Probing Large Language Model Hidden States for Adverse Drug Reaction Knowledge

Jacob Berkowitz[1] [0009-0009-8460-7240], Davy Weissenbacher[1] [0000-0001-8331-3675], Apoorva Srinivasan[1] [0009-0001-6036-7651], Nadine A. Friedrich[1] [0000-0002-6713-0614], Jose Miguel Acitores Cortina[1] [0000-0003-0937-6517], Sophia Kivelson[1][0000-0003-0937-6517], Graciela Gonzalez Hernandez[1] [0000-0002-6416-9556], Nicholas P. Tatonetti[1,2] [0000-0002-2700-2597]

[1] Department of Computational Biomedicine, Cedars-Sinai Medical Center, Los Angeles, California, USA
[2] Cedars-Sinai Cancer, Cedars-Sinai Medical Center, Los Angeles, California, USA
Jacob.berkowitz2@cshs.org, Nicholas.Tatonetti@cshs.org

Abstract. Large language models (LLMs) integrate knowledge from diverse sources into a single set of internal weights. However, these representations are difficult to interpret, complicating our understanding of the models' learning capabilities. Sparse autoencoders (SAEs) linearize LLM embeddings, creating monosemantic features that both provide insight into the model's comprehension and simplify downstream machine learning tasks. These features are especially important in biomedical applications where explainability is critical. Here, we evaluate the use of Gemma Scope SAEs to identify how LLMs store known facts involving adverse drug reactions (ADRs). We transform hidden-state embeddings of drug names from Gemma2-9b-it into interpretable features and train a linear classifier on these features to classify ADR likelihood, evaluating against an established benchmark. These embeddings provide strong predictive performance, giving AUC-ROC of 0.957 for identifying acute kidney injury, 0.902 for acute liver injury, 0.954 for acute myocardial infarction, and 0.963 for gastrointestinal bleeds. Notably, there are no significant differences ($p > 0.05$) in performance between the simple linear classifiers built on SAE outputs and neural networks trained on the raw embeddings, suggesting that the information lost in reconstruction is minimal. This finding suggests that SAE-derived representations retain the essential information from the LLM while reducing model complexity, paving the way for more transparent, compute-efficient strategies. We believe that this approach can help synthesize the biomedical knowledge our models learn in training and be used for downstream applications, such as expanding reference sets for pharmacovigilance.

Keywords: Interpretable AI, Sparse Autoencoder, Knowledge Synthesis, Adverse drug reactions

1 Introduction

Through training, large language models (LLMs) synthesize information from diverse sources into coherent representations. Large pretrained models like OpenAI's GPT-

4[1], Google's Gemma[2], etc. have demonstrated exceptional abilities in understanding context and generating human-like text, making them valuable for advancing scientific research across various domains [3].

Evaluating large language models' (LLMs) knowledge, or probing [4], solely through generation tasks can be misleading, as these tasks may not reveal the model's true internal knowledge representation. The GPT-4 technical report [1] highlights that current evaluation methods can lead to models appearing overconfident while producing inaccurate predictions [5]. This systematic bias suggests we need more sophisticated approaches to probe and understand LLM capabilities.

One promising approach is to classify based on residual stream activations in the hidden layers of LLMs. This approach considers the internal "*hidden state*" layers of the model, which may retain more accurate signals of confidence and correctness than the output layer [6]. However, a notable challenge in this approach is the issue of superposition, where features are not linearly separable and often polysemantic, responding to mixtures of unrelated inputs [7, 8]. While deep learning approaches can address this complexity, they often lack explainability and are computationally intensive[9].

Sparse autoencoders (SAEs) offer a compelling solution for extracting monosemantic features, or features dedicated to a single and specific concept, from LLM residual streams by transforming complex activations into interpretable components. The process involves transforming the activations to include a sparsity constraint on the internal activations. This encourages most neurons to remain inactive while a select few, termed feature neurons, become highly active. These active neurons are designed to represent isolated concepts, promoting monosemanticity and providing a clearer window into the model's understanding[8, 10].

We propose using this approach to evaluate LLM knowledge of adverse drug reactions (ADRs) for given drugs, an important task in pharmacovigilance. By using SAEs to distill LLM embeddings into interpretable features, we can improve our understanding of the biomedical knowledge embedded in these models.

ADRs are unfavorable reactions associated with drug use, whether preventable or not [11], and represent a significant concern in patient care, often contributing to increased morbidity, hospitalizations, and healthcare costs[12–14]. Despite advancements in pharmacovigilance, identifying and characterizing ADRs remains challenging due to the fragmented and unstructured nature of relevant data, including clinical trials, electronic health records, and social media platforms[15–17].

Current methods for ADR detection rely heavily on natural language processing (NLP) tools[18, 19], which may fail to capture the complex relationships between drugs and adverse effects. Traditional pharmacovigilance systems also depend on spontaneous reporting, which is frequently hurt by underreporting, delays, and incomplete data[20, 21].

In this study, we use SAEs to extract interpretable features from LLM embeddings for identifying ADRs. Our approach demonstrated strong classification performance across multiple health outcomes, suggesting that SAE-derived representations effectively retain critical information while simplifying model complexity. This process enhances interpretability and offers a promising route for improving pharmacovigilance applications.

2 Methods

2.1 Reference Data Source

We use a reference set of test cases from "Defining a Reference Set to Support Methodological Research in Drug Safety" by Ryan et al. [22]. A test case set in this context refers to pairs of drugs and symptoms, where the symptoms are specific adverse health outcomes. This reference set provides a benchmark across four key health outcomes: acute liver injury, acute kidney injury, acute myocardial infarction, and upper gastrointestinal bleeding.

The reference set comprises 399 test cases, including 165 positive controls (meaning **drug-adverse reaction pairs with established evidence of a causal relationship**) and 234 negative controls (meaning pairs with no evidence of a causal relationship). Positive controls are drug-symptom pairs supported by evidence from randomized clinical trials, observational studies, and case reports. Negative controls are selected based on the absence of evidence suggesting a causal relationship between the drug and the outcome.

2.2 Model Descriptions

Gemma-2-9b-it

Driven by the availability of pre-trained SAEs. we use the Gemma-2-9b-it[2] model developed by Google DeepMind. They trained the model on 8 trillion tokens, learning a diverse dataset that includes web documents, code, and scientific articles. The model's architecture includes 42 layers with a model dimension of 3584 and a context length of 8192 tokens.

Gemma-2-9b-it specifically, is an instruction-tuned, meaning it has undergone post-training fine-tuning to improve its performance on specific tasks, such as instruction following and safety. This tuning involves supervised fine-tuning and reinforcement learning from human feedback (RLHF), which helps align the model's outputs with desired behaviors and reduces the likelihood of generating harmful content.

Gemma Scope

We use pre-trained SAEs from the Gemma Scope[23] project, specifically trained on Gemma2-9b-it's residual stream after layers 9, 20, and 31, to transform the activations of drug names into interpretable features. The SAE expands the model from 3584 to 131,072 dimensions and uses a JumpReLU activation function, which enforce sparsity by zeroing out activations below a learned threshold, enhancing interpretability while maintaining reconstruction fidelity.

2.3 Classification Techniques for Probing

SAE-Driven Feature Extraction

Transforming hidden-state drug embeddings from the Gemma-2-9b-it model into interpretable features involves several key steps, as visually represented in Figure 1.

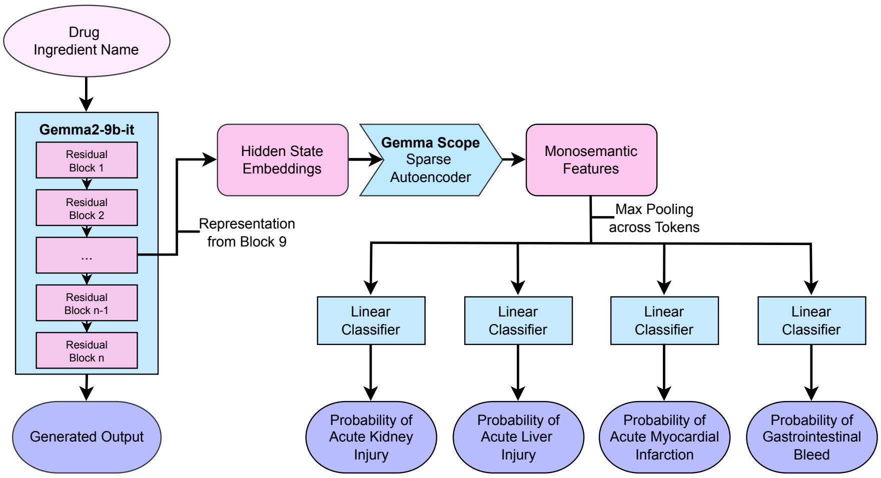

Fig. 1. Flowchart for converting drug ingredient names into monosemantic features used to classify drug-ADR relationships.

We begin by extracting activations from a specific layer of the Gemma-2-9b-it model. These activations are processed using Gemma Scope to produce monosemantic features. The SAE transformation simplifies the complex embeddings, making them more interpretable while preserving critical information necessary for downstream analysis.

The transformed features are applied to our dataset of drug names associated with specific ADRs. Each drug name may be tokenized into m tokens, where each token i is represented by a feature vector $\bar{t}_i$. To manage the dimensionality and enhance interpretability, we condense these m token-level vectors via max pooling across tokens. Concretely, if each $\bar{t}_i$ is d-dimensional, then the pooled vector $\bar{t}_{pooled} \in \mathbb{R}^d$ is computed by taking the maximum value in each dimension across all tokens:

$$\bar{t}_{pooled,j} = \max_{0 \le i \le m} (\bar{t}_{i,j}) \text{ for } j = 1,2, \dots, d \tag{1}$$

The pooled feature vector $\bar{t}_{pooled}$ is then used as input to a logistic regression model to classify the likelihood of ADRs. Denote each dimension of $\bar{t}_{pooled}$ by x_j. We predict the probability of an ADR occurring as follows:

$$P(y = 1 | \bar{t}_{pooled}) = \frac{1}{1+e^{-(\beta_0 + \sum_{j=1}^{d} \beta_j x_j)}} \tag{2}$$

where β_0 is the intercept, and β_j are the coefficients learned for each dimension j.

To encourage sparsity in the coefficients (which helps identify the most influential features), L1 regularization is applied. The penalized loss function is:

$$\text{Loss} = -\sum(y \log \hat{y} + (1-y)(\log 1 - \hat{y})) + \lambda \sum_{j=1}^{d} |\beta_\mathrm{j}| \quad (3)$$

where λ is the L1 regularization parameter.

The model's performance is evaluated using 10 repetitions of 5-fold cross-validation, and results are reported as area under the receiver operating characteristic (ROC) curve (AUC) scores across the selected health outcomes. To assess the statistical significance of the model's performance, we conduct a permutation test[24], where the labels were shuffled and the AUC scores were recalculated to generate a distribution of scores under the null hypothesis. The p-values were computed by comparing the original AUC scores to this permuted distribution. Finally, to understand which pooled features are most indicative of ADR risk, we look at the difference between normalized features with nonzero L1-regularized coefficients.

Text Generation-Based Probing

To assess the generative capabilities of the Gemma-2-9b-it model in classifying ADRs, we worked with a straightforward prompting strategy. The model was prompted with the query:
"<start_of_turn>user\nIs the drug {drug_name} known to cause {condition_name}? Answer using only 'Yes' or 'No'.<end_of_turn>\n<start_of_turn>model\n"

We then analyzed the probability assigned to the correct token (either "Yes" or "No") as the next word. This probability was used to compute the ROC AUC.

Neural Networks on Unaltered Model Activations

To explore the predictive power of the unaltered model activations (3584 dense dimensions compared to the sparse 131,072 dimensions of our SAE activations), we implemented a shallow feed-forward neural network architecture. Specifically, after each residual block of the Gemma-2-9b-it model, we trained a multilayer perceptron (MLP) with a single hidden layer comprising 10 neurons on the residual stream. This setup was chosen to maintain simplicity while capturing essential patterns in the data. The model's performance was evaluated using repeated stratified k-fold cross-validation, with 5 splits and 10 repetitions. Similarly to the other approaches, we looked at the AUC to evaluate performance.

3 Results

3.1 Classification Performance of Layer 9 Activations

The performance of the SAE-derived features from Layer 9 activations of the Gemma-2-9b-it model was evaluated for classifying ADRs. The SAE transformation of hidden-state embeddings of drug names resulted in monosemantic features that were subsequently used in a logistic regression model to classify ADRs.

For acute kidney injury, the area under the receiver operating characteristic curve (AUC) was 0.957, with a p-value of 0.020 from the permutation test. Similarly, for acute liver injury, the AUC was 0.902, with a p-value<0.001. The prediction of acute myocardial infarction yielded an AUC of 0.954, with a p-value<.001. Finally, the model achieved an AUC of 0.963 for gastrointestinal bleeds, with a p-value<.001. These results suggest that the SAE-derived features effectively capture the necessary information for accurate ADR classification.

Figure 2 illustrates the differences in expression values across the nonzero features identified by L1 regression with λ=.10.

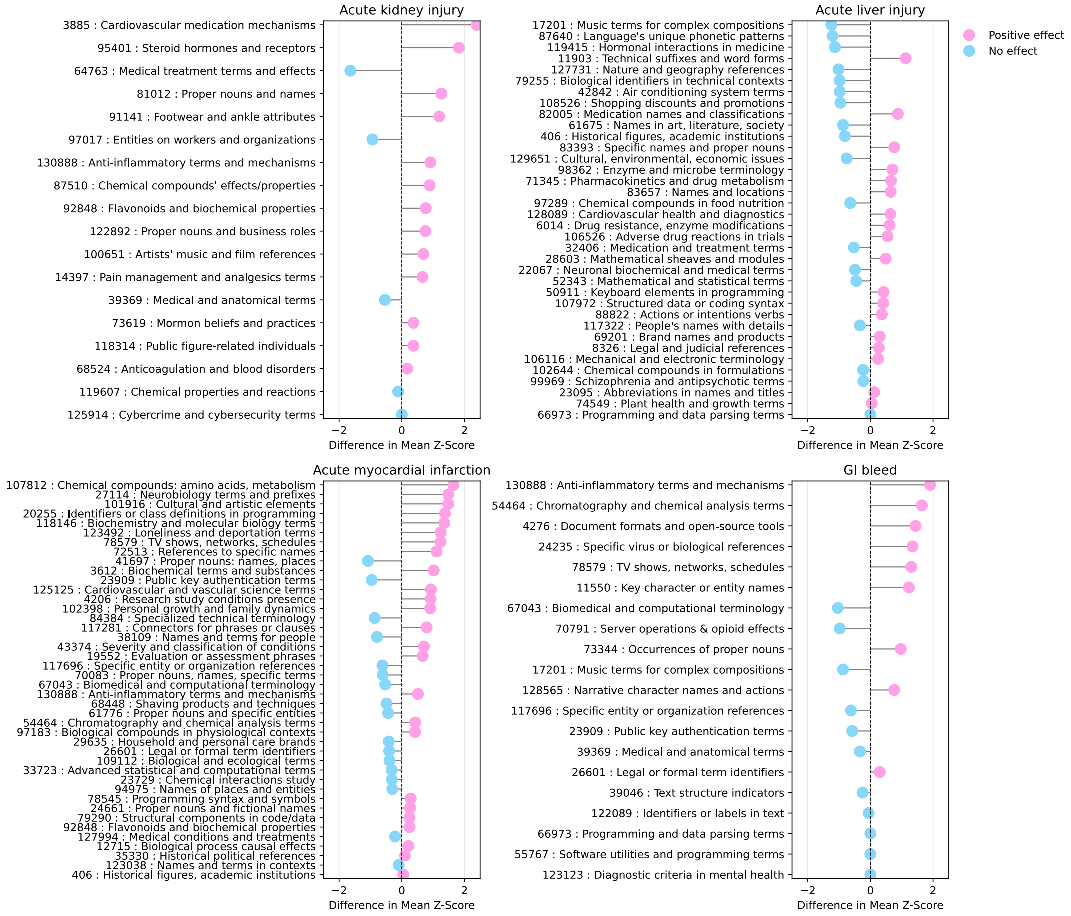

Fig. 2. Lollipop plots showing the differences in mean Z-scores for nonzero sparse autoencoder features identified by L1 logistic regression across four conditions: acute kidney injury, acute liver injury, acute myocardial infarction, and gastrointestinal bleed. Feature descriptions are from https://www.neuronpedia.org/.

3.2 Comparison Across Layers and Approaches

We extend our analysis by evaluating SAEs trained on different layers of the Gemma-2-9b-it model, specifically layers 9, 20, and 31. Although pairwise t-tests among the three layers yield p-values of 0.016 for L9 vs L20, 0.884 for L9 vs L31, and 0.016 for L20 vs L31 (suggesting that L20 may differ statistically from the other two), the ranges of the AUC distributions in the boxplots overlap substantially. In practical terms, the performances across these layers are quite similar, with no clear advantage to choosing one layer over another based on these data alone.

We compare our classification SAE-derived features to a text-based generative approach, where the model was prompted to classify ADRs for a given drug directly. The generative method showed lower AUC scores compared to the SAE classifier on layer 9, with p-values indicating significant differences for each ADR: acute kidney injury ($p = 0.001$), acute liver injury ($p = 0.789$), acute myocardial infarction ($p < 0.001$), and gastrointestinal bleeds ($p < 0.001$). These results suggest that while the text generation-based approach captures some predictive information, the structured feature extraction via SAEs offers a more robust and interpretable solution.

Furthermore, we trained neural networks on the untransformed activations from each layer to explore their classification capabilities. These networks, evaluated across all layers, demonstrated varying performance. Figure 3 visualizes the comprehensive comparison of all methods, showing the AUC scores for SAEs, generative predictions, and neural networks across the evaluated layers.

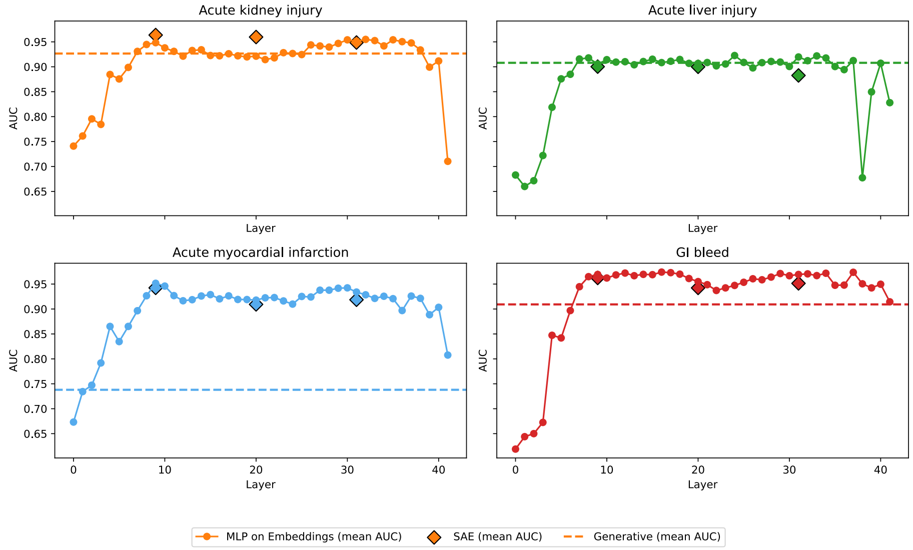

Fig. 3. Comparison of mean AUC scores for ADR classification using SAE-derived features from layers 9, 20, and 31, generative predictions, and neural networks trained on raw activations across all layers.

4 Discussion

In this study, we explore how SAEs enhance the interpretability of the information within LLMs for ADR classification, revealing insights into the Gemma-2-9b-it model's internal workings.

Examining the performance of early layers, particularly layers 0 to 9, we see that each layer demonstrates substantial gains in performance. These layers appear to capture foundational features that are crucial for identifying ADRs, suggesting that they hold generalizable patterns and representations. As we moved to later layers, we observed a plateau in performance, with the latest layers (40 to 42) showing sporadic results. This shift likely reflects the model's focus on preparing for next-token predictions, a task that likely does not align well with the requirements for ADR classification.

Interestingly, a brief manual annotation revealed that only about half (46.6%) of the features identified by SAEs were somewhat science related. This partial monosemanticity suggests that while the features are not purely isolated to single concepts, they still activate in ways that are beneficial for ADR classification suggested by their retention through L1 regularization. For example, the feature 91141 (references to footwear or ankle attributes) appears in our layer 9 acute kidney injury model. We speculate that this may reflect broader associations with conditions that affect mobility or circulation, which are relevant to kidney health. Since features appear to split with an increase in SAE dimension[23], a larger SAE may better differentiate the relevant aspects of this feature.

The comparison between SAEs and the text-generation method revealed the advantages of structured feature extraction. While generative approaches rely heavily on prompt engineering, SAEs focus on internal representations. Prompt engineering can be thought of as optimizing the selection of features in the overall model structure. Learning from SAE activations does this automatically, without the trial and error of prompt engineering[25].

On this note, when comparing the SAE output to the unaltered activations, we found no significant differences in performance. This result is particularly encouraging, as it confirms that SAEs can simplify complex model outputs without losing critical information. This type of dictionary learning offers an interpretable approach to describing text input, which could simplify the circuitry of downstream machine learning tasks.

While our SAE-based approach shows promise in making the information within LLMs more interpretable for ADR classification, it's important to acknowledge its limitations. The computational demands of training SAEs are intensive, potentially limiting their accessibility. Pretrained SAEs offer valuable insights into their parent LLMs, but training SAEs for additional models remains resource intensive[23]. Future research should focus on optimizing the training process or developing lightweight alternatives that maintain interpretability without the computational burden. Additionally, the partial monosemanticity observed in the extracted features suggests that there is room for improvement in achieving fully separable representations. Also worth noting, our generative approach was not heavily optimized through prompt engineering, which may have impacted its performance.

Looking ahead, future researchers could expand and refine our classification approach by exploring interactions and correlations between features. This would lead to a deeper understanding of how different features contribute to ADR classification and improve model accuracy. Additionally, using sparse crosscoders [26] to consider information across multiple layers could provide a more holistic view of the model's internal representations, potentially uncovering new information on the complex dynamics within transformers.

Our findings indicate that the internal representations of LLMs, as interpreted through SAEs, provide a valuable synthesis of biomedical knowledge, as demonstrated in the context of ADR detection. By building upon these representations, we were able to validate known drug-ADR pairs, showcasing the potential for SAEs to interpret and utilize the information acquired during pretraining. This approach not only confirms existing knowledge but also opens the possibility of discovering new correlations that may not yet be recognized by experts. As we continue to explore these latent features, we aim to uncover novel insights that could enhance pharmacovigilance efforts and improve patient safety.

Acknowledgments. NF is supported by NIH T32 HL116273.

Disclosure of Interests. The authors declare no conflicts of interest or disclosures related to this study.

References

1. OpenAI, Achiam J, Adler S, et al (2023) GPT-4 Technical Report
2. Gemma Team, Riviere M, Pathak S, et al (2024) Gemma 2: Improving Open Language Models at a Practical Size
3. Sallam M (2023) The Utility of ChatGPT as an Example of Large Language Models in Healthcare Education, Research and Practice: Systematic Review on the Future Perspectives and Potential Limitations
4. Ju T, Sun W, Du W, et al (2024) How Large Language Models Encode Context Knowledge? A Layer-Wise Probing Study
5. Steyvers M, Tejeda H, Kumar A, et al (2025) What large language models know and what people think they know. Nat Mach Intell. https://doi.org/10.1038/s42256-024-00976-7
6. Azaria A, Mitchell T (2023) The Internal State of an LLM Knows When It's Lying
7. Elhage N, Hume T, Olsson C, et al (2022) Toy Models of Superposition
8. Lan M, Torr P, Meek A, et al (2024) Sparse Autoencoders Reveal Universal Feature Spaces Across Large Language Models
9. Zhao H, Chen H, Yang F, et al (2023) Explainability for Large Language Models: A Survey
10. Lieberum T, Rajamanoharan S, Conmy A, et al (2024) Gemma Scope: Open Sparse Autoencoders Everywhere All At Once on Gemma 2
11. World Health Organization (2011) Patient safety curriculum guide: multi-professional edition. 272

12. Sahilu T, Getachew M, Melaku T, Sheleme T (2020) Adverse Drug Events and Contributing Factors Among Hospitalized Adult Patients at Jimma Medical Center, Southwest Ethiopia: A Prospective Observational Study. Curr Ther Res Clin Exp 93:100611. https://doi.org/10.1016/j.curtheres.2020.100611

13. Durand M, Castelli C, Roux-Marson C, et al (2024) Evaluating the costs of adverse drug events in hospitalized patients: a systematic review. Health Econ Rev 14:11. https://doi.org/10.1186/s13561-024-00481-y

14. Costa C, Abeijon P, Rodrigues DA, et al (2023) Factors associated with underreporting of adverse drug reactions by patients: a systematic review. Int J Clin Pharm 45:1349–1358. https://doi.org/10.1007/s11096-023-01592-y

15. Harpaz R, Callahan A, Tamang S, et al (2014) Text mining for adverse drug events: the promise, challenges, and state of the art. Drug Saf 37:777–90. https://doi.org/10.1007/s40264-014-0218-z

16. Liang L, Hu J, Sun G, et al (2022) Artificial Intelligence-Based Pharmacovigilance in the Setting of Limited Resources. Drug Saf 45:511–519. https://doi.org/10.1007/s40264-022-01170-7

17. Golder S, O’Connor K, Wang Y, Gonzalez Hernandez G (2023) The Role of Social Media for Identifying Adverse Drug Events Data in Pharmacovigilance: Protocol for a Scoping Review. JMIR Res Protoc 12:e47068. https://doi.org/10.2196/47068

18. Murphy RM, Klopotowska JE, de Keizer NF, et al (2023) Adverse drug event detection using natural language processing: A scoping review of supervised learning methods. PLoS One 18:e0279842. https://doi.org/10.1371/journal.pone.0279842

19. Luo Y, Thompson WK, Herr TM, et al (2017) Natural Language Processing for EHR-Based Pharmacovigilance: A Structured Review. Drug Saf 40:1075–1089. https://doi.org/10.1007/s40264-017-0558-6

20. Costa C, Abeijon P, Rodrigues DA, et al (2023) Factors associated with underreporting of adverse drug reactions by patients: a systematic review. Int J Clin Pharm 45:1349–1358. https://doi.org/10.1007/s11096-023-01592-y

21. Alomar M, Tawfiq AM, Hassan N, Palaian S (2020) Post marketing surveillance of suspected adverse drug reactions through spontaneous reporting: current status, challenges and the future. Ther Adv Drug Saf 11:2042098620938595. https://doi.org/10.1177/2042098620938595

22. Ryan PB, Schuemie MJ, Welebob E, et al (2013) Defining a reference set to support methodological research in drug safety. Drug Saf 36:. https://doi.org/10.1007/s40264-013-0097-8

23. Lieberum T, Rajamanoharan S, Conmy A, et al (2024) Gemma Scope: Open Sparse Autoencoders Everywhere All At Once on Gemma 2

24. Good PI (2004) Permutation, Parametric, and Bootstrap Tests of Hypotheses (Springer Series in Statistics)

25. Kong W, Hombaiah SA, Zhang M, et al (2024) PRewrite: Prompt Rewriting with Reinforcement Learning

26. Lindsey J, Templeton A, Marcus J, et al (2024) Sparse Crosscoders for Cross-Layer Features and Model Diffing