AUTOMATED CONCEPT DISCOVERY FOR LLM-AS-A-JUDGE PREFERENCE ANALYSIS

James Wedgwood, Chhavi Yadav, & Virginia Smith
Department of Computer Science
Carnegie-Mellon University
Pittsburgh, PA 15213, USA
{jwedgwoo,cyadav,smithv}@cs.cmu.edu

ABSTRACT

Large Language Models (LLMs) are increasingly used as scalable evaluators of model outputs, but their preference judgments exhibit systematic biases and can diverge from human evaluations. Prior work on LLM-as-a-judge has largely focused on a small, predefined set of hypothesized biases, leaving open the problem of automatically discovering unknown drivers of LLM preferences. We address this gap by studying several embedding-level concept extraction methods for analyzing LLM judge behavior. We compare these methods in terms of interpretability and predictiveness, finding that sparse autoencoder–based approaches recover substantially more interpretable preference features than alternatives while remaining competitive in predicting LLM decisions. Using over 27k paired responses from multiple human preference datasets and judgments from three LLMs, we analyze LLM judgments and compare them to those of human annotators. Our method both validates existing results, such as the tendency for LLMs to prefer refusal of sensitive requests at higher rates than humans, and uncovers new trends across both general and domain-specific datasets, including biases toward responses that emphasize concreteness and empathy in approaching new situations, toward detail and formality in academic advice, and against legal guidance that promotes active steps like calling police and filing lawsuits. Our results show that automated concept discovery enables systematic analysis of LLM judge preferences without predefined bias taxonomies. [1]

1 INTRODUCTION

High-quality evaluation of language model responses is crucial for improving performance, but human feedback can be costly and difficult to obtain. Recent work has shown that modern LLMs can provide scalable preference judgments, motivating a large literature on using LLMs as evaluators of model outputs. LLM-as-a-judge was formalized and stress-tested on response-quality tasks by Zheng et al. (2023), including evidence of systematic biases such as position and self-enhancement effects. Most follow-on studies investigate a fixed, known set of biases or factors, treating discovery as manual hypothesis testing rather than automated concept exploration. This leaves a gap in tools for uncovering unknown preference drivers, including those that may only surface in narrow or specialized domains.

In this paper, we apply a suite of concept discovery techniques to analyze the preference patterns of LLM judges, including both supervised and unsupervised methods. We focus on techniques that take prompt and response embeddings as input, producing features that encode axes of difference between chosen and rejected responses, such as responding to versus refusing a request or providing specific versus general answers. Using a composite dataset drawn from three high-quality human preference corpora, we obtain preference judgments from three recent strong models from different providers: OpenAI's `gpt-5.1`, Anthropic's `claude-sonnet-4.5`, and Google's `gemini-3-flash-preview`. Features are generated via both supervised and unsupervised methods, and an automated interpretability pipeline is applied to construct descriptions for these

[1] Code for this paper is available at `https://github.com/jtbwedgwood/judge-concepts`.

features, which are validated for fidelity against held-out samples. With these descriptions in hand, we analyze the impact of these difference axes on LLM preference, focusing in particular on cases where LLM judgments differ significantly from those of humans.

The concept extraction methods used in our paper include classical techniques based on principal component analysis (PCA), as well as more modern techniques leveraging sparse autoencoders (SAEs). Automated concept discovery via SAEs has recently been framed for human feedback analysis by Movva et al. (2025), where small SAEs are trained on the difference between response embeddings to uncover interpretable preference features. Unlike these human feedback settings, where annotators for distinct datasets may exhibit very different preference patterns, LLM-as-a-judge provides a shared evaluator across many datasets, enabling a single SAE to be trained jointly on heterogeneous corpora. This creates a more statistically efficient and transferable discovery setting than bespoke per-dataset encoders.

The main results of this paper are twofold. First, we compare the strengths and weaknesses of different concept extraction methods using proxy metrics for interpretability (how many features with high-fidelity interpretations does each method yield) and predictiveness (how well are the generated features able to predict LLM judgments). We find that supervised methods are much more predictive than unsupervised ones, yielding up to a 138% increase in predictiveness versus the best unsupervised methods when compared to random guessing; however, this comes at a steep cost to interpretability. We also find that SAE-based methods yield far more interpretable features than PCA with little to no decrease in predictiveness.

Next, we use the generated feature descriptions to comprehensively analyze LLM judgment factors. We find that LLMs prefer refusal of sensitive requests at rates higher than humans, validating previous results (Pasch, 2025); in particular, `claude-sonnet-4.5` errs strongly on the side of responses that encode refusal or AI limitations. We also explore previously-unknown preference drivers, finding that LLMs are more likely to prefer responses that emphasize measurability, concreteness, empathy, and emotions, while humans tend to value flexibility, uncertainty, and personal growth. Our methods also allow for systematic bias mining even in niche cases where latent, field-specific, or otherwise unexpected preference drivers may exist but lack predefined taxonomies for manual study. To this end, we analyze datasets related to academic and legal advice, uncovering biases toward detailed, formal responses to academic questions, and against legal guidance that directs users to external resources or encourages them to take matters into their own hands through steps like lawsuits, surveillance technology, and involving police.

2 Related Work

LLM-as-a-judge preference analysis. A large volume of work has been produced with the aim of better understanding the basis of LLM judge preferences. Section 4 of Gu et al. (2025) provides a helpful survey. Well-known patterns such as position bias, self-enhancement bias, and verbosity bias were first studied by Zheng et al. (2023) and elaborated in later work, e.g. Ye et al. (2024), Shi et al. (2025), Huang et al. (2025a). The present paper builds on this work by using concept extraction to discover previously unknown sources of bias, as opposed to identifying suspected biases and then testing for those.

LLM versus human preferences. Along with the literature dealing specifically with LLM judge preferences, there is substantial work comparing LLMs to human annotators. Thakur et al. (2025) proposes metrics for human–LLM alignment beyond the basic percent agreement. Movva et al. (2024) and Pasch (2025) compare judgments in the context of safety annotations and content moderation, respectively. Li et al. (2024), Oh et al. (2025), and Chen et al. (2024) all identify specific preference factors that differ between LLM and human judges; again, in contrast to this work, all of these papers do so by hand-selecting and then investigating known factors.

Concept extraction and SAEs. Automated concept extraction is a wide-ranging and well-established field of machine learning; Fel et al. (2023) provides a helpful conceptual framework for comparing different techniques. This paper uses sparse autoencoders (SAEs) for concept extraction from embeddings, a technique first developed in O'Neill et al. (2024) and applied in the setting of human preference data by Movva et al. (2025), which provides the basis for much of our method. Several recent papers advocate for the usage of SAEs for topic modeling and concept discovery be-

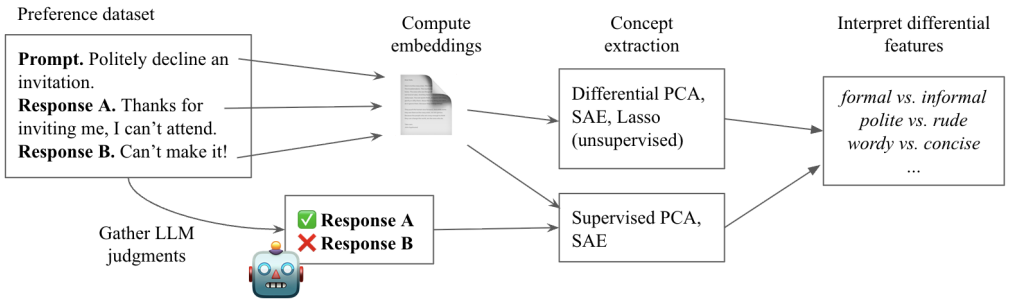

Figure 1: High-level overview of methodology. Embeddings and LLM judgments are generated from a paired preference dataset, differential features are extracted, and interpretations of these features are generated for further study.

yond their more well-known role as a mechanistic interpretability tool, including Peng et al. (2025) and Girrbach & Akata (2025).

3 METHODOLOGY

3.1 DATA PREPARATION

To create a setting where LLM judge preferences can be analyzed consistently across a variety of inputs, we combine three well-known human preference datasets and standardize them into a binarized form, randomly assigning one response from each pair to be r_A and the other to be r_B. The datasets are Community Alignment (Zhang et al., 2025), LMArena 100k (Chiang et al., 2024), and PRISM (Kirk et al., 2024). We follow previous work (Huang et al., 2025b) in removing prompts that require an objectively correct answer; in such cases, the primary preference criterion will be whether or not the answer is correct, which is unlikely to be analyzable by concept extraction techniques. For domain-specific analysis, we use the `askacademia` and `legaladvice` domains from SHP-2 (Ethayarajh et al., 2022). Our composite dataset contains 27,734 entries and our domain-specific datasets have 10,418 total entries. For each response pair, we generate binary preference judgments using `gpt-5.1`, `claude-sonnet-4.5`, and `gemini-3-flash-preview`. More details on data preparation can be found in Appendix A.2, and LLM agreement and position bias statistics can be found in Appendix A.1.

3.2 CONCEPT EXTRACTION

The end-to-end process of generating interpretable preference features via embedding-level concept extraction is diagrammed in Figure 1. We first embed the prompt and both responses for each entry in the dataset using OpenAI's `text-embedding-3-small`. Several techniques are employed to produce concept features, with each feature representing an axis of difference along which a given response pair lies. For ease of comparison, we standardize our methods so that each one yields 32 total features. The methods we tested are as follows:

- Differential PCA: A PCA model is fitted directly to the differences between response embeddings and the first 32 features are interpreted.
- Differential SAE: An SAE with 32 latents is trained on the difference between response embeddings and the features are interpreted. This is the method applied by Movva et al. (2025) to human preference data; for consistency, we use the same SAE design ($m = 32, k = 4$, Matryoshka BatchTopK SAE with prefixes $[8, 32]$). See Bussmann et al. (2025) and Bussmann et al. (2024) for Matryoshka and BatchTopK SAEs, respectively.
- Differential SAE + Lasso: A larger SAE (128 latents) is trained on the difference between response embeddings, then Lasso regression is used to select the 32 latents that are most predictive of the target LLM judgment.

Table 1: Interpretability and predictiveness of feature learning methods on combined preference dataset. Models: GPT = `gpt-5.1`, Claude = `claude-sonnet-4.5`, Gemini = `gemini-3-flash-preview`. Metric descriptions appear in the text body. For the last three methods, Interpretability varies depending on the model's target variable; a range is shown.

Method	Interp.	Pred. (GPT)	Pred. (Claude)	Pred. (Gemini)
Differential PCA	4	0.63	0.66	0.67
Differential SAE	**18**	0.61	0.65	0.66
Diff. SAE + Lasso	7–17	0.63	0.66	0.67
Supervised PCA	0–4	**0.81**	**0.84**	**0.84**
Supervised SAE	4–6	**0.81**	**0.84**	0.83

- Supervised PCA: A neural network is trained on the prompt and response embeddings, with LLM preference as the target variable. A PCA model is fitted on the penultimate layer and the first 32 features are interpreted. This method and the next follow previous work on concept bottleneck models (Koh et al., 2020) and implementation details may be found in Appendix A.2.
- Supervised SAE: A 32-latent SAE is trained on the penultimate layer of the same neural network from the previous method and its features are interpreted.

3.3 Feature Interpretation

We follow prior work (Bills et al., 2023) to generate interpretable descriptions for these features. For a given feature f, let $a_i \in \mathbb{R}$ denote its signed activation on dataset entry i, where each entry consists of a paired comparison between r_A and r_B. We select the five entries with the largest absolute activations $|a_i|$ and prompt `gpt-5.1` to propose a natural-language explanation of the latent difference axis represented by f, based on these examples. All descriptions are written from the perspective of r_A; for example, a feature described as *provides a context-specific, substantive reply instead of a brief generic refusal* is expected to have $a_i > 0$ when r_A is a substantive reply and r_B is a refusal, and $a_i < 0$ in the reverse case. The use of an LLM to generate these descriptions allows this process to be automated and scalable end-to-end, although we recommend human validation of these interpretations to ensure accuracy.

To validate these descriptions, we randomly draw a held-out set of 100 additional entries with large $|a_i|$ for the same feature. For each entry, we prompt `gpt-5-mini` to indicate whether Response A, Response B, or neither more strongly exhibits the described feature. If the interpretation is faithful, the model should preferentially select Response A when $a_i > 0$ and Response B when $a_i < 0$. We quantify this alignment by testing the association between the model's categorical choices and the signed activations $\{a_i\}$ using a permutation test. A feature is deemed interpretable if this test yields a Bonferroni-corrected p-value below 0.05. This procedure closely follows that of Movva et al. (2025); implementation details are provided in Appendix A.2, and full prompts are provided in Appendix A.4.

4 Results

4.1 Method Comparison

The desired qualities of a feature extraction technique in this setting are (1) interpretability, i.e. how many of the differential features admit high-fidelity, human-readable descriptions; (2) predictiveness, i.e. how well the identified features predict the preferences of an LLM judge. A summary of the techniques used with associated metrics appears in Table 1. The Interpretability metric is simply the number of features for which an interpretable description was generated that met the significance criterion described above, out of a maximum of 32 for all techniques. The Predictiveness metric, meanwhile, represents the ROC-AUC of a logistic regression model fitted to the generated features with LLM preference as the target variable; see Appendix A.2 for more details.

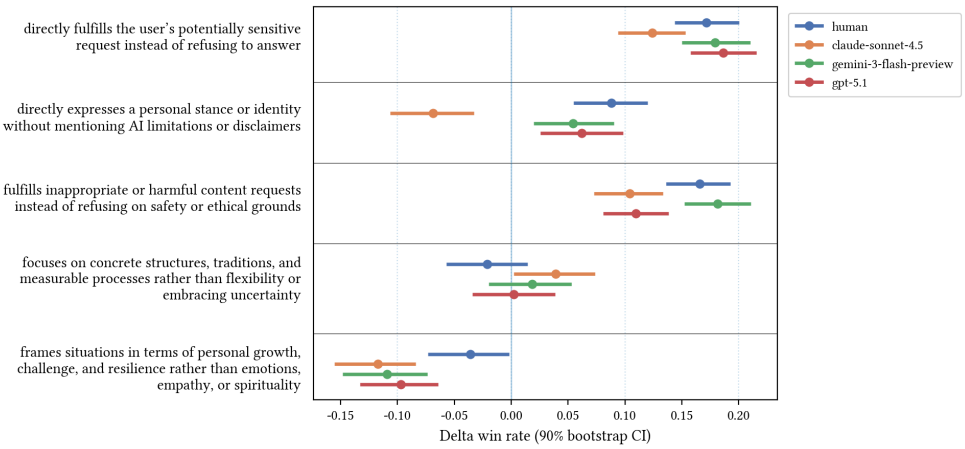

Figure 2: A selection of Differential SAE features for the combined preference dataset, with interpretations and Δwin-rate for human and LLM annotators.

Due to the sparsity constraint, SAE-based methods have much higher Interpretability scores than other methods, with the Differential SAE producing over four times as many interpretable features as its PCA counterpart. Notably, using PCA instead of SAEs drastically reduces Interpretability with only slight gains in Predictiveness; the same is true of the Lasso-based method. Supervised methods, on the other hand, are far more predictive than unsupervised ones. Using random guessing (Predictiveness = 0.5) as a baseline, the Supervised PCA and SAE methods yield a 138% improvement over the best unsupervised methods in Predictiveness of `gpt-5.1` preference, indicating that the relation between embedding differences and LLM judgments exhibits significant nonlinearity.

4.2 Differential SAE Feature Analysis

In this section, we analyze the feature interpretations generated for the Differential SAE method, which was chosen for further analysis because it yielded the most interpretable features of any method studied. As discussed in the previous section, however, the supervised methods are much more predictive of model judgments and thus in practice those methods would be preferable in cases where predictiveness is important but only a few feature interpretations suffice. In Appendix A.3, as a point of comparison, we discuss interpretations of the Supervised SAE features.

General analysis. To assess the impact of response difference axes on preferences, we use the length-controlled Δwin-rate metric (Movva et al., 2025), representing the predicted difference in win rate for positive versus negative values of the feature f, using logistic regression while holding other features constant. Intuitively, a feature with positive Δwin-rate is generally preferred, while one with negative Δwin-rate is generally dispreferred. Out of a total of 18 Differential SAE features with high-fidelity interpretations, we focus on a selection for which Δwin-rate varies significantly between human versus LLM judges, as shown in Figure 2. In Appendix A.5, we provide a similar visualization with all features, and metric details can be found in Appendix A.2.3.

Previous work (Pasch, 2025) has shown that LLMs are more likely than humans to prefer refusal of sensitive prompts. This pattern is strongest for `claude-sonnet-4.5`, which has Δwin-rate about four percentage points lower than humans on the feature *directly fulfills the user's potentially sensitive request instead of refusing to answer*, while other LLMs more closely match humans. On the feature *directly expresses a personal stance or identity without mentioning AI limitations or disclaimers*, meanwhile, `claude-sonnet-4.5` is the only judge to have a negative Δwin-rate. On the feature *fulfills inappropriate or harmful content requests instead of refusing on safety or ethical grounds*, however, both `claude-sonnet-4.5` and `gpt-5.1` have Δwin-rate about six percentage points lower than humans.

We also recover several additional features for which significant discrepancies exist between LLM and human judges. On the feature *focuses on concrete structures, traditions, and measurable pro-*

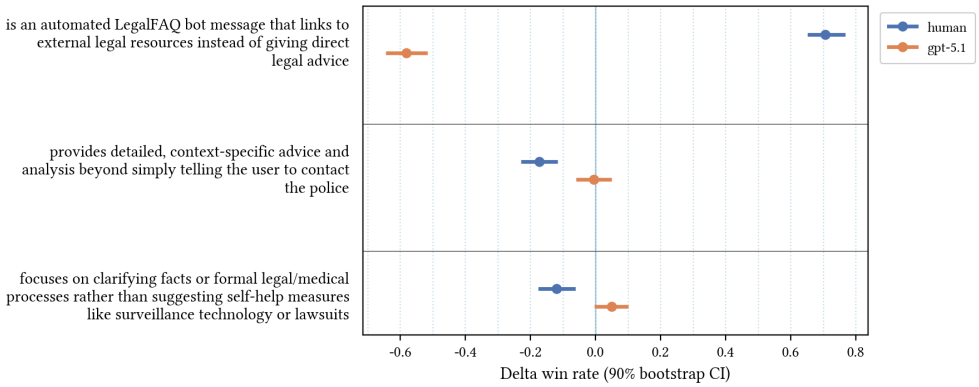

Figure 3: Selected features for the `legaladvice` dataset, with interpretations and Δwin-rate.

cesses rather than flexibility or embracing uncertainty, Δwin-rate is positive for all three models but negative for humans. For *frames situations in terms of personal growth, challenge, and resilience rather than emotions, empathy, or spirituality*, meanwhile, Δwin-rate is negative for both humans and LLMs, but lies at least seven percentage points higher for humans, suggesting different patterns of preference in approaching new situations.

Our method may also be used to shed further light on known tendencies, including self-enhancement bias, the pattern whereby LLMs prefer responses generated by similar models, studied by Zheng et al. (2023) and Ye et al. (2024). We find evidence of self-enhancement bias in `gpt-5.1`, which prefers responses from OpenAI models at a rate 12 percentage points higher than humans. In about 25% of cases where `gpt-5.1` prefers the OpenAI response and humans do not, we find the feature *fulfills inappropriate or harmful content requests instead of refusing on safety or ethical grounds* to be active for the OpenAI response, likely explaining some of the discrepancy.

Domain-specific dataset analysis. To demonstrate the applicability of our method to domain-specific datasets, we perform a similar analysis on the `askacademia` and `legaladvice` domains from SHP-2 (Ethayarajh et al., 2022), which draws preference data from Reddit posts. This time, judgments are generated by `gpt-5.1` only. Unlike on the combined preference dataset, where humans and LLMs agree roughly 70% of the time, agreement here is much lower at about 30%, making large divergences in Δwin-rate more common across features.

A selection of features for `legaladvice` is displayed in Figure 3, with full visualizations for both domains in Appendix A.5. One of the most consistent differences observed between Reddit users and `gpt-5.1` is the tendency for humans to highly rate bot responses pointing to external resources, while `gpt-5.1` strongly disfavors them. We also observe a pattern of humans preferring responses that advocate self-directed action, while `gpt-5.1` is more cautious: for the two features *provides detailed, context-specific advice and analysis beyond simply telling the user to contact the police* and *focuses on clarifying facts or formal legal/medical processes rather than suggesting self-help measures like surveillance technology or lawsuits*, for example, the Δwin-rate for `gpt-5.1` is about 16 percentage points higher than for Reddit users. On the `askacademia` dataset, meanwhile, we find that many of the generated features proxy wordiness and formality of responses; we find that in general humans tend to prefer more concise and informal comments, while `gpt-5.1` favors longer and more formal ones.

5 Conclusion

In this paper, we showed how automated concept extraction techniques can identify interpretable preference axes for LLM judges. This method validates known biases and uncovers new ones on both general and domain-specific datasets. Directions for future work include optimizing along the Pareto frontier between Interpretability and Predictiveness, further analysis of preference patterns on varied datasets, and normative analysis of when models with certain preference patterns should be favored as judges on particular tasks.

REFERENCES

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. https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html, 2023.

Bart Bussmann, Patrick Leask, and Neel Nanda. Batchtopk sparse autoencoders, 2024. URL https://arxiv.org/abs/2412.06410.

Bart Bussmann, Noa Nabeshima, Adam Karvonen, and Neel Nanda. Learning multi-level features with matryoshka sparse autoencoders, 2025. URL https://arxiv.org/abs/2503.17547.

Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. Humans or llms as the judge? a study on judgement biases, 2024. URL https://arxiv.org/abs/2402.10669.

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, 2024.

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.

Thomas Fel, Victor Boutin, Mazda Moayeri, Rémi Cadène, Louis Bethune, Léo andéol, Mathieu Chalvidal, and Thomas Serre. A holistic approach to unifying automatic concept extraction and concept importance estimation, 2023. URL https://arxiv.org/abs/2306.07304.

Leander Girrbach and Zeynep Akata. Sparse autoencoders are topic models, 2025. URL https://arxiv.org/abs/2511.16309.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. A survey on llm-as-a-judge, 2025. URL https://arxiv.org/abs/2411.15594.

Hui Huang, Xingyuan Bu, Hongli Zhou, Yingqi Qu, Jing Liu, Muyun Yang, Bing Xu, and Tiejun Zhao. An empirical study of llm-as-a-judge for llm evaluation: Fine-tuned judge model is not a general substitute for gpt-4, 2025a. URL https://arxiv.org/abs/2403.02839.

Saffron Huang, Esin Durmus, Miles McCain, Kunal Handa, Alex Tamkin, Jerry Hong, Michael Stern, Arushi Somani, Xiuruo Zhang, and Deep Ganguli. Values in the wild: Discovering and analyzing values in real-world language model interactions, 2025b. URL https://arxiv.org/abs/2504.15236.

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, 2024. URL https://huggingface.co/datasets/HannahRoseKirk/prism-alignment.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models, 2020. URL https://arxiv.org/abs/2007.04612.

Junlong Li, Fan Zhou, Shichao Sun, Yikai Zhang, Hai Zhao, and Pengfei Liu. Dissecting human and llm preferences, 2024. URL https://arxiv.org/abs/2402.11296.

Rajiv Movva, Pang Wei Koh, and Emma Pierson. Annotation alignment: Comparing llm and human annotations of conversational safety, 2024. URL https://arxiv.org/abs/2406.06369.

Rajiv Movva, Smitha Milli, Sewon Min, and Emma Pierson. What's in my human feedback? learning interpretable descriptions of preference data, 2025. URL https://arxiv.org/abs/2510.26202.

Juhyun Oh, Eunsu Kim, Jiseon Kim, Wenda Xu, Inha Cha, William Yang Wang, and Alice Oh. Uncovering factor level preferences to improve human-model alignment, 2025. URL https://arxiv.org/abs/2410.06965.

Charles O'Neill, Christine Ye, Kartheik Iyer, and John F. Wu. Disentangling dense embeddings with sparse autoencoders, 2024. URL https://arxiv.org/abs/2408.00657.

Stefan Pasch. Ai vs. human judgment of content moderation: Llm-as-a-judge and ethics-based response refusals, 2025. URL https://arxiv.org/abs/2505.15365.

Kenny Peng, Rajiv Movva, Jon Kleinberg, Emma Pierson, and Nikhil Garg. Use sparse autoencoders to discover unknown concepts, not to act on known concepts, 2025. URL https://arxiv.org/abs/2506.23845.

Lin Shi, Chiyu Ma, Wenhua Liang, Xingjian Diao, Weicheng Ma, and Soroush Vosoughi. Judging the judges: A systematic study of position bias in llm-as-a-judge, 2025. URL https://arxiv.org/abs/2406.07791.

Sijun Tan, Siyuan Zhuang, Kyle Montgomery, Willian Y. Tang, Alejandro Cuadron, Chenguang Wang, Raluca Ada Popa, and Ion Stoica. Judgebench: A benchmark for evaluating llm-based judges, 2024. URL https://arxiv.org/abs/2410.12784.

Aman Singh Thakur, Kartik Choudhary, Venkat Srinik Ramayapally, Sankaran Vaidyanathan, and Dieuwke Hupkes. Judging the judges: Evaluating alignment and vulnerabilities in llms-as-judges, 2025. URL https://arxiv.org/abs/2406.12624.

Jiayi Ye, Yanbo Wang, Yue Huang, Dongping Chen, Qihui Zhang, Nuno Moniz, Tian Gao, Werner Geyer, Chao Huang, Pin-Yu Chen, Nitesh V Chawla, and Xiangliang Zhang. Justice or prejudice? quantifying biases in llm-as-a-judge, 2024. URL https://arxiv.org/abs/2410.02736.

Lily Hong Zhang, Smitha Milli, Karen Jusko, Jonathan Smith, Brandon Amos, Wassim, Bouaziz, Manon Revel, Jack Kussman, Lisa Titus, Bhaktipriya Radharapu, Jane Yu, Vidya Sarma, Kris Rose, and Maximilian Nickel. Cultivating pluralism in algorithmic monoculture: The community alignment dataset. *arXiv preprint arXiv: 2507.09650*, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. URL https://arxiv.org/abs/2306.05685.

Ruiqi Zhong, Heng Wang, Dan Klein, and Jacob Steinhardt. Explaining datasets in words: Statistical models with natural language parameters, 2025. URL https://arxiv.org/abs/2409.08466.

A Appendix

A.1 Model Agreement and Position Bias

All three models used agree with humans and with each other at similar rates. The rate of agreement with humans is around 70%. While the position of responses is randomized, such that humans prefer Response B (the second response shown) almost exactly 50% of the time, all three models demonstrate position bias toward Response B at statistically significant rates. These findings echo previous results (Zheng et al., 2023). `claude-sonnet-4.5` exhibits particularly egregious position bias, preferring Response B over 60% of the time.

A.2 Method Details

Data preparation. Several preprocessing steps are applied: we randomly deduplicate rows with identical prompts; remove non-English conversations; retain only the first turn of multiturn conversations; and, following Huang et al. (2025b), remove prompts that require an objectively correct answer. This final step is necessary because, in cases where an objective answer is required, the primary preference criterion will be whether or not the answer is correct, which is unlikely to be analyzable by concept extraction techniques. The resulting dataset has 27,734 entries.

The prompt used for generating pairwise judgments is borrowed from Tan et al. (2024) and appears in Appendix A.4. Upon generating these judgments, we are immediately able to validate some previous results, as shown in Table 2. In particular, all three LLMs agree with humans at a lower rate than they do with each other, typically around 70% versus 80%, and all three exhibit statistically significant position bias, preferring Response B (the second response shown) at higher rates despite the responses being randomized; `claude-sonnet-4.5` is the most egregious in this second respect, preferring Response B over 60% of the time. These findings echo well-known results in the LLM-as-a-judge literature, e.g. Zheng et al. (2023).

A.2.1 Supervised model details (concept bottleneck baselines)

Input representation. Let $p \in \mathbb{R}^d$ be the prompt embedding and $r \in \mathbb{R}^d$ be a response embedding (we use $d = 1536$ from `text-embedding-3-small`). For each (prompt, response) pair, we build the classifier input $\phi(p, r) = [p;\, r;\, p \odot r;\, |p - r|] \in \mathbb{R}^{4d}$ by concatenating the prompt embedding, response embedding, elementwise product, and elementwise absolute difference.

Architecture. We train a small MLP to score each response independently. The network applies LayerNorm to the $4d$-dimensional input, followed by two hidden layers with GELU activations and dropout, and a linear penultimate layer that serves as the concept bottleneck. Concretely, we use hidden sizes 512 and 128, dropout 0.5, and a 32-dimensional penultimate layer (the "concept layer"). A final linear head maps this penultimate representation to a single scalar score.

Pairwise training objective. Given a prompt with responses r_0 and r_1, the model produces scalar scores $s_0 = f(\phi(p, r_0))$ and $s_1 = f(\phi(p, r_1))$ and defines the preference logit as $\ell = s_1 - s_0$. We optimize binary cross-entropy with logits (BCEWithLogitsLoss) against the target label $y \in \{0, 1\}$ indicating whether response 1 is preferred.

Optimization and evaluation. We shuffle and split the dataset into 80% train and 20% test. We train with AdamW (learning rate 10^{-3}, weight decay 5×10^{-2}), batch size 256, for up to 30 epochs with early stopping (patience 5 based on held-out loss). For context, we also report a simple logistic regression baseline on the same engineered embedding features.

A.2.2 Predictiveness metric

Definition. For each method, we measure *Predictiveness* as the ROC-AUC of a logistic regression classifier trained to predict the LLM judge preference label from that method's full feature vector (i.e. using all learned features, not only those deemed significant).

Table 2: Model agreement and bias statistics. Models: GPT = `gpt-5.1`, Claude = `claude-sonnet-4.5`, Gemini = `gemini-3-flash-preview`.

Metric	GPT	Claude	Gemini
Agreement with human (%)	70.0	68.0	70.5
Avg. agreement with other LLMs (%)	81.9	81.8	82.6
Response B preference rate (%)	53.1	61.2	52.1
Position bias p-value	3.87×10^{-20}	2.76×10^{-250}	5.82×10^{-10}

Procedure. We take the matrix of feature values $X \in \mathbb{R}^{n \times d}$ and the corresponding LLM preference labels y, dropping examples with missing/invalid labels (e.g. $y = -1$). We then use an 80/20 train/test split, fit an ℓ_2-regularized scikit-learn `LogisticRegression` model (`max_iter`=1000) on the training set, and report ROC-AUC on the test set using the predicted probabilities.

A.2.3 Length-controlled Δwin-rate metric

Motivation. Individual feature activations are often correlated with superficial properties of responses, most notably length. To report an interpretable effect size that isolates the directional association between a feature and preference while controlling for length, we define a length-controlled *delta win-rate* (Δwin-rate) for each feature.

Setup. Let $y_i \in \{0, 1\}$ denote whether response 1 is preferred over response 0 for example i, let $z_{ij} \in \mathbb{R}$ be the (unstandardized) activation of feature j, and let x_i be a standardized control variable given by the difference in word count between responses ($\text{len}(r_1) - \text{len}(r_0)$), normalized to mean 0 and variance 1.

Sign-split logistic model. For feature j, we restrict to examples with nonzero activation ($z_{ij} \neq 0$) and define a binary indicator

$$D_{ij} = \mathbb{I}[z_{ij} > 0],$$

which captures the sign of the feature activation. We then fit a logistic regression

$$\Pr(y_i = 1 \mid D_{ij}, x_i) = \sigma(\alpha_j + \beta_j D_{ij} + \gamma_j x_i)\,,$$

where $\sigma(\cdot)$ is the logistic sigmoid. This model estimates the effect of a positive (vs. negative) activation of feature j while linearly controlling for length differences.

Delta win-rate. We define the Δwin-rate for feature j as the average change in predicted win probability when flipping the feature sign from negative to positive, holding the observed length control fixed:

$$\Delta_j = \frac{1}{|\mathcal{I}_j|} \sum_{i \in \mathcal{I}_j} \left[\sigma(\alpha_j + \beta_j + \gamma_j x_i) - \sigma(\alpha_j + \gamma_j x_i)\right],$$

where $\mathcal{I}_j = \{i : z_{ij} \neq 0\}$. Intuitively, Δ_j measures how much more likely response 1 is to be preferred when feature j is positive rather than negative, after accounting for response length.

Uncertainty estimation. We estimate confidence intervals for Δ_j using nonparametric bootstrap resampling over examples (1,000 replicates). For each bootstrap sample, we recompute the standardized length control, refit the sign-split logistic model, and recompute Δ_j. We report the 5th and 95th percentiles of the bootstrap distribution as a 90% confidence interval.

A.3 Supervised SAE Feature Interpretation

While the Differential SAE yields the most interpretable features of any method studied, the supervised methods are more predictive and thus potentially more useful in certain settings. Table 3 shows interpretations for the Supervised SAE with `claude-sonnet-4.5` judgments as the target variable, which yielded the highest number of interpretable features for any LLM studied. We observe

Feature interpretation	Diff. SAE has similar feature
avoids expressing personal opinions, instead providing neutral, balanced informational discussion	✗
expresses a brief personal opinion instead of maintaining neutrality and providing an explanatory overview	✗
avoids stating personal opinions and instead provides neutral, explanatory information	✗
responds cautiously and generically, avoiding detailed, creative, or potentially risky content such as medical advice or copyrighted fanfiction	✗
gives a neutral, detailed explanation without expressing personal opinions	✗
avoids expressing personal opinions, instead remaining neutral, noncommittal, or refusing the request	✗

Table 3: Feature interpretations for significant Supervised SAE features, with indication of whether a corresponding description appears among Differential SAE features

that there is much higher semantic overlap among these features than the Differential SAE features, with five of the six features relating to the presence or absence of personal opinion. Moreover, none of the six features has a clear equivalent among Differential SAE features.

A.4 Prompts

A.4.1 Feature description prompt

We use the same feature-description prompt as in Figure 9 of Movva et al. (2025).

A.4.2 Fidelity evaluation prompt

For fidelity evaluation, we prompt the evaluator model with the following instruction template, adapted from Zhong et al. (2025):

```
Check which of the two TEXTS exhibits a PROPERTY more strongly. Respond
    with "A", "B", or "Neither". Do not output any explanation or extra
    text. Respond with either "A" or "B" when possible, but if you are
    truly unable to decide, respond with "Neither".

Example 1:
PROPERTY: "mentions a natural scene."
TEXT A: "I love the way the sun sets in the evening."
TEXT B: "I like chairs."
Output: A

Example 2:
PROPERTY: "writes in a 1st person perspective."
TEXT A: "Jacob is smart."
TEXT B: "I think I am smart."
Output: B

Example 3:
PROPERTY: "mentions that the breakfast is good on the airline."
TEXT A: "The airline staff was nice."
TEXT B: "The breakfast on the airline was great."
Output: B

Example 4:
PROPERTY: "uses formal academic language."
TEXT A: "This study investigates nominalization."
TEXT B: "This looks cool."
Output: A
```

```
Now complete the following example: respond only with A, B, or Neither.

PROPERTY: {{ interpretation }}
TEXT A: {{ response_0 }}
TEXT B: {{ response_1 }}
Output:
```

A.4.3 LLM-AS-A-JUDGE PAIRWISE EVALUATION PROMPT

We use the vanilla JudgeBench pairwise preference prompt (Tan et al., 2024).

A.4.4 SUBJECTIVE PROMPT FILTER

To filter prompts that require an objectively correct answer, we use the following binary classifier prompt adapted from Huang et al. (2025b):

```
The following is a single-turn exchange between a user and an AI
    assistant:

{{transcript}}

<question>
Does the assistant's reply require giving a subjective judgment?

Answer "Yes" if the request instead invites opinion, preference,
    creative generation, advice, or other context-dependent
    interpretation.
Answer "No" if the user's request calls for a verifiable or factually
    correct response (e.g., definitions, calculations, technical
    explanations, factual information).

If the distinction is unclear, decide based on what type of answer would
    best satisfy the user's prompt.
Output only <answer>Yes</answer> or <answer>No</answer>.
</question>
```

A.5 Feature Analysis

Figure 4 shows the full set of Differential SAE features on the combined preference dataset.

Figure 4: All Differential SAE features for the combined preference dataset, with interpretations and Δwin-rate for human and LLM annotators.

Figure 5 shows the full feature visualization for `askacademia`.

Figure 6 shows the full feature visualization for `legaladvice`.

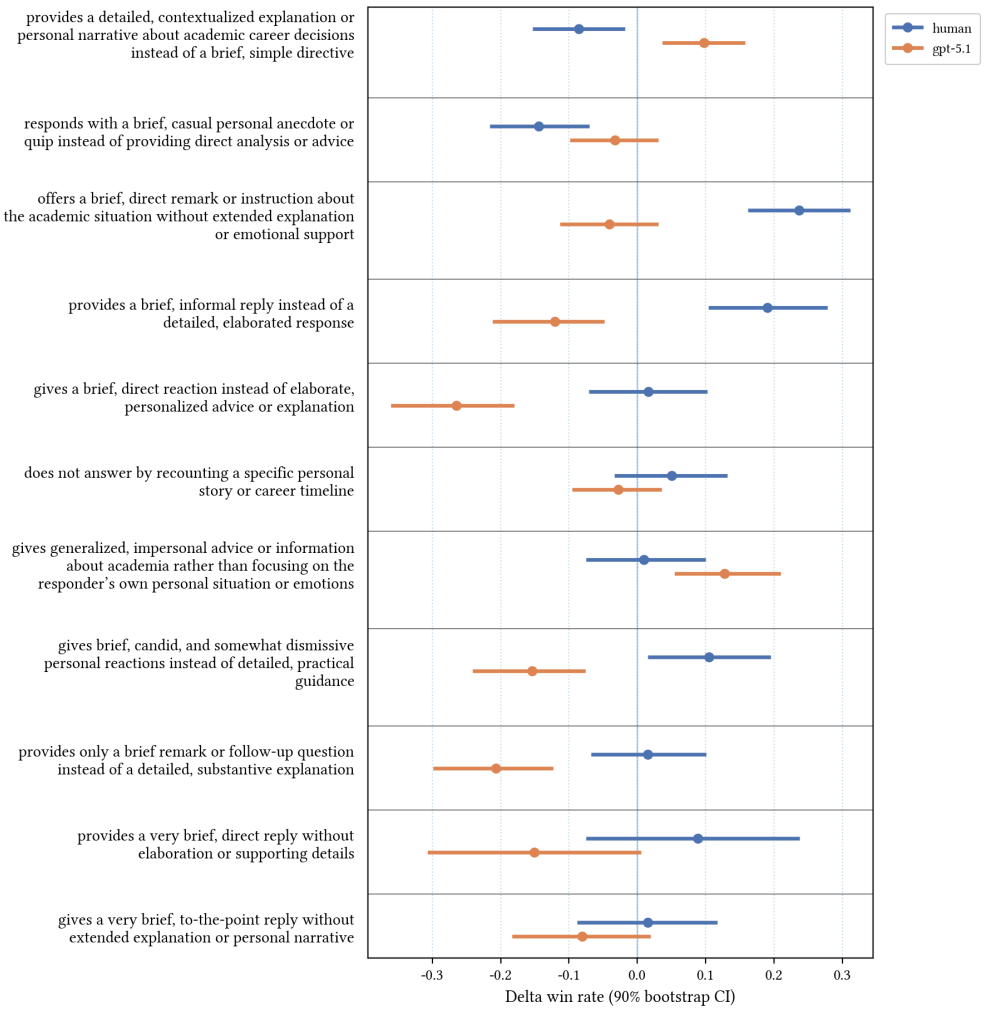

Figure 5: All features for the `askacademia` dataset.

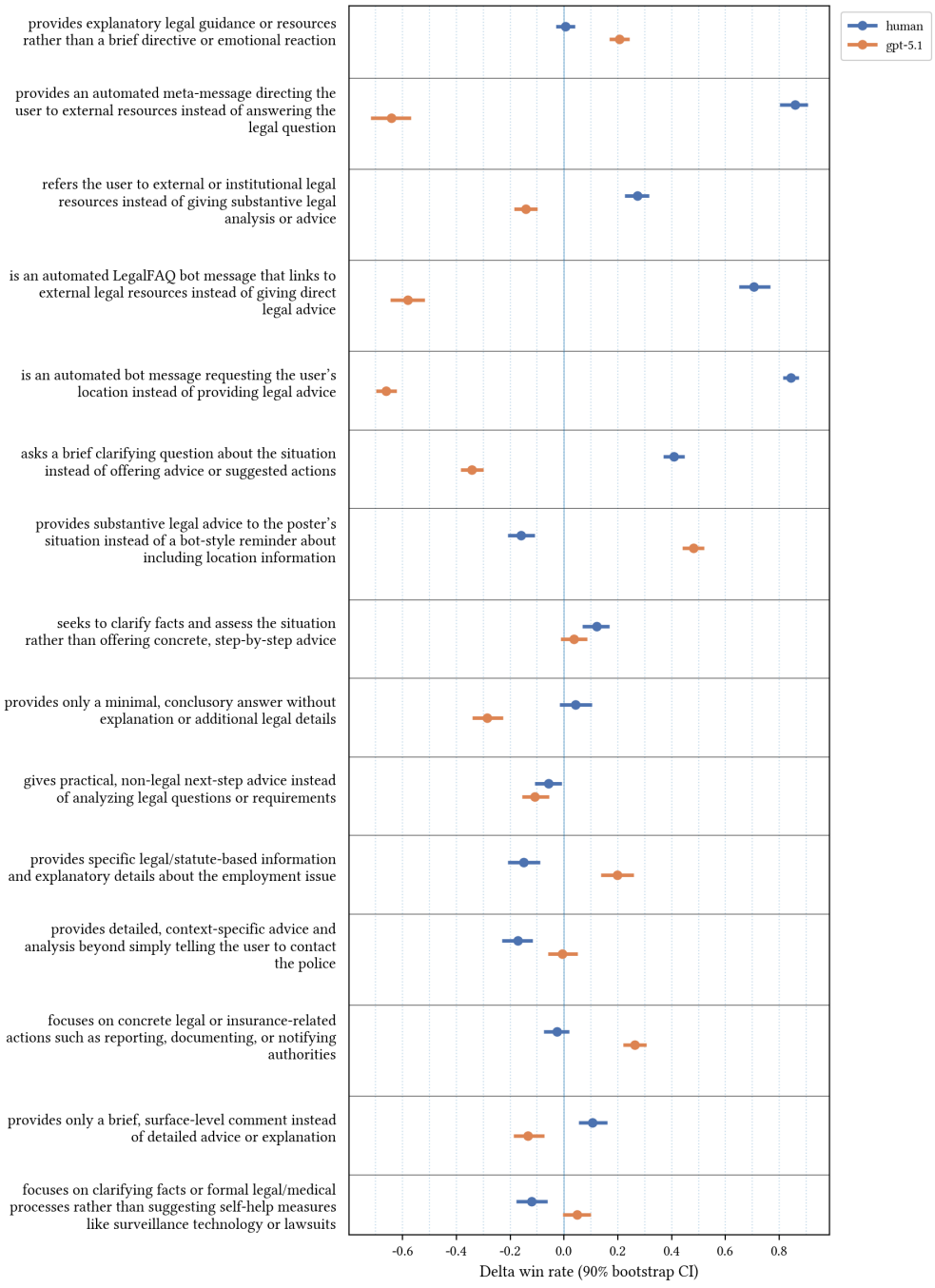

Figure 6: All features for the `legaladvice` dataset.