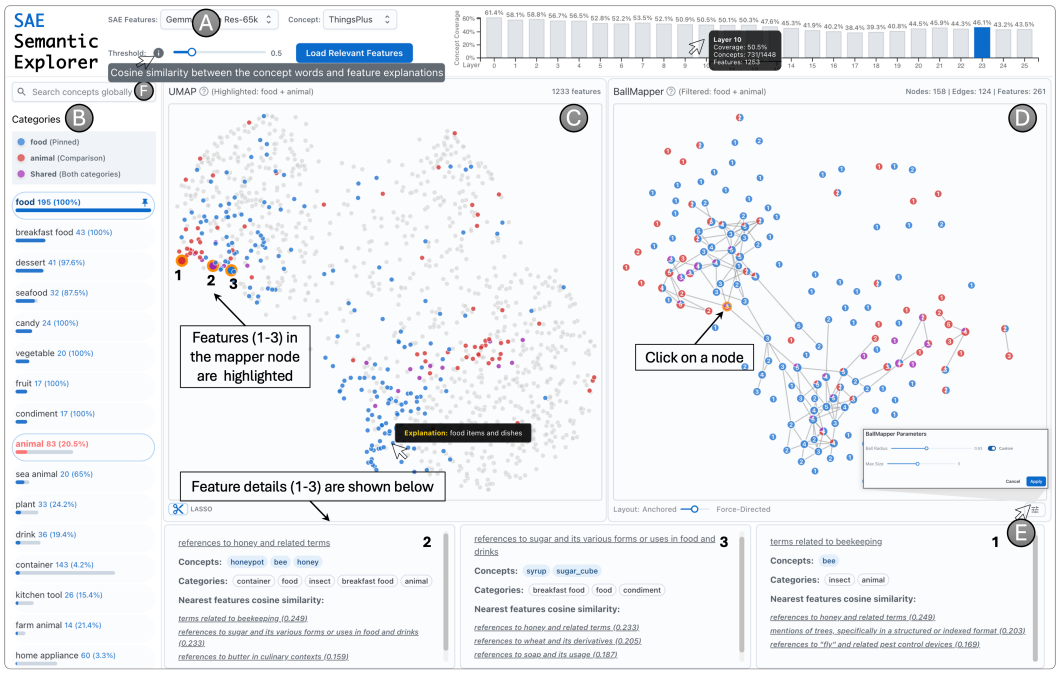

Figure 1: *SAE Semantic Explorer* interface. A. **Data view**. Left: SAE features, a concept set (words with assigned categories), and a cosine-similarity threshold for retrieving relevant features. Right: bar chart showing the number of discovered concepts per layer. B. **Category view**. For the selected layer (23), each row displays a category's feature count and its overlap with the pinned category *food*, facilitating comparison with *animal*. C. **UMAP view**. Retrieved features from the selected layer, with *food* and *animal* categories highlighted. D. **Ball Mapper view**. Topological graph showing the structural relationships among *food* and *animal* features. E. **Feature view**. Interactive panel displaying details of selected features via click or lasso selection. F. **Concept query**. Search interface for locating specific concepts.

SAE feature space, assessing whether models exhibit organizational patterns consistent with human understanding [14, 15]. Third, we analyze how both local and global structures evolve across model layers, as feature representations undergo systematic transformations with network depth [7, 16].

Existing SAE visualizations commonly employ dimensionality reduction methods such as UMAP [17], which compress thousands of dimensions into two. However, such extreme compression often introduces severe structural distortions that misrepresent neighborhood relationships [18], which are critical for interpretability and understanding semantic similarities between features. For example, similar features may appear artificially separated, while dissimilar ones may appear spuriously clustered, leading to misleading conclusions about feature organization.

To address these limitations, we introduce a topology-based visual encoding inspired by topological data analysis. Building on the ball mapper algorithm [19], our method constructs a network representation that preserves both local and global structural properties. Unlike projection-based embeddings that force features into 2D space, our topological representation maintains discrete feature clusters and explicitly encodes their interconnections, providing a more faithful depiction of similarity relationships among SAE features.

Finally, we present *SAE Semantic Explorer*, a visual analytics tool that integrates this topology-based encoding with dimensionality reduction techniques. Through multiple coordinated views, our tool enables interactive exploration of local, global, and cross-layer feature relationships, supporting flexible and hypothesis-driven analysis of SAEs. The code, datasets, and video demonstration are publicly available at `https://github.com/tdavislab/SAEExploration.git`.

2 Related Works

Foundations, evaluation, and analysis of SAEs. Sparse autoencoders (SAEs) have emerged as an unsupervised approach for decomposing latent representations into approximately monosemantic and interpretable features [20, 21]. Early studies demonstrate that SAEs can recover interpretable

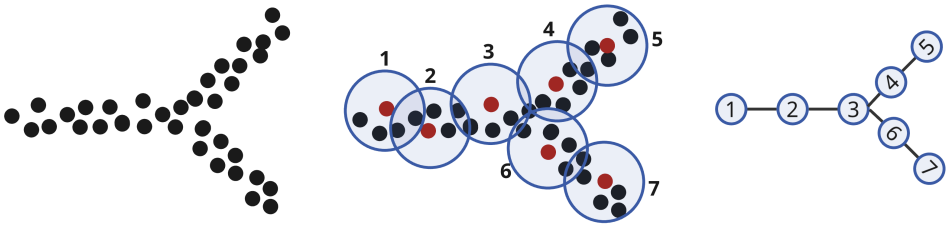

Figure 2: Ball mapper construction example. Left: Original point cloud. Middle: For a given radius ϵ, a subset of points (red) is selected as ball centers such that the resulting balls (1–7, blue) cover the entire dataset. Right: The resulting ball mapper graph represents each ball as a node, with an edge between two nodes if their corresponding balls share data points.

directions in both toy transformers and LLMs [1, 2, 4]. Subsequent research has advanced and scaled SAEs through architectural modifications [9], alternative activation functions [6, 22], and enhanced loss formulations [23, 24]. To interpret individual features, automated methods employ LLMs to summarize the tokens that most strongly activate each feature [22, 25], while SAEBench [26] provides a standardized benchmark for comprehensive quantitative evaluation. Complementary studies further analyze how SAE features evolve across network layers [7, 14] and across model scales [27], revealing characteristic trends in representation structure. Despite these interpretability gains, recent findings highlight instability in the learned features [28] and show that SAEs may underperform simpler baselines in representation steering tasks [29], raising questions about their practical reliability.

Visualizing SAE features. An emerging ecosystem of interactive visualization platforms supports the exploration and manipulation of SAE-learned features, including SAELens [10], Neuronpedia [11], and the interactive analyses presented in Anthropic's interpretability reports [30]. These systems demonstrate both the utility and the usability challenges of exploring thousands to millions of discovered directions. Such challenges are compounded by the use of dimensionality-reduction methods like UMAP [17], where distance distortion and overplotting can obscure meaningful structure [18]. In this work, we instead enable a more *focused* exploration of selected, conceptually meaningful features through curated concepts and a topology-driven analysis that faithfully captures structural relationships.

Mapper graph for model interpretability. The mapper graph [31] is a topological data analysis technique that summarizes the structure of a point cloud as a graph, where each node represents a cluster of points, and edges connect nodes with overlapping members. Mapper graphs have recently been leveraged to interpret model embeddings by revealing the topology of latent spaces. For instance, they have been used to capture the structural organization of hidden representations in image classifiers and large language models (LLMs) [32, 33, 34], to analyze how representations evolve across layers and during fine-tuning [32, 33], and to characterize the effects of adversarial perturbations [35]. In this work, we employ *ball mapper* [19], a variant of mapper with fewer parameters, to complement dimensionality reduction methods, providing a view that better preserves both local neighborhood relations and global structural patterns.

3 *SAE Semantic Explorer*: Interactive Concept Exploration

We develop a visual analytics tool, *SAE Semantic Explorer*, that enables human-in-the-loop exploration of SAE features (Figure 1).

Data description and preprocessing. The input data comprise SAE features (high-dimensional vectors) across layers, LLM-generated textual explanations for each feature [36], and a concept dataset containing concept words and their associated categories (e.g., the category *animal* includes *dog*, *cat*, etc.; a concept may belong to multiple categories). For each layer, relevant features for each concept are retrieved via the Neuronpedia API [11], which computes cosine similarity between concept and feature explanations in a sentence embedding space. By default, features with similarity scores above 0.5 are visualized, though users can adjust this threshold interactively.

Embedding view based on ball mapper. To better preserve local structure and mitigate distortions introduced by traditional dimensionality reduction, we introduce an embedding view grounded in

the ball mapper framework. Ball mapper [19] encodes the topology of a high-dimensional point cloud as a graph, where nodes represent local neighborhoods (balls) of points and edges denote overlaps between them. As illustrated in Figure 2, given a user-defined radius ϵ, a subset of points is selected (e.g., via a greedy procedure) such that balls centered at these points cover the entire dataset. Each ball defines a node, and two nodes are connected if their corresponding balls share points. By default, ϵ is estimated by computing all pairwise cosine distances and selecting the elbow point of the resulting distribution, following common practice [37]. To reduce visual clutter, we employ an adaptive variant: for a specified maximum node size, the ball radius is iteratively reduced by a factor η (default 0.9) until the constraint is satisfied. In our system, the maximum node size defaults to 5, though users can adjust this and other parameters interactively.

Interface design. The interface integrates multiple coordinated views. As shown in Figure 1, View A allows users to select SAE feature data, concept data, and a similarity threshold to retrieve concept-relevant features per layer. The resulting distribution of discovered concepts is displayed as a bar chart; clicking a bar loads the corresponding layer data in Views B–D.

For a selected layer, View B lists category information, including the number of relevant features associated with each category. View C presents a UMAP projection of features, supporting zoom, pan, hover, lasso, and selection, with the three nearest features highlighted on click. View D shows the ball mapper view, where users can drag, adjust parameters, and select nodes or edges. The ball mapper graph supports a smooth transition between two layouts: an *anchored* layout, in which nodes are positioned by the average location of their UMAP points for alignment, and a *force-directed* layout for a clearer, aesthetically pleasing visualization.

View E displays detailed information about selected features, including textual explanations, related concepts and categories, and nearby features within the same layer. Each explanation links directly to the Neuronpedia feature detail page [11]. View F provides a search interface for retrieving features associated with specific concepts.

To aid navigation, tooltips are available throughout the interface. Selecting a category in View B highlights its corresponding features in both the UMAP and ball mapper views. Users can pin a category to explore relationships among categories, after which others are sorted by shared features. When a comparison category is selected, features are consistently color-coded across UMAP and ball mapper views to reveal overlaps and distinctions.

4 Exploratory Analysis Results

In this section, we present insights obtained through the proposed visualization framework, highlighting how it facilitates understanding of the relationships between learned SAE features and human-interpretable concepts.

Concept sets. We analyze two hierarchical concept sets: (i) `THINGSplus` [38], a human-curated dataset containing 1,448 concepts organized into 53 categories (e.g., *mammals*, *food*), where each category includes a list of relevant concepts (e.g., *panda*, *cat*, *breadsticks*); and (ii) `Subjects`, a field-of-study-oriented set curated via an LLM, containing 1,683 concepts spanning eight disciplines (e.g., *mathematics*, *physics*) and their subtopics (e.g., *algebra*, *geometry*, *calculus* for mathematics). While most SAE visualization frameworks [6] focus on providing overviews of learned features, they often lack mechanisms for connecting those features to practitioner-relevant concepts. Our framework addresses this gap by allowing users to load concept sets and retrieve features aligned with specific concepts of interest, enabling targeted exploration. To demonstrate this capability, we analyze 65k open-source SAE features from each layer of the `gemma-2-2b` model [39], as provided by Gemma Scope [40]. Neuronpedia [11] supplies corresponding textual explanations for each feature, generated via the Auto-Interp framework [25], across 26 residual-stream layers.

Evolution of concept relationships across layers. Using our visualization framework, we observe several patterns in how concept relationships evolve across layers. For the `THINGSplus` concept set, early layers produce UMAP representations that are densely clustered; these gradually separate into two major clusters in the middle layers and later recombine into more integrated structures in deeper layers. This trajectory suggests that the model first learns broad, general representations, refines them into specialized concepts, and subsequently integrates them into cohesive, high-level abstractions—a pattern consistent with prior findings on the evolution of factual knowledge in LLMs [41]. The

ball mapper view, however, reveals a more nuanced picture: multiple smaller clusters emerge in the middle layers, indicating a diversity of feature specialization not fully captured by UMAP embedding.

For the `Subjects` concept set, this layer-wise differentiation is less pronounced: the UMAP embeddings remain relatively clustered, likely due to the abstract nature of academic domains, which exhibit weaker separability in feature space. Additionally, ball mapper neighborhoods highlight persistent local semantic relationships across layers, consistent with prior observations [42]. For instance, in Figure 3(A, bottom), querying *fox* consistently retrieves features related to *Fox News*, with *wolf* appearing as its nearest neighbor within the same mapper node.

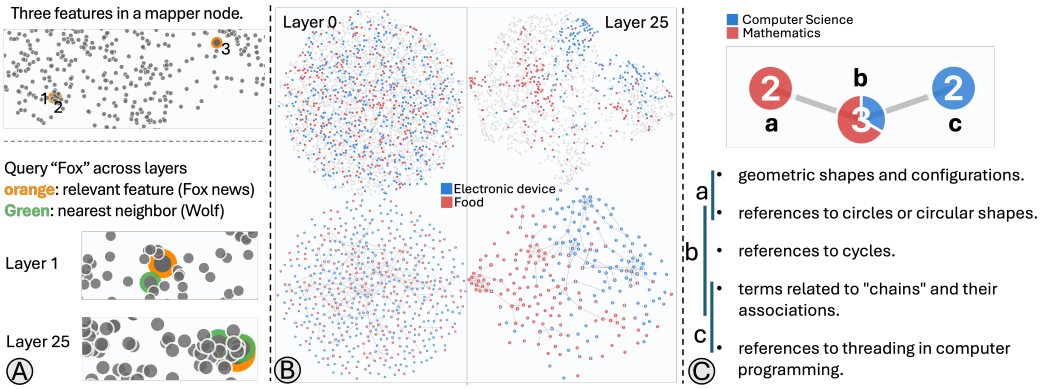

Figure 3: **A.** Top: a mapper node containing features 1–3, all related to the concept *music album*; bottom: querying *fox* across layers consistently yields *Fox News*, with *wolf* as its nearest neighbor. **B.** UMAP and ball mapper views for layers 0 and 25, highlighting features associated with *food* and *electronic devices*. **C.** A ball mapper path of subject concepts illustrating a transition from *Mathematics* to *Computer Science*.

Complementary local and global analysis. Our tool supports complementary local and global analyses of the feature space: ball mapper view exposes topological structure and reveals distortions introduced by dimensionality reduction that may be obscured in the UMAP view. This complementarity is illustrated in Figure 3(A, top), where features 1–3 within a ball mapper node all correspond to the concept *music album*. While features 1 and 2 are true nearest neighbors in the original space, their proximity is distorted in the UMAP view. As discussed earlier, our visualization also shows that local semantic proximity often persists across layers.

At the global scale, Figure 3(B) highlights category features associated with *electronic devices* and *food*. In early layers, these features appear mixed and spatially dispersed in both views, whereas deeper layers exhibit clearer intra-category grouping and stronger inter-category separation. Overlaps between categories also emerge; for instance, Figure 1(B) shows food-related features intersecting with *breakfast* and *dessert*, while Figure 1(D) reveals a ball mapper node containing *sugar*, *honey*, and *bee*, effectively marking a semantic boundary between *food* and *animal*.

Our framework further enables exploration of transitions between related concepts. In Figure 3(C), a path in the ball mapper graph (i.e., a sequence of overlapping local neighborhoods) traces a progression from *Computer Science* to *Mathematics*, illustrating how the model encodes relationships across fields of study. This observation aligns with prior work showing that mapper graphs can capture the evolution of linguistic phenomena along trajectories in embedding space [33].

Although we present only a few representative cases, these examples demonstrate that our framework provides a scalable and interpretable means of exploring how user-defined concepts relate to the structured knowledge encoded in SAEs across layers. We anticipate that such targeted exploration will facilitate deeper insight into model organization and support downstream applications such as feature-level intervention [43], model editing [44], and unlearning [45].

Limitations and discussion. Despite our exploration of curated concepts, a key limitation remains the inherent challenge of *auto-interpretability* in SAE features, that is, the reliability of identified features depends heavily on the quality of their automatically generated explanations. We plan to enhance the UMAP visualization by integrating scalable and annotated visualization toolkits [46, 47], and to leverage LLMs to automatically generate richer, context-aware explanations for mapper nodes [32].

Acknowledgments and Disclosure of Funding

This work was partially supported by the U.S. National Science Foundation (NSF) under grants DMS-2134223 and IIS-2205418. It was performed under the auspices of the U.S. Department of Energy (DOE) by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344, with additional support from the LLNL LDRD program (23-ERD-029, 25-SI-001) and the DOE Early Career Research Program (ECRP, SCW1885). This work has been reviewed and released under LLNL-CONF-2010620.

References

[1] Lee Sharkey, Dan Braun, and Beren Millidge. Taking features out of superposition with sparse autoencoders. In *AI Alignment Forum*, volume 6, pages 12–13, 2022.

[2] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *Transformer Circuits Thread*, 2022.

[3] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nicholas L Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023.

[4] 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.

[5] Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*, 2024.

[6] 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.

[7] Daniel Balcells, Benjamin Lerner, Michael Oesterle, Ediz Ucar, and Stefan Heimersheim. Evolution of sae features across layers in llms. *arXiv preprint arXiv:2410.08869*, 2024.

[8] Joshua Engels, Eric J Michaud, Isaac Liao, Wes Gurnee, and Max Tegmark. Not all language model features are one-dimensionally linear. *arXiv preprint arXiv:2405.14860*, 2024.

[9] Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Tom Lieberum, Vikrant Varma, Janos Kramar, Rohin Shah, and Neel Nanda. Improving sparse decomposition of language model activations with gated sparse autoencoders. *Advances in Neural Information Processing Systems*, 37:775–818, 2024.

[10] Joseph Bloom, Curt Tigges, Anthony Duong, and David Chanin. SAELens. https://github.com/decoderesearch/SAELens, 2024.

[11] Johnny Lin. Neuronpedia: Interactive reference and tooling for analyzing neural networks, 2023. Software available from neuronpedia.org.

[12] Sonia Joseph, Praneet Suresh, Lorenz Hufe, Edward Stevinson, Robert Graham, Yash Vadi, Danilo Bzdok, Sebastian Lapuschkin, Lee Sharkey, and Blake Aaron Richards. Prisma: An open source toolkit for mechanistic interpretability in vision and video. *arXiv preprint arXiv:2504.19475*, 2025.

[13] Samuel Marks, Can Rager, Eric J Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models. *arXiv preprint arXiv:2403.19647*, 2024.

[14] Tatsuro Inaba, Go Kamoda, Kentaro Inui, Masaru Isonuma, Yusuke Miyao, Yohei Oseki, Yu Takagi, and Benjamin Heinzerling. How a bilingual lm becomes bilingual: Tracing internal representations with sparse autoencoders. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 13458–13470, 2025.

[15] Angie Boggust, Hyemin Bang, Hendrik Strobelt, and Arvind Satyanarayan. Abstraction alignment: Comparing model-learned and human-encoded conceptual relationships. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–20, 2025.

[16] Tim Lawson, Lucy Farnik, Conor Houghton, and Laurence Aitchison. Residual stream analysis with multi-layer saes. *arXiv preprint arXiv:2409.04185*, 2024.

[17] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[18] Shusen Liu, Bei Wang, P-T Bremer, and Valerio Pascucci. Distortion-guided structure-driven interactive exploration of high-dimensional data. In *Computer Graphics Forum*, volume 33, pages 101–110. Wiley Online Library, 2014.

[19] Paweł Dłotko. Ball mapper: A shape summary for topological data analysis. *arXiv preprint arXiv:1901.07410*, 2019.

[20] Alireza Makhzani and Brendan Frey. K-sparse autoencoders. *arXiv preprint arXiv:1312.5663*, 2013.

[21] Zheng Zhang, Yong Xu, Jian Yang, Xuelong Li, and David Zhang. A survey of sparse representation: algorithms and applications. *IEEE Access*, 4:490–530, 2016.

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

[23] Luke Marks, Alasdair Paren, David Krueger, and Fazl Barez. Enhancing neural network interpretability with feature-aligned sparse autoencoders. *arXiv preprint arXiv:2411.01220*, 2024.

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

[25] Gonçalo Santos Paulo, Alex Troy Mallen, Caden Juang, and Nora Belrose. Automatically interpreting millions of features in large language models. In *42nd International Conference on Machine Learning*, 2025.

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

[27] Patrick Leask, Bart Bussmann, Michael Pearce, Joseph Bloom, Curt Tigges, Noura Al Moubayed, Lee Sharkey, and Neel Nanda. Sparse autoencoders do not find canonical units of analysis. *arXiv preprint arXiv:2502.04878*, 2025.

[28] Gonçalo Paulo and Nora Belrose. Sparse autoencoders trained on the same data learn different features. *arXiv preprint arXiv:2501.16615*, 2025.

[29] Zhengxuan Wu, Aryaman Arora, Atticus Geiger, Zheng Wang, Jing Huang, Dan Jurafsky, Christopher D Manning, and Christopher Potts. Axbench: Steering llms? even simple baselines outperform sparse autoencoders. *arXiv preprint arXiv:2501.17148*, 2025.

[30] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.

[31] Gurjeet Singh, Facundo Mémoli, and Gunnar E Carlsson. Topological methods for the analysis of high dimensional data sets and 3D object recognition. *Eurographics Symposium on Point-Based Graphics*, pages 91–100, 2007.

[32] Xinyuan Yan, Rita Sevastjanova, Sinie van der Ben, Mennatallah El-Assady, and Bei Wang. Explainable Mapper: Charting llm embedding spaces using perturbation-based explanation and verification agents. *arXiv preprint arXiv:2507.18607*, 2025.

[33] Archit Rathore, Yichu Zhou, Vivek Srikumar, and Bei Wang. TopoBERT: Exploring the topology of fine-tuned word representations. *Information Visualization*, 22(3):186–208, 2023.

[34] Archit Rathore, Nithin Chalapathi, Sourabh Palande, and Bei Wang. TopoAct: Exploring the shape of activations in deep learning. *Computer Graphics Forum*, 40(1):382–397, 2021.

[35] Youjia Zhou, Yi Zhou, Jie Ding, and Bei Wang. Visualizing and analyzing the topology of neuron activations in deep adversarial training. *Topology, Algebra, and Geometry in Machine Learning (TAGML) Workshop at ICML*, 2023.

[36] Gonçalo Paulo Caden Juang, Jacob Drori, and Nora Belrose. Open source automated interpretability for sparse autoencoder features. *EleutherAI Blog, July*, 30, 2024.

[37] Youjia Zhou, Nithin Chalapathi, Archit Rathore, Yaodong Zhao, and Bei Wang. Mapper Interactive: A scalable, extendable, and interactive toolbox for the visual exploration of high-dimensional data. *IEEE 14th Pacific Visualization Symposium (PacificVis)*, 2021.

[38] Laura M Stoinski, Jonas Perkuhn, and Martin N Hebart. Thingsplus: New norms and metadata for the things database of 1854 object concepts and 26,107 natural object images. *Behavior Research Methods*, 56(3):1583–1603, 2024.

[39] Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

[40] Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2. *arXiv preprint arXiv:2408.05147*, 2024.

[41] Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

[42] Nikita Balagansky, Ian Maksimov, and Daniil Gavrilov. Mechanistic permutability: Match features across layers. *arXiv preprint arXiv:2410.07656*, 2024.

[43] Sviatoslav Chalnev, Matthew Siu, and Arthur Conmy. Improving steering vectors by targeting sparse autoencoder features. *arXiv preprint arXiv:2411.02193*, 2024.

[44] Song Wang, Yaochen Zhu, Haochen Liu, Zaiyi Zheng, Chen Chen, and Jundong Li. Knowledge editing for large language models: A survey. *ACM Computing Surveys*, 57(3):1–37, 2024.

[45] Aashiq Muhamed, Jacopo Bonato, Mona T Diab, and Virginia Smith. Saes can improve unlearning: Dynamic sparse autoencoder guardrails for precision unlearning in llms. In *ICML 2025 Workshop on Reliable and Responsible Foundation Models*, 2025.

[46] Zijie J. Wang, Fred Hohman, and Duen Horng Chau. WizMap: Scalable interactive visualization for exploring large machine learning embeddings. In Danushka Bollegala, Ruihong Huang, and Alan Ritter, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 516–523, Toronto, Canada, July 2023. Association for Computational Linguistics.

[47] Donghao Ren, Fred Hohman, Halden Lin, and Dominik Moritz. Embedding atlas: Low-friction, interactive embedding visualization. *arXiv preprint arXiv:2505.06386*, 2025.