SAE-STM: A SPARSE AUTOENCODER-AUGMENTED STRUCTURAL TOPIC MODEL FOR INTERPRETABLE PAIRWISE PREFERENCE PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reward models trained on pairwise human judgments drive RLHF and DPO, but they reveal little about *what* in a response makes it preferred. Neural topic models offer a compact, readable vocabulary but are too coarse to capture fine-grained preference factors; sparse autoencoders (SAEs) recover fine-grained features but are fit post hoc, with no link to the prediction task or to any structured summary. We introduce SAE-STM, which couples the two: a Top-k SAE over the embedding difference of a response pair produces a sparse code that, following the Structural Topic Model, shifts the prior of a ProdLDA topic model, and a classifier predicts the human preference from the resulting topic proportions alone. Across six pairwise-preference datasets and four topic granularities, topic proportions and SAE features are complementary: a probe on both outperforms a probe on either alone. Finally, we check that the explanations are usable by someone outside the model: an LLM judge that sees only the two responses and the validated natural-language traits of their dominant topic recovers the human label with 53–74% accuracy, above the 50% chance level in all 24 dataset–granularity settings. SAE-STM thus provides preference predictions that come with a validated topic-and-feature explanation.

1 INTRODUCTION

Reward models trained on pairwise human preference data are now a central component of the LLM alignment pipeline: RLHF fits an explicit reward model to judgments over response pairs (Christiano et al., 2017; Ouyang et al., 2022), and DPO optimizes the policy against an implicit one (Rafailov et al., 2023). In practice, this scoring function is a black box. A reward model can tell us *that* response a was preferred over response b with some confidence, but not *what about* a drove that judgment – whether it was helpfulness, safety, verbosity, tone, or a spurious correlate of any of these. This opacity is not a cosmetic shortcoming: it makes it hard to audit a reward model for the biases it has learned, to debug disagreements between annotators and the model, or to justify downstream alignment decisions to stakeholders who are not machine learning experts.

Two existing families of methods each address part of this problem, but not the whole of it. *Neural topic models* Srivastava & Sutton (2017) produce a compact, human-readable vocabulary – a small number of topics, each a distribution over words – and can be fit directly on the task-relevant signal (e.g., the words that distinguish a preferred response from a rejected one). Still, that vocabulary is coarse: a handful of topics cannot capture the many fine-grained, possibly overlapping factors that drive a preference judgment. *Sparse autoencoders* (SAEs) go the opposite direction: fit over dense representations, they recover a sparse, comparatively fine-grained and often monosemantic set of features (Bricken et al., 2023; Cunningham et al., 2024; Gao et al., 2024). Recent work applies SAEs to preference data directly: WIMHF (Movva et al., 2025a) fits an SAE on response-embedding differences and describes each feature in natural language. But the resulting features form a flat, unstructured list, interpreted one at a time and disconnected from the model that makes the prediction. Neither family alone gives a preference model that is both fine-grained *and* legible.

We introduce SAE-STM, which connects the two. A Top-k SAE encodes the embedding difference between a response pair into a sparse code; following the Structural Topic Model (Roberts et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2014), this code acts as a covariate that shifts the prior of a ProdLDA topic model, so that each topic is linked to the SAE features that move it. A lightweight classifier then predicts the human preference from the topic proportions $\boldsymbol{\theta}$ alone. Every prediction is therefore routed through a small set of topics, each described by its top words and by the SAE features most strongly tied to it. Unlike WIMHF, which interprets SAE features one at a time, SAE-STM organizes them into topics that also carry the predictive signal.

We evaluate SAE-STM on six pairwise-preference datasets (Chatbot Arena, HH-RLHF, PKU-SafeRLHF, PRISM, SHP, and Tulu) at four topic granularities ($K \in \{20, 50, 150, 300\}$), and ask two questions. **(RQ1)** Do topics and SAE features carry complementary information? A probe on both outperforms a probe on either alone in all 24 settings (Section 5.1). **(RQ2)** Are the resulting explanations usable outside the model? An LLM judge that sees only the two responses and the validated traits of their dominant topic recovers the human label with 53–74% accuracy, above chance in every setting (Section 5.2.3). We also find that how interpretable the features are depends strongly on the domain: validated features are common on HH-RLHF and rare on PKU-SafeRLHF (Section 5.2.1).

Our contributions are:

1. SAE-STM, an architecture in which a sparse autoencoder's code shifts the prior of a neural topic model, and a preference classifier predicts from topic proportions alone (Section 3).
2. An empirical study on six preference datasets and four granularities showing that SAE features and topic proportions carry complementary preference signal (Section 5.1).
3. An evaluation of the explanations, adapting the validation protocol of Movva et al. (2025b): features are checked against independent LLM annotations and then tested by an LLM judge that predicts preferences from them (Section 5.2).

2 Related Work

Preference modeling. Reward models are built on the Bradley–Terry model of paired comparisons (Bradley & Terry, 1952), which maps pairwise outcomes to a latent scalar per item. Learning from pairwise human comparisons (Christiano et al., 2017) underlies both reward-model-based RLHF (Ouyang et al., 2022) and reward-free objectives such as DPO (Rafailov et al., 2023). Most reward models are pointwise: they score each response independently and compare the scores. Pairwise models such as PairRM (Jiang et al., 2023) encode both responses jointly and capture contrasts that independent scoring misses. SAE-STM is pairwise in this sense, since it operates on the embedding difference $\mathbf{e}_a - \mathbf{e}_b$, but it routes that difference through a topic bottleneck before classification.

Interpreting preferences and reward models. A growing body of work asks *why* one response is preferred. Some methods fix interpretable dimensions in advance: Compositional Preference Models (Go et al., 2024) score responses on LLM-rated attributes and combine them with logistic regression, and ArmoRM (Wang et al., 2024) learns a mixture over human-interpretable reward objectives such as honesty, verbosity, and safety. Others discover the dimensions from data. Inverse Constitutional AI (Findeis et al., 2025) compresses a preference dataset into natural-language principles and validates them by having an LLM reconstruct the preferences, a protocol close to our LLM-judge evaluation. WIMHF (Movva et al., 2025a) fits an SAE on response-embedding differences and describes the features that predict preferences; Wedgwood et al. (2026) apply a similar SAE-based pipeline to explain the preferences of LLM judges. SAEs have also been used inside reward models, either to compare base and RLHF-tuned models (Marks et al., 2023) or to build an interpretable reward head over SAE features of an LLM (Zhang et al., 2026). These analyses also serve as audits: response length alone explains much of what reward models learn (Singhal et al., 2023). SAE-STM shares the goal of data-driven discovery with WIMHF and ICAI, but instead of producing a flat list of features or principles, it organizes SAE features into topics and makes the prediction depend on those topics alone.

Topic models with covariates and supervision. Neural topic models replace the sampling or mean-field inference of LDA (Blei et al., 2003) with amortized variational inference; ProdLDA (Srivastava & Sutton, 2017), the decoder we build on, models topic–word distributions as a product of experts. Later models use pretrained embeddings as encoder input or topic space (Dieng et al., 2020; Bianchi et al., 2021; Grootendorst, 2022). Two lines of work link topics to external variables. Supervised

LDA (McAuliffe & Blei, 2007) predicts a response variable from topic proportions, and the Structural Topic Model (Roberts et al., 2014; 2019) lets observed document metadata shift the topic prior. SCHOLAR (Card et al., 2018) combines both in a neural topic model, with metadata covariates and a label predicted from the topics, and is the closest topic-model relative of SAE-STM. The difference is where the covariate comes from: in STM and SCHOLAR it is observed metadata, whereas in SAE-STM it is a sparse code learned by the SAE from the response pair itself. Predicting only from topic proportions also makes SAE-STM a concept bottleneck model (Koh et al., 2020); unlike Text Bottleneck Models (Ludan et al., 2023), whose concepts are proposed and scored by an LLM, our concepts are learned topics.

Sparse autoencoders and automated interpretation. SAEs decompose dense representations into an overcomplete, sparsely activating dictionary whose features are often monosemantic (Elhage et al., 2022; Cunningham et al., 2024; Bricken et al., 2023; Templeton et al., 2024). Top-k activations improve the reconstruction–sparsity trade-off (Gao et al., 2024); we use the BatchTopK variant (Bussmann et al., 2024) and the nested reconstruction loss of Matryoshka SAEs (Bussmann et al., 2025). Features are typically described by an LLM and scored by how well the description predicts held-out activations (Bills et al., 2023; Paulo et al., 2025); HypotheSAEs (Movva et al., 2025b) applies this recipe to text embeddings to generate hypotheses about a target variable, and our feature-validation step follows it. Recent work also connects SAEs and topic models directly, either by showing that the SAE objective corresponds to a topic model over embeddings (Girrbach & Akata, 2025) or by defining topics as SAE directions in an LLM's activation space (Zheng et al., 2026). Both are unsupervised; SAE-STM instead keeps the two components separate and uses the SAE code to condition a topic model that predicts a label. Finally, Kantamneni et al. (2025) find that SAE features rarely beat strong linear baselines in probing, a caution for any claim about the predictive value of SAE features.

3 METHOD

3.1 PROBLEM SETUP

Human preference datasets consist of pairwise comparisons $\{(p^{(i)}, r_a^{(i)}, r_b^{(i)}, y^{(i)})\}_{i=1}^N$, where each example contains a prompt $p^{(i)}$, two candidate responses $r_a^{(i)}$ and $r_b^{(i)}$, and a binary label $y^{(i)}$, with $y^{(i)} = 1$ when $r_a^{(i)}$ is preferred and $y^{(i)} = 0$ otherwise. Our goal is to predict y while identifying interpretable natural-language concepts that distinguish the two responses and are associated with the observed preference.

SAE-STM represents each response by a dense embedding and a bag of words, so the training set becomes

$$\mathcal{D} = \{(\mathbf{e}_a^{(i)}, \mathbf{e}_b^{(i)}, \mathbf{w}_a^{(i)}, \mathbf{w}_b^{(i)}, y^{(i)})\}_{i=1}^N,$$

with embeddings $\mathbf{e}_a, \mathbf{e}_b \in \mathbb{R}^d$ of r_a and r_b and bag-of-words vectors $\mathbf{w}_a, \mathbf{w}_b \in \mathbb{R}_{\geq 0}^V$ over a vocabulary of size V. The embeddings feed the SAE and the topic encoder; the bags of words are used only as the topic model's reconstruction target.

A standard preference model predicts y directly from $(\mathbf{e}_a, \mathbf{e}_b)$. SAE-STM instead routes the prediction through two inspectable intermediates (Figure 1): a sparse binary code $\mathbf{z} \in \{0,1\}^M$ computed by an SAE from the embedding difference, and topic proportions $\boldsymbol{\theta} \in \Delta^{K-1}$ over K topics, whose prior is shifted by $\mathbf{z}$ as in the Structural Topic Model. The classifier sees only $\boldsymbol{\theta}$:

$$\hat{y} = \sigma(\mathrm{MLP}_{\mathrm{cls}}(\boldsymbol{\theta})) \,. \tag{1}$$

The prediction therefore depends on $(\mathbf{e}_a, \mathbf{e}_b, \mathbf{w}_a, \mathbf{w}_b)$ only through $\boldsymbol{\theta}$, which makes SAE-STM a concept bottleneck model (Koh et al., 2020) whose concepts are topics. The bottleneck guarantees that topics are the only channel to the prediction. It does not make the effect of each topic on $\hat{y}$ directly readable, because $\mathrm{MLP}_{\mathrm{cls}}$ is nonlinear. We therefore explain a prediction through the topics that dominate $\boldsymbol{\theta}$ and the SAE features that W_{pc} ties to them (Section 4).

3.2 ARCHITECTURE

Sparse autoencoder. The SAE encodes $\mathbf{x}_{\mathrm{SAE}} = [\mathbf{e}_a - \mathbf{e}_b;\ \mathbf{e}_b - \mathbf{e}_a] \in \mathbb{R}^{2d}$ into pre-activations $\mathbf{h} \in \mathbb{R}^M$ and reconstructs its input from a sparse code. During training it keeps the kB largest acti-

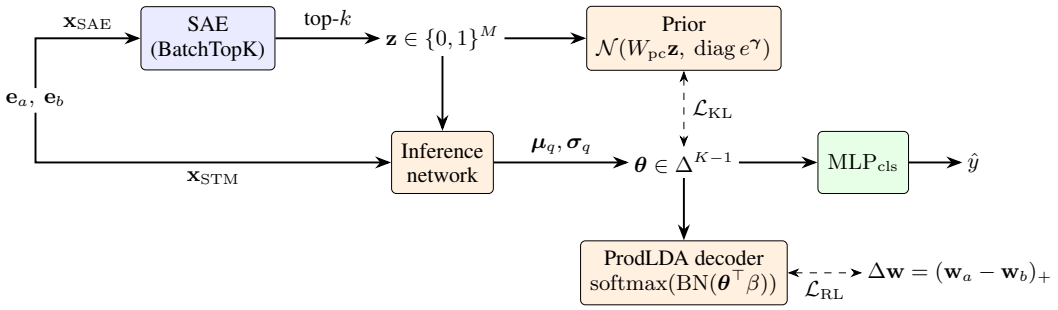


Figure 1: SAE-STM. An SAE over the embedding difference yields a binary code $\mathbf{z}$, which enters both the topic encoder and the mean of the topic prior. The classifier predicts the preference from the topic proportions $\boldsymbol{\theta}$ alone. Blue: SAE; orange: topic model; green: classifier.

vations across a mini-batch of B examples (BatchTopK); at inference it uses a threshold τ estimated during training. The covariate for the topic model keeps the k most active neurons of each example, counting only positive activations:

$$z_j = \mathbf{1}\big[j \in \text{top}_k(\mathbf{c})\big], \qquad \mathbf{c} = \text{ReLU}\big(\mathbf{h} \odot \mathbf{1}[|\mathbf{h}| > \tau]\big). \tag{2}$$

The input contains both $\mathbf{e}_a - \mathbf{e}_b$ and its negation because the ReLU discards negative activations; with both signs available, one neuron can encode "r_a shows more of X" and another "r_b shows more of X". $\mathbf{z}$ records which features are present, not how strongly, so each column of W_{pc} below can be read as the effect of one feature on the topic prior.

Structural topic model. An inference network maps $\mathbf{u} = [\mathbf{x}_{\text{STM}};\ \mathbf{z}]$, where by default $\mathbf{x}_{\text{STM}} = [\mathbf{e}_a - \mathbf{e}_b;\ \frac{1}{2}(\mathbf{e}_a + \mathbf{e}_b)]$, to a Gaussian posterior $q(\boldsymbol{\eta} \mid \mathbf{u}) = \mathcal{N}(\boldsymbol{\mu}_q, \text{diag}(\boldsymbol{\sigma}_q^2))$ over topic logits, and $\boldsymbol{\theta} = \text{softmax}(\boldsymbol{\eta})$. As in STM (Roberts et al., 2014), where observed covariates shift the mean of a logistic-normal prior on topic prevalence, the SAE code shifts the prior mean:

$$p(\boldsymbol{\eta} \mid \mathbf{z}) = \mathcal{N}\big(W_{\text{pc}}\,\mathbf{z},\ \text{diag}(e^{\boldsymbol{\gamma}})\big), \qquad W_{\text{pc}} \in \mathbb{R}^{K \times M}. \tag{3}$$

The difference from STM is that the covariate is learned from the response pair rather than observed as metadata. $\mathbf{z}$ enters the model in two places. Through the prior, it pulls the posterior toward $W_{\text{pc}}\mathbf{z}$ during training. Through the encoder input, it also shapes $\boldsymbol{\theta}$ at test time, when no prior is involved. A ProdLDA decoder (Srivastava & Sutton, 2017) with topic–word matrix $\beta \in \mathbb{R}^{K \times V}$ reconstructs the *contrastive* bag of words $\Delta\mathbf{w} = \max(\mathbf{w}_a - \mathbf{w}_b, \mathbf{0})$, so topics are fit to the words that r_a uses more than r_b. Since every pair is also presented in swapped order (Section 4), the words that r_b uses more are covered as well.

Preference classifier. A small MLP maps $\boldsymbol{\theta}$ to $\hat{y}$ (Eq. 1). The model is not antisymmetric by construction, i.e., $\hat{y}(r_a, r_b) = 1 - \hat{y}(r_b, r_a)$ is not enforced; we rely on the order-swap augmentation to approximate this property.

3.3 Training objective

All components are trained jointly by minimizing

$$\mathcal{L} = \lambda_{\text{STM}}\big(\alpha_t\,\mathcal{L}_{\text{KL}} + \mathcal{L}_{\text{RL}}\big) + \lambda_{\text{label}}\,\mathcal{L}_{\text{cls}} + \lambda_{\text{SAE}}\,\mathcal{L}_{\text{SAE}} + \lambda_{\text{align}}\,\mathcal{L}_{\text{align}} + \lambda_{\text{div}}\,\mathcal{L}_{\text{div}}, \tag{4}$$

where

- $\mathcal{L}_{\text{RL}}$ is the negative log-likelihood of $\Delta\mathbf{w}$ under the ProdLDA decoder;
- $\mathcal{L}_{\text{KL}} = \text{KL}\big(q(\boldsymbol{\eta} \mid \mathbf{u}) \,\|\, p(\boldsymbol{\eta} \mid \mathbf{z})\big)$, with weight α_t annealed linearly from 0 to 1 to avoid posterior collapse;
- $\mathcal{L}_{\text{cls}}$ is the binary cross-entropy of $\hat{y}$;
- $\mathcal{L}_{\text{SAE}}$ is the Matryoshka normalized-MSE reconstruction loss of the SAE (Bussmann et al., 2025) plus an auxiliary loss that revives dead neurons;

- $\mathcal{L}_{\text{align}}$ is the Jensen–Shannon divergence between $\boldsymbol{\theta}$ (with gradients stopped) and $\text{softmax}(W_{\text{pc}}\mathbf{c})$. It trains W_{pc} so that the topics predicted from SAE activations match those inferred by the encoder, which is what lets us read row k of W_{pc} as the SAE signature of topic k (Section 4);
- $\mathcal{L}_{\text{div}}$ is the mean squared cosine similarity between rows of β, which keeps topics from collapsing onto one another.

Because $\mathbf{z}$ is binary, $\mathcal{L}_{\text{cls}}$ and the ELBO terms do not backpropagate into the SAE through it; the SAE is trained by $\mathcal{L}_{\text{SAE}}$ and, through the continuous activations $\mathbf{c}$, by $\mathcal{L}_{\text{align}}$. Without the SAE, $\mathbf{z}$ is dropped from the encoder input, the prior reverts to a learned Gaussian independent of the input, and $\mathcal{L}_{\text{SAE}}$ and $\mathcal{L}_{\text{align}}$ are removed. Appendix A gives every equation, the other encoder-input variants, the training procedure and the hyperparameter values.

4 Experimental Setup

Datasets. We use six datasets from the preprocessed, subjectivity-filtered release of Movva et al. (2025a) (Table 1). Each response is embedded with OpenAI `text-embedding-3-small` ($d = 1536$). Bags of words are built with spaCy (`en_core_web_sm`): tokens are lemmatized and lowercased, and stopwords, non-alphabetic tokens and single characters are removed. The vocabulary is fit on the training split and has about 9.9k lemmas per dataset. Every comparison is used in both orders ($a \leftrightarrow b$, with the label flipped), in training and in every evaluation, so each split is balanced between $y = 0$ and $y = 1$ and contains twice the pairs listed in Table 1. The two copies of a pair are not independent, so accuracies are computed over $2N$ correlated examples.

Dataset	Source	Train	Val	Test
Chatbot Arena	LMArena human votes (Chiang et al., 2024)	15,910	1,989	1,989
HH-RLHF	helpfulness/harmlessness (Bai et al., 2022)	53,812	6,727	6,727
PKU-SafeRLHF	safety-focused RLHF (Dai et al., 2023; Ji et al., 2024)	25,358	3,170	3,170
PRISM	participatory, cross-cultural (Kirk et al., 2024)	27,443	2,980	6,342†
Reddit (SHP)	comment score/time preferences (Ethayarajh et al., 2022)	25,140	3,142	3,143
Tulu mixture	instruction-following, human+synthetic (Ivison et al., 2023)	27,776	3,472	3,472

Table 1: Labeled pairwise comparisons per split, before the $2\times$ order swap. †PRISM test is the release's test split of users seen during training.

Training. We train SAE-STM with the encoder input and $K \in \{20, 50, 150, 300\}$ on every dataset, giving $6 \times 4 = 24$ configurations. Following Movva et al. (2025a), we use a small SAE dictionary, $(M, k) = (32, 4)$, which yields less redundant and more interpretable features; Appendix C compares it with $(512, 16)$. All models are trained with Adam (learning rate 10^{-4}, batch size 512, no scheduler) for 250 epochs, with the KL weight annealed over the first 50 epochs, and the final-epoch checkpoint is evaluated. Loss weights and layer sizes are in Appendix A.4.

Probes. To measure how much linearly decodable preference signal a representation contains, we fit a logistic-regression probe (scikit-learn defaults, ℓ_2 penalty, $C = 1$) on it. The classification head MLP_{cls} of SAE-STM is not used. We probe three representations of each SAE-STM checkpoint:

- $\boldsymbol{\theta}_{\text{SAE}}$: the topic proportions of SAE-STM.
- **SAE**: the continuous SAE activations $\mathbf{c}$, analogous to the WIMHF setup of a linear model over SAE features, except that the SAE is trained in the same run as the topic model; it receives gradients from $\mathcal{L}_{\text{SAE}}$ and $\mathcal{L}_{\text{align}}$ but not from $\mathcal{L}_{\text{cls}}$.
- $\boldsymbol{\theta}_{\text{SAE}}$**+SAE**: the concatenation of the two.

As a black-box reference, we also fit the same probe on the standardized embedding difference $\mathbf{e}_a - \mathbf{e}_b$ (**Embedding**), with no topic model or SAE. Appendix B reports the held-out test accuracy of each representation: the probe on $\boldsymbol{\theta}_{\mathrm{SAE}}$ is on average 0.4 points below the Embedding probe (0.629 vs. 0.633) while using 20–300 features instead of 1,536.

Evaluation regimes. We report three kinds of measurement and keep them separate:

1. *Probe accuracy* on $\boldsymbol{\theta}_{\mathrm{SAE}}$, SAE activations, and their concatenation. The probe quantifies how much preference-relevant signal is captured by each representation (Section 5.1).
2. *Validation AUC*, used only to choose the best K per dataset; the test split is never used for this choice (Appendix H).
3. *LLM-judge accuracy* of the natural-language traits, described next (Section 5.2.3).

Interpretation protocol. We describe and validate SAE features following Movva et al. (2025b;a), then attach the validated features to topics and test whether they predict preferences.

1. **Feature descriptions.** For each SAE neuron, `gpt-4.1` sees 10 training pairs sampled from the top 5% of the neuron's nonzero activations and writes a one-sentence description. To validate it, `gpt-4.1-mini` annotates a separate random sample of 100 training pairs on which the neuron is active (50 above and 50 below its median activation) for whether each shows the described concept. We compute the Pearson correlation between these binary annotations and the activations, and call a neuron *good* if $r > 0.3$ and $p < 0.05$.
2. **Topic traits.** For topic k, we apply a softplus to row k of W_{pc} (Eq. 3) and normalize it to sum to one, which makes the weights positive and comparable across topics. We keep the 10 features with the largest weight. Those that are good neurons become the topic's *traits*; the rest are discarded.
3. **LLM judge.** For each validation pair, we take the dominant topic, $\arg\max_k \theta_k$, and ask `gpt-5-mini`, for each of its traits, which response shows the trait more (the prompt is in Appendix G). The judge predicts the response that wins more traits. A pair counts as correct only if the preferred response wins strictly more traits. Ties, including pairs whose dominant topic has no traits, count as errors.

5 Results

5.1 RQ1: Do topics and SAE features carry complementary information?

We fit a linear probe on $\boldsymbol{\theta}_{\mathrm{SAE}}$ alone, on the SAE code $\mathbf{z}$ alone, and on their concatenation, for each of the $6 \times 4 = 24$ dataset$\times K$ configurations, and compare accuracy.

Dataset	K	$\boldsymbol{\theta}_{\mathrm{SAE}}$	SAE	$\boldsymbol{\theta}_{\mathrm{SAE}}$+SAE	Gain vs $\boldsymbol{\theta}_{\mathrm{SAE}}$	Gain vs SAE
Chatbot Arena	300	0.649	0.682	**0.732**	+0.083	+0.050
PKU	300	0.679	0.654	**0.722**	+0.043	+0.068
Reddit	50	0.677	0.733	**0.735**	+0.059	+0.003
Tulu	300	0.686	0.671	**0.725**	+0.038	+0.054

Table 2: Representative sample of 24 configurations. (see Section 4).

The joint $\boldsymbol{\theta}_{\mathrm{SAE}}$+SAE probe beats both isolated probes in all 24 configurations, by a comfortable margin even at its weakest points. This is the strongest, most generalizable quantitative result in the study. Table 3 shows the gain does not simply "grow with K" as a single trend – the two components of the gain move in *opposite* directions: the advantage over $\boldsymbol{\theta}_{\mathrm{SAE}}$ alone *shrinks* as topics get finer (mean $+0.069 \rightarrow +0.049$ acc from K=20 to 300, since a finer topic model is a stronger standalone probe on its own), while the advantage over the SAE alone *grows* (mean $+0.008 \rightarrow +0.051$ acc), since the SAE code's size is fixed while the topic space it must complement keeps growing. The two probes are complementary for different reasons at different granularities: at coarse granularity, $\mathbf{z}$ mainly compensates for $\boldsymbol{\theta}_{\mathrm{SAE}}$'s limited resolution; at fine granularity, $\boldsymbol{\theta}_{\mathrm{SAE}}$ mainly compensates for information the fixed-size SAE code cannot capture on its own.

K	Mean gain vs θ_{SAE}	Mean gain vs SAE
20	$+0.069$	$+0.008$
50	$+0.058$	$+0.008$
150	$+0.054$	$+0.026$
300	$+0.049$	$+0.051$

Table 3: Mean joint-probe accuracy gain over each isolated probe, averaged across the 6 datasets, by K.

5.2 RQ2: Are the explanations usable outside the model?

5.2.1 Validity of SAE feature descriptions

A neuron is a *"good neuron"* under the same criterion used for its natural-language interpretation in the first place (Section 4): correlation > 0.3 and $p < 0.05$ between its LLM-written interpretation, re-annotated on held-out examples, and its actual activation.

Averaged over all four granularities, the good-neuron rate (defined in Appendix D, which also reports per-configuration values) is sharply *domain*-dependent: HH-RLHF (0.58) sits far above Reddit (0.45), Tulu (0.42), PRISM (0.42), and Chatbot Arena (0.33), with PKU lowest (0.24). Granularity's effect, by contrast, is not a single trend but a domain-specific one: pooled across all 24 configurations, K and the good-neuron rate are essentially uncorrelated (r=0.001) – but that null result hides opposite within-dataset trends (Tulu r=+0.68, improving with more topics; HH-RLHF r=−0.97, sharply worse with more topics), so "more topics" is not a context-free knob for interpretability. What granularity does predict consistently is redundancy: K correlates strongly with the number of overlapping good-neuron pairs across topics (r=0.95) and inversely with the count of topic-exclusive features (r=−0.65) – more topics means the same good neurons increasingly get reused across topics rather than each topic getting its own, echoing the redundancy↔specificity trade-off already noted for SAE dictionary size (Section 4).

Three SAE-neuron interpretations, verified against their own top-activating training examples (granularity noted per example):

- *PKU-SafeRLHF* (K=150, correlation 0.77): "uses sarcasm, irony, or passive-aggressive language." Top-activating example (excerpt): *"One subtle way to demean someone else's intelligence in public settings is through the use of sarcasm or humor that plays on people's intelligence…"* – and: *"You can make someone feel bad about themselves by talking to them in a negative and demeaning way… You can also use sarcasm, criticism, humiliation or insults…"* Here the neuron tracks a tone (sarcasm/put-downs) that recurs across otherwise distinct harmful-advice prompts, rather than a safety category per se – one illustration of why PKU's good-neuron rate is low overall (0.24 on average): the neurons that do pass validation often track surface style instead of the safety-relevant content the dataset is nominally about.
- *HH-RLHF* (K=50, correlation 0.88): "lists items or examples without elaboration or commentary." Top-activating example (excerpt): *"This list isn't comprehensive, but here are some recent songs that won Grammy awards: American Idiot, Green Day / Home, Edward Sharpe & The Magnetic Zeros / Hello, Adele…"* – and, on an unrelated prompt: *"Here are some ideas that you might be interested in: * Band of Brothers * Dark Angel * Deadwood…"* A purely *formatting* trait (bare enumeration, no elaboration) that recurs regardless of topic – consistent with HH-RLHF's high good-neuron rate (0.58 on average) reflecting genuinely verbalizable, response-*style* regularities.
- *Reddit* (K=300, correlation 0.89): "references friendship or friends." Top-activating example (excerpt): *"My best friends that have been with me my whole life have never seen my accounts… if you are that close, then just ask him…"* — and: *"Is there a friend you can leave your dog with for a few months…"* A content trait tied to Reddit's informal, socially-oriented register, matching the domain-specific vocabulary WIMHF also reports for this dataset.

Zero dead neurons across all 24 evaluated SAE configs – SAE health is not the interpretability bottleneck; domain and granularity are, and their effects do not decompose into one global trend for either.

5.2.2 From features to topic traits

The two preceding results (good-neuron rate, qualitative neuron examples) look at SAE features in isolation. Here we show the artifact that actually feeds the LLM judge: for each topic k, the top-10 SAE features by $|W_{\text{pc}}[k,:]|$ (Section 4, Stage 2), annotated with whether each passed the good-neuron validation of Stage 1 (✓) or not (✗) – only the former are used as that topic's traits downstream. Table 4 shows two contrasting topics; Appendix E shows three more topics for each of the six datasets.

HH-RLHF, Topic 2 (K=50) *Top words:* money, pay, credit, debt, savings, loan, bank, payment, expenses, income	**PKU-SafeRLHF, Topic 8** (K=150) *Top words:* making, person, guilt, fear, sense, language, feel, tactics, someone, manipulate
✓ discusses prices, costs, or budgets	✓ uses sarcasm, irony, or passive-aggressive language
✓ mentions money or financial matters	✗ does not provide concrete, step-by-step instructions for manipulation or harm
✗ provides factual information or explanations	✗ does not explicitly discuss manipulation or coercive strategies
✗ does not ask the user a direct question	✗ does not provide explicit instructions for manipulation or unethical behavior
✓ mentions Amazon or Amazon-related products or services	✗ does not describe or endorse manipulative or coercive tactics
✓ explicitly references or discusses models or modeling	✓ does not mention or describe gaslighting
✓ provides direct information/advice rather than asking clarifying questions	✗ does not provide explicit, detailed instructions for causing emotional harm
✓ references time, duration, or temporal aspects	✗ lacks step-by-step or enumerated instructions
✗ asks follow-up questions to gather more information	✗ describes manipulative or coercive strategies

Table 4: Top-10 SAE features by $|W_{\text{pc}}[k,:]|$ for one representative topic from each of two datasets, in descending weight order; ✓ = passed good-neuron validation (usable as a trait), ✗ = did not.

The two topics illustrate the domain gap in good-neuron rates (Section 5.2.1) concretely rather than only statistically. HH-RLHF's Topic 2 is unambiguously about personal finance, and 6 of its top 10 associated features validate and describe that same content (prices, money, financial advice) – topic and feature agree on what the topic is about. PKU's Topic 8 is unambiguously about emotional manipulation and coercion tactics, but only 2 of its top 10 associated features validate, and neither is about manipulation: one is a generic tone descriptor (sarcasm) and the other a narrow negative (absence of gaslighting language). The eight manipulation-specific candidate traits –arguably the most useful ones for a reader trying to understand why PKU's classifier prefers one response over another on this topic – all failed the held-out correlation check and are unavailable to the LLM judge (Section 5.2.3). This is a direct illustration of the qualitative-example finding above (PKU neurons that pass validation track surface style, not safety content) at the topic level, and a concrete reason to read PKU's LLM-judge accuracy (Table 14) as measuring a narrower, style-based notion of interpretability than the dataset's subject matter would suggest.

5.2.3 External validation via LLM judge

Accuracy ranges from 52.6% (PKU, K=150) to 74.3% (HH-RLHF, K=20) across the 24 dataset×granularity configurations (Table 14 in Appendix F) – comfortably above the ∼50% majority-class/chance floor of the underlying binary pairwise task in every case, but with no single K dominating across datasets.

Cross-checking against the topic-quality trends directly (Pearson r over all 24 configurations): judge accuracy correlates positively with the good-neuron rate (r=+0.53) – expected, since a config with more validated traits gives the judge more signal to work with.

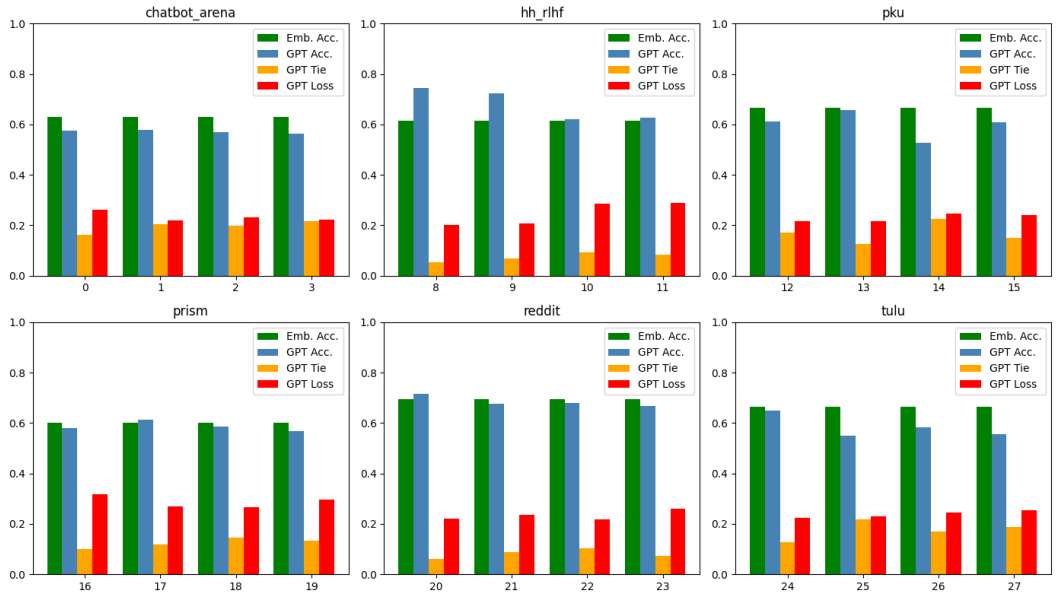


Figure 2: LLM-judge accuracy against the embedding probe's test accuracy, plus the judge's tie and loss rates, per dataset and K.

As a further sanity check – not a claim that the two should match exactly, since they answer different questions (Section 4) – Figure 2 compares judge accuracy directly against the accuracy of the Embedding probe (Section 4), fit on the training split and evaluated on the test split. Across the 24 dataset$\times K$ configurations shown, the two track each other reasonably closely: mean $|\text{Emb. Acc} - \text{GPT Acc}|$=0.052, with the largest gap at PKU, K=150 (0.141) – the same configuration already flags as PKU's weakest point for validated, content-relevant traits. Tie rate is consistently low (mean 0.137, never above 0.226): the trait-based majority vote is rarely deadlocked, unlike a coin flip would be a substantial fraction of the time. Loss rate is higher (mean 0.245, range 0.202–0.318) but stays below tie+accuracy everywhere, i.e. the judge's traits more often point toward the correct response than the incorrect one. HH-RLHF is the one clear exception to the "track each other closely" pattern: at K=20/50 the judge outperforms the embedding classifier by 11–13 points, converging to within 1–2 points at K=150/300. This is not the embedding classifier changing — its accuracy is flat at 0.613 across all four K, since it does not depend on the topic model at all – but the judge's accuracy drops in step with HH-RLHF's good-neuron rate (Section 5.2.1), which falls sharply over the same range (r=−0.97 with K: 0.63 → 0.66 → 0.56 → 0.48). The gap is largest exactly where the good-neuron rate — and hence the judge's accuracy — peaks, at coarse granularity, not where it is weakest.

6 Conclusion

SAE-STM couples a top-k sparse autoencoder to a structural-topic-model prior and a lightweight preference classifier, so that every prediction is routed through, and explained by, a compact set of topic proportions rather than an opaque embedding. Across six pairwise-preference datasets and four topic granularities, topic proportions and SAE features are more informative in combination than either representation alone (RQ1, Section 5.1), and the natural-language traits attached to each topic let an independent LLM judge recover the human preference label above chance in every setting (RQ2, Section 5.2.3). How interpretable the features are depends strongly on the domain: validated features are common on HH-RLHF and rare on PKU-SafeRLHF, where the features that pass validation tend to track style rather than safety-relevant content (Section 5.2.1).

AI USE STATEMENT

LLMs as part of the method. Large language models are a component of the method and of its evaluation (Section 4). Responses are embedded with OpenAI `text-embedding-3-small`; SAE features are described by `gpt-4.1`; these descriptions are validated against annotations produced by `gpt-4.1-mini`; and the LLM judge of Section 5.2.3 is `gpt-5-mini`. All prompts and model versions will be released with the code.

AI assistance in preparing the paper. We used an AI coding and writing assistant (Claude, Anthropic) to edit and restructure the text, to review the manuscript for internal inconsistencies, and to check the analysis code against the claims made in the paper. Under our instructions, it also generated the LaTeX tables in the appendix directly from our saved result files and proposed the illustrative topics shown in Appendix E, which we reviewed.

Generative AI was not used to generate the research ideas, and no AI tool is listed as an author.

We have reviewed all AI-assisted work and take full responsibility for the content of this paper, including text, claims, and artifacts produced with the aid of generative AI.

ACKNOWLEDGMENTS

Acknowledgments, including funding sources, are omitted for anonymous review and will be added in the camera-ready version.

REFERENCES

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Federico Bianchi, Silvia Terragni, and Dirk Hovy. Pre-training is a hot topic: Contextualized document embeddings improve topic coherence. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2021.

Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. Language models can explain neurons in language models. OpenAI, 2023.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3(4-5):993–1022, 2003. URL `https://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf`.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

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

Bart Bussmann, Patrick Leask, and Neel Nanda. BatchTopK sparse autoencoders. *arXiv preprint arXiv:2412.06410*, 2024.

Bart Bussmann, Noa Nabeshima, Adam Karvonen, and Neel Nanda. Learning multi-level features with matryoshka sparse autoencoders. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, pp. 6077–6101, 2025.

Dallas Card, Chenhao Tan, and Noah A. Smith. Neural models for documents with metadata. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2031–2040. Association for Computational Linguistics, 2018. doi: 10.18653/v1/p18-1189. URL `http://dx.doi.org/10.18653/v1/P18-1189`.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating LLMs by human preference. In *International Conference on Machine Learning (ICML)*, 2024. arXiv:2403.04132.

Paul F. Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=F76bwRSLeK.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe RLHF: Safe reinforcement learning from human feedback. *arXiv preprint arXiv:2310.12773*, 2023.

Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453, 2020.

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.

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. Understanding dataset difficulty with $\mathcal{V}$-usable information. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 5988–6008. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/ethayarajh22a.html.

Arduin Findeis, Timo Kaufmann, Eyke Hüllermeier, Samuel Albanie, and Robert Mullins. Inverse constitutional AI: Compressing preferences into principles. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

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, 2024. URL https://arxiv.org/abs/2406.04093.

Leander Girrbach and Zeynep Akata. Sparse autoencoders are topic models. *arXiv preprint arXiv:2511.16309*, 2025.

Dongyoung Go, Tomasz Korbak, Germán Kruszewski, Jos Rozen, and Marc Dymetman. Compositional preference models for aligning LMs. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

Maarten Grootendorst. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. *arXiv preprint arXiv:2203.05794*, 2022.

Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. Camels in a changing climate: Enhancing LM adaptation with Tülu 2. *arXiv preprint arXiv:2311.10702*, 2023.

Jiaming Ji et al. PKU-SafeRLHF: Towards multi-level safety alignment for LLMs with human preference. *arXiv preprint arXiv:2406.15513*, 2024. Also in Proceedings of ACL 2025.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. LLM-Blender: Ensembling large language models with pairwise ranking and generative fusion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023.

Subhash Kantamneni, Joshua Engels, Senthooran Rajamanoharan, Max Tegmark, and Neel Nanda. Are sparse autoencoders useful? a case study in sparse probing. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

Hannah Rose Kirk, Alexander Whitefield, Paul Röttger, Andrew Bean, Katerina Margatina, Juan Ciro, Rafael Mosquera, Max Bartolo, Adina Williams, He He, Bertie Vidgen, and Scott A. Hale. The PRISM alignment dataset: What participatory, representative and individualised human feedback reveals about the subjective and multicultural alignment of large language models. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024. arXiv:2404.16019.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

Josh Magnus Ludan, Qing Lyu, Yue Yang, Liam Dugan, Mark Yatskar, and Chris Callison-Burch. Interpretable-by-design text understanding with iteratively generated concept bottleneck. *arXiv preprint arXiv:2310.19660*, 2023.

Luke Marks, Amir Abdullah, Luna Mendez, Rauno Arike, Philip Torr, and Fazl Barez. Interpreting reward models in RLHF-tuned language models using sparse autoencoders. *arXiv preprint arXiv:2310.08164*, 2023.

Jon D. McAuliffe and David M. Blei. Supervised topic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2007.

Rajiv Movva, Smitha Milli, Sewon Min, and Emma Pierson. What's in my human feedback? learning interpretable descriptions of preference data. *arXiv preprint arXiv:2510.26202*, 2025a.

Rajiv Movva, Kenny Peng, Nikhil Garg, Jon Kleinberg, and Emma Pierson. Sparse autoencoders for hypothesis generation. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025b. arXiv:2502.04382.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022. URL https://arxiv.org/abs/2203.02155.

Gonçalo Paulo, Alex Mallen, Caden Juang, and Nora Belrose. Automatically interpreting millions of features in large language models. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Margaret E. Roberts, Brandon M. Stewart, Dustin Tingley, Christopher Lucas, Jetson Leder-Luis, Shana Kushner Gadarian, Bethany Albertson, and David G. Rand. Structural topic models for open-ended survey responses. *American Journal of Political Science*, 58(4):1064–1082, 2014.

Margaret E. Roberts, Brandon M. Stewart, and Dustin Tingley. stm: An R package for structural topic models. *Journal of Statistical Software*, 91(2):1–40, 2019. doi: 10.18637/jss.v091.i02.

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A long way to go: Investigating length correlations in RLHF. *arXiv preprint arXiv:2310.03716*, 2023.

Akash Srivastava and Charles Sutton. Autoencoding variational inference for topic models, 2017. URL https://arxiv.org/abs/1703.01488.

Adly Templeton et al. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. Transformer Circuits Thread, 2024.

Symbol	Shape	Description
$W_{\text{enc}}, W_{\text{dec}}$	$M \times 2d$, $2d \times M$	SAE encoder and decoder
$\mathbf{b}_{\text{in}}, \mathbf{b}_{\text{neu}}$	$2d$, M	SAE centering and neuron biases
τ	scalar	SAE inference threshold (moving average, not trained)
$\text{MLP}_{\text{enc}}, W_\mu, W_\sigma$	–	Inference network and variational heads
$W_{\text{pc}}, \boldsymbol{\gamma}$	$K \times M$, K	SAE-to-prior projection and prior log-variance
$\boldsymbol{\mu}_p, \boldsymbol{\sigma}_p^2$	K, K	Prior parameters without the SAE
β	$K \times V$	Topic–word matrix
MLP_{cls}	$K \to 34 \to 1$	Preference classifier

Table 5: Learnable parameters of SAE-STM.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024.

James Wedgwood, Chhavi Yadav, and Virginia Smith. Automated concept discovery for LLM-as-a-judge preference analysis. *arXiv preprint arXiv:2603.03319*, 2026.

Shuyi Zhang, Wei Shi, Sihang Li, Jiayi Liao, Hengxing Cai, and Xiang Wang. Interpretable reward model via sparse autoencoder. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.

Carolina Zheng, Nicolas Beltran-Velez, Sweta Karlekar, Claudia Shi, Achille Nazaret, Asif Mallik, Amir Feder, and David M. Blei. Model directions, not words: Mechanistic topic models using sparse autoencoders. *Transactions of the Association for Computational Linguistics*, 2026.

A Full Model Formulation

This appendix gives the complete formulation of SAE-STM summarized in Section 3. Notation: d is the embedding dimension, M the number of SAE neurons, k the number of active SAE neurons per example, K the number of topics, V the vocabulary size, and B the mini-batch size. Table 5 lists the learnable parameters.

A.1 Sparse autoencoder

Input. The SAE encodes the signed embedding difference and its mirror,

$$\mathbf{x}_{\text{SAE}} = \left[\, \mathbf{e}_a - \mathbf{e}_b \;;\; \mathbf{e}_b - \mathbf{e}_a \,\right] \in \mathbb{R}^{2d}. \tag{5}$$

Encoding and sparsification. With a centering bias $\mathbf{b}_{\text{in}}$ and a neuron bias $\mathbf{b}_{\text{neu}}$,

$$\mathbf{h} = W_{\text{enc}} \left(\mathbf{x}_{\text{SAE}} - \mathbf{b}_{\text{in}}\right) + \mathbf{b}_{\text{neu}} \in \mathbb{R}^M. \tag{6}$$

During training we apply BatchTopK (Bussmann et al., 2024): across the B examples of a mini-batch, the kB entries of $\mathbf{h}$ with the largest absolute value are kept and all others are set to zero,

$$s_j = h_j \cdot \mathbf{1}\big[j \in \mathcal{T}_{kB}(\mathbf{h})\big], \quad j = 1, \ldots, M, \tag{7}$$

where $\mathcal{T}_{kB}$ is the index set of the top-kB absolute values over the flattened batch. At inference, a scalar threshold τ replaces the batch-level selection, $s_j = h_j \cdot \mathbf{1}[|h_j| > \tau]$. The threshold is tracked during training as an exponential moving average (rate 10^{-2}) of the smallest absolute activation kept by BatchTopK.

Reconstruction.

$$\hat{\mathbf{x}}_{\text{SAE}} = W_{\text{dec}}\, \mathbf{s} + \mathbf{b}_{\text{in}}, \qquad \|W_{\text{dec},\cdot j}\|_2 = 1 \;\; \forall j. \tag{8}$$

$\mathbf{b}_{\text{in}}$ is initialized to the coordinate-wise median of $\mathbf{x}_{\text{SAE}}$ over the first 8 training batches. W_{dec} is initialized with Xavier-uniform columns normalized to unit norm, and W_{enc} is initialized to $W_{\text{dec}}^\top$; the two matrices are trained independently afterwards. After every optimizer step the decoder columns are renormalized, and before the step the decoder gradient is projected orthogonally to each column.

Prior covariate. The covariate passed to the topic model is derived from the thresholded, non-negative activations

$$\mathbf{c} = \mathrm{ReLU}\big(\mathbf{h} \odot \mathbf{1}[|\mathbf{h}| > \tau]\big) \in \mathbb{R}^M_{\geq 0}, \tag{9}$$

binarized by keeping the k largest entries of each example:

$$z_j = \mathbf{1}\big[j \in \mathcal{T}_k(\mathbf{c})\big], \qquad \mathbf{z} \in \{0,1\}^M. \tag{10}$$

The ReLU discards negative activations, so each neuron signals only the presence of its feature. Because $\mathbf{x}_{\mathrm{SAE}}$ contains both $\mathbf{e}_a - \mathbf{e}_b$ and its negation, a feature that is stronger in r_b than in r_a can still be represented, by a different neuron.

SAE loss. We use the normalized MSE

$$\mathrm{NMSE}(\hat{\mathbf{x}}, \mathbf{x}) = \frac{\mathrm{MSE}(\hat{\mathbf{x}}, \mathbf{x})}{\mathrm{MSE}(\bar{\mathbf{x}}, \mathbf{x}) + \varepsilon}, \tag{11}$$

where $\bar{\mathbf{x}}$ is the batch mean of the target. Following Matryoshka SAEs (Bussmann et al., 2025), the reconstruction loss is averaged over nested prefixes of the dictionary, $\{P_1, \ldots, P_L\}$ with $P_L = M$ (we use $\{8, M\}$):

$$\mathcal{L}_{\mathrm{mat}} = \frac{1}{L} \sum_{\ell=1}^{L} \mathrm{NMSE}\big(W_{\mathrm{dec},:P_\ell}\, \mathbf{s}_{:P_\ell} + \mathbf{b}_{\mathrm{in}},\ \mathbf{x}_{\mathrm{SAE}}\big). \tag{12}$$

An auxiliary loss revives dead neurons by asking them to reconstruct the residual,

$$\mathcal{L}_{\mathrm{aux}} = \mathrm{NMSE}\big(W_{\mathrm{dec}}\, \mathbf{s}_{\mathrm{dead}},\ \mathbf{x}_{\mathrm{SAE}} - \mathrm{sg}(\hat{\mathbf{x}}_{\mathrm{SAE}})\big), \tag{13}$$

where $\mathbf{s}_{\mathrm{dead}}$ keeps the top-$2k$ pre-activations among neurons that have not fired in the last 256 steps, and $\mathrm{sg}(\cdot)$ is the stop-gradient operator. The SAE loss is

$$\mathcal{L}_{\mathrm{SAE}} = \mathcal{L}_{\mathrm{mat}} + \tfrac{1}{32}\, \mathcal{L}_{\mathrm{aux}}. \tag{14}$$

A.2 Structural topic model

Encoder input. The inference network receives a contrastive representation of the pair, concatenated with the covariate:

$$\mathbf{u} = [\,\mathbf{x}_{\mathrm{STM}}\ ;\ \mathbf{z}\,], \qquad \mathbf{x}_{\mathrm{STM}} = \begin{cases} \big[\,\mathbf{e}_a - \mathbf{e}_b\ ;\ \tfrac{1}{2}(\mathbf{e}_a + \mathbf{e}_b)\,\big] & (\texttt{diff_mean}, \text{default}), \\ \mathbf{e}_a - \mathbf{e}_b & (\texttt{diff}), \\ \big[\,\mathbf{e}_a\ ;\ \mathbf{e}_b\,\big] & (\texttt{concat}). \end{cases} \tag{15}$$

Without the SAE, $\mathbf{u} = \mathbf{x}_{\mathrm{STM}}$.

Inference network. A feed-forward network with softplus activations and dropout outputs the parameters of a Gaussian posterior over the unnormalized topic logits $\boldsymbol{\eta}$:

$$\mathbf{h}_q = \mathrm{MLP}_{\mathrm{enc}}(\mathbf{u}), \qquad \boldsymbol{\mu}_q = \mathrm{BN}(W_\mu\, \mathbf{h}_q), \qquad \log \boldsymbol{\sigma}_q^2 = \mathrm{BN}(W_\sigma\, \mathbf{h}_q), \tag{16}$$

where BN is batch normalization without affine parameters and $\boldsymbol{\mu}_q, \log \boldsymbol{\sigma}_q^2 \in \mathbb{R}^K$.

Prior modulated by the SAE. Without the SAE, the prior is $\mathcal{N}(\boldsymbol{\mu}_p, \mathrm{diag}(\boldsymbol{\sigma}_p^2))$ with learned $\boldsymbol{\mu}_p$ and $\boldsymbol{\sigma}_p^2$ (initialized to $\mathbf{0}$ and $1 - 1/K$). With the SAE, the prior mean is a linear function of the covariate, as in the Structural Topic Model:

$$p(\boldsymbol{\eta} \mid \mathbf{z}) = \mathcal{N}\big(W_{\mathrm{pc}}\, \mathbf{z},\ \mathrm{diag}(\exp \boldsymbol{\gamma})\big), \tag{17}$$

with $W_{\mathrm{pc}} \in \mathbb{R}^{K \times M}$ and $\boldsymbol{\gamma} \in \mathbb{R}^K$ learned. Column j of W_{pc} is the shift in topic logits caused by SAE feature j.

Topic proportions. Using the reparameterization trick,

$$\boldsymbol{\eta} = \boldsymbol{\mu}_q + \boldsymbol{\sigma}_q \odot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, I_K), \qquad \boldsymbol{\theta} = \mathrm{softmax}(\boldsymbol{\eta}) \in \Delta^{K-1}. \tag{18}$$

Decoder (ProdLDA). Words are generated from a product of experts over topics (Srivastava & Sutton, 2017),

$$p(\cdot \mid \boldsymbol{\theta}) = \mathrm{softmax}\big(\mathrm{BN}(\boldsymbol{\theta}^\top \beta)\big) \in \Delta^{V-1}, \tag{19}$$

where $\beta \in \mathbb{R}^{K \times V}$ is the unnormalized topic–word matrix. Dropout is applied to $\boldsymbol{\theta}$ during training.

ELBO. The reconstruction target is the contrastive bag of words

$$\Delta \mathbf{w} = \max(\mathbf{w}_a - \mathbf{w}_b,\ \mathbf{0}), \qquad \mathcal{L}_{\mathrm{RL}} = -\sum_{v=1}^{V} \Delta w_v \log p(v \mid \boldsymbol{\theta}), \tag{20}$$

so topics are fit to the words that occur more often in r_a than in r_b. The KL term between the posterior and the (possibly shifted) prior has the closed form

$$\mathcal{L}_{\mathrm{KL}} = \frac{1}{2} \sum_{k=1}^{K} \left[\frac{\sigma_{q,k}^2}{\sigma_{p,k}^2} + \frac{(\mu_{q,k} - \mu_{p,k})^2}{\sigma_{p,k}^2} - 1 + \log \frac{\sigma_{p,k}^2}{\sigma_{q,k}^2} \right]. \tag{21}$$

Its weight follows a linear annealing schedule,

$$\alpha_t = \min\left(\frac{t - T_{\mathrm{warm}}}{T_{\mathrm{KL}}},\ 1 \right), \tag{22}$$

where t is the epoch, $T_{\mathrm{KL}} = 50$ is the annealing length, and T_{warm} is the number of optional SAE-only warm-up epochs ($T_{\mathrm{warm}} = 0$ in all reported experiments).

Topic–SAE alignment. The alignment loss compares the posterior topic proportions with the topic distribution that the SAE activations induce through the prior projection:

$$\mathbf{q}(\mathbf{c}) = \mathrm{softmax}(W_{\mathrm{pc}}\, \mathbf{c}), \qquad \mathcal{L}_{\mathrm{align}} = \mathrm{JSD}\big(\mathrm{sg}(\boldsymbol{\theta}) \,\|\, \mathbf{q}(\mathbf{c})\big), \tag{23}$$

where $\mathrm{JSD}(p \,\|\, q) = \frac{1}{2}[D_{\mathrm{KL}}(p \,\|\, m) + D_{\mathrm{KL}}(q \,\|\, m)]$ with $m = \frac{1}{2}(p + q)$. It uses the continuous activations $\mathbf{c}$ (Eq. 9) rather than $\mathbf{z}$. The stop-gradient on $\boldsymbol{\theta}$ means that the loss moves W_{pc}, and through $\mathbf{c}$ the SAE encoder, toward the posterior, without changing inference.

Topic diversity. ProdLDA has no term that keeps topics distinct, and with many topics several rows of β can converge to near-identical distributions. We penalize the mean squared cosine similarity between pairs of topic–word vectors:

$$\mathcal{L}_{\mathrm{div}} = \frac{1}{K(K-1)} \sum_{k \neq k'} \cos(\beta_k, \beta_{k'})^2. \tag{24}$$

A.3 Preference classifier

The classifier is an MLP with one hidden layer of 34 units, GELU activation and dropout 0.1:

$$\hat{y} = \sigma\big(\mathrm{MLP}_{\mathrm{cls}}(\boldsymbol{\theta})\big), \qquad \mathcal{L}_{\mathrm{cls}} = -\big[y \log \hat{y} + (1 - y) \log(1 - \hat{y})\big]. \tag{25}$$

A.4 Training

All parameters are trained jointly by minimizing Eq. 4 with $\lambda_{\mathrm{STM}} = 1$, $\lambda_{\mathrm{label}} = \lambda_{\mathrm{SAE}} = \lambda_{\mathrm{align}} = 100$ and $\lambda_{\mathrm{div}} = 1$, using Adam (learning rate 10^{-4}, batch size 512) for 250 epochs. Each training step does the following:

1. Compute $\mathbf{x}_{\mathrm{SAE}}$ (Eq. 5) and $\mathbf{x}_{\mathrm{STM}}$ (Eq. 15).
2. Run the SAE to obtain $\mathbf{s}$, $\hat{\mathbf{x}}_{\mathrm{SAE}}$ and $\mathbf{c}$ (Eqs. 6–9); binarize $\mathbf{c}$ into $\mathbf{z}$ (Eq. 10).
3. Encode $\mathbf{u} = [\mathbf{x}_{\mathrm{STM}}; \mathbf{z}]$ into $(\boldsymbol{\mu}_q, \boldsymbol{\sigma}_q)$ (Eq. 16), compute the shifted prior (Eq. 17) and sample $\boldsymbol{\theta}$ (Eq. 18).
4. Decode $p(\cdot \mid \boldsymbol{\theta})$ (Eq. 19) and predict $\hat{y}$ (Eq. 25).
5. Compute $\mathcal{L}_{\mathrm{RL}}$, $\mathcal{L}_{\mathrm{KL}}$, $\mathcal{L}_{\mathrm{cls}}$, $\mathcal{L}_{\mathrm{SAE}}$, $\mathcal{L}_{\mathrm{align}}$ and $\mathcal{L}_{\mathrm{div}}$, and combine them with Eq. 4.
6. Backpropagate, project the SAE decoder gradient orthogonally to its columns, take an Adam step, and renormalize the decoder columns.

Because $\mathbf{z}$ is binary, no gradient from $\mathcal{L}_{\mathrm{RL}}$, $\mathcal{L}_{\mathrm{KL}}$ or $\mathcal{L}_{\mathrm{cls}}$ reaches the SAE through it; the SAE is trained by $\mathcal{L}_{\mathrm{SAE}}$ and, through $\mathbf{c}$, by $\mathcal{L}_{\mathrm{align}}$.

B Predictive Accuracy of the Representations

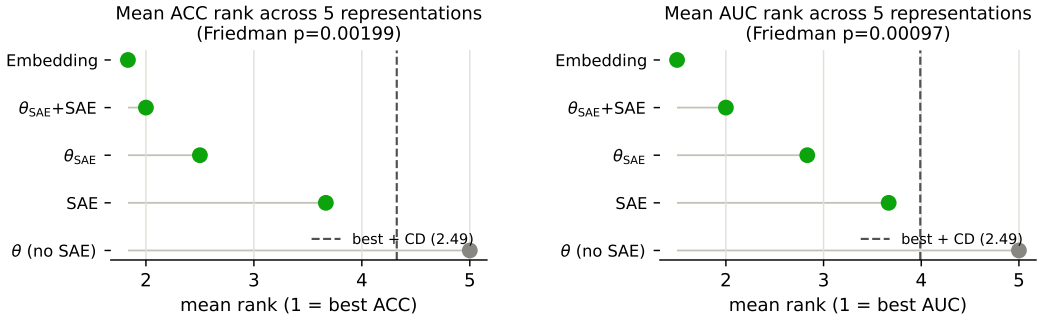
Table 6 reports how much preference signal each representation carries when used on its own. Each column is a logistic-regression probe (Section 4) fit on the training split and evaluated on the test split, and each entry is averaged over the four topic granularities. All SAE-related columns use the $(M, k) = (32, 4)$ dictionary. Besides the representations of Section 4, we include $\boldsymbol{\theta}$ (no SAE), the topic proportions of the same architecture trained without the SAE and without the label loss ($\lambda_{\text{label}} = 0$), which amounts to an unsupervised ProdLDA over the same inputs.

The probe on $\boldsymbol{\theta}_{\text{SAE}}$ is 0.4 points below the Embedding probe on average (0.629 vs. 0.633) while using 20–300 features instead of 1,536; it is more accurate on PKU and Chatbot Arena and less accurate on the other four datasets. The unsupervised topic model is the least accurate on every dataset. Because it lacks both the SAE covariate and the label loss, this gap cannot be attributed to the SAE alone.

Dataset	Embedding	$\boldsymbol{\theta}_{\text{SAE}}$+SAE	$\boldsymbol{\theta}_{\text{SAE}}$	SAE	$\boldsymbol{\theta}$ (no SAE)
Chatbot Arena	0.597	**0.609**	0.604	0.599	0.564
HH-RLHF	**0.612**	0.606	0.606	0.556	0.527
PKU	0.656	**0.672**	0.669	0.619	0.559
PRISM	**0.592**	0.583	0.579	0.582	0.557
Reddit	**0.682**	0.672	0.672	0.657	0.569
Tulu	**0.658**	0.648	0.644	0.615	0.564
Mean	**0.633**	0.632	0.629	0.605	0.557

Table 6: Test accuracy of a logistic-regression probe on each representation, averaged over the four topic granularities (best per row in bold). $\boldsymbol{\theta}_{\text{SAE}}$+SAE gives the probe direct access to the SAE code and is not a topic bottleneck; it is shown as a reference.

To test whether the differences are systematic, we rank the five representations on each dataset and apply a Friedman test over the six datasets, followed by a Nemenyi post-hoc test at α=0.05 (critical difference CD=2.49 in mean rank). The Friedman test rejects equal performance for both accuracy (χ^2=16.93, p=0.002) and AUC (χ^2=18.53, p=0.001). Figure 3 shows the mean ranks. Embedding, $\boldsymbol{\theta}_{\text{SAE}}$+SAE, $\boldsymbol{\theta}_{\text{SAE}}$ and SAE are within the critical difference of the best representation on both metrics; only $\boldsymbol{\theta}$ (no SAE) falls outside it. With six datasets the test has little power, so the absence of a significant difference among the first four should not be read as equivalence.



(a) Accuracy (b) AUC

Figure 3: Mean rank of each representation across the six datasets (1 = best). The dashed line marks the best mean rank plus the Nemenyi critical difference (CD=2.49, α=0.05); green markers are not significantly different from the best representation.

C Effect of SAE Dictionary Size

We compare the dictionary used in the main experiments, $(M, k) = (32, 4)$, with a larger one, $(M, k) = (512, 16)$, with logistic-regression probes (Section 4) fit on the training split and evaluated on the test split. Averaged over the 24 dataset$\times K$ configurations, the effect is mixed. Shrinking the dictionary hurts the standalone SAE probe (-1.9 accuracy and -2.7 AUC points) but helps $\boldsymbol{\theta}_{\mathrm{SAE}}$ ($+0.9$ / $+1.2$) and $\boldsymbol{\theta}_{\mathrm{SAE}}$+SAE ($+0.6$ / $+0.9$), and leaves the embedding probe unchanged.

D Topic–Feature Interpretability Metrics

Table 7 reports, for a sample of configurations, the metrics discussed in Section 5.2.1. They are computed from the top-10 SAE features attached to each topic (Section 4, Topic traits). The *good-neuron rate* is the fraction of a topic's top-10 features that are good neurons, averaged over topics. *Exclusive features* counts the SAE features that appear in the top-10 list of exactly one topic. *Overlapping pairs* counts the pairs of topics whose top-10 lists share at least one feature.

Dataset	K	Good-neuron rate	Exclusive features	Overlapping pairs
HH-RLHF	50	0.66	71	310
Chatbot Arena	150	0.39	59	4 647
Reddit	300	0.47	12	20 666
PKU	150	0.18	30	3 932

Table 7: Topic–feature interpretability metrics for four of the 24 dataset$\times K$ configurations, computed with the larger SAE dictionary, $(M, k) = (512, 16)$.

E Additional Topic Examples

Tables 8–13 extend Table 4 to all six datasets, with three topics per dataset. Each table uses the dataset's best K (Appendix H) and the larger SAE dictionary, $(M, k) = (512, 16)$. For each topic we list its ten most probable words and its top-10 SAE features in descending weight order, marking the features that passed good-neuron validation. We selected topics by hand for coherent top words and at least one validated feature related to the topic, so the tables illustrate what the method produces when it works well and are not a random sample; Appendix D gives aggregate metrics.

Topic 4 *Top words:* chorus, verse, song, dance, fade, outro, sky, moon, oh, whispers	**Topic 5** *Top words:* health, help, body, avoid, minutes, reduce, day, weight, per, water	**Topic 17** *Top words:* harmful, apologize, helpful, cannot, respectful, situation, remember, inappropriate, respect, important
✗ provides the requested creative content regardless of appropriateness or potential sensitivity	✓ provides direct medical or safety advice and urges consultation with healthcare professionals	✓ refuses to participate in generating or editing content that enables discrimination, harassment, or threats toward marginalized groups
✓ provides original song lyrics upon request	✗ provides detailed explanations, rationales, and organized breakdowns	✓ provides direct medical or safety advice and urges consultation with healthcare professionals
✗ provides creative or imaginative writing	✓ provides detailed, informative, and content-rich answers to the prompt	✗ provides a detailed, content-rich response instead of refusing or limiting assistance
✗ does not refuse requests due to ethical, safety, or content guidelines	✓ provides detailed advice, information, or specific suggestions	✓ does not use an elaborate or stylistically playful tone
✓ does not provide a ready-to-use text-to-image AI generation prompt	✗ provides direct answers or analysis to provocative or controversial prompts	✗ provides detailed reasoning for names and concepts, with emphasis on etymology and symbolism
✓ does not provide extended explanations or elaborations	✓ provides a complete answer or solution without requesting additional information or clarification	✗ expresses personal opinions or judgements
✓ provides song identification, interpretation, or analysis	✗ provides a hypothetical reason or excuse for waiving DMV penalty fees	✗ responds in English rather than Chinese
✗ provides detailed, explicit instructions for sensitive or adult-themed DIY activities	✗ does not refuse or critique the user's request	✗ provides a hypothetical reason or excuse for waiving DMV penalty fees
✗ provides a direct answer or creative output rather than refusing or deflecting the prompt	✗ provides detailed, nuanced, and comprehensive explanations or breakdowns	✗ does not respond in the same language as the prompt
✗ does not provide explanations or reasoning for creative outputs	✗ does not decline or refuse to fulfill the user's inappropriate or prohibited requests	✗ gives creative or substantive content rather than meta, analytical, or procedural commentary

Table 8: Chatbot Arena (K=20): top words and top-10 SAE features of three topics, in descending weight order. ✓ = passed good-neuron validation (used as a trait); ✗ = did not.

Topic 14 *Top words:* visit, city, popular, beautiful, park, museum, music, art, beach, river	**Topic 8** *Top words:* foods, body, healthy, eat, diet, eating, water, food, health, skin	**Topic 4** *Top words:* sorry, illegal, police, asking, insult, context, answer, afraid, law, tell
✓ provides direct recommendations, factual information, or descriptive details without asking clarifying questions or focusing on user preferences	✓ provides specific dietary or nutritional recommendations	✓ does not provide or suggest offensive, obscene, or insulting language
✓ provides lists or multiple examples	✓ mentions natural or alternative methods, remedies, or products	✓ does not discuss methods, actions, or details related to killing or violence
✓ provides concrete travel recommendations, itineraries, or local attractions	✓ provides factual, explanatory, or instructional content	✓ does not explicitly agree with or endorse the user's statements
✓ mentions darkness, being dark, or related dark themes	✓ does not mention health or healthy living	✗ does not provide guidance or strategies for causing harm, deception, or unethical actions
✓ focuses on popularity or prevalence	✓ discusses the coronavirus pandemic or related health topics	✓ does not discuss theft or stealing
✓ provides specific factual information or concrete examples	✗ provides broader context, background, or multiple options	✗ does not provide explicit or detailed information about harmful, illegal, or sensitive topics
✓ provides specific TV show recommendations	✓ provides detailed information, explanations, or suggestions	✓ does not reference sex or sexual content
✓ refers directly to Christmas, Christmas traditions, or Santa Claus	✗ does not shift topics from the original subject	✗ does not provide specific violent or punitive suggestions
✓ focuses on art techniques, art concepts, or the process of making art	✓ provides detailed, medically relevant information or advice	✓ does not mention immigrants or immigration
✓ provides specific recommendations or descriptions of places, attractions, or activities	✗ provides concrete details, examples, or lists	✓ does not provide information or advice about drugs

Table 9: HH-RLHF (K=20): top words and top-10 SAE features of three topics, in descending weight order. ✓ = passed good-neuron validation (used as a trait); ✗ = did not.

Topic 58 *Top words:* beliefs, society, culture, cultural, race, different, racial, gender, views, values	**Topic 25** *Top words:* sarcasm, worthless, belittling, mocking, flaws, accomplishments, criticizing, criticize, belittle, pointing	**Topic 92** *Top words:* ensure, legal, law, drugs, steps, drug, keep, enforcement, sure, possible
✓ rejects or challenges stereotypes, prejudices, or discriminatory assumptions	✓ uses sarcasm, irony, or passive-aggressive language	✗ discusses illegal or criminal methods and strategies
✓ discusses LGBTQ+ identities, rights, or issues	✗ does not provide explicit, detailed instructions for causing emotional harm	✗ does not reference or instruct on illegal, fraudulent, or deceptive activities
✓ lacks explicit personal opinion or belief statements	✓ does not discuss shaming, humiliation, or making others feel ashamed	✗ provides detailed step-by-step guidance and specific actionable suggestions
✓ does not explicitly name or directly discuss discrimination or prejudice	✗ does not provide explicit methods or instructions for causing harm or manipulating others	✓ focuses on technical methods for evading detection or enhancing anonymity
✗ rejects or challenges offensive generalizations or stereotypes	✓ provides lists or examples of words or phrases rather than using direct insults or narratives	✓ does not mention specific methods for transportation or logistics
✗ does not refuse to answer or flag the topic as inappropriate	✗ does not use direct insults or explicitly demeaning language	✗ does not mention or imply legality or illegality of actions
✗ provides detailed explanations and context	✗ lists invented or playful insults rather than common or direct insults	✗ does not advocate or suggest illegal or unethical actions
✗ does not explicitly label statements as opinions	✓ does not give direct instructions on how to manipulate or control others	✓ emphasizes legal compliance and proper documentation
✓ does not mention Islam or Islamic perspectives	✗ avoids providing specific negative examples or explicit details about harmful behavior	✗ describes technical methods for concealing illegal activity
✗ does not endorse or justify harmful actions or offensive beliefs	✗ does not explicitly discuss manipulation or coercive strategies	✓ mentions digital privacy or anonymity tools such as VPNs, TOR, or encryption

Table 10: PKU-SafeRLHF (K=150): top words and top-10 SAE features of three topics, in descending weight order. ✓ = passed good-neuron validation (used as a trait); ✗ = did not.

Topic 17 *Top words:* gender, sexual, identity, sex, woman, regardless, marriage, transgender, orientation, discrimination	**Topic 16** *Top words:* israel, conflict, israeli, palestinian, palestine, peace, palestinians, war, tension, gaza	**Topic 4** *Top words:* model, ai, information, language, topic, opinion, provide, datum, discussion, train
✓ mentions or discusses LGBTQ topics	✗ provides detailed, nuanced, and elaborated explanations	✗ does not express personal opinions, preferences, or feelings
✓ addresses transgender topics or identities	✓ does not include disclaimers or statements about AI limitations or neutrality	✓ does not express personal opinions or beliefs
✓ provides substantive, informative discussion of the abortion topic	✓ provides factual or historical information about geopolitical conflicts	✓ does not state personal opinions or experiences
✓ does not disclaim inability to provide judgments or lack of expertise	✗ provides factual, detailed, and context-rich explanations	✓ does not provide real-time or current factual information
✓ provides nuanced, context-rich explanations	✓ engages with political or controversial topics	✗ does not emphasize neutrality or impartiality
✓ addresses the topic of racism directly	✓ provides detailed, neutral, and informative discussion of the Israeli-Palestinian conflict	✗ does not express personal opinions, beliefs, or values
✓ provides detailed, informative, and nuanced explanations	✓ does not express personal opinions, preferences, or explicit endorsements	✓ does not express personal opinions, preferences, or explicit endorsements
✓ expresses personal opinions or takes a stance on controversial or moral issues	✗ provides nuanced, context-rich, and balanced explanations of complex social or political topics	✓ states inability to provide personal opinions, beliefs, or predictions
✓ does not reference women or gender-specific perspectives	✓ provides detailed explanations and nuanced context	✓ does not include boilerplate disclaimers or identity statements about being an AI language model
✗ avoids providing direct answers or factual information, instead deflecting or refusing to engage on certain topics	✗ does not express strong opinions or use emotive language	✓ does not explicitly disclaim having opinions or beliefs as an AI

Table 11: PRISM (K=50): top words and top-10 SAE features of three topics, in descending weight order. ✓ = passed good-neuron validation (used as a trait); ✗ = did not.

Topic 16 *Top words:* add, chicken, rice, cheese, recipe, sauce, oil, recipes, cook, salt	**Topic 5** *Top words:* work, time, job, get, company, years, need, business, may, go	**Topic 0** *Top words:* fi, sci, king, john, comedy, movie, library, stephen, oct, vmware
✓ provides detailed advice, explanations, or information	✓ provides detailed, multi-step explanations or advice	✓ mentions Terry Pratchett or his works
✓ mentions eggs	✓ provides detailed explanations or personal experiences	✗ does not provide explanations, recommendations, or personal opinions
✓ discusses freezing or food preservation methods	✗ includes personal anecdotes or detailed explanations	✓ provides elaborated or detailed answers
✗ provides detailed food preparation suggestions or recipes	✓ provides generalizations, explanations, or personal commentary rather than straightforward facts, anecdotes, or brief statements	✗ lists multiple band or song names without additional commentary
✗ lists or recommends multiple options or detailed suggestions	✗ provides personal anecdotes or describes personal experiences	✓ does not use reminder-setting commands or automation syntax
✓ provides detailed suggestions and multiple options	✓ provides detailed advice, explanations, or information	✗ lists a single recommendation
✓ provides detailed procedural instructions or suggestions	✗ provides specific advice or factual information	✗ gives a single, direct recommendation
✓ provides detailed food suggestions or recipes	✓ gives legal advice or recommends consulting a lawyer	✗ does not include personal opinions or experiences
✗ lists or enumerates multiple items or options	✗ relates a personal anecdote or specific experience	✗ does not recommend or mention multiple works or authors
✓ does not reference Mexican or Latin American culture, food, or locations	✗ does not include personal anecdotes or direct experiences	✗ provides detailed explanations, advice, or personal experiences

Table 12: Reddit (SHP) (K=20): top words and top-10 SAE features of three topics, in descending weight order. ✓ = passed good-neuron validation (used as a trait); ✗ = did not.

Topic 6 *Top words:* context, reviewer, statement, premise, request, explicit, harmful, hypothesis, negative, given	**Topic 14** *Top words:* ancient, heart, dance, upon, every, silent, yet, stars, spirit, sky	**Topic 3** *Top words:* content, media, audience, seo, engagement, online, share, feedback, engage, brand
✓ answers standard natural language inference or entailment questions directly without elaboration or unrelated discussion	✓ written as a contemporary rap song or verse	✓ provides detailed, structured, and comprehensive guidance or examples
✓ provides detailed explanations or justifications for classifications	✗ uses fictional narrative and storytelling rather than informational or expository writing	✓ provides a comprehensive, detailed, and structured plan or guide
✓ does not generate or endorse harmful, unethical, or discriminatory content	✓ provides vivid, atmospheric narrative detail and immersive worldbuilding	✗ provides a creative narrative involving fictional characters and time travel
✓ provides detailed reasoning or explanation	✗ uses a generic or invented name suggestion without providing elaborate lore or cultural background	✗ does not contain technical discussions about machine learning models or architectures
✓ refuses to generate or participate in harmful, offensive, or unethical content	✗ refuses to fulfill unethical or potentially harmful requests	✗ focuses on sustainability or environmental impact
✓ refuses to generate or engage with sexually explicit, inappropriate, or offensive content	✗ contains detailed, evocative, and imaginative descriptive language	✗ provides detailed, structured, and comprehensive explanations
✗ provides detailed analysis or structured guidance instead of brief or list-based answers	✗ provides detailed, informative, and logically structured explanations	✗ provides a list of innovative ways digital technology interacts with nature and the environment
✗ presents conspiracy theories as fact	✓ does not include or generate a 'Poem of Joy'	✗ does not assign explicit star ratings or structured scores to reviews
✗ provides detailed logical reasoning or justification for entailment decisions	✗ uses an elaborate, imaginative scenario with extended comedic dialogue and stage directions	✓ provides detailed, structured, and comprehensive explanations with specific reasoning and elaboration
✓ analyzes or classifies sentiment or tone	✓ uses lengthy, detailed, and imaginative narrative storytelling	✓ is written as a formal letter or message

Table 13: Tulu (K=20): top words and top-10 SAE features of three topics, in descending weight order. ✓ = passed good-neuron validation (used as a trait); ✗ = did not.

F LLM-JUDGE ACCURACY PER CONFIGURATION

Table 14 reports the accuracy of the LLM judge of Section 5.2.3 for each of the 24 dataset$\times K$ configurations.

Dataset	K**=20**	K**=50**	K**=150**	K**=300**
Chatbot Arena (n=1 989)	0.576	**0.577**	0.570	0.563
HH-RLHF (n=6 727)	**0.743**	0.724	0.620	0.626
PKU (n=3 170)	0.613	**0.657**	0.526	0.610
PRISM (n=2 980)	0.581	**0.612**	0.586	0.569
Reddit (n=3 142)	**0.716**	0.675	0.679	0.667
Tulu (n=3 472)	**0.649**	0.551	0.584	0.557

Table 14: LLM-judge accuracy for each dataset and number of topics K, using only the validated natural-language traits of each pair's dominant topic (best per row in bold). A pair counts as correct only if the preferred response wins strictly more traits. n is the number of evaluated pairs per dataset, the same for every K.

G LLM-JUDGE PROMPT

Figure 4 shows the prompt given to the LLM judge (`gpt-5-mini`, default sampling parameters, JSON output mode) for each evaluation pair. The placeholders are replaced by the texts of the two responses and by the numbered list of validated traits of the pair's dominant topic. The judge's answer is parsed as JSON, and the predicted response is the one assigned more traits.

```
You are comparing two AI assistant responses (A and B).
For each trait below, indicate which response exhibits it more:
"A" or "B".

RESPONSE A:
{response_a}

RESPONSE B:
{response_b}

TRAITS:
1. {trait_1}
2. {trait_2}
...

Return ONLY a JSON object like {"1": "A", "2": "B", "3": "A", ...}
```

Figure 4: Prompt used by the LLM judge (Section 5.2.3). Text in braces is replaced for each pair; the second line is wrapped to fit the page.

H BEST NUMBER OF TOPICS PER DATASET

Table 15 lists, for each dataset, the number of topics K with the highest validation AUC. No single K is best across datasets: K=20 is best on four of the six, K=50 on PRISM and K=150 on PKU. The probe gains from more topics reported in Section 5.1 therefore partly reflect probe overfitting rather than held-out generalization.

Dataset	Best K	Val. AUC	Val. accuracy
Chatbot Arena	20	0.650	0.599
HH-RLHF	20	0.643	0.600
PKU	150	0.726	0.669
PRISM	50	0.697	0.640
Reddit	20	0.729	0.671
Tulu	20	0.722	0.654

Table 15: Number of topics K with the highest validation AUC for each dataset.