

Embedding Orrery: Visual Exploration of Vector Spaces and Sparse Autoencoder Features

Giacomo De Luca

Department of Enterprise Engineering
University of Rome Tor Vergata
Via del Politecnico 1, 00133, Rome, Italy
deluca@ing.uniroma2.it

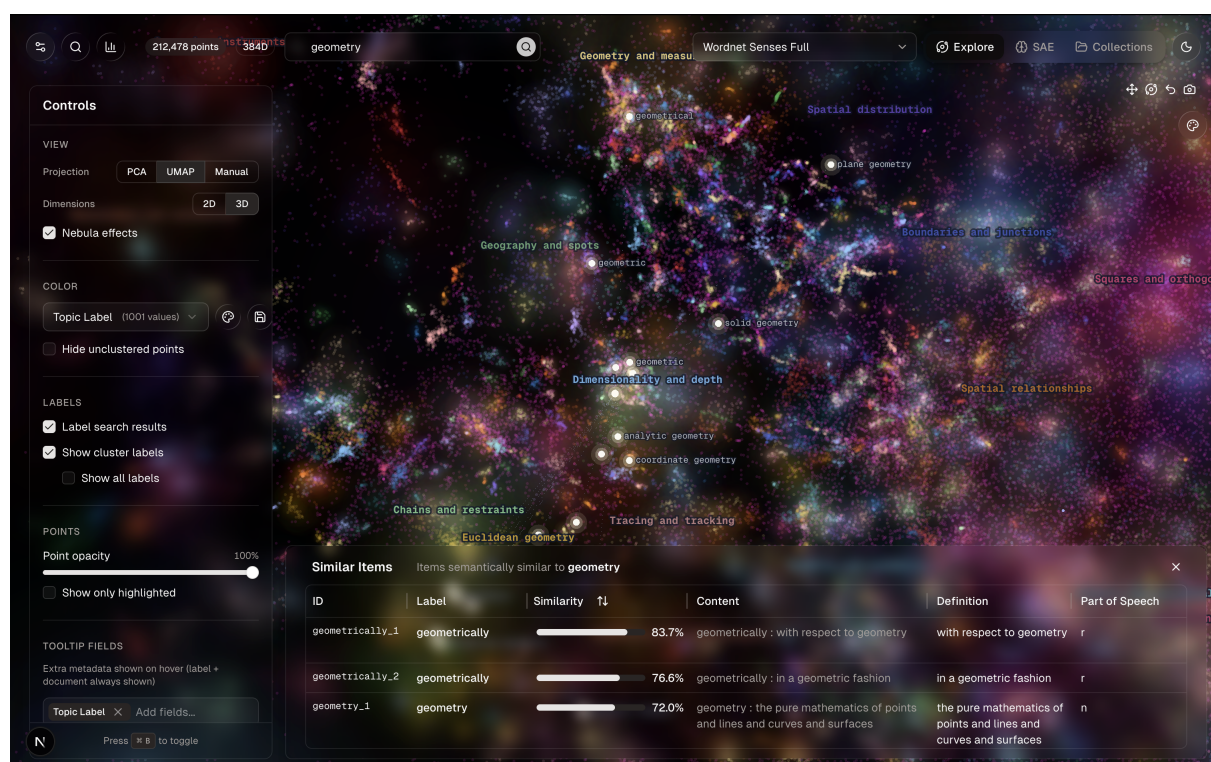


Figure 1: A screenshot of Orrery displaying the nebula mode on the WordNet dataset. Each point represents a gloss, and each color denotes an extracted topic. The table shows the top semantic-search results for “geometry”, also highlighted in the scatterplot. On the left, the visualization controls panel. Search and analytics panels can be accessed using the icons in the header, and the Collections and SAE pages through the navigation bar.

Abstract

Embedding Orrery is an open-source platform for interactive 3-dimensional visualization of embedding spaces, with native support for Sparse Autoencoders (SAEs), built for data exploration and interpretability research. Given any textual, image, or vector dataset, a single pipeline - embedding, projection, clustering, automatic topic labeling and, optionally, the collection of per-document SAE feature activations - turns it into a 3D constellation that can be searched visually, semantically, lexically, or by SAE feature names; probes can be trained, configured and scored directly in the interface.

SAEs are automatically downloaded and rendered as navigable constellations in which entering a prompt highlights the features it activates. Individual features can be inspected inside a dashboard and injected into the model for causal steering. The star-cartography-inspired interface sustains up to 500,000 points, including WebGL animations and volumetric nebula shaders, on 8 GB of RAM. Embedding Orrery is released under the Apache 2.0 license. Code: <https://github.com/Giacomo-De-Luca/embedding-orrery>. HF Spaces: <https://huggingface.co/spaces/GiacomoDeLuca/orrery-demo>.

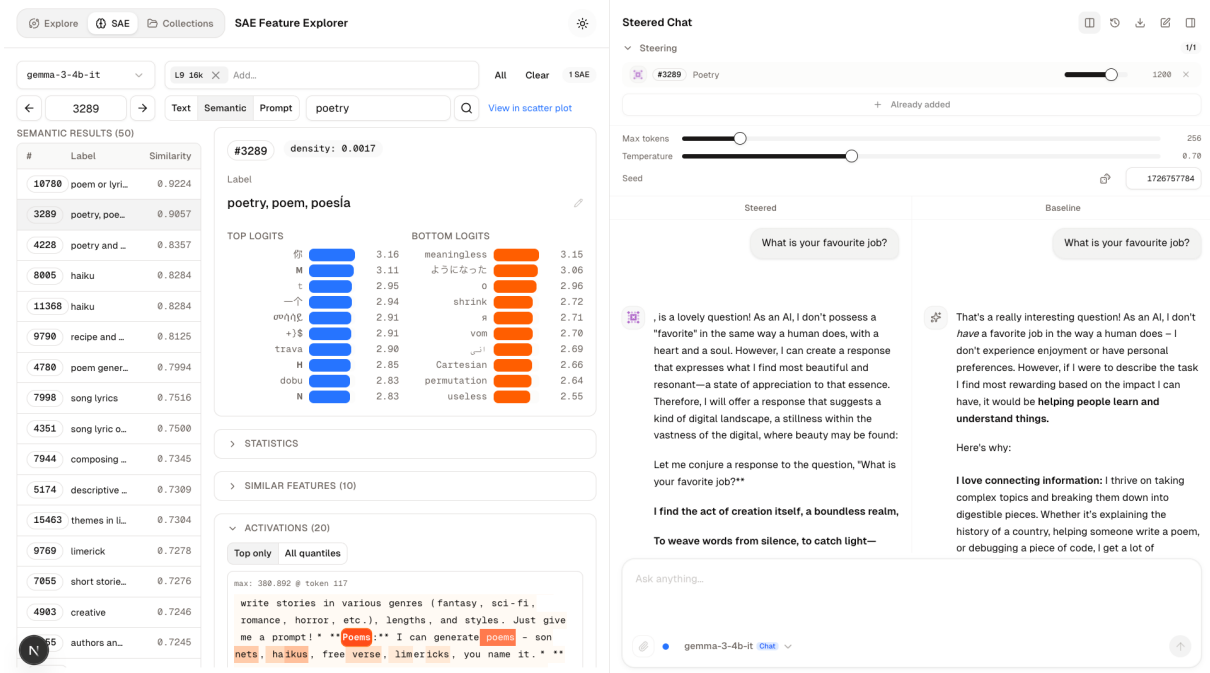


Figure 2: A screenshot of the SAE dashboard, with features related to “poetry” searched semantically. The selected feature’s panel shows label, density histogram, top/bottom logits, similar features, and top-activating documents. Right: steered vs. baseline response with single-vector steering on “What is your favorite job?”

1 Introduction

Ever since Mikolov et al. (2013) made meaningful textual embedding spaces widely available, multiple platforms have been developed to explore them visually. The Embedding Projector (Smilkov et al., 2016) let users explore embedding spaces as 3D scatterplots, reduced with PCA or t-SNE (Van der Maaten and Hinton, 2008). Its accompanying paper named three tasks for visualizers to solve: (1) *exploring local neighborhoods*, (2) *viewing global geometry and finding clusters*, and (3) *finding meaningful directions*. The Projector solved only the first task: it offered no way to cluster vectors, while direction-finding was limited to constructing an axis from the results of two textual queries.

After the popularization of contextualized embeddings (Devlin et al., 2019), new platforms (Wang et al., 2023; Nomic AI, 2022; Johnson, 2024; Ren et al., 2025) brought scale, clustering and automated topic labeling, solving the second task proposed in the Projector’s list, but reverted to 2D, trading navigation for scalability.

The third task, *finding directions in the original space*, remains crucial in interpretability research, yet no visualization platform learns directions from data by training probes. Linear structures emerge unsupervised in LLMs: the separation

between factual true and false can be visualized and recovered through linear probes (Marks and Tegmark, 2023); refusal is mediated by a single direction in the residual stream (Arditi et al., 2024).

Additionally, since the Projector’s release, new interpretability techniques built on vector spaces have emerged, most notably Sparse Autoencoders (SAE) (Bricken et al., 2023). SAEs expand the dimensionality of the model’s representation space to disentangle superposition (Elhage et al., 2022), recover interpretable features (Templeton et al., 2024), and can be used in downstream tasks for steering, classification and retrieval (Jiang et al., 2025; Deng et al., 2026).

Since each row of the SAE decoder matrix is a vector in the original space, SAE features can also be visualized: Anthropic published a dashboard and 2D scatterplots to navigate SAE features in a UMAP-projected space (Bricken et al., 2023; Templeton et al., 2024).

Embedding Orrery (abbreviated Orrery) aims to integrate embedding visualization with interpretability, solving all three of the Projector’s tasks. To embedding-space exploration it brings interpretability tooling: corpora become searchable by SAE features, and hypotheses about linear structure can be tested by training probes directly in the interface. To interpretability it brings the capabilities

of modern visualization platforms: SAE feature spaces are downloaded, projected, clustered, and labeled; they are also connected in the same interface to feature inspection and causal steering.

2 Related Work

Embedding-visualization platforms. The platforms that followed the Embedding Projector (§1) converged on two design choices: 2D projections, and scale as the primary goal. Nomic Atlas (Nomic AI, 2022) renders millions of embeddings with clustering, topic labels, filtering and search, but is a closed commercial service with a fixed embedding model and projection models. WizMap (Wang et al., 2023) and Embedding Atlas (Ren et al., 2025) reach similar scale in the browser, adding density contours, cross-filtering, and analytical charts. Latent Scope (Johnson, 2024), an open-source local pipeline, is the only platform with any SAE integration: documents can be filtered by features from an SAE trained on nomic-embed-text-v1.5 embeddings¹, but feature spaces are never visualized.

Three-dimensional embedding views survived mainly as demonstrations: Google’s *t-SNE Map* of artworks² and the Hugging Face community’s *Semantic Galaxy*,³ an in-browser 3D view of EmbeddingGemma (Vera et al., 2025) vectors and the main inspiration for Orrery’s cosmic theme. The third dimension is not cosmetic: rotation-based navigation, as opposed to pure zooming, increases immersion and supports cluster understanding (Yang et al., 2021).

SAE feature visualization. Activation atlases (Carter et al., 2019) mapped the activation space of vision models on a projected grid. For SAEs, Bricken et al. (2023) established the visual vocabulary still in use: a per-feature dashboard (activation density, top and bottom logits, activating examples) accompanied by a 2D UMAP map of 4,608 features, later extended to Claude Sonnet (Templeton et al., 2024). Both interfaces cover a single layer and are closed-source. Open-source tooling, such as SAE-Vis (McDougall, 2024), SAEDashboard (Decode Research, 2024), and Neuronpedia (Lin, 2023) replicated the dashboard leaving the navigable space behind. Neuronpedia offers search

¹<https://enjalot.github.io/latent-taxonomy>

²<https://artsexperiments.withgoogle.com/tsnemap>

³<https://huggingface.co/spaces/webml-community/semantic-galaxy>

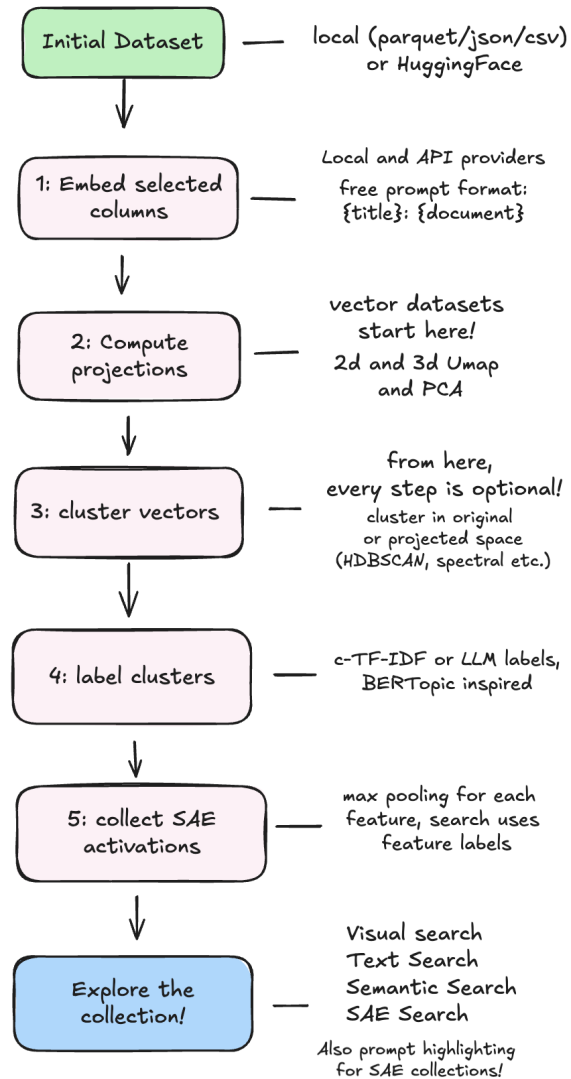


Figure 3: Collection pipeline, moving from a dataset to an explorable constellation. The entire pipeline can be set-up and executed visually in the collections page.

over feature explanations, activation testing, and in-browser steering, but its only spatial view is a 2D UMAP of GPT-2 Small, without clustering or navigation aids. Google’s PAIR published a pedagogical explorable of a 2D visualization of a single Gemma Scope (Lieberum et al., 2024) SAE⁴. Existing feature maps are one-off showcases of a fixed SAE rather than research platforms.

3 System Design

Orrery’s design goal is seamless exploration of embedding spaces: frictionless (Ren et al., 2025) movement from a raw dataset to a navigable constellation, and from a promising SAE feature to its causal effect on the model (Templeton et al., 2024).

⁴<https://pair.withgoogle.com/explorables/sae>

The platform allows two workflows over one shared backend. The *collection workflow* (embed, project, cluster, label, explore) turns any textual, image, or vector dataset into a searchable space. The *SAE workflow* (visualize, inspect, steer) reuses the same engine for interpretability.

Datasets, collections, storage. Orrery separates two concepts. A *dataset* is a set of documents (text, images, or pre-computed vectors) stored and indexed in DuckDB. A *collection* is one representation of that dataset: an embedding, its 2D/3D projections, optional topic extractions and SAE activations. One dataset can therefore have many collections. DuckDB acts as the central orchestrator (documents, metadata, projections, SAE activations), while ChromaDB (Chroma, 2023) stores only IDs and dense vectors for approximate nearest-neighbor search. A GraphQL API serves the Next.js frontend; long-running jobs stream progress over WebSockets and are resumable after interruption.

Ingestion. A dataset is previewed from HuggingFace or from local JSON, CSV, or Parquet files. The columns to embed are specified with a free-form template that supports both custom prompts and column concatenation (e.g. {word}:{definition} for WordNet). Eight embedding providers are supported (Appendix A). All features can run fully offline selecting local models. *Projections*, the only mandatory step to visualize a collection, are computed with PCA and UMAP (McInnes et al., 2018), in both 2D and 3D.

Topic modeling. Topic extraction follows and forks BERTopic (Grootendorst, 2022): since embedding and reduction already exist in the platform, we reuse only its clustering, c-TF-IDF, and labeling components, including its LLM labeling prompt. Clustering (HDBSCAN, k-means, Gaussian mixtures, or spectral) runs on the stored 2D/3D projections, on 5-D UMAP as in BERTopic, or on the original vectors; with BERTopic’s defaults, topic assignments are identical (Table 1 in the Appendix). Labels come from c-TF-IDF keywords or LLMs.

Per-document SAE activations. Any stored collection can optionally be run through Gemma Scope 2 SAEs (McDougall et al., 2025). Following Jiang et al. (2025), token-level activations are max-pooled into a single sparse vector per document. Because each dimension carries an auto-interpreted

feature label, the corpus becomes searchable by feature names.

Search and filtering. A collection can be searched four ways, with results rendered as a highlighted constellation centered on the first result and colored by similarity (Figure 1): (i) **semantic** search over the original embedding space, also triggered by clicking any point; (ii) **text** search (exact, partial, or BM25, with column selection); (iii) **SAE feature search**: querying an SAE feature name (e.g. *football*) ranks documents by their max activation over matching features; (iv) **prompt highlighting**, on SAE collections (§3.1): the prompt is run through the model, encoded by the SAE at the selected layer, and every activated feature lights up, ranked and colored by activation strength. This inverts the SAE search: there, features index documents; here, a document indexes features.

Probing. Probes can be trained and configured directly in the interface on the original embeddings: any metadata field serves as target, fitted with a linear probe (mass mean, logreg, ridge), SVR or a small MLP. Linear probes double as directions: the learned weight vector is stored with the collection, and points can be colored by their projection onto it (§4.1).

Rendering 3D galaxies at scale. The scatterplots are rendered through WebGL using a fork of Plotly.js: on every trace update, Plotly re-reads all n points across all m attributes of every existing trace, an $O(n \cdot m)$ burden that makes real-time overlays and filters impossible at scale. Plotly is used only for camera control and point rendering: tooltips, legend, labels, and animations are re-implemented on an overlay canvas that reads the WebGL context directly. Beyond d3’s scales, Orrery employs Crameri’s perceptually uniform color maps (Crameri, 2018; Crameri et al., 2020), and numeric legends render as interactive histograms of the field’s distribution, with draggable handles to re-anchor scales skewed by outliers (Figure 5). Label placement ports the Embedding Projector’s collision-avoidance code (Smilkov et al., 2016): 3D anchors are projected to screen space each frame and packed on a spatial grid, extended with a priority order and rank-dependent opacity. Since no scatterplot library offers volumetric rendering, the *nebula* mode approximates it: we render feathered sprites around cluster centroids, spread according

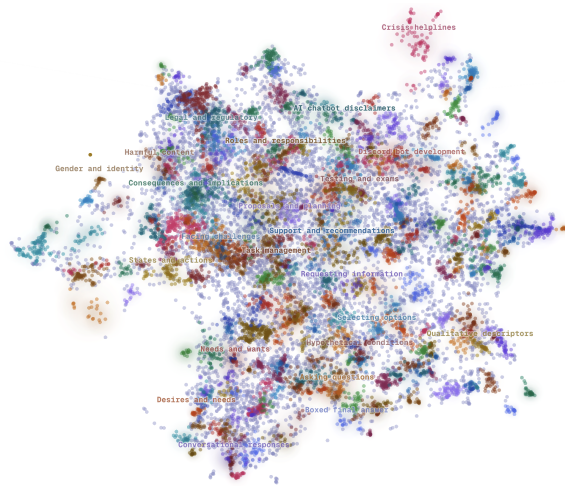


Figure 4: The Gemma Scope 2 16k SAE (Gemma-3-4B-it, layer 29) as a 3D galaxy of its Neuronpedia-label embeddings. Topic labels appear on zoom and rotation; right-clicking a feature opens its dashboard.

to cluster extent, in the style of GalaxyThreeJS⁵. For 2D density contouring we port the code from Embedding Atlas (Ren et al., 2025), adding coloring according to the categories (Figure 6). The interface was tested rendering up to 500k points with 8 GB RAM (Table 2 in the Appendix).

3.1 SAE collections

Feature spaces as collections. Each SAE feature is a row of the decoder matrix W_{dec} , a vector in the model’s original space, so feature sets can be visualized like any vector dataset (Figure 4), by projecting the decoder vectors directly or by embedding their auto-interpreter labels (Bills et al., 2023; Paulo et al., 2025) with any provider from §3. From the SAE tab in the *Collections* page, the user selects the target model, layer, and width; the SAE is downloaded and registered as a dataset together with its feature labels and activation examples from Neuronpedia (Lin, 2023). Since these are ordinary vector collections with metadata, the entire collection pipeline (Figure 3) is reused with no additional code.

Feature dashboard and steering. Any feature in the SAE galaxy can be right clicked to examine it in the SAE page: a dashboard inspired by Bricken et al. (2023) and Neuronpedia (Lin, 2023) (Figure 2). For each feature, the dashboard displays top and bottom logits, activation-density histograms, features with similar activation patterns, top or

decile based activation samples. Features are located through textual or semantic search, or by prompting the model and examining token-level activations in a heatmap. The found features can be directly injected into the model with ActAdd (Turner et al., 2024), selecting the activation coefficient. Steering with multiple features is supported, as is split view to compare with baseline response.

Interpretability backend. SAELens (Bloom et al., 2024) supports neither multimodal inputs nor streaming generation with interventions active. The requirements for SAEs are small, - read/write caches at three sites per layer (residual, attention output, MLP output), and a single encode/decode pair for each SAE, - so we implement it directly. Gemma-pytorch was forked to add activation caches and streaming generation. JumpReLU (Rajamanoharan et al., 2024) for Gemma Scope and TopK (Gao et al., 2024) SAEs for Qwen-Scope (Deng et al., 2026) are written in plain PyTorch. Qwen-Scope SAEs are only for the residual stream, so they are run via *register_forward_hook* on the Transformers model.

4 Case Studies

We validate Orrery with two replication studies, each exercising a different subsystem end-to-end: finding linear directions visually in the embedding space and validating them through probe training in the collection workflow (§4.1); activation caching and steering in the interpretability backend (§4.2).

4.1 Psycholinguistic Norms are Linear Directions

LLMs encode semantic dimensions linearly: a simple mass mean probe recovers a factual true/false direction across models (Marks and Tegmark, 2023). The probe is the vector between the two class means, $\theta = \Sigma^{-1}(\mu^+ - \mu^-)$, with Σ being the within class covariance. Whether this extends to sentence-embedding spaces remains open.

We probe two sets of human psycholinguistic norms in the embedding space of EmbeddingGemma and MiniLM from SentenceTransformers (Reimers and Gurevych, 2019).

1. **Concreteness** ratings for about 40k English words and two-word expressions obtained from 4,000 participants (Brysbaert et al., 2014).

⁵<https://github.com/pickles976/GalaxyThreeJS>

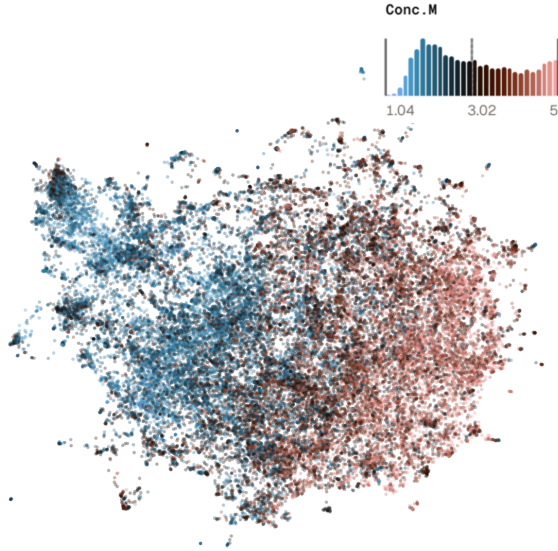


Figure 5: 3D UMAP of MiniLM embeddings of the Concreteness dataset. The concrete/abstract direction appears visually separated, with a mixed region at the center.

2. **Glasgow norms** $\sim 5.5k$ words rated on nine dimensions (arousal, valence, dominance concreteness, imageability, familiarity, age of acquisition, semantic size, and gender association).

The collections are first embedded, projected and colored using diverging color scales to find which dimensions appear to be linearly separated (Figure 5). We train mass mean, Ridge Regression and SVR probes on the datasets, using 5-fold cross-validation with an 80/20 split in each fold. Every dimension visually separable is linearly recoverable: concreteness 40k is linearly encoded in both models (Ridge $R^2=0.69/0.83$; mass mean $R^2=0.61/0.71$ for MiniLM/EmbeddingGemma). Glasgow norms are all linearly encoded (ridge ranges from $R^2=0.59$ to $R^2=0.85$ on EmbeddingGemma). Nonlinear probes improve slightly the results (SVR–Ridge ≤ 0.1). Per-dimension results and the full protocol are in Appendix B.

4.2 Refusal is a Single Vector Away

To validate the steering backend (§3.1), we replicate Arditi et al. (2024) on Gemma-3-4b-it (the reference uses Gemma-2). Candidate refusal directions r_ℓ are extracted by difference-in-means between activations over $n=128$ harmful prompts (AdvBench (Zou et al., 2023), MaliciousInstruct (Huang et al., 2024)) and harmless Alpaca instruc-

tions (Taori et al., 2023), at each layer and each of the five end-of-instruction template tokens.

Arditi et al. (2024) remove refusal by *directional ablation*, projecting the unit-norm $\hat{r}$ out of the residual stream at every layer, $h \leftarrow h - (\hat{r}^\top h) \hat{r}$; in our evaluation, this collapses the model to degenerate token loops, possibly caused by the residual stream growth over layers (Heimersheim and Turner, 2023).

We instead use ActAdd, the same additive mechanism as our SAE steering (§3.1), adding the unnormalized r_{ℓ^*} at a single layer: $h_{\ell^*} \leftarrow h_{\ell^*} + \alpha r_{\ell^*}$, with $\alpha = -1$ suppressing refusal and $\alpha = +1$ inducing it. At $\ell^* = 13$, ActAdd(+1) induces refusal on 100/100 entirely benign prompts using substring matching from (Arditi et al., 2024) for comparable results. Using Llama-Guard-4-12B (Inan et al., 2023) as a harmfulness classifier attack success rises from 0.11 to 0.83 on JailbreakBench (Chao et al., 2024) and from 0.21 to 0.86 on HarmBench-test (Mazeika et al., 2024). All completions were manually reviewed for coherence. A single vector at a single layer still suffices to remove refusal or to induce it, suggesting that mid-generation steering is a promising avenue (Deng et al., 2026). Vector selection details and per-category results are in Appendix C.

5 Conclusions

Orrery connects embedding visualization and interpretability: corpus exploration gains SAE-feature search and in-interface probing; interpretability gains the interactivity of modern visualization platforms, with feature spaces rendered as navigable collections wired to inspection and causal steering. The case studies display how the platform enables research end-to-end: linear structure spotted visually and confirmed by probes (§4.1) and a refusal direction for safety-testing extracted, injected, and evaluated without leaving the platform (§4.2).

Beyond interpretability research, the coloring, temporal, and filtering tools target the navigation of digital humanities corpora: by breaking the linear order of texts and drawing similar documents together in space, connections across authors, periods, and passages surface visually. The nebula shaders, highlight traces, and 3D labels are not ornaments but navigation aids, inviting users to treat a corpus not as a flat canvas but as a place to explore with curiosity and a sense of wonder.

Limitations

Orrery currently fully supports only SAEs from Gemma Scope 2 and Qwen-Scope. Support for additional SAE families requires only an encode/decode implementation and a *register_forward_hook attachment*. SAE search is also enabled only for the Gemma Scope 2 family, since the current application downloads labels from Neuronpedia. The backend hosts a working autointerpreter pipeline to generate native labels, which is still pending validation. SAE searching is also limited by the noisiness of the labels (Jiang et al., 2025). While Orrery has been provided to interpretability researchers to debug and experiment, a systematic user evaluation study has not yet been conducted. Finally, Orrery has a scale limitation compared to the millions of points that Nomic Atlas or Embedding Atlas render. This was a deliberate design trade-off in favor of 3D navigation, real-time overlay and volumetric shaders.

Ethics Statement

Our refusal case study (§4.2) replicates a published method (Arditi et al., 2024), which can be run using the `refusal_direction` code available on GitHub⁶. Our experiment reproduces it using Gemma-3 instead of Gemma-2, and employs the fork of Gemma-pytorch instead of using Transformers, so the release provides no capability beyond what is already public. Our contribution is reproducibility and tooling for inspecting the safety-relevant direction. The released SAE tooling supports an emerging line of safety work that monitors and intervenes on model internals, using input classification and activation-level steering via SAEs or steering vectors (Lu et al., 2026), rather than easily broken post-hoc alignment. The refusal vectors themselves are not distributed: the released code contains the extraction and evaluation pipeline, a download script to retrieve the prompts (again adapted from (Arditi et al., 2024)), and no harmful completions. The Hugging Face space contains no steering interface. The psycholinguistic norms (§4.1) are public datasets of aggregate human ratings, used under their licenses.

References

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda.

⁶https://github.com/andyrdt/refusal_direction

2024. Refusal in language models is mediated by a single direction. *Preprint*, arXiv:2406.11717.

Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. 2023. Language models can explain neurons in language models. <https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html>.

Joseph Bloom, Curt Tigges, Anthony Duong, and David Chanin. 2024. Saelens. <https://github.com/decoderresearch/SAELens>.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermy, 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, and 6 others. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*. <https://transformer-circuits.pub/2023/monosemantic-features/index.html>.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known english word lemmas. *Behav. Res. Methods*, 46(3):904–911.

Shan Carter, Zan Armstrong, Ludwig Schubert, Ian Johnson, and Chris Olah. 2019. *Activation atlas*. *Distill*. <https://distill.pub/2019/activation-atlas>.

Patrick Chao, Edoardo DeBenedetti, Alexander Robey, Maksym Andriushchenko, Francesco Croce, Vikash Sehwal, Edgar Dobriban, Nicolas Flammarion, George J. Pappas, Florian Tramèr, Hamed Hassani, and Eric Wong. 2024. Jailbreakbench: An open robustness benchmark for jailbreaking large language models. In *NeurIPS Datasets and Benchmarks Track*.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. *M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation*. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2318–2335, Bangkok, Thailand. Association for Computational Linguistics.

Chroma. 2023. Chroma: The AI-native open-source embedding database. <https://github.com/chroma-core/chroma>. Accessed: 2026-07-09.

Fabio Crameri. 2018. *Scientific colour maps*.

Fabio Crameri, Grace E. Shephard, and Philip J. Heron. 2020. *The misuse of colour in science communication*. *Nature Communications*, 11:5444.

Decode Research. 2024. SAE Dashboard. <https://github.com/jbloomAus/sae-dashboard>.

- Boyi Deng, Xu Wang, Yaoning Wang, Yu Wan, Yubo Ma, Baosong Yang, Haoran Wei, Jialong Tang, Huan Lin, Ruizhe Gao, Tianhao Li, Qian Cao, Xuancheng Ren, Xiaodong Deng, An Yang, Fei Huang, Dayiheng Liu, and Jingren Zhou. 2026. [Qwen-scope: Turning sparse features into development tools for large language models](#). *Preprint*, arXiv:2605.11887.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [Bert: Pre-training of deep bidirectional transformers for language understanding](#). *Preprint*, arXiv:1810.04805.
- Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. 2022. Toy models of superposition. *Transformer Circuits Thread*. https://transformer-circuits.pub/2022/toy_model/index.html.
- Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2024. [Scaling and evaluating sparse autoencoders](#). *Preprint*, arXiv:2406.04093.
- Maarten Grootendorst. 2022. [Bertopic: Neural topic modeling with a class-based tf-idf procedure](#). *Preprint*, arXiv:2203.05794.
- Stefan Heimersheim and Alex Turner. 2023. [Residual stream norms grow exponentially over the forward pass](#). AI Alignment Forum. Accessed: 2026-06-09.
- Yangsibo Huang, Samyak Gupta, Mengzhou Xia, Kai Li, and Danqi Chen. 2024. [Catastrophic jailbreak of open-source llms via exploiting generation](#). In *International Conference on Learning Representations*, volume 2024, pages 13707–13727.
- Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabsa. 2023. [Llama guard: Llm-based input-output safeguard for human-ai conversations](#). *Preprint*, arXiv:2312.06674.
- Nick Jiang, Xiaoqing Sun, Lisa Dunlap, Lewis Smith, and Neel Nanda. 2025. [Interpretable embeddings with sparse autoencoders: A data analysis toolkit](#). *Preprint*, arXiv:2512.10092.
- Ian Johnson. 2024. Latent scope: A scientific instrument for investigating latent spaces. <https://github.com/enjalot/latent-scope>. Accessed: 2026-07-09.
- Tom Lieberum, Senthooan Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. 2024. [Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2](#). *Preprint*, arXiv:2408.05147.
- Johnny Lin. 2023. [Neuronpedia: Interactive reference and tooling for analyzing neural networks](#). Software available from neuronpedia.org.
- Christina Lu, Jack Gallagher, Jonathan Michala, Kyle Fish, and Jack Lindsey. 2026. [The assistant axis: Situating and stabilizing the default persona of language models](#). *Preprint*, arXiv:2601.10387.
- Samuel Marks and Max Tegmark. 2023. [The geometry of truth: Emergent linear structure in large language model representations of true/false datasets](#). *ArXiv*, abs/2310.06824.
- Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, David Forsyth, and Dan Hendrycks. 2024. [Harmbench: A standardized evaluation framework for automated red teaming and robust refusal](#). *Preprint*, arXiv:2402.04249.
- Callum McDougall. 2024. SAE Visualizer. https://github.com/callummcdougall/sae_vis.
- Callum McDougall, Arthur Conmy, János Kramár, Tom Lieberum, Senthooan Rajamanoharan, and Neel Nanda. 2025. [Gemma scope 2 – technical paper](#). Technical report, Google DeepMind. Technical report released alongside the Gemma Scope 2 open suite of sparse autoencoders and transcoders for the Gemma 3 model family.
- Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. 2018. [Umap: Uniform manifold approximation and projection](#). *Journal of Open Source Software*, 3(29):861.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.
- Nomic AI. 2022. Nomic atlas. <https://atlas.nomic.ai/>. Accessed: 2026-07-09.
- Gonçalo Paulo, Alex Mallen, Caden Juang, and Nora Belrose. 2025. [Automatically interpreting millions of features in large language models](#). *Preprint*, arXiv:2410.13928.
- Senthooan Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. 2024. [Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders](#). *Preprint*, arXiv:2407.14435.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-BERT: Sentence embeddings using Siamese BERT-networks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

- Donghao Ren, Fred Hohman, Halden Lin, and Dominik Moritz. 2025. [Embedding atlas: Low-friction, interactive embedding visualization](#). *Preprint*, arXiv:2505.06386.
- Daniel Smilkov, Nikhil Thorat, Charles Nicholson, Emily Reif, Fernanda B. Viégas, and Martin Wattenberg. 2016. [Embedding projector: Interactive visualization and interpretation of embeddings](#). *Preprint*, arXiv:1611.05469.
- Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.
- Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, and 3 others. 2024. [Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet](#). *Transformer Circuits Thread*.
- Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2024. [Steering language models with activation engineering](#). *Preprint*, arXiv:2308.10248.
- Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).
- Henrique Schechter Vera, Sahil Dua, Biao Zhang, Daniel Salz, Ryan Mullins, Sindhu Raghuram Panayam, Sara Smoot, Iftekhar Naim, Joe Zou, Feiyang Chen, Daniel Cer, Alice Lisak, Min Choi, Lucas Gonzalez, Omar Sanseviero, Glenn Cameron, Ian Ballantyne, Kat Black, Kaifeng Chen, and 70 others. 2025. [Embeddinggemma: Powerful and lightweight text representations](#). *Preprint*, arXiv:2509.20354.
- Zijie J. Wang, Fred Hohman, and Duen Horng Chau. 2023. [Wizmap: Scalable interactive visualization for exploring large machine learning embeddings](#). *Preprint*, arXiv:2306.09328.
- Yalong Yang, Maxime Cordeil, Johanna Beyer, Tim Dwyer, Kim Marriott, and Hanspeter Pfister. 2021. [Embodied navigation in immersive abstract data visualization: Is overview+detail or zooming better for 3d scatterplots?](#) *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1214–1224.
- Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. [Qwen3 embedding: Advancing text embedding and reranking through foundation models](#). *Preprint*, arXiv:2506.05176.
- Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. 2023. [Universal and transferable adversarial attacks on aligned language models](#). *Preprint*, arXiv:2307.15043.

Dataset	Pipeline	Topics	Sil.	C_v
emotion	Orrery	22	0.5259	0.3279
(1k)	BERTopic	22	0.5259	0.3431
20 Newsgroups	Orrery	214	0.6166	0.6113
(18.8k)	BERTopic	214	0.6166	0.6128

Table 1: Parity with BERTopic on identical inputs (same MiniLM embeddings and UMAP/HDBSCAN configuration). Document–topic assignments are identical (ARI = 1.0), making silhouette match exactly; the small C_v gap stems from minor differences in c-TF-IDF keyword extraction, not clustering.

Points	3D FPS	2D FPS	Heap (MB)	Load (s)
1,000	88.8	101.7	114	1.7
10,000	91.3	102.9	169	1.7
19,537	84.7	98.1	189	1.7
39,954	78.6	93.7	360	2.8
50,000	74.5	90.7	470	3.0
65,536	71.6	94.3	542	4.2
153,772	51.6	80.9	1,050	7.7
212,478	43.3	74.8	1,441	7.9
250,000	34.5	76.6	1,664	6.4
500,000	24.9	57.4	3,243	6.4

Table 2: Rendering performance vs. collection size: mean FPS during continuous camera interaction (orbit in 3D, pan in 2D), measured on the live application (Chrome/Metal, MacBook Pro M3, 8 GB RAM, 120 Hz display). Heap is the renderer’s JS heap after load (3D).

A Embedding Settings

The platform supports 8 embedding providers for sentence embeddings. Local: SentenceTransformers (default), Ollama, FlagEmbedding for BGE (Chen et al., 2024), and PyTorch for Qwen (Zhang et al., 2025); via API key: OpenAI, Cohere, Gemini, and HuggingFace. The settings include task-type prompts for models that use them, such as EmbeddingGemma (Vera et al., 2025).

B Probe Settings and Per-Dimension Results

Probe settings. We evaluate all probes with five-fold cross-validation, using an 80/20 train–test split (seed 7) on per-dimension z -scored embeddings. RIDGE: ℓ_2 linear regression, $\alpha = 1$. SVR: support-vector regression, RBF kernel, $C = 1$. MASS MEAN: the covariance corrected direction $\mu_+ - \mu_-$ of Marks and Tegmark (2023) on a median split, scored by projection. All probes report R^2 .

Norm	Ridge R^2	SVR R^2	Mass mean R^2
<i>MiniLM</i>			
concreteness	0.72 ± 0.02	0.77 ± 0.02	0.64 ± 0.04
imageability	0.66 ± 0.02	0.71 ± 0.02	0.56 ± 0.03
valence	0.62 ± 0.02	0.70 ± 0.02	0.50 ± 0.03
aoa	0.56 ± 0.02	0.63 ± 0.02	0.44 ± 0.01
semsize	0.58 ± 0.01	0.67 ± 0.01	0.47 ± 0.02
gender	0.51 ± 0.04	0.61 ± 0.03	0.39 ± 0.05
familiarity	0.46 ± 0.03	0.52 ± 0.02	0.36 ± 0.02
arousal	0.47 ± 0.02	0.57 ± 0.01	0.35 ± 0.03
dominance	0.46 ± 0.03	0.55 ± 0.02	0.35 ± 0.03
concreteness-40k	0.69 ± 0.01	0.74 ± 0.00	0.61 ± 0.01
<i>EmbeddingGemma</i>			
concreteness	0.85 ± 0.01	0.87 ± 0.01	0.74 ± 0.02
imageability	0.79 ± 0.01	0.82 ± 0.01	0.69 ± 0.02
valence	0.83 ± 0.01	0.86 ± 0.01	0.73 ± 0.01
aoa	0.70 ± 0.01	0.72 ± 0.02	0.35 ± 0.01
semsize	0.75 ± 0.01	0.80 ± 0.01	0.54 ± 0.03
gender	0.72 ± 0.03	0.77 ± 0.02	0.42 ± 0.04
familiarity	0.59 ± 0.03	0.60 ± 0.02	0.35 ± 0.04
arousal	0.66 ± 0.03	0.71 ± 0.02	0.50 ± 0.01
dominance	0.64 ± 0.02	0.70 ± 0.02	0.46 ± 0.05
concreteness-40k	0.83 ± 0.00	0.84 ± 0.00	0.71 ± 0.00

Table 3: Per-dimension recovery of the psycholinguistic norms at the output embedding; 5-fold cross-validation, mean $\pm$ std. Mass mean is the covariance-corrected difference-of-means direction ($\mu_+ - \mu_-$), with R^2 calibrated via a train-split. *Concreteness-40k* is the full Brysbaert set.

C Refusal-direction evaluation

To extract the refusal vectors, we use the difference-in-means equation from (Arditi et al., 2024), with $h_\ell(x)$ the residual-stream activation at layer ℓ at the selected token position for input x , and H^+ , H^- the harmful and harmless instruction sets on the residual stream:

$$r_\ell = \frac{1}{|H^+|} \sum_{x \in H^+} h_\ell(x) - \frac{1}{|H^-|} \sum_{x \in H^-} h_\ell(x). \quad (1)$$

This returns a candidate direction per layer and token position. To select among vector candidates we rank candidates on the generations of 32 held-out prompts. To evaluate the generations, following (Arditi et al., 2024), we score refusal by substring matching. Since substring matching cannot detect broken or looped output, we additionally compute the KL divergence between the steered and unsteered next-token distributions on a set of neutral prompts. Since substring matching can only measure overt refusal, not effective harmful content ASR is measured by Llama-Guard-4-12B on the selected vector, residual, token position -3 , layer 13 ($\ell^*=13$).

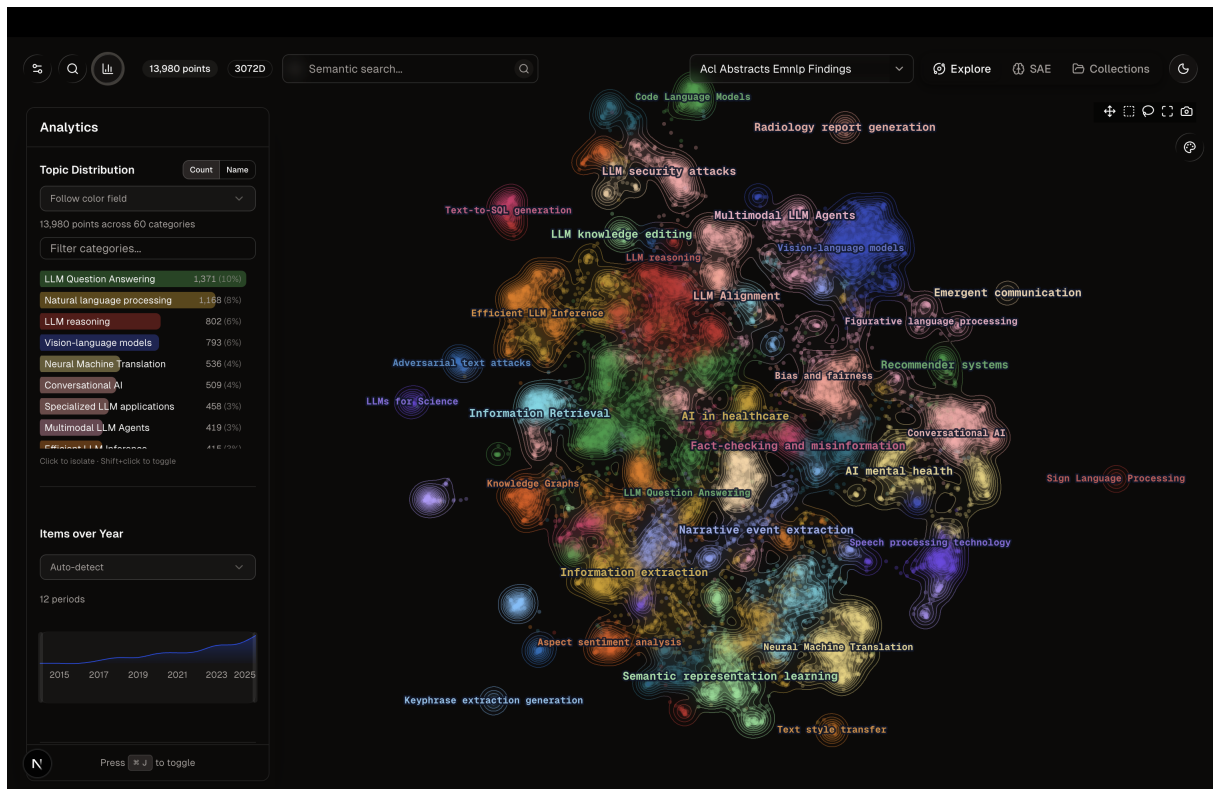


Figure 6: UMAP projection of Gemini Embedding 2 embeddings of the EMNLP abstracts dataset 2014–2025. The image displays the density contouring, colored by the topic extraction pipeline. The analytics panel is open, allowing users to filter by category or drag the temporal-histogram handles.

Set (n)		base	ActAdd ₋₁	ActAdd ₊₁
JBB (100)	ref.	0.84	0.00	—
	ASR	0.11	0.83	—
HB-test (159)	ref.	0.75	0.00	—
	ASR	0.21	0.86	—
Harmless (100)	ref.	0.04	—	1.00

Table 4: Refusal rate (ref.) and harm attack-success rate (ASR) under activation addition. ActAdd₋₁ suppresses refusal, ActAdd₊₁ induces it; a single direction controls both. JBB: JailbreakBench; HB-test: HarmBench-test. Refusal (ref.) is scored by substring matching following (Arditi et al., 2024); ASR is scored by Llama-Guard-4-12B.

HarmBench category	n	base	ActAdd ₋₁
Chem./bio.	19	0.37	1.00
Cybercrime	33	0.45	1.00
Harassment	16	0.00	0.88
Illegal	47	0.15	0.87
Misinfo	27	0.15	0.74
Harmful (gen.)	17	0.00	0.59

Table 5: Per-category harm ASR (HarmBench-test) at $\ell^* = 13$. The weakest cells are disinformation and generic “harmful”, where the model complies but Llama-Guard under-scores the output.