Evaluating the Utility of Sparse Autoencoders for Interpreting a Pathology Foundation Model

Nhat Minh Le[*]
PathAI Inc,
Boston, USA
nhatmle.mail@gmail.com

Ciyue Shen
PathAI Inc,
Boston, USA
judy.shen@pathai.com

Neel Patel[*]
PathAI Inc,
Boston, USA
neelnil14@gmail.com

Chintan Shah
PathAI Inc,
Boston, USA
chintan.shah@pathai.com

Darpan Sanghavi[*]
PathAI Inc,
Boston, USA
darpansanghavi1993@gmail.com

Blake Martin
PathAI Inc,
Boston, USA
blake.martin@pathai.com

Alfred Eng[*]
PathAI Inc,
Boston, USA
theblueman@gmail.com

Daniel Shenker[*]
PathAI Inc,
Boston, USA
dan.shenker@gmail.com

Harshith Padigela
PathAI Inc,
Boston, USA
harshith.padigela@pathai.com

Raymond Biju[*]
PathAI Inc,
Boston, USA
raymondbiju@gmail.com

Syed Ashar Javed
PathAI Inc,
Boston, USA
ashar.javed@pathai.com

Jennifer Hipp
PathAI Inc,
Boston, USA
jennifer.hipp@pathai.com

John Abel[*]
PathAI Inc,
Boston, USA
jhabel01@gmail.com

Harsha Pokkalla
PathAI Inc,
Boston, USA
harsha.pokkalla@pathai.com

Sean Grullon[*]
PathAI Inc,
Boston, USA
sean.grullon@gmail.com

Dinkar Juyal[*]
PathAI Inc,
Boston, USA
dinkarjuyal@gmail.com

Abstract

Pathology plays an important role in disease diagnosis, treatment decision-making, and drug development. Previous works on interpretability for machine learning models on pathology images have revolved around methods such as attention value visualization and deriving human-interpretable features from model heatmaps. Mechanistic interpretability is an emerging area of model interpretability that focuses on reverse-engineering neural networks. Sparse Autoencoders (SAEs) have strong potential for extracting monosemantic concepts from polysemantic model

---

[*] Employee of PathAI at the time of study

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: The 3rd Workshop on Imageomics: Discovering Biological Knowledge from Images Using AI.

activations. In this work, we train a Sparse Autoencoder on the embeddings of a pathology-pretrained foundation model. We find that Sparse Autoencoder features represent interpretable and monosemantic biological concepts. In particular, individual SAE dimensions show strong correlations with the counts of individual cell types, such as plasma cells and lymphocytes. These biological representations are unique to the pathology-pretrained model and are not found in a self-supervised model pretrained on natural images. These biologically grounded monosemantic representations evolve across the model's depth, and the pathology foundation model eventually gains robustness to non-biological factors, such as scanner type. The emergence of these biologically-relevant SAE features are generalizable to an out-of-domain dataset. Finally, we highlight certain limitations of SAEs and why more work is needed towards achieving complete monosemanticity. Our work paves the way for further exploration around interpretable feature dimensions and their utility for medical and clinical applications.

1 Introduction

Artificial Intelligence (AI) has made significant strides in various domains, including healthcare and pathology. As AI systems become more complex and widely adopted, understanding their internal mechanisms becomes crucial for ensuring reliability, addressing biases, and fostering trust among potential users in the medical community. This paper focuses on the application of mechanistic interpretability (MI) techniques, particularly sparse autoencoders (SAEs), to neural networks used in pathology.

Mechanistic interpretability aims to study neural networks by reverse-engineering them, providing insights into their internal workings [1, 2, 3, 4]. SAEs are important tools for mechanistic interpretability, being used in NLP [5, 6] to achieve a more monosemantic unit of analysis compared to the model neurons. In vision datasets, SAEs trained on layers of convolutional neural nets have uncovered interpretable features such as curve detectors [7, 8]. Various improvements to SAEs have been suggested, including k-sparse [9] and gated sparse [10] autoencoders, and using JumpReLU [11] instead of ReLU as the activation function.

Histopathology, a term often used interchangeably with pathology, is the diagnosis and study of diseases through microscopic examination of cells and tissues. It plays a critical role in disease diagnosis and grading, treatment decision-making, and drug development [12, 13]. Digitized whole-slide images (WSIs) of pathology samples can be gigapixel-sized, containing millions of areas of interest and biologically relevant entities across a wide range of characteristic length scales.

Machine learning has been applied to pathology images for tasks such as segmentation and classification of biological entities, and end-to-end weakly supervised prediction at a WSI level [14, 15, 16]. Work on interpretability in pathology has focused on assigning spatial credit to WSI-level predictions [17, 18], computing human-interpretable features from model output [19], and visualization of multi-head self-attention values on image patches [20].

Foundation Models (FMs) are promising for pathology as they can take advantage of large amounts of unlabeled data to build rich representations which can be easily adapted for downstream tasks in a data-efficient manner [21, 22, 23, 24, 20]. The diversity of pre-training data powers these models to generate robust representations, enabling them to generalize better than individual task-specific models trained on smaller datasets. Additionally, these models can be used as a universal backbone across different tasks, reducing the development and maintenance overhead associated with bespoke task-specific models.

MI is particularly interesting in histopathology, where understanding the decision-making process of AI systems can promote trustworthiness of models in clinical settings. In addition, pathology images are susceptible to high-frequency artifacts and systematic confounders in image acquisition [25]. MI can help disentangle biological content from incidental attributes, leading to more robust models for real-world applications. With a forward-looking perspective, MI may lead to new biological insights or hypotheses that were not apparent through traditional analysis methods.

This work evaluates the utility of sparse autoencoders (SAEs) for interpretability analysis of the embedding dimensions derived from a vision foundation model trained on histopathology images.

We train SAEs on the pathology foundation model embeddings and examine four key properties of the trained SAEs which provide evidence for their utility in downstream interpretability analysis. We then evaluate the trained SAEs as well as the original FM embedding on which these SAEs are trained on two key criteria (1) monosemanticity of the neurons within the embeddings and (2) predictive performance of sparse linear probes. Our study provides the first detailed characterization and evaluation of sparse autoencoders in pathology.

The main contributions of our work are as follows:

- We train the first sparse autoencoders (SAEs) on the embeddings of a pathology foundation model and evaluate the interpretability of the dimensions in the SAE latent space.
- We discover the following key properties of the trained SAEs:
    - Features that activate individual neurons in the SAE latent space are highly biological, including cell and tissue characteristics, geometric structures, and image artifacts.
    - Pathology-specific SAEs capture biological concepts better than natural-image SAEs.
    - A small proportion of SAE features are universal.
    - SAE latent dimensions representing biological features are robust to sources of variations like scanners and stains.
- We conduct a detailed evaluation of SAEs and original FM embeddings on which they are trained in two key metrics (1) monosemanticity of the neurons, and (2) predictive power in sparse probes. We find mixed evidence regarding the utility of SAEs in these two metrics.
    - SAE dimensions are much more monosemantic than original FM embeddings.
    - Monosemanticity in SAE features emerges in later layers of the FM.
    - Monosemantic behavior of SAEs generalizes to new datasets with unseen cancer types.
    - Partial monosemanticity limits SAE utility.
    - Sparse probes trained on SAE latents do not always outperform those trained on FM embeddings in predicting biological concepts.

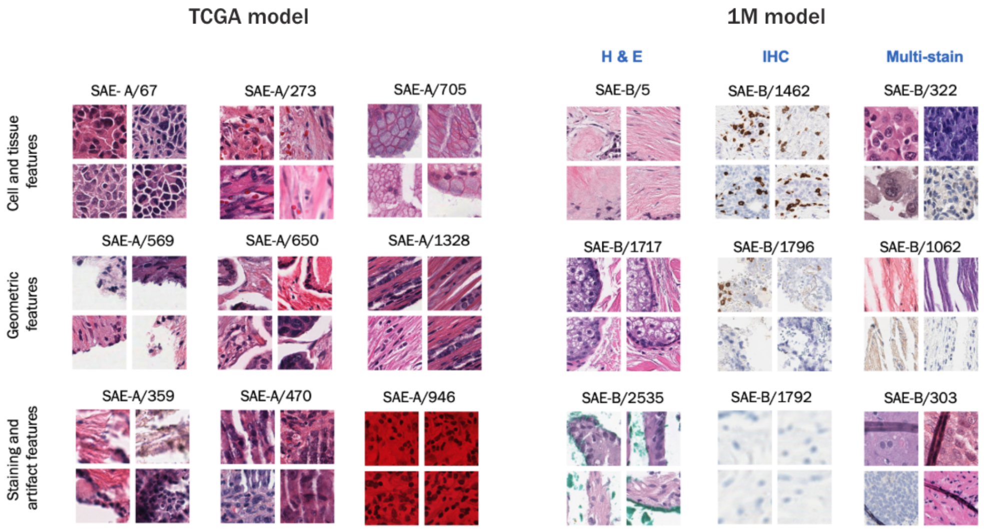


Figure 1: Feature visualization of SAE hidden dimensions revealed interpretable dictionary of pathology features. Manual examination reveals interpretable features represented by these dimensions. For the TCGA model, these include cell and tissue features specific to H & E stain (top: poorly differentiated carcinoma, red blood cells, mucin); geometric features (middle: edge of tissue, clefting, diagonal fibers); staining and artifact features (bottom: blur, sectioning artifact, red stain). For the 1M model, some SAE dimensions are specific to H & E stain (left column), or specific to IHC stain (middle column), or generalizable across stains (right column: large cancer cells, vertical structures, tissue folds).

2 Experimental Setup

2.1 Datasets

For the training dataset, we use 1.1 million image patches (1M dataset) (224 x 224 pixels at resolution of 0.25 microns per pixel) including both haematoxylin & eosin (H & E) and immunohistochemistry (IHC) stains, covering oncology, IBD (inflammatory bowel disease), and MASH (metabolic dysfunction-associated steatohepatitis).

Two different datasets are used for evaluation. The first ('TCGA dataset') consists of three publicly available TCGA (The Cancer Genome Atlas) [26] cohorts containing H & E-stained histology images from three organs: breast (TCGA-BRCA, 951 WSIs), lung (TCGA-LUAD, 493 WSIs), and prostate (TCGA-PRAD, 488 WSIs). The second dataset ('CPTAC') consists of two publicly available CPTAC cohorts (Clinical Proteomic Tumor Analysis Consortium) [27] containing H & E-stained histology images from two cancer types: cutaneous melanoma (CPTAC-CM, 256 WSIs), and head and neck cancer (CPTAC-HNSCC, 228 WSIs).

2.2 Embedding extraction

All the images in the train and evaluation datasets are passed through a frozen ViT-S encoder taken from 'PLUTO' - a pathology pretrained foundation model [28]. For each image patch, we extract 384-dimensional embedding vectors corresponding to the CLS token residual stream in layers 1-12 with 12 being the output layer. CLS tokens are chosen as they better capture global context and predict total cell counts compared to average of patch tokens. For baseline comparison, we extract embeddings from a self-supervised vision transformer DINO [29] that is also 384-dimensional.

2.3 Sparse autoencoder training

We use a standard autoencoder architecture defined by [5]. The encoder and decoder are defined by

$$\begin{aligned} \mathbf{z} &= \mathrm{ReLU}\left(W_{\mathrm{enc}}(\mathbf{x} - \mathbf{b}_{\mathrm{pre}})\right) + \mathbf{b}_{\mathrm{enc}} \\ \hat{\mathbf{x}} &= W_{\mathrm{dec}}\mathbf{z} + b_{\mathrm{pre}} \end{aligned}$$

where $\mathbf{x}$ is the input PLUTO embedding and $\mathbf{z}$ is the SAE latent.

During training, encoder and decoder weights are updated to minimize the loss function $\mathcal{L} = \frac{1}{k}(\sum_{i=1}^{k} ||\mathbf{x}_i - \hat{\mathbf{x}}_i||_2 + \lambda \sum_{i=1}^{k} ||\mathbf{z}_i||_1)$, where k is the batch size. The first loss term is the reconstruction MSE, and the second is an L1 loss on the latent space to promote sparsity. [5, 30]. We use Adam optimizer with a learning rate of 0.001, expansion factors of 1, 8, 16, 32; and L1-penalty weight in 0.001, 0.004, 0.006, 0.008, 0.01. SAE training is performed on A40 GPU nodes and each model training takes about 10 min. Results are reported for models with an expansion factor of 8 and L1-penalty weight of 0.004.

A known problem during SAE training is the presence of dead neurons [5, 30], neurons that fail to activate for all input images. These dead neurons do not represent any useful features in the data. Previous work [5, 30] used dead neuron resampling to reduce the fraction of dead neurons. This involves identifying neurons that have not activated for a number of steps and resetting their encoder weights, increasing the number of likely interpretable features. We use the same approach during our SAE training.

2.4 Probing strategy

Probing is a mechanistic interpretability technique commonly used to examine the features that can be extracted from the activations of individual neurons in neural networks [31, 32, 33, 34, 35]. We train and evaluate sparse linear probes to evaluate the usefulness of SAEs in pathology.

Probe construction: We extract a number of human-interpretable features (HIFs) corresponding to biological concepts from the TCGA dataset (Section 2.1). These HIFs are extracted using PathExplore, a set of machine-learning models that detect and classify tissue and cells in tumors [36, 37] (PathExplore is for research use only; not for use in diagnostic procedures). HIFs quantifying count of cancer cells, lymphocytes, macrophages, fibroblasts, and plasma cells, as well as area, eccentricity, and orientation of cell nuclei are computed. We also extract a set of other generic image features to serve as controls for the probing. These features include gray-scale intensity, LAB colorspace, and saturation, computed by taking the average or standard deviation of feature values across all the pixels in the image at its original resolution.

Probe training: We split the TCGA dataset into a train set (80%) and a test set (20%). For each HIF, we train a k-sparse linear probe on the train set, where a k-sparse probe refers to a linear regression model with at most k non-zero coefficients. We repeat the sparse probe training for $k = 1, ..., 10$, and report results for $k = 3$. k-sparse linear probe fitting is known to be an NP-hard problem, and several approximate solutions have been proposed [38]. Here, we approximate the solution by determining the top k neurons with the highest Pearson correlation with the given HIF, and use these k neurons in the regression model.

Probe evaluation: Performance of the sparse linear probes is evaluated as the R^2 coefficient of the probe predictions on the test set.

2.5 Quantification of monosemanticity

We also test if SAE latent activations demonstrate utility compared to original PLUTO embeddings in terms of increased *monosemanticity*. A monosemantic neuron activates for a single feature, in contrast to a polysemantic neuron which activates for a large number of features. To quantify monosemanticity with respect to a set of N probes, we compute the entropy of a probability distribution $p_i = \frac{|\rho_i|}{\sum_j |\rho_j|}$ where $\rho_1, ..., \rho_N$ are the Pearson's correlations with the individual HIFs. A monosemantic neuron would have a highly-peaked distribution (high correlation for one feature and low correlations for others), with low entropy, while a polysemantic neuron would have a distribution with high entropy.

3 Characteristics of SAE Interpretability

In this section, we present our findings on the training of a sparse autoencoder model for disentangling features from a pathology foundation model. We train two sets of sparse autoencoder models using the CLS token embeddings in the output layer of PLUTO. One model is trained on the TCGA dataset, which consists of whole-slide images from a single stain (H&E), and another model is trained on a more diverse dataset of 1 million samples spanning multiple stains and diseases (1M). We will refer to these two models the "TCGA model" and the "1M model". We highlight the following four key properties that provide initial evidence for utility of SAEs

3.1 SAE dimensions are activated by interpretable biological concepts

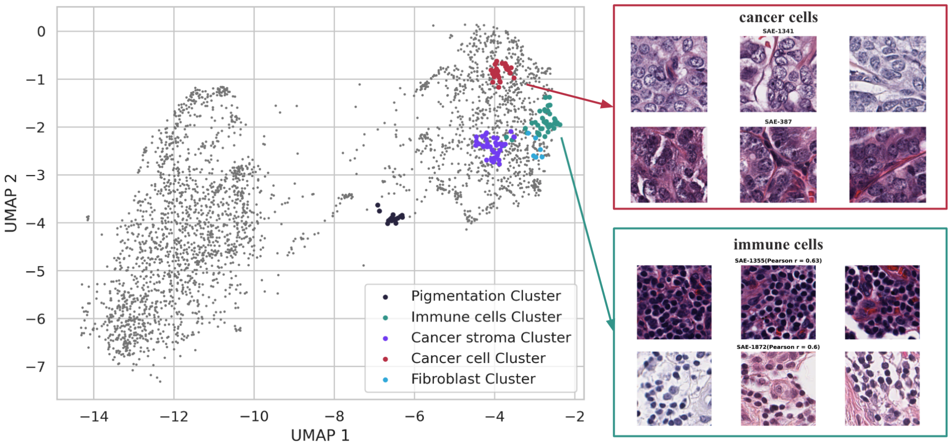
Visualization of images that strongly activate each SAE latent reveals highly interpretable features in both models, as shown in Figure 1. These features include cell and tissue features such as poorly differentiated carcinoma, geometric structures such as vertical fibers, and staining and artifact features. Importantly, SAE dimensions from the 1M model represent stain-specific features and exhibit cross-stain generalization (Fig. 1B).

To better categorize the features represented by the 1M SAE model, we extract SAE latents on the TCGA dataset (used as an independent evaluation set for the 1M model). We perform unsupervised clustering on the UMAP representations of the SAE dimensions using HDBSCAN, following the analysis strategy of [5] (Figure 2).

By manually examining image patches activating the SAE dimensions within clusters, we find clusters containing SAE features correlated with unique histological concepts such as immune cell presence (Cluster 27), cancer stroma (Cluster 33), fibroblast cells (Cluster 37) and circular cancer cells (Cluster 41). The presence of these interpretable clusters highlight the ability of SAEs to learn interpretable biological concepts from the training data.

3.2 Pathology-specific SAEs capture biological concepts better than general natural-image SAEs

We compare the representations of the 1M model against those from a baseline ViT-S pretrained on ImageNet-1k using the self-supervised DINO method (obtained from the timm library [39, 40, 41]).



Cluster ID	**Cluster name**	**Histological concepts represented in cluster**
0	Abnormal pigmentation	Carbon accumulating black anthracotic macrophages, artifactual pigmentations from residual brown stain, and from marker inks
27	Immune cells	Immune cells such as lymphocytes, plasma cells and macrophages.
33	Cancer stroma	Cancer-associated stroma
37	Fibroblast cells	Fibroblast cells
41	Cancer cells	Circular cancer cells

Figure 2: UMAP of 3072 SAE dimensions from the 1M model. Feature clusters are identified by HDBSCAN and are interpreted by manual inspection. Several clusters clearly associated with histological concepts are highlighted. For cancer and immune cell clusters, visualizations of top 3 patches that maximally activate the SAE dimension are shown.

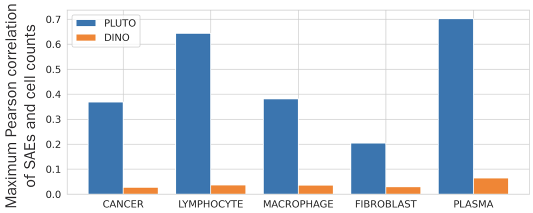


Figure 3: Pearson correlations of SAE dimensions of PLUTO and DINO models with counts of pathology-relevant cell types, showing much higher correlations of the PLUTO SAE dimensions with the cell count features.

We choose the same input patch size as PLUTO and SAE training methodology as in Section 2 to ensure a fair comparison.

To evaluate these two models, we extract human-interpretable features (HIFs) [19] quantifying the counts of cancer cells, plasma cells, lymphocytes, macrophages, and fibroblasts (Section 2.4), and correlate these features against dimensions in the SAE latents of the two models. We find SAE dimensions in PLUTO that strongly correlate with cell counts: plasma cells ($\rho = 0.7$), lymphocytes ($\rho = 0.63$), cancer cells ($\rho = 0.37$), macrophages ($\rho = 0.38$), and fibroblasts ($\rho = 0.21$). In contrast, SAE dimensions of DINO show weak association with these cell count features (Figure 3).

3.3 A small proportion of SAE features are universal

Previous work suggests that SAE dimensions are more likely to be useful (representing true monosemantic features in the real world) if they display *universality* (the same feature is discovered across independently trained SAE models [5]). We examine feature universality by comparing the SAE activations from the two independently trained SAE models (TCGA and 1M). We define "universality" as pairs of SAE features with Pearson's ρ above 0.5, where the correlation is computed between the activation of those two features across all images in the TCGA dataset. Using the Hungarian matching algorithm, we identify 5% (152) SAE dimensions exhibiting feature universality between

	Scanners					Stains			
	AT2	C9600	DP200	GT450	UFS	CD8	H&E	HER2 485	PD-L1
Plasma-SAE	0.517	0.508	0.505	0.510	0.509	0.510	0.509	0.509	0.509
Macrophage-SAE	0.513	0.537	0.511	0.497	0.574	0.504	0.508	0.509	0.495
Lymphocyte-SAE	0.513	0.537	0.511	0.497	0.574	0.504	0.508	0.509	0.495
Fibroblast-SAE	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Cancer-SAE	0.505	0.494	0.503	0.504	0.504	0.503	0.503	0.502	0.504

Table 1: AUROC of biological SAEs predicting scanners and stains. SAE dimensions representing cell types do not encode non-biological information. Test ROC scores for predicting scanners and stains are close to the chance level of 0.5

the two models. For example, SAE-1736 from the 1M model and SAE-2541 from the TCGA model are highly correlated ($\rho = 0.96$), and both represent plasma cells. Similarly, both SAE-1745 from the 1M model and SAE-1667 from the TCGA model ($\rho = 0.91$) represent anthracotic macrophages. This universality property of the SAEs suggests generalizability of the learned SAE features.

3.4 SAE latent dimensions representing biological features are robust to sources of variations like scanner and stain

In section 3.2, we have identified five SAE dimensions in the PLUTO-based SAE that correlate with cell count features (cancer cells, fibroblasts, lymphocytes, macrophages and plasma cells). To confirm that these features do not also encode non-biological information, we examine the predictive power of these SAEs for various stains or scanners.

For each SAE dimension that best correlates with a given cell type, we examine how well the activation of that dimension predicts 9 binary variables corresponding to 5 scanners and 4 stain types (Table 1). We fit linear logistic regression models on the train set (80%) for predicting these 9 variables, and evaluate the performance of each model on the test set (20%) using the ROC metric. ROC scores for these models are close to 0.5, showing that scanner and stain information cannot be linearly retrieved from the cell-type specific SAEs.

4 Monosemanticity in SAEs

In this section, we use two orthogonal approaches to study the usefulness of SAEs. First, we evaluate the degree of *monosemanticity* in the SAE latents - the ability of single neurons in the SAE latent space to represent single biological concepts.

As an orthogonal evaluation of SAEs, motivated by recent investigations of SAE neurons through probing [5, 31, 42], we construct sparse linear probes based on 8 HIFs, and test the performance of SAE latents and the original FM embeddings in predicting these interpretable concepts.

4.1 SAE features are more monosemantic than the FM embeddings they are trained on

Using biological interpretable features as probes, we evaluate and quantify the increase in monosemanticity in the SAE latents compared to PLUTO embedding space. We use a set of 5 cell count features and 3 nuclear features (mean nuclear size, orientation and eccentricity) derived using PathExplore (Section 2.4) to quantify monosemanticity (Section 2.5). By computing entropy for all SAE and FM embedding dimensions, we find a number of SAE dimensions with low entropy, forming a long tail of the entropy distribution (Fig 4A). These low-entropy dimensions are consistently seen in the 10 independently trained SAE models with different seeds, and are not observed in the FM embedding dimensions. The entropies of the SAE features are significantly lower than the FM features (p<0.001 for all 10 SAE-PLUTO comparisons, Mann-Whitney U test). While none of the FM features had $S < 0.6$, $0.8 \pm 0.1\%$ of the SAE features had entropy $S < 0.6$.

To see which interpretable feature these low-entropy SAE dimensions correspond to, we focus on neurons in the SAE latents with entropy $S < 0.6$, determine the concept that each neuron best correlates with, and count the number of monosemantic neurons that best correlate with each concept (Fig 4B). The total number of monosemantic dimensions in SAEs is higher than the PLUTO

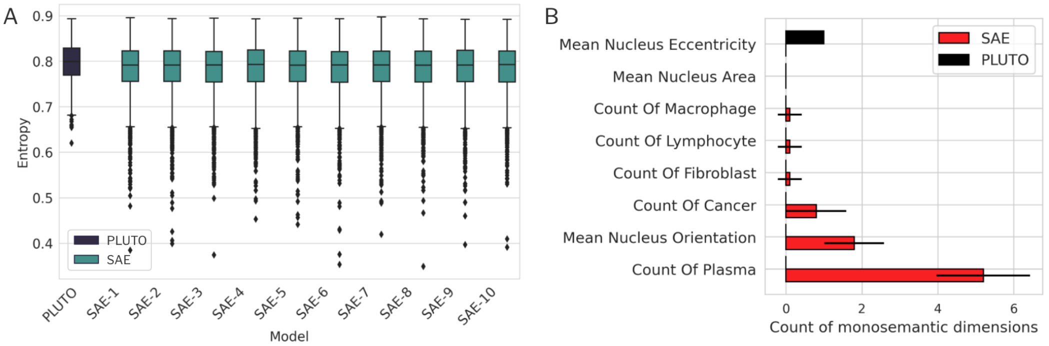


Figure 4: SAE features are more monosemantic than the FM features they are trained on. A) Entropy distribution of PLUTO embedding dimensions (left, gray box), and the embedding dimensions of the 10 independently trained SAEs (right, light green boxes). B) Number of monosemantic dimensions for each HIF in the SAE latent space (red), or PLUTO embedding space (black). Note that count of monosemantic dimensions for PLUTO is zero for 7/8 features.

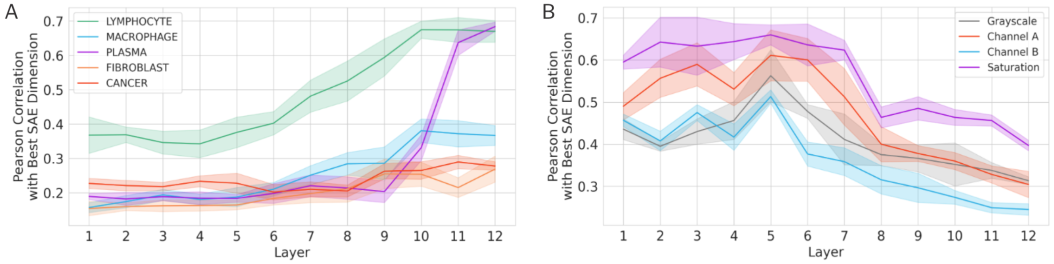


Figure 5: Monosemanticity emerges in later layers of PLUTO. A) SAE dimensions with the highest correlation with each cell count features across layers. B) Correlation of color features with SAE dimensions.

embedding on which it is trained, suggesting that the SAE transforms the highly polysemantic original embeddings into a space with higher monosemanticity.

We investigate SAE-1736 as an illustration of a monosemantic dimension in the SAE latent space. SAE-1736 highly correlates with plasma cell counts ($\rho = 0.70$) while showing minimal correlation ($\rho < 0.1$) with other cell types, suggesting high specificity of the activation. In contrast, no such monosemantic plasma cell feature is found in the PLUTO embedding space. The strongest plasma cell-associated PLUTO dimension, 148 ($\rho = 0.29$) also correlates with counts of other cell types.

4.2 Monosemanticity in SAE features emerges in later layers of the FM

We investigate how monosemanticity of SAEs evolves across layers by measuring the monosemanticity of SAE latent dimensions for models trained on CLS tokens from different layers of PLUTO. In earlier layers, SAE dimensions correlate with low-level color features such as intensity, hue and saturation (Figure 5B). Correlations of these SAE dimensions with cellular features are low ($\rho < 0.5$ for lymphocytes and $\rho < 0.3$ for the other four cell types). At later layers , association of SAE dimensions with color features decreases, while association with cell features increases (Figure 5A). The increase in association is cell type is robust across 10 independently trained SAEs with different random seeds. Furthermore, SAE neurons with low entropy ($S < 0.6$) start to emerge in layers 11 and 12 of the model.

4.3 Monosemantic behavior of SAEs generalizes to new datasets with unseen cancer types

We verify that our results in sections 4.1 and 4.2 generalize to an independent dataset. We extract FM embeddings and deploy the 10 trained SAEs on the CPTAC dataset, which includes two cancer types not included in TCGA (see Methods section 2.1). We confirm that (1) there are more monosemantic

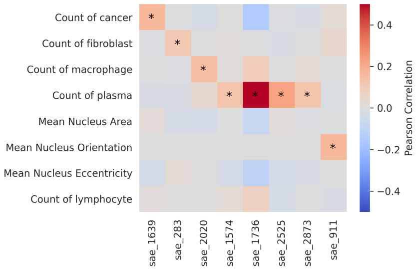


Figure 6: Illustration of all monosemantic SAE dimensions of the 1M model, and their correlations with 8 human-interpretable features. Stars indicate the features with the highest correlation for each dimension. Multiple monosemantic SAEs (1574, 1736, 2525 and 2873) can be seen correlating with plasma cell counts.

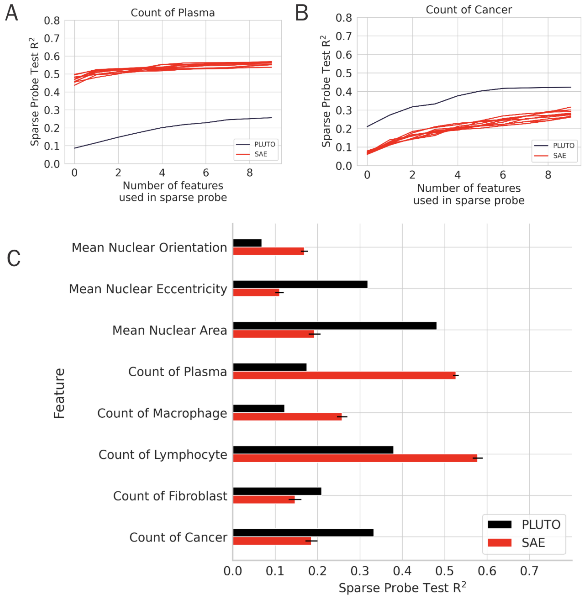


Figure 7: Performance of sparse probes trained on SAE latent embeddings is not always better than those trained on FM embeddings. A-B) Test R^2 of probes for predicting count of plasma and count of cancer from the SAE latent space (red), or the FM embeddings (black). Probes trained on SAE latents achieve higher test R^2 for predicting plasma counts, but not cancer cell counts. C) Sparse probe test R^2 for all 8 human-interpretable features for $k = 3$.

dimensions from the 10 independently trained SAEs than from the FM embeddings, and (2) neurons in the SAE latent space also correlate with cell count features, particularly in SAEs trained on embeddings from the later layers of the FM.

4.4 Partial monosemanticity limits SAE utility

Our results in Figure 4B suggests that not all interpretable features have an associated monosemantic SAE dimension. We perform a further investigation by analyzing the monosemantic SAE units of the 1M model and the human interpretable features they best correlate with (Fig 6). We find 4 features with associated monosemantic neurons (count of cancer, count of fibroblast, count of macrophage, mean nucleus orientation), and 3 features that do not correlate with any monosemantic SAE neurons (mean nucleus area, mean nucleus eccentricity, count of lymphocyte). Notably, plasma cell counts is correlated with multiple monosemantic SAE units, suggesting potential feature splitting. These results point to a key limitation of SAEs since the monosemantic SAE units don't have a one-to-one mapping with the HIFs.

4.5 SAE sparse probes have mixed performance as compared to FM embeddings

As an orthogonal approach for evaluating the utility of SAE in interpretability analyses, we use k-sparse probes (see Methods section 2.4) to quantify how well a small number of dimensions in SAEs (or dimensions in the original FM embedding space) can be used to predict the 8 HIFs (Fig 7). We find mixed evidence regarding the utility of the SAE latent space, as SAE embeddings show better performance than the FM embeddings in only 4/8 probes. These findings are consistent with a recent study that investigated SAE utility in large language models [42].

5 Limitations and future work

In this work, we restrict our analysis to a vanilla SAE. We leave the application of newer variations such as gated SAE and k-sparse SAE in pathology to future work. Similarly, analysis around how these findings translate to SAEs trained on other pathology foundation models can be the subject of further studies. The exploration of probing is limited to 8 features representing 5 different cell types. Future studies might examine the performance of SAEs when evaluated on a larger number of probes corresponding to a more diverse selection of biological features. Our evaluation is performed on oncology datasets only. Future studies should test the generalizability of the findings to other disease types.

6 Conclusion

We train the first sparse autoencoders on the embeddings of a pathology vision transformer model, and investigate the features represented in the embedding space of the model. Sparse autoencoder training enables the extraction of interpretable features corresponding to distinct biological characteristics, geometric features and image acquisition artifacts. Dimensions in the SAE latent space are more monosemantic than those in the FM embeddings, potentially facilitating downstream interpretability analyses. We find SAE dimensions that correlate with cell count features and are robust to non-biological factors like scanners and stains, suggesting the SAE has learned generalizable biological features. These learned biological features can be starting points for downstream interpretability analyses of the FM, similar to work in other domains [5, 31, 43, 44].

Through an evaluation of the SAE in two metrics (monosemanticity and predictive performance with sparse linear probing), we also highlight strengths and limitations of the SAE latent space for interpretability analyses. First, while SAE embeddings are much more monosemantic than original FM embeddings, monosemanticity can be partial, as there is not a one-to-one mapping between the monosemantic SAE dimensions and HIFs. Second, while sparse probes trained on SAE latents sometimes outperform those trained on FM embeddings in predicting some biological concepts, they underperform in many other cases (consistent with recent observations [42]). Future studies might explore methods for sparse autoencoder training that address these limitations. Overall, investigation of sparse features is a promising direction and motivates further work in discovering explainable, generalizable features of pathology foundation models.

References

[1] Christopher Olah. Mechanistic interpretability, variables, and the importance of interpretable bases, 2022.

[2] Nick Cammarata, Shan Carter, Gabriel Goh, Chris Olah, Michael Petrov, Ludwig Schubert, Chelsea Voss, Ben Egan, and Swee Kiat Lim. Thread: Circuits. *Distill*, 2020. https://distill.pub/2020/circuits.

[3] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021.

[4] Leonard Bereska and Efstratios Gavves. Mechanistic interpretability for ai safety – a review, 2024.

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

[6] Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models, 2023.

[7] Liv Gorton. The missing curve detectors of inceptionv1: Applying sparse autoencoders to inceptionv1 early vision, 2024.

[8] Nick Cammarata, Gabriel Goh, Shan Carter, Ludwig Schubert, Michael Petrov, and Chris Olah. Curve detectors. *Distill*, 2020. https://distill.pub/2020/circuits/curve-detectors.

[9] Alireza Makhzani and Brendan Frey. k-sparse autoencoders, 2014.

[10] Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Tom Lieberum, Vikrant Varma, János Kramár, Rohin Shah, and Neel Nanda. Improving dictionary learning with gated sparse autoencoders, 2024.

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

[12] Eric E Walk. The role of pathologists in the era of personalized medicine. *Archives of pathology & laboratory medicine*, 133(4):605–610, 2009.

[13] Anant Madabhushi and George Lee. Image analysis and machine learning in digital pathology: Challenges and opportunities. *Medical Image Analysis*, 33:170–175, 2016. 20th anniversary of the Medical Image Analysis journal (MedIA).

[14] Wouter Bulten, Hans Pinckaers, Hester van Boven, Robert Vink, Thomas de Bel, Bram van Ginneken, Jeroen van der Laak, Christina Hulsbergen-van de Kaa, and Geert Litjens. Automated deep-learning system for Gleason grading of prostate cancer using biopsies: a diagnostic study. *The Lancet Oncology*, 21(2):233–241, 2020.

[15] Gabriele Campanella, Matthew G Hanna, Luke Geneslaw, Allen Miraflor, Vitor Werneck Krauss Silva, Klaus J Busam, Edi Brogi, Victor E Reuter, David S Klimstra, and Thomas J Fuchs. Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature medicine*, 25(8):1301–1309, 2019.

[16] Dayong Wang, Aditya Khosla, Rishab Gargeya, Humayun Irshad, and Andrew H Beck. Deep learning for identifying metastatic breast cancer. *arXiv preprint arXiv:1606.05718*, 2016.

[17] Syed Ashar Javed, Dinkar Juyal, Harshith Padigela, Amaro Taylor-Weiner, Limin Yu, and Aaditya Prakash. Additive mil: Intrinsically interpretable multiple instance learning for pathology, 2022.

[18] Ming Y. Lu, Drew F. K. Williamson, Tiffany Y. Chen, Richard J. Chen, Matteo Barbieri, and Faisal Mahmood. Data efficient and weakly supervised computational pathology on whole slide images, 2020.

[19] James A Diao, Jason K Wang, Wan Fung Chui, Victoria Mountain, Sai Chowdary Gullapally, Ramprakash Srinivasan, Richard N Mitchell, Benjamin Glass, Sara Hoffman, Sudha K Rao, et al. Human-interpretable image features derived from densely mapped cancer pathology slides predict diverse molecular phenotypes. *Nature communications*, 12(1):1–15, 2021.

[20] Richard J Chen, Tong Ding, Ming Y Lu, Drew FK Williamson, Guillaume Jaume, Bowen Chen, Andrew Zhang, Daniel Shao, Andrew H Song, Muhammad Shaban, et al. Towards a general-purpose foundation model for computational pathology. *Nature Medicine*, 2024.

[21] Mingu Kang, Heon Song, Seonwook Park, Donggeun Yoo, and Sergio Pereira. Benchmarking self-supervised learning on diverse pathology datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3344–3354, June 2023.

[22] Jonas Dippel, Barbara Feulner, Tobias Winterhoff, Simon Schallenberg, Andreas Kunft Gabriel Dernbach, Stephan Tietz, Timo Milbich, Simon Heinke, Marie-Lisa Eich, Julika Ribbat-Idel, Rosemarie Krupar, Philipp Jurmeister, David Horst, Lukas Ruff, Klaus-Robert Müller, Frederick Klauschen, and Maximilian Alber. RudolfV: A Foundation Model by Pathologists for Pathologists, 2024.

[23] Eugene Vorontsov, Alican Bozkurt, Adam Casson, George Shaikovski, Michal Zelechowski, Siqi Liu, Kristen Severson, Eric Zimmermann, James Hall, Neil Tenenholtz, Nicolo Fusi, Philippe Mathieu, Alexander van Eck, Donghun Lee, Julian Viret, Eric Robert, Yi Kan Wang, Jeremy D. Kunz, Matthew C. H. Lee, Jan Bernhard, Ran A. Godrich, Gerard Oakley, Ewan Millar, Matthew Hanna, Juan Retamero, William A. Moye, Razik Yousfi, Christopher Kanan, David Klimstra, Brandon Rothrock, and Thomas J. Fuchs. Virchow: A Million-Slide Digital Pathology Foundation Model, 2023.

[24] Alexandre Filiot, Ridouane Ghermi, Antoine Olivier, Paul Jacob, Lucas Fidon, Alice Mac Kain, Charlie Saillard, and Jean-Baptiste Schiratti. Scaling Self-Supervised Learning for Histopathology with Masked Image Modeling. *medRxiv*, 2023.

[25] Frederick M. Howard, James Dolezal, Sara Kochanny, Jefree Schulte, Heather Chen, Lara Heij, Dezheng Huo, Rita Nanda, Olufunmilayo I. Olopade, Jakob N. Kather, Nicole Cipriani, Robert Grossman, and Alexander T. Pearson. The impact of digital histopathology batch effect on deep learning model accuracy and bias. *bioRxiv*, 2020.

[26] John N Weinstein, Eric A Collisson, Gordon B Mills, Kenna R Shaw, Brad A Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, and Joshua M Stuart. The cancer genome atlas pan-cancer analysis project. *Nature genetics*, 45(10):1113–1120, 2013.

[27] Ratna R. Thangudu, Michael Holck, Deepak Singhal, Alexander Pilozzi, Nathan Edwards, Paul A. Rudnick, Marcin J. Domagalski, Padmini Chilappagari, Lei Ma, Yi Xin, Toan Le, Kristen Nyce, Rekha Chaudhary, Karen A. Ketchum, Aaron Maurais, Brian Connolly, Michael Riffle, Matthew C. Chambers, Brendan MacLean, Michael J. MacCoss, Peter B. McGarvey, Anand Basu, John Otridge, Esmeralda Casas-Silva, Sudha Venkatachari, Henry Rodriguez, and Xu Zhang. Nci's proteomic data commons: A cloud-based proteomics repository empowering comprehensive cancer analysis through cross-referencing with genomic and imaging data. *Cancer Research Communications*, 4(9):2480–2488, 09 2024.

[28] Dinkar Juyal, Harshith Padigela, Chintan Shah, Daniel Shenker, Natalia Harguindeguy, Yi Liu, Blake Martin, Yibo Zhang, Michael Nercessian, Miles Markey, Isaac Finberg, Kelsey Luu, Daniel Borders, Syed Ashar Javed, Emma Krause, Raymond Biju, Aashish Sood, Allen Ma, Jackson Nyman, John Shamshoian, Guillaume Chhor, Darpan Sanghavi, Marc Thibault, Limin Yu, Fedaa Najdawi, Jennifer A. Hipp, Darren Fahy, Benjamin Glass, Eric Walk, John Abel, Harsha Pokkalla, Andrew H. Beck, and Sean Grullon. Pluto: Pathology-universal transformer, 2024.

[29] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *CoRR*, abs/2104.14294, 2021.

[30] AI Safety Foundation. Sparse autoencoder, 2024. Accessed: 2024-08-29.

[31] Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. Finding neurons in a haystack: Case studies with sparse probing. *arXiv preprint arXiv:2305.01610*, 2023.

[32] Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, Anthony Bau, and James Glass. What is one grain of sand in the desert? analyzing individual neurons in deep nlp models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6309–6317, 2019.

[33] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes, 2017. In *URL https://openreview. net/forum*, 2018.

[34] Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

[35] Nadir Durrani, Hassan Sajjad, Fahim Dalvi, and Yonatan Belinkov. Analyzing individual neurons in pre-trained language models. *arXiv preprint arXiv:2010.02695*, 2020.

[36] Miles Markey, Juhyun Kim, Zvi Goldstein, Ylaine Gerardin, Jacqueline Brosnan-Cashman, Syed Ashar Javed, Dinkar Juyal, Harshith Padigela, Limin Yu, Bahar Rahsepar, et al. Abstract b010: Spatially-resolved prediction of gene expression signatures in h&e whole slide images using additive multiple instance learning models. *Molecular Cancer Therapeutics*, 22(12_Supplement):B010–B010, 2023.

[37] John Abel, Suyog Jain, Deepta Rajan, Harshith Padigela, Kenneth Leidal, Aaditya Prakash, Jake Conway, Michael Nercessian, Christian Kirkup, Syed Ashar Javed, Raymond Biju, Natalia Harguindeguy, Daniel Shenker, Nicholas Indorf, Darpan Sanghavi, Robert Egger, Benjamin Trotter, Ylaine Gerardin, Jacqueline A. Brosnan-Cashman, Aditya Dhoot, Michael C. Montalto, Chintan Parmar, Ilan Wapinski, Archit Khosla, Michael G. Drage, Limin Yu, and Amaro Taylor-Weiner. Ai powered quantification of nuclear morphology in cancers enables prediction of genome instability and prognosis. *npj Precision Oncology*, 8(1):134, Jun 2024.

[38] Andreas M Tillmann, Daniel Bienstock, Andrea Lodi, and Alexandra Schwartz. Cardinality minimization, constraints, and regularization: a survey. *SIAM Review*, 66(3):403–477, 2024.

[39] Mathilde Caron, Hugo Touvron, Ishan Misra, Herv'e J'egou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[40] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

[41] Ross Wightman. Pytorch image models. `https://github.com/huggingface/pytorch-image-models`, 2019.

[42] Subhash Kantamneni, Joshua Engels, Senthooran Rajamanoharan, Max Tegmark, and Neel Nanda. Are sparse autoencoders useful? a case study in sparse probing. *arXiv preprint arXiv:2502.16681*, 2025.

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

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

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract clearly summarizes the work. Main contributions are included in the last part of the introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: A separate limitation section (section 5) is included.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: No theoretical results included.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.

- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All the methods used are clearly explained. Experimental results can be reproduced given the data.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The method section provides sufficient information to generate the code. The data cannot be shared for licensing.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.

- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: All the experimental setups are explained in Section 2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Statistical significance was reported where needed and rrror bars are included in figures.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: Compute resources were not included due to low relevance.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The authors have reviewed the NuerIPS Code of Ethics and confirmed conformity.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: The work is foundational research and the authors do not expect any significant societal impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: No such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The original papers were properly cited and data resources were referenced.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: No new assets were released.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The work does not involve crowdsourcing or human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

Justification: The work does not involve crowdsourcing or human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: No LLM used.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.