PULSE: Identifying Demonstration-Utility Features with Sparse Autoencoders

Chenduo Hao[1,†], **Chuanbao Gao**[1,†], **Pinjun Zeng**[1], **Jingze Zhu**[1], **Chonghan Liu**[2], **Zidong Liu**[3], **Xu Yang**[1,*]
[1]Southeast University
{220242441,220242345,213242848,220242456,101013120}@seu.edu.cn
[2]University of California, Los Angeles
khazzz1c@gmail.com
[3]School of Architecture, The University of Texas at Austin
zidong.liu.22@utexas.edu

Abstract

In-context learning is highly sensitive to demonstration choice, yet most methods select demonstrations using external query–demonstration similarity. Such criteria can miss model-specific signals: Similar demonstrations may activate different internal features and downstream behaviors. We introduce **PULSE** (**P**aired **U**tility **L**ocalization over **S**parse **E**ncodings), an SAE-based framework for identifying model-internal features associated with demonstration utility and using them for demonstration selection. Using a small labeled discovery set, PULSE samples candidate demonstration sets, measures their zero-shot-relative utility under the target model, and scores SAE features by how their activation differences align with utility differences. The top positive and negative coordinates form a sparse utility-localization vector. We use this vector in two complementary ways: as a signed score for controlled complete-set ranking, and as **PULSE-Retriever**, which converts its magnitude into a feature-relevance mask for scalable pool-scale retrieval. Across classification, generation, and reasoning benchmarks, PULSE-Retriever improves over the strongest baseline by 2–3 accuracy points, 0.6–0.9 BLEU-4, and 3.2 exact-match points, respectively, while controlled ranking validates the identified features encode a predictive set-level utility signal. Feature inspection and cross-dataset experiments suggest that the identified features capture task-relevant, dataset-conditioned patterns, yet retain utility signals that partially transfer across datasets. Our code is available at the GitHub repository.

1 Introduction

In-context learning (ICL) enables large language models (LLMs) to adapt to new tasks at inference time by conditioning on a small set of demonstrations, without any parameter updates [3]. Despite its practical success, ICL is notoriously sensitive to how demonstrations are configured: for the same query, some demonstrations substantially improve performance while others offer little benefit or even degrade it [50, 26, 45]. This sensitivity makes demonstration selection a central problem.

Early work observed that demonstrations similar to the query often work better, motivating retrieval methods based on lexical or semantic proximity [22, 25]. While useful, these methods capture only superficial similarity and leave much of the variation in demonstration utility unexplained. Later approaches introduced learnable retrievers trained from richer supervision, including language-model feedback or ranking signals [36, 19], and more recent methods optimize entire demonstration sets for

[†]Chenduo Hao and Chuanbao Gao contributed equally.
[*]Xu Yang is the corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

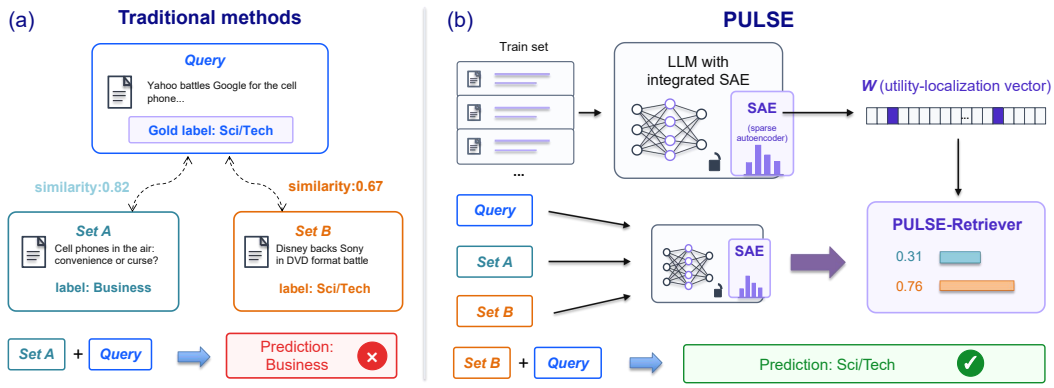


Figure 1: Overview of the motivation and workflow of PULSE. Traditional methods still study these signals mainly at the level of query–demonstration matching. In contrast, PULSE identifies utility-related internal features and uses them for demonstration selection.

diversity, coverage, or set-level quality [17, 46, 44, 33, 41]. The success of these methods suggests that good demonstrations depend on latent, non-obvious signals beyond superficial similarity.

However, as illustrated in Figure 1(a), existing methods study these signals primarily at the level of query–demonstration matching, but do not ask which internal features of the model give rise to those signals. This distinction matters because recent work shows that different LLMs exhibit different preferences over demonstrations [30, 39], utility cannot be determined solely by the external query–demonstration relation, it also depends on which internal features those demonstrations activate in the model. Mechanistic studies support this interpretation: intervening on a small fraction of in-context heads can substantially change ICL performance [1, 29, 47]. Taken together, these findings suggest that the key question is not only which demonstrations match the query, but also which internal features of the model make demonstrations useful.

The challenge, then, is that such utility-related features are difficult to isolate in dense activations. Neurons and activation directions are often polysemantic, so signals related to demonstration utility are entangled with many other factors in raw representations [14]. Sparse autoencoders (SAEs) provide a natural tool for addressing this challenge: prior work shows that SAE-style decompositions can recover sparse, interpretable features and can now be scaled and deployed broadly across models and layers [14, 2, 11, 20, 13]. This creates an opportunity to move from black-box demonstration scoring to explicit identification of the internal features associated with useful demonstrations.

Inspired by this, we propose **PULSE** (**P**aired **U**tility **L**ocalization over **S**parse **E**ncodings), an SAE-based framework that identifies utility-related internal features and uses them for demonstration selection, as shown in Figure 1(b). For each query, PULSE samples multiple candidate demonstration sets and forms pairs with different downstream utilities. It then identifies SAE features whose activation differences consistently track these utility differences, yielding a sparse utility-localization vector over SAE features. This vector emphasizes internal features predictive of demonstration utility while suppressing variation unrelated to ICL utility.

We then design a PULSE-Ranking experiment to test whether the identified features encode genuine internal utility signals. For each evaluation query, we rank randomly sampled demonstration sets by their alignment with the sparse utility-localization vector. Top-ranked sets consistently outperform lexical, embedding-based, and SAE-space baselines, confirming that the identified features distinguish useful contexts. Building on this, we design **PULSE-Retriever**, a pool-scale demonstration retriever to select demonstrations for each query. Because exhaustively scoring all k-shot subsets of a large support pool is combinatorially infeasible, PULSE-Retriever uses the sparse utility-localization vector as a feature-relevance mask, combining utility-weighted SAE similarity with standard SAE similarity to retrieve the final context. Experiments across classification, generation, and reasoning tasks validate that this approach improves demonstration selection across different datasets, shot settings, and backbone models. Furthermore, we explore what kinds of factors these identified features capture, and investigate the cross-task transferability of these features.

Overall, our main contributions are as follows:

- We reframe ICL demonstration selection as a model-internal sparse feature identification problem, and propose PULSE to identify SAE features whose activation shifts are associated with useful demonstration sets.
- We show, through the PULSE-Ranking experiment, that the identified features contain a predictive utility signal for ranking demonstration configurations, and introduce PULSE-Retriever, a pool-scale demonstration retriever to select demonstrations.
- We demonstrate across extensive experiments that PULSE improves demonstration selection over lexical, embedding-based, and SAE-space baselines. Furthermore, we explore the interpretability and transferability of the features identified by PULSE.

2 Related Work

In-context learning and demonstration selection. Large language models can perform in-context learning (ICL) by conditioning on a few demonstrations without parameter updates [3], but their performance is highly sensitive to the choice, order, and composition of demonstrations [50, 26, 24, 45]. Existing selection methods estimate demonstration utility from lexical or semantic similarity [22, 25], learned retrievers [36, 49, 19, 42], or set-level criteria such as diversity, coverage, representativeness, and context quality [17, 46, 44, 33, 41, 6]. Structure-aware retrieval further incorporates reasoning steps and their dependencies: GraphIC constructs thought graphs for queries and candidate examples and retrieves demonstrations through graph-based similarity [10]. These approaches are effective, but they mainly operate on external query–demonstration or demonstration-set signals.

Recent work shows that demonstration utility can be model-dependent: effective demonstrations may vary across inference models, motivating model-aware scoring or retrieval criteria [30, 39]. Mechanistic analyses further support this view, showing that ICL can depend on specific internal components, such as in-context heads [47], and on the balance between learning from demonstrations and retrieving internal knowledge [27]. Related work also studies internal representations of demonstration and reasoning effects. LIVE distills task information from demonstrations into learnable in-context vectors for visual question answering [31], while MimIC learns to approximate demonstration-induced hidden-state shifts in multimodal models [15]. CoT Vectors extract or learn compact internal representations for transferring and probing multi-step reasoning [18]. PULSE focuses on identifying SAE features associated with differences in demonstration-set utility and using these features to guide demonstration selection.

Sparse autoencoders for interpretable model features. A key obstacle to analyzing model-internal utility is that neural activations are dense and entangled. Under the superposition hypothesis, many features may be represented in overlapping activation directions, making neurons or raw directions difficult to interpret [9, 14]. Sparse autoencoders (SAEs) decompose activations into sparse feature representations that are often more interpretable [28, 2, 14]. Recent SAE scaling efforts and public SAE releases across model families further make SAE features a practical space for studying model-specific behavior [11, 38, 34, 20]. PULSE uses this space to identify features whose activation shifts distinguish higher-utility from lower-utility demonstration sets, and then uses the resulting utility-localization vector for demonstration selection.

Capability-based selection and feedback efficiency. Related selection problems also arise at the model and agent level. Constructive specifications use limited behavioral evaluations to build hierarchical capability representations for plug-and-play agent matching [23]. SaveRouter selectively acquires feedback for LLM routing and evaluates both supervision expenditure and subsequent serving-time savings [16]. These studies address model or agent selection. At the demonstration level, PULSE uses target-model feedback to identify utility-related SAE features for retrieval with a fixed target model.

3 Method

We propose PULSE, a framework that first estimates the utility-localization vector to identify utility-relevant SAE features, then validates the identified features through the PULSE-Ranking experiment, and finally uses PULSE-Retriever to select demonstrations. PULSE consists of three components, as shown in Figure 2: estimating a sparse utility-localization vector from paired demonstration

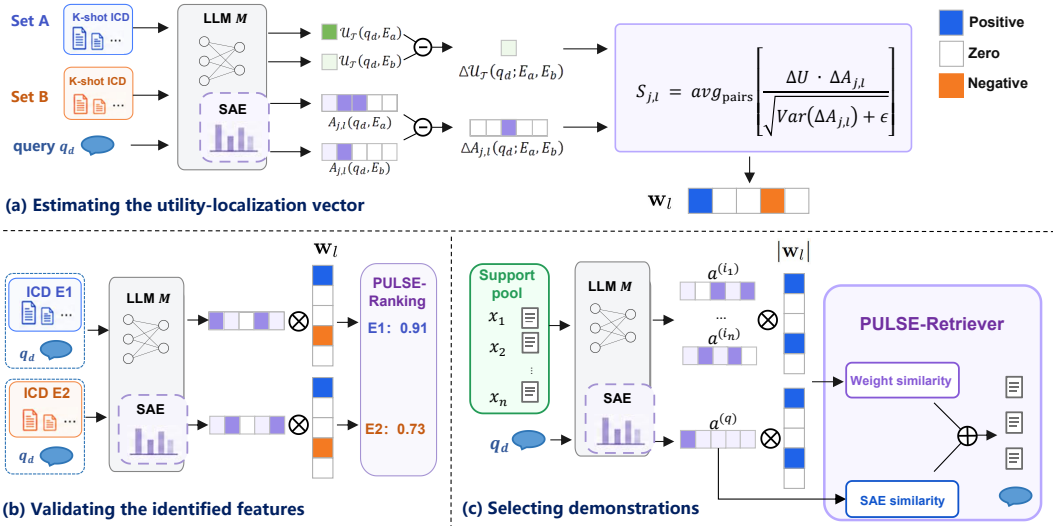


Figure 2: **Overview of PULSE.** Here, k-shot ICD denotes a k-example in-context demonstration set. (a) PULSE estimates a utility-localization vector by associating utility differences $\Delta\mathcal{U}$ with SAE activation differences ΔA across paired demonstration sets. (b) PULSE validates the identified features by ranking pre-sampled complete demonstration sets in the PULSE-Ranking experiment. (c) PULSE selects demonstrations using PULSE-Retriever, which uses the utility-localization vector as a feature-relevance mask for scalable pool-scale retrieval.

sets, validating the identified features with PULSE-Ranking experiment, and selecting demonstrations by PULSE-Retriever.

Setup. Let $\mathcal{S}$ be the supervised support pool, where each example is (x_i, z_i). Here z_i is the task target: a class label for classification (drawn from label set $\mathcal{Y}$), a reference sentence for CommonGen, or a supervised solution sequence for GSM8K. For classification, we write $y^\star(x) \in \mathcal{Y}$ for the gold label of input x. For an input x, q_x denotes the prediction query with the target hidden. A k-shot demonstration set is $E = \{(x_1, z_1), \ldots, (x_k, z_k)\} \subset \mathcal{S}$, and $P_\mathcal{T}(E, q_x)$ denotes the task prompt containing E followed by q_x; $P_\mathcal{T}(\varnothing, q_x)$ denotes the corresponding zero-shot prompt.

Let M be the backbone language model with output distribution $p_M(\cdot \mid \cdot)$, and a pretrained SAE attached to residual-stream layer l. We pass the prompt through M and the SAE, mean-pool SAE activations over prompt tokens, and denote the sparse feature vector by $\mathbf{A}_l(q_x, E) = (A_{j,l}(q_x, E))_j$.

Utility definition. PULSE uses labels only in a small discovery set to estimate which SAE features are associated with demonstration utility. For classification tasks, we use the gold-label margin:

$$\mathcal{G}_{\text{cls}}(q_x, E) = \log p_M(y^\star(x) \mid P_\mathcal{T}(E, q_x)) - \max_{y \in \mathcal{Y},\, y \neq y^\star(x)} \log p_M(y \mid P_\mathcal{T}(E, q_x)). \tag{1}$$

For generative tasks, we use the length-normalized teacher-forced likelihood of the target sequence $z^\star = (z_1^\star, \ldots, z_T^\star)$:

$$\mathcal{G}_{\text{gen}}(q_x, E) = \frac{1}{T} \sum_{t=1}^{T} \log p_M\left(z_t^\star \mid P_\mathcal{T}(E, q_x), z_{<t}^\star\right). \tag{2}$$

This utility proxy is used only to estimate the utility-localization vector; final performance is evaluated with the standard task metric. We define zero-shot-relative utility as

$$\mathcal{U}_\mathcal{T}(q_x, E) = \mathcal{G}_\mathcal{T}(q_x, E) - \mathcal{G}_\mathcal{T}(q_x, \varnothing), \tag{3}$$

where $\mathcal{G}_\mathcal{T}$ is instantiated as $\mathcal{G}_{\text{cls}}$ for classification and $\mathcal{G}_{\text{gen}}$ for generation.

Estimating the utility-localization vector. We estimate $\mathbf{w}_l$ using a labeled discovery set $\mathcal{D}_{\text{disc}}$ sampled from the training split and disjoint from evaluation queries. Each discovery example $d = (x_d, z_d)$ is treated as a query $q_d = q_{x_d}$. For each q_d, we sample N candidate demonstration sets from $\mathcal{S}$, excluding d itself if present, and form within-query pairs (E_a, E_b). Within-query pairing

controls for query-specific difficulty, so $\Delta\mathcal{U}_{\mathcal{T}}$ mainly reflects relative demonstration-set utility rather than differences across queries. For each pair, we compute the utility difference

$$\Delta\mathcal{U}_{\mathcal{T}}(q_d; E_a, E_b) = \mathcal{U}_{\mathcal{T}}(q_d, E_a) - \mathcal{U}_{\mathcal{T}}(q_d, E_b), \tag{4}$$

and the SAE activation difference

$$\Delta A_{j,l}(q_d; E_a, E_b) = A_{j,l}(q_d, E_a) - A_{j,l}(q_d, E_b). \tag{5}$$

PULSE scores each SAE feature by its activation-normalized association with utility differences. Let $n_{\text{pairs}} = |\mathcal{D}_{\text{disc}}| \cdot \binom{N}{2}$ denote the total number of unordered pairs, and let $\text{Var}(\Delta A_{j,l})$ denote the variance of $\Delta A_{j,l}$ computed globally over all pairs. The feature score is

$$S_{j,l} = \frac{\sum_{(q_d,E_a,E_b)} \Delta\mathcal{U}_{\mathcal{T}}(q_d; E_a, E_b)\, \Delta A_{j,l}(q_d; E_a, E_b)}{n_{\text{pairs}} \sqrt{\text{Var}(\Delta A_{j,l}) + \varepsilon}}, \tag{6}$$

where ε is a small stability constant. This score measures the normalized alignment between $\Delta A_{j,l}$ and $\Delta\mathcal{U}_{\mathcal{T}}$. We define $\mathbf{w}_l$ coordinate-wise by setting $w_{j,l} = S_{j,l}$ for features whose scores are among the K^+ largest positive or K^- most negative values, and $w_{j,l} = 0$ otherwise.

Validating the identified features. To test whether the identified features encode an internal utility signal, we design a PULSE-Ranking experiment. For each evaluation query, all methods rank the same pre-sampled complete demonstration sets. PULSE ranks a candidate set E by the alignment between its complete-context activation shift and the signed utility-localization vector:

$$s_{\text{rank}}(q_x, E) = \mathbf{w}_l^\top \big[\mathbf{A}_l(q_x, E) - \mathbf{A}_l(q_x, \varnothing)\big]. \tag{7}$$

A higher score indicates that the context activates features associated with higher utility. This experiment is a controlled validation of the identified features, not the practical pool-scale retriever.

Selecting demonstrations. Equation (7) scores complete demonstration sets with the signed utility-localization vector, but applying it to all $\binom{|\mathcal{S}|}{k}$ candidate sets is combinatorially infeasible. We therefore design PULSE-Retriever as a scalable proxy that ranks individual support examples before constructing the final k-shot context.

To adapt the set-level signal to pointwise retrieval, we use $|\mathbf{w}_l|$ as a feature-relevance mask. In complete-set scoring, the sign of each weight captures the direction of association between activation shifts and utility differences, while its magnitude measures the strength of this association. For retrieval, the magnitude mask weights both positive- and negative-weight features according to their utility relevance when comparing query and candidate representations. Let $\mathbf{a}^{(q)} = \mathbf{A}_l(q_x, \varnothing)$ and $\mathbf{a}^{(i)} = \mathbf{A}_l(q_{x_i}, \varnothing)$ be zero-shot SAE encodings of the query and candidate input, both with targets hidden. The PULSE-masked similarity is

$$s_{\text{mask}}(q_x, x_i) = \cos\big(|\mathbf{w}_l| \odot \mathbf{a}^{(q)},\ |\mathbf{w}_l| \odot \mathbf{a}^{(i)}\big). \tag{8}$$

We blend this masked similarity with standard SAE-space cosine similarity:

$$s_{\text{blend}}(q_x, x_i) = (1-\beta)\, \hat{s}_{\text{mask}}(q_x, x_i) + \beta\, \hat{s}_{\cos}(q_x, x_i), \tag{9}$$

where $s_{\cos}(q_x, x_i) = \cos(\mathbf{a}^{(q)}, \mathbf{a}^{(i)})$, $\hat{\cdot}$ denotes per-query z-score normalization, and $\beta \in [0, 1]$. The final k-shot context is greedily selected from a top-L shortlist with a redundancy penalty in SAE space. Thus, PULSE-Retriever uses the identified internal features as a scalable retrieval proxy, rather than as an exact decomposition of complete-context utility.

4 Experiments

4.1 Experiment Setup

Models and SAEs. Because PULSE identifies utility-related features in SAE feature space, we evaluate it on models with publicly available SAEs: Gemma2-2B [37] and Llama3.1-8B [12]. Unless otherwise specified, experiments use both backbones; GSM8K is evaluated only with Llama3.1-8B. Additional 4-shot evaluations on Qwen3-8B and Gemma2-9B extend the evaluation to three model families spanning 2B–9B parameters (Appendix F).

Table 1: **Candidate-set scoring accuracy (%) on Gemma2-2B and Llama3.1-8B.** Downstream accuracy after using each score to select the top-1 candidate demonstration set. The Avg. columns report the average across four datasets under each shot setting.

Backbone	Method	AGNews				REST14				LAP14				EMOC				Avg.			
		1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8
Gemma2-2B	Random	40.62	66.99	75.00	77.93	60.94	65.82	73.83	73.05	54.21	65.44	76.03	73.87	44.92	53.52	54.66	60.16	50.17	62.94	69.89	71.25
	Lex-Sim	43.75	68.75	77.34	82.03	66.41	72.46	78.71	75.39	64.79	70.84	78.40	77.75	52.93	54.88	56.64	60.94	56.97	66.73	72.77	74.03
	SBERT	55.47	77.15	78.12	82.23	60.74	69.14	76.17	73.83	66.52	71.49	75.81	77.11	51.56	55.86	56.05	61.33	58.57	68.41	71.54	73.62
	CEIL	55.47	71.48	78.12	78.12	60.74	69.73	76.76	75.59	66.52	70.41	79.70	78.19	51.56	**56.64**	56.25	60.94	58.57	67.06	72.71	73.21
	KNN-SAE	54.69	77.54	76.95	81.84	63.67	69.53	78.12	76.17	65.87	72.35	79.05	76.24	48.63	52.93	58.01	60.55	58.22	68.09	73.03	73.70
	PULSE-Ranking	**57.03**	**78.12**	**82.03**	**82.62**	**75.59**	**77.34**	**80.47**	**78.71**	**73.43**	**74.95**	**81.86**	**79.48**	**54.69**	55.66	**60.94**	**61.52**	**65.18**	**71.52**	**76.32**	**75.58**
Llama3.1-8B	Random	43.95	60.16	73.24	82.23	75.39	74.41	76.76	77.15	77.11	76.89	79.91	80.35	45.12	57.03	60.16	63.28	60.39	67.12	72.52	75.75
	Lex-Sim	53.32	66.99	74.61	75.20	76.17	77.34	79.49	77.93	**78.19**	78.19	79.05	79.27	50.20	58.59	60.94	62.50	64.47	70.28	73.52	73.72
	SBERT	59.77	74.02	78.32	81.25	75.78	74.80	78.52	78.12	77.97	78.83	78.40	79.05	55.47	60.35	63.28	64.45	67.25	72.00	74.63	75.72
	CEIL	59.77	75.59	78.91	82.42	75.78	76.37	**80.08**	78.71	77.97	79.05	78.83	79.70	55.47	60.74	66.60	67.77	67.25	72.94	76.10	77.15
	KNN-SAE	55.66	74.41	80.47	83.20	74.61	74.61	78.12	78.52	76.67	**79.70**	79.70	78.40	50.98	59.77	60.94	64.84	64.48	72.12	74.81	76.24
	PULSE-Ranking	**62.50**	**82.03**	**84.38**	**85.94**	**76.56**	**80.47**	**80.08**	**80.27**	77.97	**79.70**	**80.13**	**80.56**	**56.25**	**64.06**	**67.58**	**69.34**	**68.32**	**76.56**	**78.04**	**79.03**

Datasets and metrics. We evaluate on six benchmarks spanning classification, generation, and reasoning. For classification, we use AGNews [48] (4-class topic classification), REST14 and LAP14 (3-class aspect-based sentiment) [32], and EMOC (4-class emotion recognition) [4], reporting classification accuracy. For generation, we use CommonGen [21], reporting BLEU-4. For reasoning, we use GSM8K [7], reporting exact match accuracy. Across all tasks, utility-related features are identified only from the training split, and held-out test data are used for ICL evaluation.

Protocol. PULSE estimates $\mathbf{w}_l$ from $|\mathcal{D}_{\text{disc}}| = 64$ discovery queries sampled from the training split and disjoint from evaluation queries, with $N = 32$ candidate demonstration sets per query. We evaluate the identified features in two settings: **PULSE-Ranking experiment**, a controlled validation where all methods rank the same pre-sampled complete k-shot sets, and **PULSE-Retriever**, a practical pool-scale retriever that uses the same features as a feature-relevance mask. We further evaluate PULSE-Retriever on CommonGen and GSM8K for generation and reasoning. Unless otherwise specified, $K^+ = K^- = 512$ for Gemma2-2B and $K^+ = K^- = 2048$ for Llama3.1-8B; PULSE-Retriever uses $L = 50$ and $\lambda_r = 0.3$. All hyperparameters are calibrated on a held-out validation split and then fixed before evaluation on the test set.

Baselines. We compare PULSE against five baselines covering random, lexical, dense-embedding, SAE-space, and diversity-aware selection: **Random** (uniform sampling); **Lex-Sim** (Jaccard lexical similarity); **SBERT** [35], cosine similarity over `all-MiniLM-L6-v2` [40] embeddings; **KNN-SAE**, cosine similarity in SAE feature space; and **CEIL** [46], DPP-based diversity selection over pretrained embeddings. Since PULSE uses a small labeled discovery set, we additionally compare against an EPR-style learned retriever [36] under two feedback allocations, including one that matches PULSE in both the number of discovery queries and the number of scored candidate contexts (Appendix G). A singleton-feedback PULSE variant further examines the role of supervision granularity.

4.2 Main Results

PULSE-Ranking Experiment. To evaluate whether the utility-localization vector learned by PULSE captures a set-level utility signal, we design a PULSE-Ranking experiment, with results shown in Table 1. Across 32 dataset–shot–backbone cells, PULSE-Ranking achieves the best or tied-best accuracy in 30 cases, outperforming lexical similarity, sentence embeddings, SAE nearest neighbors, and diversity-aware selection. Because all methods rank identical pre-sampled candidate configurations, this directly shows that PULSE provides a stronger score for identifying high-utility demonstration sets. The gain over KNN-SAE is substantial—3.89 points on Gemma2-2B and 3.58 points on Llama3.1-8B—indicating that the improvement stems from the utility-related SAE feature weights identified by PULSE, not merely from using SAE representations. These results validate that the identified features encode a predictive set-level utility signal: they distinguish higher-utility from lower-utility complete demonstration configurations under a controlled ranking protocol.

Pool-Scale Demonstration Retrieval. We evaluate the effectiveness of PULSE-Retriever, the pool-scale demonstration retriever derived from PULSE, with results reported in Table 2. Unlike the candidate-set scoring setup, this setting does not pre-enumerate complete k-shot configurations; instead, each method retrieves individual examples from a large pool to form the final context. PULSE-

Table 2: **Pool-scale retrieval accuracy (%) on Gemma2-2B and Llama3.1-8B.** Downstream accuracy after retrieving individual examples from the support set and composing the final k-shot context. The Avg. columns report the average across four datasets under each shot setting.

Backbone	Method	AGNews				REST14				LAP14				EMOC				Avg.			
		1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8
Gemma2-2B	Random	41.80	66.99	72.66	78.52	57.62	64.65	70.90	72.46	56.59	60.91	68.25	71.71	41.60	54.69	56.25	60.94	49.40	61.81	67.02	70.91
	Lex-Sim	68.16	78.32	79.88	86.13	64.45	66.80	73.63	76.76	70.63	75.16	74.30	77.54	62.70	62.50	65.82	71.68	66.48	70.69	73.41	78.03
	SBERT	76.76	78.52	79.49	84.18	65.62	66.99	71.68	74.22	68.47	73.87	75.81	78.62	63.48	64.65	66.80	72.46	68.58	71.01	73.45	77.37
	CEIL	76.76	78.91	81.25	87.11	65.62	67.97	75.39	76.95	68.47	76.03	76.03	78.83	63.48	65.62	66.02	71.09	68.58	72.13	74.67	78.50
	KNN-SAE	79.69	78.71	81.05	87.50	**66.99**	65.43	75.59	75.59	70.63	76.46	75.16	77.75	50.20	60.74	66.80	**72.46**	66.88	70.33	74.65	78.32
	PULSE-Retriever	**81.25**	**82.62**	**85.94**	**88.09**	64.65	**71.68**	**77.73**	**77.73**	**72.79**	**77.11**	**77.32**	**80.35**	**63.87**	**65.82**	**68.36**	72.27	**70.64**	**74.31**	**77.34**	**79.61**
Llama3.1-8B	Random	50.39	69.92	81.45	83.40	75.00	75.59	76.37	77.54	77.11	77.11	77.32	77.32	48.83	55.47	62.11	65.23	62.83	69.52	74.31	75.87
	Lex-Sim	69.53	81.84	87.70	87.50	75.20	78.12	79.30	79.10	**77.75**	79.05	80.99	81.43	61.91	69.92	62.70	67.19	71.10	77.23	77.67	78.81
	SBERT	80.08	80.86	86.72	87.70	75.59	80.08	80.47	80.27	71.06	80.35	81.43	81.43	**65.23**	68.95	65.23	69.34	72.99	77.56	78.46	79.68
	CEIL	80.08	85.16	85.55	88.48	75.59	80.66	79.88	80.47	71.06	78.83	81.64	81.21	**65.23**	71.09	68.95	74.22	72.99	78.94	79.00	81.10
	KNN-SAE	**81.25**	84.77	86.13	88.09	75.39	76.56	80.66	81.25	68.03	**81.86**	**82.72**	82.29	61.13	68.36	66.02	73.83	71.45	77.89	78.88	81.36
	PULSE-Retriever	79.88	**86.52**	**88.09**	**89.84**	**77.34**	**81.45**	**81.84**	**81.45**	76.03	80.56	82.07	**82.72**	64.26	**71.29**	**71.09**	**75.78**	**74.38**	**79.96**	**80.77**	**82.45**

Table 3: Generation and reasoning results. Scores are BLEU-4 for CommonGen and exact match accuracy for GSM8K. The Avg. columns report the average over 1-, 2-, 4-, and 8-shot settings.

Method	CommonGen / Gemma2-2B / BLEU-4					CommonGen / Llama3.1-8B / BLEU-4					GSM8K / Llama3.1-8B/ EM				
	1	2	4	8	Avg.	1	2	4	8	Avg.	1	2	4	8	Avg.
Random	8.42	8.35	8.10	8.86	8.43	9.02	9.22	9.64	9.44	9.33	42.61	47.92	49.20	50.72	47.61
Lex-Sim	8.22	7.74	9.20	10.56	8.93	9.68	8.91	9.19	9.16	9.24	38.29	48.67	50.04	52.31	47.33
SBERT	8.41	8.36	9.08	9.82	8.92	9.41	9.84	10.21	9.91	9.84	39.73	49.20	43.82	52.92	46.42
KNN-SAE	9.00	8.34	8.93	7.55	8.46	8.91	9.64	9.96	9.71	9.56	39.12	46.10	46.93	53.53	46.42
CEIL	8.41	8.62	10.07	9.24	9.09	9.41	9.68	10.01	9.47	9.64	39.73	50.42	50.80	52.08	48.26
PULSE-Retriever	**9.05**	**9.34**	**11.09**	**10.59**	**10.02**	**9.91**	**10.91**	**10.90**	**10.09**	**10.45**	**43.37**	**50.72**	**54.66**	**56.94**	**51.42**

Retriever achieves the best accuracy in 25 out of 32 dataset–shot–backbone cells, demonstrating that the utility-related features identified by PULSE can be converted into a practical retrieval signal.

Compared to KNN-SAE, PULSE-Retriever improves average accuracy by 2.93 points on Gemma2-2B and 1.99 points on Llama3.1-8B. These results demonstrate the effectiveness of utility-guided SAE feature weighting for pool-scale demonstration retrieval. Appendix K reports additional results across three random seeds.

Generalization to Generation and Reasoning. Beyond classification datasets, we further evaluate PULSE-Retriever on generation (CommonGen) and reasoning (GSM8K) tasks. Unlike the previous classification tasks, these benchmarks require open-ended generation or multi-step mathematical reasoning, offering a stronger test of whether the utility-related features identified by PULSE can support broader demonstration selection. As reported in Table 3, PULSE-Retriever achieves the best result in all 12 setting–shot cells: on CommonGen, it improves over the strongest baseline by $+0.93$ BLEU-4 on Gemma2-2B and $+0.61$ BLEU-4 on Llama3.1-8B when averaged over shots; on GSM8K, it improves by $+3.16$ exact match. These results show the utility-related internal features identified by PULSE are not limited to classification but also benefit generation and reasoning tasks.

4.3 Ablation Studies

We ablate the SAE basis and the identified PULSE features in the 4-shot retriever setting. Appendix I further examines sensitivity to the number of selected features, the sparse–dense blend weight β, the number of discovery queries, and SAE dictionary width. Appendix H evaluates whether a utility-localization vector identified at one shot count can be reused at other shot counts.

Do SAE features provide a better internal basis?

To test whether SAE features provide a better basis for PULSE identification, we replace them with two alternatives while keeping the paired identification objective and retrieval pipeline unchanged: Raw uses the mean-pooled hidden state at the same layer, and PCA uses principal components of those hidden states. As shown in Table 4, SAE features outperform both alternatives on all four classification datasets, improving the 4-shot classification average by $+3.47$ points over PCA and $+3.96$ points over Raw. On CommonGen, where performance is measured by BLEU-4 rather than accuracy, SAE also outperforms PCA and Raw, improving from 10.23 and 10.34 to 11.09. The Raw control is especially diagnostic: applied directly to entangled hidden dimensions, the paired identification objective alone is insufficient. Sparse SAE coordinates therefore provide a more effective feature space for isolating utility-related internal features than either raw residual dimensions or variance-based PCA components.

Table 4: SAE-basis ablation on Gemma2-2B in the 4-shot pool-scale retrieval setting.

Dataset	SAE	PCA	Raw
AGNews	**85.94**	81.25	80.47
REST14	**77.73**	75.78	76.76
LAP14	**77.32**	73.22	74.95
EMOC	**68.36**	65.23	61.33
Cls. Avg.	**77.34**	73.87	73.38
CommonGen	**11.09**	10.23	10.34

Are the PULSE-identified features useful for retrieval? We evaluate PULSE-MASK, a pure feature-masked retriever that uses $|\mathbf{w}_l|$ without the standard SAE cosine component in PULSE-Retriever. As shown in Table 5, PULSE-MASK clearly outperforms ANTI-RETRIEVAL, which uses the same mask but selects the lowest-scoring pool items, improving average accuracy from 48.52 to 73.46 ($+24.94$ points). This shows that the PULSE-induced retrieval ranking is highly informative. PULSE-MASK also outperforms BOTTOM-FEATS and SPARSE-RANDOM by $+2.15$ and $+1.32$ points on average, indicating that the gain comes from the identified utility-related features rather than sparsity alone. Finally, its performance is comparable to PULSE-NOTOPK (73.46 vs. 73.36), showing that the selected sparse feature subset retains most of the useful retrieval signal in the broader SAE feature space.

5 Discussion

The results in Section 4.2 show that PULSE improves both candidate-set scoring and pool-scale demonstration retrieval. These gains indicate that the utility-related internal features identified by PULSE provide a stronger selection criterion than external query–demonstration similarity alone. We now ask two questions to better understand these features: (1) What kinds of internal factors do high-weight PULSE features capture? (2) To what extent is the identified feature structure partially transferable across tasks? We first inspect representative high-weight features, and then use cross-dataset transfer to test whether the identified feature structure contains components that generalize beyond the dataset on which it is estimated. The following analyses focus on the classification setting, where feature inspection and cross-dataset transfer can be compared under a shared evaluation protocol. Additional diagnostic analyses, including full top-feature tables and feature-overlap statistics, are provided in Appendix D and Appendix E.

Table 5: Controls for the PULSE feature mask on Gemma2-2B in the 4-shot pool-scale retrieval setting. PULSE-MASK uses only the magnitude of the PULSE-identified feature vector, without the standard SAE cosine component used in PULSE-Retriever. ANTI-RETRIEVAL uses the same mask but selects the lowest-scoring pool items. SPARSE-RANDOM uses randomly selected sparse features, BOTTOM-FEATS uses features with the smallest absolute identification scores, and PULSE-NOTOPK removes the top-feature restriction. Accuracy is reported as a percentage.

Dataset	PULSE-MASK	ANTI-RETRIEVAL	SPARSE-RANDOM	BOTTOM-FEATS	PULSE-NOTOPK
AGNews	**83.01**	51.95	82.81	82.03	81.64
REST14	76.17	63.67	75.39	75.00	**78.12**
LAP14	**74.51**	48.38	72.14	70.41	74.29
EMOC	**60.16**	30.08	58.20	57.81	59.38
Avg.	**73.46**	48.52	72.14	71.31	73.36

Table 6: Representative high-$|w|$ PULSE features. P1/N1 denotes the top positive or negative feature after activation filtering; the sign reflects association with paired utility differences.

Dataset	Feature	Rank	Top-activating snippets	Interpretation / Role
AGNews	#8186	P1	"Open Source: Balancing Innovation and Risk" "Citigroup ... institutional investors" "MFS Institutional Emerging Equities Fund"	Institutional and technology-industry framing; task-conditioned topic anchor.
REST14	#7875	P1	"Aspect: desserts; got favorable reviews" "Aspect: service; good things to say" "Aspect: staff; good things to say"	Aspect-specific evaluative expression; aspect–opinion binding.
LAP14	#15777	N1	"Aspect: works; Toshiba works great" "Aspect: power; power, memory and speed" "Aspect: memory; power, memory and speed"	Generic capability or off-aspect framing; lower-utility pattern.
EMOC	#7063	P1	"make others cry / i know whats wrong now" "give love to others ... happy always" "especially yours / mine is so so / amazing"	Self/other dialogue with affective reasoning or topical pivot; task-conditioned emotion cue.

5.1 What Do High-Weight PULSE Features Capture?

Because PULSE identifies utility-related features in SAE space, we can inspect the features that receive the largest absolute weights. This analysis does not claim that individual features causally mediate ICL performance; it asks whether the features identified by PULSE correspond to recognizable factors associated with demonstration utility. Table 6 reports the highest-$|w|$ feature for each dataset, together with its top-activating snippets and a post-hoc interpretation.

On AGNews, a topic classification task, the highest-weight feature activates on technology and industry-news phrasing (e.g., "Open Source," "institutional investors"), acting as a topic anchor that directly matches the dataset's category structure. On REST14, an aspect-based sentiment task for restaurant reviews, the selected feature captures aspect-specific evaluative expressions such as "desserts; got favorable reviews," reflecting an aspect–opinion binding. On EMOC, an emotion recognition task over dialogues, the feature fires on fragments containing affective reasons and topical pivots (e.g., "make others cry / i know whats wrong now"), both relevant cues for emotion classification. LAP14, an aspect-based sentiment task for laptop reviews, is especially instructive: its highest-$|w|$ feature has a negative weight despite top-activating snippets containing positive phrases such as "works great" and "power, memory and speed." Under the paired identification objective, a negative weight means that higher activation is associated with *lower* utility, suggesting generic or off-aspect phrasing rather than negative sentiment. This is consistent with PULSE-Retriever's retrieval design, where $|\mathbf{w}_l|$ serves as a feature-relevance mask: both positive- and negative-weight features are treated as utility-informative, differing only in the direction of association.

Overall, high-weight PULSE features capture recognizable internal factors—topic anchors, aspect–opinion bindings, affective cues, and off-aspect patterns—but these examples alone do not imply that all such features are task-specific. The next subsection tests whether this identified feature structure also contains components that transfer across tasks.

5.2 Cross-Dataset Transfer

Cross-dataset transfer tests whether PULSE-identified features remain useful beyond the dataset on which they are identified. Since each utility-localization vector is estimated from demonstration-utility differences on one dataset, its high-weight features may include dataset-specific information and partially transferable context-use factors. If the vector only captured source-specific surface patterns, it should degrade toward generic retrieval baselines when applied to another dataset.

Table 7 reports single-source transfer and a leave-one-out variant, $\text{LOO}_\cup$ (Leave-One-Out Union): for each target dataset, this variant excludes that target from the feature-identification stage and takes the union of PULSE-identified features from the remaining three source datasets, thereby testing whether utility-relevant features discovered on other tasks can substitute for target-specific ones. The results show the identified features are partially transferable. Averaged over all off-diagonal source–target pairs, single-source transfer reaches 76.05, only 1.29 points below the within-dataset

Table 7: Cross-dataset transfer of PULSE utility-localization vectors in the 4-shot retrieval setting on Gemma2-2B. Scores are accuracy (%). *Single-source*: a utility-localization vector learned on the source dataset is applied directly to the target dataset. *LOO*$_\cup$: for each target dataset, the target excludes the dataset used for feature identification, and the PULSE-identified features from the three non-target datasets are combined by union. *Base*$_{\max}$: best non-PULSE baseline per target. Δ_{Base} = LOO$_\cup$ − Base$_{\max}$; Δ_{Self} = LOO$_\cup$ − Self.

	Single-source transfer				**LOO transfer vs. baseline**			**Target PULSE**	
Target	AGNews	REST14	LAP14	EMOC	LOO$_\cup$	Base$_{\max}$	Δ_{Base}	Self	Δ_{Self}
AGNews	–	84.77	83.59	85.55	83.59	81.25	$+2.34$	85.94	-2.35
REST14	76.56	–	79.69	78.52	77.73	75.59	$+2.14$	77.73	0.00
LAP14	78.19	74.73	–	76.46	77.54	76.03	$+1.51$	77.32	$+0.22$
EMOC	60.94	66.80	66.80	–	67.97	66.80	$+1.17$	68.36	-0.39
Avg.	71.90	75.43	76.69	80.18	76.71	74.92	$+1.79$	77.34	-0.63

average of 77.34. LOO$_\cup$ is competitive with the target-trained vector: it obtains 76.71 on average, only 0.63 points below Self, while outperforming the best non-PULSE baseline by $+1.79$ points.

Transfer, however, is not uniform. LOO$_\cup$ is 2.35 points below Self on AGNews, suggesting that topic classification relies more on target-specific utility information. In contrast, it matches Self on REST14, slightly exceeds it on LAP14, and is only 0.39 points lower on EMOC, indicating stronger partial transfer among sentiment- and emotion-style tasks. Together with the small cross-dataset feature overlap and mixed signs reported in Appendix E, these results argue against a single task-independent utility direction. Instead, PULSE identifies a mixture of partially transferable and task-conditioned utility-related internal features.

6 Conclusion

We presented PULSE, an SAE-based framework for identifying model-internal sparse features associated with in-context demonstration utility. From paired comparisons of candidate demonstration sets, PULSE estimates a sparse utility-localization vector over SAE features. The PULSE-Ranking experiment validates that these identified features encode internal utility signals through controlled complete-set ranking, while PULSE-Retriever uses them as a feature-relevance mask for scalable pool-scale demonstration retrieval. Experiments across classification, generation, and reasoning tasks show that PULSE consistently improves demonstration selection over lexical, embedding-based, SAE-space, and diversity-aware baselines. Ablations and analyses show that the gains come from utility-aware sparse feature identification, and that the identified features are interpretable and partially transferable. These results support our claim that ICL demonstration selection benefits from explicit model-internal utility-related features, not only external query–demonstration similarity. Extending PULSE to multimodal ICL is a possible direction; such an extension could also test whether the identified utility signals remain reliable under misleading visual inputs, a robustness setting evaluated by MVI-Bench [5].

Acknowledgments and Disclosure of Funding

This work was supported by the National Natural Science Foundation of China (Grant No. 62576091); the Jiangsu Province Carbon Peak Carbon Neutrality Science and Technology Innovation Special Fund Project (Grant No. BT2025029); the Southeast University Big Data Computing Center; and the Southeast University Kunpeng & Ascend Center of Cultivation. The authors declare no competing interests.

References

[1] Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? investigations with linear models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*, 2023.

[2] 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. https://transformer-circuits.pub/2023/monosemantic-features/index.html.

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[4] Ankush Chatterjee, Kedhar Nath Narahari, Meghana Joshi, and Puneet Agrawal. Semeval-2019 task 3: Emocontext contextual emotion detection in text. In *Proceedings of the 13th international workshop on semantic evaluation*, pages 39–48, 2019.

[5] Huiyi Chen, Jiawei Peng, Dehai Min, Changchang Sun, Kaijie Chen, Yan Yan, Xu Yang, and Lu Cheng. Mvi-bench: A comprehensive benchmark for evaluating robustness to misleading visual inputs in lvlms. *arXiv preprint arXiv:2511.14159*, 2025.

[6] Huiyi Chen, Jiawei Peng, Kaihua Tang, Xin Geng, and Xu Yang. Enhancing multimodal in-context learning for image classification through coreset optimization. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 5130–5139, 2025.

[7] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[8] Boyi Deng, Xu Wang, Yaoning Wang, Yu Wan, Yubo Ma, Baosong Yang, Haoran Wei, Jialong Tang, Huan Lin, Ruize Gao, et al. Qwen-scope: Turning sparse features into development tools for large language models. *arXiv preprint arXiv:2605.11887*, 2026.

[9] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022.

[10] Jiale Fu, Yaqing Wang, Simeng Han, Jiaming Fan, and Xu Yang. Graphic: A graph-based in-context example retrieval model for multi-step reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 30753–30761, 2026.

[11] Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. *ArXiv preprint*, 2024.

[12] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[13] Zhengfu He, Wentao Shu, Xuyang Ge, Lingjie Chen, Junxuan Wang, Yunhua Zhou, Frances Liu, Qipeng Guo, Xuanjing Huang, Zuxuan Wu, et al. Llama scope: Extracting millions of features from llama-3.1-8b with sparse autoencoders. *arXiv preprint arXiv:2410.20526*, 2024.

[14] Robert Huben, Hoagy Cunningham, Logan Riggs, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

[15] Yuchu Jiang, Jiale Fu, Chenduo Hao, Xinting Hu, Yingzhe Peng, Xin Geng, and Xu Yang. Mimic in-context learning for multimodal tasks. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 29825–29835. IEEE, 2025.

[16] Guannan Lai, Gelin Bian, Hao-Xuan Ma, Jun-Peng Jiang, Long Chen, Jian-Dong Liu, Zhi-Hao Tan, and Han-Jia Ye. Routing should pay for itself: Sparse supervision for economical llm routing. *arXiv preprint arXiv:2609.37402*, 2026.

[17] Itay Levy, Ben Bogin, and Jonathan Berant. Diverse demonstrations improve in-context compositional generalization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1401–1422, 2023.

[18] Li Li, Ziyi Wang, Yongliang Wu, Jianfei Cai, et al. Cot vectors: Transferring and probing the reasoning mechanisms of llms. In *International Conference on Learning Representations*, volume 2026, pages 108745–108767, 2026.

[19] Xiaonan Li, Kai Lv, Hang Yan, Tianyang Lin, Wei Zhu, Yuan Ni, Guotong Xie, Xiaoling Wang, and Xipeng Qiu. Unified demonstration retriever for in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4644–4668, 2023.

[20] Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 278–300, 2024.

[21] Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. Commongen: A constrained text generation challenge for generative commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1823–1840, 2020.

[22] Jiachang Liu, Dinghan Shen, Yizhe Zhang, William B Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for gpt-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd workshop on knowledge extraction and integration for deep learning architectures*, pages 100–114, 2022.

[23] Jian-Dong Liu, Zi-Chen Zhao, Hao Sun, Lin-Xing Wu, Huan Zhang, Pengyuan Wang, Ming Zhao, Xinyu Chu, Shu Yan, Yongbei Zhu, et al. Constructive specification for plug-and-play learnware agents. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 3222–3233, 2026.

[24] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, 2022.

[25] Man Luo, Xin Xu, Zhuyun Dai, Panupong Pasupat, Mehran Kazemi, Chitta Baral, Vaiva Imbrasaite, and Vincent Y Zhao. Dr. icl: Demonstration-retrieved in-context learning. *arXiv preprint arXiv:2305.14128*, 2023.

[26] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pages 11048–11064, 2022.

[27] Aliakbar Nafar, K Brent Venable, and Parisa Kordjamshidi. Learning vs retrieval: The role of in-context examples in regression with large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8206–8229, 2025.

[28] Bruno A Olshausen and David J Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

[29] Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.

[30] Keqin Peng, Liang Ding, Yancheng Yuan, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. Revisiting demonstration selection strategies in in-context learning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9090–9101, 2024.

[31] Yingzhe Peng, Chenduo Hao, Xinting Hu, Jiawei Peng, Xin Geng, and Xu Yang. Live: Learnable in-context vector for visual question answering. *Advances in Neural Information Processing Systems*, 37:9773–9800, 2024.

[32] Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. SemEval-2014 task 4: Aspect based sentiment analysis. In Preslav Nakov and Torsten Zesch, editors, *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland, August 2014. Association for Computational Linguistics.

[33] Chengwei Qin, Aston Zhang, Chen Chen, Anirudh Dagar, and Wenming Ye. In-context learning with iterative demonstration selection. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7441–7455, 2024.

[34] Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders. *arXiv preprint arXiv:2407.14435*, 2024.

[35] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 3982–3992, 2019.

[36] Ohad Rubin, Jonathan Herzig, and Jonathan Berant. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 2655–2671, 2022.

[37] Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

[38] Adly Templeton. *Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet*. Anthropic, 2024.

[39] Huazheng Wang, Jinming Wu, Haifeng Sun, Zixuan Xia, Daixuan Cheng, Jingyu Wang, Qi Qi, and Jianxin Liao. Mdr: Model-specific demonstration retrieval at inference time for in-context learning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4189–4204, 2024.

[40] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788, 2020.

[41] Xubin Wang, Jianfei Wu, Yuan Yichen, Deyu Cai, Mingzhe Li, and Weijia Jia. Demonstration selection for in-context learning via reinforcement learning. In *International Conference on Machine Learning*, pages 64937–64954. PMLR, 2025.

[42] Zhiyong Wu, Yaoxiang Wang, Jiacheng Ye, and Lingpeng Kong. Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1423–1436, 2023.

[43] An Yang, Anfeng Li, Baosong Yang, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[44] Zhao Yang, Yuanzhe Zhang, Dianbo Sui, Cao Liu, Jun Zhao, and Kang Liu. Representative demonstration selection for in-context learning with two-stage determinantal point process. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5443–5456, 2023.

[45] Zhe Yang, Damai Dai, Peiyi Wang, and Zhifang Sui. Not all demonstration examples are equally beneficial: Reweighting demonstration examples for in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13209–13221, 2023.

[46] Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. Compositional exemplars for in-context learning. In *International Conference on Machine Learning*, pages 39818–39833. PMLR, 2023.

[47] Zeping Yu and Sophia Ananiadou. How do large language models learn in-context? query and key matrices of in-context heads are two towers for metric learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3281–3292, 2024.

[48] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28, 2015.

[49] Yiming Zhang, Shi Feng, and Chenhao Tan. Active example selection for in-context learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9134–9148, 2022.

[50] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pages 12697–12706. Pmlr, 2021.

A Experimental Details

A.1 Model and SAE

Because PULSE operates in SAE feature space, we evaluate it on backbone models with publicly available SAEs: Gemma2-2B and Llama3.1-8B. Unless otherwise specified, experiments use both backbones; GSM8K is evaluated only with Llama3.1-8B. Table 8 reports the exact backbone identifiers, SAE checkpoints, layers, dictionary sizes, and decoding settings.

A.2 Datasets

We evaluate on six benchmarks spanning classification, generation, and reasoning. For classification, we use AGNews (4-class topic classification), REST14 and LAP14 (3-class aspect-based sentiment), and EMOC (4-class emotion recognition), reporting classification accuracy. To keep the evaluation-query budget comparable across classification datasets, AGNews, REST14, and EMOC are evaluated on fixed 512-query held-out evaluation subsets, while LAP14 uses its full 463-query test split because its test set is already of comparable size. The same evaluation queries are used for all methods, shots, and backbones, making all method comparisons paired within each dataset. For generation, we use CommonGen [21], reporting BLEU-4, we use the official validation split for final evaluation. For reasoning, we use GSM8K [7], reporting exact match accuracy. Across all tasks, utility features are identified only from the training split, and held-out evaluation queries are used only for downstream ICL evaluation. Table 9 summarizes the split sizes, evaluation-query counts, input lengths, label distributions, and support-pool sizes.

A.3 Hyperparameters

Hyperparameter summary. Table 10 centralizes the default values for all PULSE hyperparameters used in the main experiments.

A.4 Baseline Protocol and Tuning

Table 11 summarizes the supervision and tuning used by each method. PULSE uses a small labeled discovery set to estimate $\mathbf{w}_l$, while Random, Lex-Sim, SBERT, KNN-SAE, and CEIL do not use labeled discovery queries to learn task-specific feature weights. All methods are evaluated on the same held-out evaluation queries, and all hyperparameters are fixed before test evaluation.

For pool-scale retrieval, all pointwise retrieval methods use the same shortlist-and-greedy context composition procedure with the same L and λ_r; only the relevance score used to rank candidates differs. This ensures that differences between SBERT, KNN-SAE, and PULSE-Retriever are due to the scoring signal rather than to a different context-composition algorithm.

For the PULSE-Ranking experiment, all methods rank the same pre-sampled complete k-shot candidate sets for each query. Pointwise baselines score a complete set by averaging query–example similarities:

$$s_{\text{base}}(q_x, E) = \frac{1}{|E|} \sum_{(x_i, z_i) \in E} \text{sim}(q_x, x_i).$$

Because all candidate sets have the same shot count, averaging and summing produce the same ranking. CEIL uses its set-level diversity-aware objective to score the same pre-sampled candidate sets.

B Interpreting PULSE-Retriever as a Scalable Proxy

B.1 From Complete-Set Scoring to Per-Example Retrieval

PULSE-Ranking uses the signed utility-localization vector at the same scale at which it is estimated. For a complete demonstration set E, the ranking score is

$$s_{\text{rank}}(q_x, E) = w_l^\top \left[A_l(q_x, E) - A_l(q_x, \emptyset) \right].$$

Table 8: **Backbone, SAE, and decoding details.** All SAE features are taken from residual-stream SAEs at the listed layer. The Gemma-Scope checkpoint is the layer-12 width-16k average-L_0-82 dictionary; the Llama-Scope LXR-32x checkpoints use $d_{\text{model}} = 4096$ and $d_{\text{SAE}} = 131{,}072$. For classification, downstream decisions are made by teacher-forced label likelihood rather than free-form generation.

Backbone	SAE release/checkpoint	Layer(s)	d_{SAE}	Inference and decoding settings
Gemma2-2B	`gemma-scope-2b-pt-res-canonical` `layer_12/width_16k/average_l0_82`	12	16,384	Classification: teacher-forced label likelihood, no sampling. CommonGen: greedy generation, `do_sample=False`, `max_new_tokens`=32; temperature/top-p disabled.
Llama3.1-8B	`llama_scope_lxr_32x` `fnlp/Llama3_1-8B-Base-LXR-32x` [13]	12 (classification) 16 (CommonGen) 24 (GSM8K)	131,072	Classification: teacher-forced label likelihood, no sampling. CommonGen: greedy generation, `do_sample=False`, `max_new_tokens`=32; temperature/top-p disabled. GSM8K: greedy generation, `do_sample=False`, `max_new_tokens`=220; temperature/top-p disabled.

Table 9: **Dataset and pool statistics.** Train/test sizes and support-pool sizes are read from the experiment loaders. Average input length is computed on the input side using whitespace tokenization (concept-list length for CommonGen). Generation and reasoning experiments use smaller pools because each selected context requires autoregressive evaluation.

Dataset	Task	Train	Test	Avg. len.	Pool	Label distribution
AGNews	Topic cls.	120,000	7,600	37.8	2,000	train: Business=30,000, Sci/Tech=30,000 train: Sports=30,000, World=30,000 test: Business=1,900, Sci/Tech=1,900 test: Sports=1,900, World=1,900
REST14	Aspect sentiment	2,881	721	21.9	2,000	train: Negative=633, Neutral=513, Positive=1,735 test: Negative=172, Neutral=120, Positive=429
LAP14	Aspect sentiment	1,850	463	23.8	1,850	train: Negative=686, Neutral=384, Positive=780 test: Negative=180, Neutral=76, Positive=207
EMOC	Emotion cls.	30,160	5,509	17.6	2,000	train: angry=5,506, happy=4,243 train: others=14,948, sad=5,463 test: angry=298, happy=284 test: others=4,677, sad=250
CommonGen	Constrained gen.	67,389	4,018	3.3	2,000	–
GSM8K	Math reasoning	7,473	1,319	45.3	2,000	–

This score directly measures whether the complete context activates SAE features whose paired activation shifts were associated with higher or lower discovery-time utility. Applying it to all k-shot subsets of a large support pool is infeasible because the search space grows combinatorially.

A more faithful per-candidate alternative would be to score each labeled singleton demonstration by a full prompt forward pass,

$$s_{\text{single}}(q_x, x_i) = w_l^\top \left[A_l(q_x, \{(x_i, z_i)\}) - A_l(q_x, \emptyset) \right].$$

However, this would require a separate full-prompt forward pass for every query-candidate pair. Selecting the top-k singleton scores would also approximate set-level utility as an additive sum of individual utilities, ignoring interactions among examples in the final k-shot context.

PULSE-Retriever therefore uses a scalable proxy. Instead of using the sign of w_l as an individual-candidate preference direction, it uses the magnitude $|w_l|$ as a feature-relevance mask. Let

$$a^{(q)} = A_l(q_x, \emptyset), \qquad a^{(i)} = A_l(q_{x_i}, \emptyset)$$

be zero-shot SAE encodings of the evaluation query and candidate input, respectively, with the target omitted. The PULSE-masked similarity is

$$s_{\text{mask}}(q_x, x_i) = \cos\left(|w_l| \odot a^{(q)}, |w_l| \odot a^{(i)} \right).$$

Table 10: **Hyperparameter summary.** Defaults are the values used in the main experiments unless a task-specific exception is shown.

Symbol	Meaning	Default value
K^+	Positive-weight SAE features retained	Gemma2-2B: 512 Llama3.1-8B: 2,048
K^-	Negative-weight SAE features retained	Gemma2-2B: 512 Llama3.1-8B: 2,048
β	Sparse–dense blend weight in the retrieval score	0.3
L	Retrieval shortlist size	50
λ_r	SAE-space redundancy penalty	0.3
$\lvert\mathcal{D}_{\text{disc}}\rvert$	Discovery queries	64
N	Candidate sets per discovery query	Classification: 32 CommonGen/GSM8K: 32
ε	Variance-normalization stabilizer	10^{-6}
$seed$	random seed	42

Table 11: Supervision and tuning resources used by each retrieval method. PULSE uses a small labeled discovery split to estimate a utility-localization vector; the non-PULSE retrievers do not learn task-specific utility weights from labeled discovery queries. For pool-scale retrieval, pointwise methods use the same shortlist and greedy redundancy procedure unless otherwise stated.

Method	Labeled discovery data	Validation tuning	Model-internal SAE features	Greedy redundancy
Random	No	No	No	No
Lex-Sim	No	Yes	No	Yes
SBERT	No	Yes	No	Yes
KNN-SAE	No	Yes	Yes	Yes
CEIL	No	Yes	No	Built-in diversity objective
PULSE-Retriever	Yes, 64 queries	Yes	Yes	Yes

Equivalently, this is cosine similarity under the diagonal metric

$$D_w = \mathrm{diag}(|w_l|^2),$$

whose numerator is $(a^{(q)})^\top D_w a^{(i)}$. Thus, PULSE-Retriever compares the query and candidate along SAE dimensions that the discovery procedure identified as utility-related.

This proxy differs from the complete-set score in two ways. First, it uses an input-only, zero-shot representation $A_l(q_{x_i}, \emptyset)$ of the candidate rather than the activation shift produced by inserting the labeled demonstration into the query prompt. This makes candidate encodings cacheable and avoids a query-candidate full-prompt forward pass. Second, it uses $|w_l|$ rather than signed w_l, because a sign learned from set-level contrasts is not calibrated as a per-candidate preference direction. PULSE-Retriever should therefore be interpreted as a utility-informed retrieval metric, not as an exact decomposition of complete-context utility.

B.2 Why the Retriever Uses a Magnitude Mask

The distinction between PULSE-Ranking and PULSE-Retriever is a scale distinction. At the complete-set scale, the signed vector $\mathbf{w}_l$ is meaningful because Equation (7) evaluates the activation shift of the whole context, which matches the discovery-time contrast used to estimate $\mathbf{w}_l$. At the per-example retrieval scale, however, the retriever only has access to input-only zero-shot encodings of the query and candidate before the labeled demonstration is inserted into the final context. Therefore, the set-level sign of $\mathbf{w}_l$ is not calibrated as a one-sided individual-candidate preference direction.

PULSE-Retriever uses $|\mathbf{w}_l|$ as a feature-relevance mask:

$$s_{\text{mask}}(q_x, x_i) = \cos\big(|\mathbf{w}_l| \odot \mathbf{a}^{(q)}, |\mathbf{w}_l| \odot \mathbf{a}^{(i)}\big).$$

This should be interpreted as cosine similarity under a diagonal relevance metric over utility-informative SAE dimensions, not as a signed utility score.

Table 12: Empirical comparison of the magnitude mask and a sign-sensitive retrieval score in PULSE-Retriever. Results are from the 4-shot pool-scale retrieval setting. The metric is accuracy for classification, BLEU-4 for CommonGen, and exact match for GSM8K. $\Delta = |\mathbf{w}|$ mask $-$ signed score. Bold marks $|\Delta| \geq 0.8$. "–" indicates GSM8K is not evaluated on Gemma2-2B.

Backbone	Variant	AGNews	REST14	LAP14	EMOC	CommonGen	GSM8K
Gemma2-2B	$\lvert\mathbf{w}\rvert$ mask	85.94	77.73	77.32	68.36	11.09	–
	signed score	84.38	76.56	77.54	65.23	10.25	–
	Δ	**+1.56**	**+1.17**	-0.22	**+3.13**	**+0.84**	–
Llama3.1-8B	$\lvert\mathbf{w}\rvert$ mask	88.09	81.84	82.07	71.09	10.90	54.66
	signed score	85.16	82.03	81.86	72.66	10.64	51.53
	Δ	**+2.93**	-0.19	$+0.21$	-1.57	$+0.26$	**+3.13**

A symmetric signed-mask cosine is not a distinct alternative:

$$\cos(\mathbf{w}_l \odot \mathbf{a}^{(q)}, \mathbf{w}_l \odot \mathbf{a}^{(i)}) = \cos(|\mathbf{w}_l| \odot \mathbf{a}^{(q)}, |\mathbf{w}_l| \odot \mathbf{a}^{(i)}),$$

because the sign disappears from both the numerator and denominator. To test a genuinely sign-sensitive retrieval rule, we therefore compare against a signed bilinear score,

$$s_{\text{signed}}(q_x, x_i) = (\mathbf{a}^{(q)})^\top \operatorname{diag}(\mathbf{w}_l)\mathbf{a}^{(i)}.$$

The signed score is normalized per query and uses the same shortlist and greedy redundancy procedure as PULSE-Retriever.

Table 12 compares the magnitude mask with this sign-sensitive alternative in the 4-shot pool-scale retrieval setting. The magnitude mask performs better on average, supporting our interpretation that the sign of $\mathbf{w}_l$ is reliable for complete-set activation shifts but not generally calibrated as a per-example retrieval preference direction.

C Greedy Demonstration Selection

Given the blended score s_{blend} from Equation (9), each pointwise retrieval method first ranks candidate support demonstrations and keeps a fixed-size shortlist of size L. We then greedily compose the final k-shot demonstration context with an SAE-space redundancy penalty.

Let Sel_{t-1} be the set of demonstrations selected before step t. For a candidate i, define

$$R_t(i) = \begin{cases} 0, & t = 1, \\ \max\limits_{j \in \text{Sel}_{t-1}} \cos\big(\mathbf{a}^{(i)}, \mathbf{a}^{(j)}\big), & t > 1. \end{cases}$$

At step t, the next demonstration is selected as

$$i_t = \arg \max_{i \in \text{Shortlist} \setminus \text{Sel}_{t-1}} \left[s_{\text{blend}}(q_x, x_i) - \lambda_r R_t(i) \right]. \tag{10}$$

The redundancy term is cosine similarity in SAE feature space, and $\lambda_r \geq 0$ controls the penalty strength. The base case $R_1(i) = 0$ means that the first selected example is the highest-scoring candidate in the shortlist. The exclusion $i \in \text{Shortlist} \setminus \text{Sel}_{t-1}$ prevents selecting the same demonstration more than once.

D Feature Semantics and Visualization Details

This appendix provides additional details for the feature-semantic analysis in Section 5.1. The goal of this analysis is to understand what the high-weight coordinates in the PULSE utility-localization vector tend to represent. We describe these features as *associated with* or *aligned with* discovery-time utility, rather than as causal mediators of downstream accuracy.

Table 13: Top PULSE-identified features on AGNews.

Sign	Feature	Weight	Max-activating snippet	Interpretation
+	8186	7.48	Open Source: Balancing Innovation and Risk	tech-industry commentary
+	11980	6.16	Climate fears on sharp CO_2 rise	scientific finding / climate framing
+	14321	5.75	Women Voters Enter San Diego Mayoral Fray	civic / institutional reference
+	4297	4.35	British-Irish national kidnapped in Afghanistan	named geopolitical event
+	1398	3.70	Australia struggling against New Zealand	sports-news event
−	12684	-3.65	Bond Prices Edge Higher at Close of Day	market-summary phrasing
−	12664	-3.52	Keep low profile, US citizens urged	security-advisory boilerplate
−	7429	-2.38	Owens' comments weren't necessary	sports-quote commentary
−	8762	-2.32	Dying star creates fantasy-like sculpture	descriptive scientific narrative
−	13690	-2.24	IRL: La der pour Castroneves	non-English race recap

D.1 Visualization Protocol

All feature visualizations in Section 5.1 are conducted on Gemma2-2B with the Gemma-Scope residual-stream SAE at layer 12, matching the main Gemma retrieval experiments. PULSE weights are learned from the training split using 64 discovery queries. For each discovery query, candidate discovery sets are sampled from the training support pool, and the test split is never used during discovery-time feature identification.

For each prompt, we encode the layer-12 residual-stream activations with the SAE and mean-pool SAE feature activations over all non-BOS prompt tokens. The PULSE utility-localization vector is then computed from paired utility differences and SAE activation differences, as described in Section 3. For visualization, we search over training-pool examples and report, for each feature, the example with the largest mean SAE activation. These snippets are therefore illustrative max-activating examples, not manually selected interpretations.

We apply an activation filter when constructing the displayed feature tables: a feature is included only if its maximum activation in the train search pool is at least 0.5. This avoids showing high-weight features that receive large PULSE weights during discovery but do not fire clearly on the broader visualization corpus. Such features may still affect retrieval through small activation changes, but they are less suitable for snippet-based semantic interpretation.

D.2 Representative Features Across Datasets

Tables 13, 14, 15, and 16 report the top five positive-weight and top five negative-weight PULSE-identified features for each classification dataset after the activation filter. These tables expand the qualitative summary in Section 5.1 by showing max-activating snippets for the identified features. The interpretation column is post-hoc and describes the visible pattern in the corresponding snippet.

Across the inspected high-weight features, AGNews features often correspond to topic-anchoring cues, such as technology, geopolitical, business, or sports-related phrasing. EMOC features often activate on dialogue fragments with explicit affective reasons, conversational pivots, or stylistically distinct emotional expressions. For REST14 and LAP14, several high-weight features capture aspect–opinion bindings, while others activate on generic, off-aspect, or decontextualized expressions.

Importantly, the sign of a feature reflects its association with paired utility differences under the PULSE identification objective; it should not be interpreted as sentiment polarity, label polarity, or a causal effect. Negative weights therefore indicate features whose higher activation is associated with lower discovery-time utility, not necessarily negative task labels.

Table 14: Top PULSE-identified features on EMOC.

Sign	Feature	Weight	Max-activating snippet	Interpretation
+	7063	11.19	no / make others cry / i know whats wrong now	dialogue with topical pivot
+	15322	11.18	shut up / my exam is on tuesday	direct-address with factual follow-up
+	9565	10.81	you are very bad / sorry / because…	confrontation with stated reason
+	5466	10.68	what else do you know about me?	introspective Q–A small talk
+	15273	9.54	what happened / net problems on my phone	factual problem report and topic shift
−	7657	-8.59	me happy / happy happy warm warm	repetitive affect tokens
−	5671	-8.40	turtles are stupid / i am also a turtle	absurdist syllogism
−	5435	-8.38	feeling good, feeling great !!!	repeated affect words and emoji
−	12481	-8.33	nice joke / mom you are perfect	joke-request loop
−	3537	-8.32	i don't like much / neither do i	flat agreement / topic decline

Table 15: Top PULSE-identified features on LAP14.

Sign	Feature	Weight	Max-activating snippet	Interpretation
+	11485	3.13	Aspect: screen; big screen allows you to enjoy movies	aspect to enjoyment
+	5828	1.68	Aspect: Pages; design tools make it easier	software aspect to ease of use
+	13435	1.61	Aspect: PRICE; LOVE THIS LAPTOP WONDERFUL PRICE	aspect to strong positive judgment
+	16061	1.56	Aspect: battery; why not give you a better battery?	aspect to complaint
+	9749	1.46	Aspect: price; iWork is not worth the price	aspect to negative judgment
−	15777	-3.82	Aspect: works; my new Toshiba works great on both	generic verb / off-aspect framing
−	15339	-3.11	same Toshiba sentence	duplicate off-aspect framing
−	517	-2.94	Aspect: Powerpoint program; PC users use Powerpoint	descriptive use-case
−	16208	-2.12	Aspect: zooming; Wonderful zooming	decontextualized short judgment
−	11584	-2.09	Aspect: wireless mouse; an absolute must	necessity claim without opinion detail

E Cross-Dataset Transfer

This appendix expands on the cross-dataset transfer summary in Section 5.2. We evaluate three questions: whether PULSE weights learned on one dataset transfer to another, how much active feature support is shared across datasets, and whether shared features are used in the same direction.

E.1 Source–Target Transfer

We learn the PULSE utility-localization vector on a source dataset and apply it directly to pool-scale retrieval on a different target dataset, in the 4-shot retriever setting with $\beta = 0.3$. We additionally evaluate $\text{LOO}_{\cup}$, which merges the vector learned from the three non-target datasets.

Table 7 shows substantial cross-task reuse. Averaged over all off-diagonal source–target pairs, transfer accuracy is 76.05, only 1.29 points below the within-dataset SELF average of 77.34. $\text{LOO}_{\cup}$ matches or nearly matches SELF on three targets: it equals SELF on REST14, slightly exceeds it on LAP14, and is only 0.39 points lower on EMOC. The larger gap on AGNews suggests that topic classification benefits more from task-specific utility features.

Table 16: Top PULSE-identified features on REST14.

Sign	Feature	Weight	Max-activating snippet	Interpretation
+	7875	0.61	Aspect: desserts; got favorable reviews	committed positive judgment
+	5435	0.56	Aspect: owner; always has a smile	warm person-description
+	1512	0.53	Aspect: food; good food and ok service	committed mixed judgment
+	3486	0.44	Aspect: Decor; leaves something to be desired	committed negative judgment
+	8374	0.43	Aspect: comfort food; hot and large portions	simple positive description
−	9781	-0.57	Aspect: ambience; I care more about the food and ambience	meta-commentary
−	13165	-0.56	Aspect: sour spicy soup; drumsticks over rice and soup	enumerative menu list
−	9472	-0.54	Aspect: thai; definitely not great	generic-aspect reference
−	11584	-0.54	Aspect: privacy; corner booth	parenthetical aside / off-aspect framing
−	12741	-0.52	Aspect: meal; amazing meal and experience	generic aspect with strong sentiment

Table 17: Pairwise overlap of active PULSE features across datasets. The two aspect-based sentiment datasets, REST14 and LAP14, have the largest overlap, but the overall Jaccard values remain small.

Dataset pair	Intersection	Union	Jaccard
AGNews–EMOC	94	1954	0.048
AGNews–REST14	90	1958	0.046
AGNews–LAP14	87	1961	0.044
EMOC–REST14	140	1908	0.073
EMOC–LAP14	95	1953	0.049
REST14–LAP14	207	1841	0.112

E.2 Feature Overlap and Sign Consistency

To understand how transfer is possible despite task-specific weights, we examine the support and signs of active PULSE features across datasets. Table 17 shows that active-feature overlap is generally small. The largest overlap appears between REST14 and LAP14, which share 207 active features with a Jaccard overlap of 0.112, consistent with their shared aspect-based sentiment structure. All other dataset pairs have Jaccard overlaps below 0.075.

Sign consistency further shows that shared features are not used in the same way across tasks. As shown in Table 18, 44.1% of features active in exactly two datasets have mixed signs; this ratio rises to 60.9% for features active in three datasets and 75.0% for features active in all four. Thus, the same SAE coordinate can be utility-positive for one task but utility-negative for another.

Together, these results suggest that PULSE uses reusable SAE coordinates but assigns task-dependent signs and magnitudes. The learned signal is therefore partly transferable, but not a single task-independent utility direction.

F Generalization to Additional Model Families and Scales

To examine whether the benefits of PULSE extend to additional model families and larger backbones, we conduct 4-shot retrieval experiments on Qwen3-8B [43] and Gemma2-9B across four classification datasets and CommonGen. For Qwen3-8B, we use the Qwen-Scope `W64K-L0_100` SAE [8] at layer 12, with a dictionary width of 65,536. For Gemma2-9B, we use the Gemma Scope canonical 16K SAE [20] at layer 20, with a dictionary width of 16,384. As shown in Table 19, PULSE achieves the highest score among the evaluated methods in all ten model–dataset combinations. Together with the Gemma2-2B and Llama3.1-8B experiments, these results extend

Table 18: Sign consistency among active PULSE features across datasets. A mixed-sign feature is utility-positive on at least one dataset and utility-negative on at least one other dataset.

# active datasets	# features	Consistent +	Consistent −	Mixed	% mixed
1	2894	1401	1493	–	–
2	497	145	133	219	44.1
3	64	19	6	39	60.9
4	4	1	0	3	75.0

Table 19: Additional 4-shot pool-scale retrieval results on Qwen3-8B and Gemma2-9B. Classification datasets use accuracy (%), and CommonGen uses BLEU-4. Bold marks the best result for each dataset within each model.

	Qwen3-8B					Gemma2-9B				
Dataset	ICL	KNN-SAE	SBERT	CEIL	PULSE	ICL	KNN-SAE	SBERT	CEIL	PULSE
AGNews	72.66	84.18	85.94	86.33	**86.91**	75.59	86.52	86.91	87.30	**87.89**
REST14	77.34	79.49	79.69	78.91	**80.47**	74.80	76.56	77.15	77.73	**78.13**
LAP14	78.19	81.43	82.07	82.72	**83.15**	77.11	81.20	81.43	81.64	**82.51**
EMOC	59.96	73.63	71.68	72.07	**74.22**	57.81	72.46	70.12	71.29	**74.22**
CommonGen	11.48	12.45	11.83	11.81	**12.47**	9.54	10.47	11.03	11.00	**11.73**

our evaluation to three model families spanning 2B–9B parameters and support the effectiveness of utility-guided SAE retrieval across the tested backbone configurations.

G Learned Retriever Baseline and Feedback Budget

G.1 Feedback Allocation and Supervision Granularity

The main experiments compare PULSE-Retriever against unsupervised baselines that do not consume target-model feedback. To examine whether its advantage persists when the baseline also receives such feedback, we additionally compare against an EPR-style learned retriever [36], which uses target-model feedback for gradient-based retriever training.

We consider two feedback allocations for EPR. The original configuration retrieves BM25 top-16 candidates for each of 128 training queries, evaluates each singleton query–demonstration pair with target-model feedback, and trains a dense retriever head on the resulting 2,048 labeled pairs. We additionally evaluate a matched configuration with 64 queries and 32 candidates per query. This configuration preserves the total of 2,048 feedback contexts while matching PULSE in both the number of discovery queries and the number of scored candidate contexts per query.

To further examine the role of supervision granularity, we evaluate two PULSE configurations. PULSE ($4 \rightarrow 4$) is the main setting: it estimates the utility-localization vector from complete 4-shot candidate sets and uses the vector for 4-shot retrieval. PULSE ($1 \rightarrow 4$) instead estimates the vector from singleton candidate contexts, as in EPR's feedback collection, and then uses it for the same 4-shot retrieval evaluation. Both configurations use 64 discovery queries and 32 candidate contexts per query, require no gradient-based retriever training, and produce a closed-form utility-localization vector over SAE features.

As shown in Table 20, under the original EPR allocation of 128×16, PULSE ($4 \rightarrow 4$) achieves a higher classification average on both backbones: 77.34 versus 77.27 on Gemma2-2B and 80.77 versus 80.22 on Llama3.1-8B. The methods show complementary per-dataset strengths: EPR leads on AGNews and LAP14, while PULSE leads on REST14 and EMOC. PULSE also improves CommonGen by 1.31 BLEU-4 on Gemma2-2B and 0.71 BLEU-4 on Llama3.1-8B, and improves GSM8K by 10.13 exact-match percentage points. These results are obtained with half as many discovery queries, the same number of scored feedback contexts, and no gradient-based retriever training.

Under the matched 64×32 allocation, PULSE ($4 \rightarrow 4$) outperforms EPR in 10 of the 11 evaluated task–backbone combinations. Its classification averages are 77.34 versus 75.01 on Gemma2-2B and 80.77 versus 78.36 on Llama3.1-8B. PULSE ($1 \rightarrow 4$) also outperforms the matched EPR baseline in

Table 20: Comparison with an EPR-style learned retriever under different feedback allocations and supervision granularities. All methods are evaluated in the 4-shot pool-scale retrieval setting. Q denotes the number of feedback queries and C the number of scored candidate contexts per query; all configurations use $Q \times C = 2{,}048$ feedback contexts. For PULSE, the arrows denote discovery shot count $\rightarrow$ evaluation shot count. Classification metrics are accuracy (%); CommonGen is BLEU-4; GSM8K is exact match (%). Cls. Avg. is the mean over the four classification datasets. Bold marks the best result in each metric column within each backbone.

		Classification (Acc.)						Generation & Reasoning	
Backbone	Method	AGNews	REST14	LAP14	EMOC	Cls. Avg.	$Q \times C$	CommonGen	GSM8K
Gemma2-2B	EPR-style learned retriever	**87.50**	75.00	**80.35**	66.21	77.27	128×16	9.78	–
	EPR-style learned retriever	85.55	74.80	74.08	65.62	75.01	64×32	10.21	–
	PULSE ($4 \rightarrow 4$)	85.94	**77.73**	77.32	**68.36**	**77.34**	64×32	**11.09**	–
	PULSE ($1 \rightarrow 4$)	85.74	75.20	74.73	67.77	75.86	64×32	10.32	–
Llama3.1-8B	EPR-style learned retriever	**89.26**	77.73	**83.00**	70.89	80.22	128×16	10.19	44.53
	EPR-style learned retriever	87.50	79.30	74.95	**71.67**	78.36	64×32	10.65	47.85
	PULSE ($4 \rightarrow 4$)	88.09	**81.84**	82.07	71.09	**80.77**	64×32	**10.90**	**54.66**
	PULSE ($1 \rightarrow 4$)	85.35	80.08	81.21	70.31	79.24	64×32	10.34	53.90

Table 21: Representation and learned-head controls on Gemma2-2B in the 4-shot pool-scale retrieval setting. Cls. Avg. is the mean classification accuracy (%) over AGNews, REST14, LAP14, and EMOC; CommonGen uses BLEU-4. Bold marks the best result in each column.

Method	Cls. Avg.	CommonGen
Hidden states without paired utility	73.19	10.15
Hidden states with paired utility	73.38	10.34
SAE features with a learned diagonal head	77.18	10.97
PULSE-Retriever	**77.34**	**11.09**

eight combinations and retains higher classification averages on both backbones (75.86 and 79.24, respectively). Thus, PULSE's advantage is not solely attributable to access to complete 4-shot feedback during discovery. The stronger results of PULSE ($4 \rightarrow 4$) further indicate an additional benefit from set-level estimation.

Together, these comparisons show that PULSE can effectively use target-model feedback without training a retriever. Moreover, its utility-localization vector supports the feature inspection (Section 5.1) and cross-task transfer (Section 5.2) analyses presented in this work.

G.2 Representation and Learned-Head Controls

To further examine the roles of SAE representations, paired utility supervision, and retriever training, we conduct additional comparisons on Gemma2-2B in the 4-shot pool-scale retrieval setting. We consider three controls: hidden-state retrieval without paired utility supervision, hidden-state retrieval with paired utility supervision, and SAE-based retrieval with a learned diagonal head. The hidden-state comparison examines whether paired supervision remains effective outside the SAE feature space. The learned-head comparison evaluates whether training an additional scoring function over SAE features offers an advantage over PULSE's closed-form utility identification.

PULSE-Retriever uses paired utility comparisons to identify utility-related SAE features and converts the resulting localization vector into a feature-relevance mask. Unlike the learned diagonal head, this procedure requires no gradient-based retriever training. We report the average accuracy over the four classification datasets and BLEU-4 on CommonGen.

As shown in Table 21, applying paired utility supervision to hidden-state retrieval improves the classification average from 73.19 to 73.38 and CommonGen from 10.15 to 10.34 BLEU-4. These gains are modest, indicating that paired supervision alone provides limited improvement in the tested hidden-state configuration. Both SAE-based configurations achieve higher scores on classification and CommonGen, complementing the SAE-basis ablation in Table 4 and supporting the use of SAE features for utility-guided retrieval.

Table 22: Cross-shot transfer of a utility-localization vector estimated with 4-shot contexts. Each cell reports the result using the *reused 4-shot vector / shot-specific vector*. Column headings indicate the evaluation shot count. Classification datasets use accuracy (%), and CommonGen uses BLEU-4.

	Gemma2-2B			Llama3.1-8B		
Dataset	1-shot	2-shot	8-shot	1-shot	2-shot	8-shot
AGNews	79.10 / 81.25	82.81 / 82.62	86.91 / 88.09	78.91 / 79.88	86.13 / 86.52	88.28 / 89.84
REST14	63.87 / 64.65	70.70 / 71.68	77.34 / 77.73	76.95 / 77.34	81.05 / 81.45	83.20 / 81.45
LAP14	71.92 / 72.79	76.24 / 77.11	79.70 / 80.35	75.59 / 76.03	80.78 / 80.56	82.51 / 82.72
EMOC	63.67 / 63.87	65.82 / 65.82	68.55 / 72.27	64.06 / 64.26	69.92 / 71.29	74.80 / 75.78
CommonGen	8.90 / 9.05	9.23 / 9.34	10.53 / 10.59	9.67 / 9.91	10.74 / 10.91	10.01 / 10.09

The learned diagonal head is competitive with PULSE, obtaining 77.18 classification accuracy and 10.97 BLEU-4 on CommonGen. PULSE achieves slightly higher scores of 77.34 and 11.09, corresponding to differences of 0.16 accuracy points and 0.12 BLEU-4, respectively. Thus, the learned head provides a strong alternative within the SAE feature space, while PULSE attains comparable performance without gradient-based retriever training.

Together, these results suggest that the representation used for utility-guided retrieval is important: applying paired supervision directly to hidden states yields limited gains, whereas both SAE-based approaches perform substantially better in this comparison. PULSE combines the SAE representation with closed-form feature identification, providing a competitive retrieval metric while retaining an interpretable utility-localization vector.

H Cross-Shot Transfer of the Utility-Localization Vector

Our default setting estimates a separate utility-localization vector for each shot count, since the utility of a demonstration set can depend on context length and interactions among demonstrations. To examine whether the learned vector can be reused across shot counts, we estimate a vector using 4-shot discovery contexts and apply it to 1-, 2-, and 8-shot retrieval without re-estimating the vector. We compare this transferred vector with the corresponding shot-specific vector for each backbone and dataset. This experiment evaluates whether cross-shot reuse can reduce the need for repeated offline discovery while retaining downstream performance.

As shown in Table 22, the transferred vector preserves much of the shot-specific performance in most settings. The differences are particularly small on CommonGen, where all six comparisons are within 0.24 BLEU-4. For classification, reuse can occasionally improve performance: on Llama3.1-8B REST14 at 8 shots, accuracy increases from 81.45 to 83.20. However, the effect is not uniform. On Gemma2-2B EMOC at 8 shots, accuracy decreases from 72.27 to 68.55, indicating that some settings benefit more from shot-specific estimation.

These results suggest that the utility-localization vector captures signals that partially transfer across shot counts, while retaining sensitivity to the evaluation context. Cross-shot reuse therefore offers a practical option for reducing offline discovery requirements, whereas shot-specific estimation remains our default.

I Hyperparameter Sensitivity

We report sensitivity analyses for five design choices in PULSE: the retained SAE feature budget K, the number of discovery queries $|\mathcal{D}_{\text{disc}}|$, the sparse–dense blend weight β, the SAE layer, and SAE dictionary width. All experiments use Gemma2-2B. Unless otherwise specified, evaluations use the 4-shot pool-scale retrieval setting; the SAE layer sweep covers 1-, 2-, 4-, and 8-shot retrieval.

Number of identified features. Table 23 shows the effect of varying the number of PULSE-identified features. Performance is stable across a wide range of feature budgets, with the average accuracy ranging from 76.62 to 77.65. This suggests that PULSE does not rely on a finely tuned feature count; once enough high-utility SAE features are retained, the retrieval signal remains stable.

Table 23: Retriever accuracy as a function of selected feature count K in the 4-shot pool-scale retrieval setting. Bold indicates the best result in each row.

Dataset	K=256	K=512	K=1024	K=2048	K=4096
AGNews	85.16	85.94	**86.13**	85.94	84.38
REST14	78.12	77.73	79.30	**80.08**	79.69
LAP14	77.97	77.32	76.67	78.19	**78.62**
EMOC	65.23	**68.36**	68.16	66.41	66.80
Avg.	76.62	77.34	77.57	**77.65**	77.37

Table 24: Retriever accuracy as a function of discovery query count $|\mathcal{D}_{\text{disc}}|$ in the 4-shot pool-scale retrieval setting. Bold indicates the best result in each row.

Dataset	16	32	64	128
AGNews	82.42	83.98	85.94	**86.72**
REST14	77.73	**78.52**	77.73	77.34
LAP14	**79.48**	78.19	77.32	78.40
EMOC	62.50	64.45	**68.36**	66.80
Avg.	75.53	76.28	**77.34**	77.32

Discovery set size. Table 24 reports performance as the number of discovery queries varies from 16 to 128. The average accuracy improves from 75.53 with 16 discovery queries to 77.34 with 64 queries, and then saturates at 128 queries. This indicates that PULSE can identify a useful sparse utility vector from a relatively small labeled discovery set.

Sparse–dense blend weight. Table 25 evaluates the blend weight β between PULSE-masked similarity and standard SAE-space cosine similarity. Pure PULSE weight retrieval ($\beta = 0$) and pure SAE cosine retrieval ($\beta = 1$) are both weaker than intermediate blend settings on average. The best average performance is obtained at $\beta = 0.3$, suggesting that the learned utility-related feature mask and broader SAE-space similarity provide complementary signals.

SAE layer choice. We further evaluate the sensitivity of PULSE-Retriever to the SAE layer on Gemma2-2B. Table 26 summarizes the layer sweep by reporting the best layer, the best average accuracy, and the performance of the default layer used in the main experiments. Tables 27–30 provide the complete per-layer results on AGNews, LAP14, REST14, and EMOC.

Overall, PULSE-Retriever is stable across a broad range of SAE layers. Although the best layer varies across shot settings, the default Layer 12 remains close to the best average performance, with gaps of at most 1.14 percentage points. This suggests that PULSE does not rely on a narrowly tuned layer choice.

SAE Width and Retained-Feature Budget

To examine the sensitivity of PULSE to SAE dictionary width, we evaluate Gemma2-2B under the 4-shot pool-scale retrieval setting using SAE widths of 16,384, 32,768, and 65,536. We consider two feature-retention policies. The fixed-budget policy retains $K^+ = K^- = 512$ features regardless of dictionary width. The proportional-budget policy retains 3.125% of the dictionary from each sign tail, giving $K^+ = K^- = 512$, 1,024, and 2,048 for the three widths, respectively. Here, K^+ and K^- denote the numbers of retained positive- and negative-weight features, so the total retained budget is $K^+ + K^-$.

As shown in Table 31, increasing SAE width does not yield uniform improvements across datasets. With the fixed feature budget, wider dictionaries improve LAP14 accuracy over the 16K configuration, while the default 16K SAE performs better on AGNews, REST14, and CommonGen. Scaling the retained feature budget with dictionary width improves several results relative to the fixed-budget setting. For example, LAP14 accuracy increases from 80.13 to 81.21 at 32K and from 79.91 to 81.43 at 65K. The proportional policy also improves REST14 and CommonGen at both wider configurations, although the effect on EMOC is mixed.

Table 25: Retriever accuracy as a function of the sparse–dense blend weight β in the 4-shot pool-scale retrieval setting. $\beta = 0$ denotes pure PULSE-weighted cosine similarity, while $\beta = 1$ denotes pure SAE-space cosine similarity. Bold indicates the best result in each row.

Dataset	$\beta = 0.0$	$\beta = 0.1$	$\beta = 0.2$	$\beta = 0.3$	$\beta = 0.5$	$\beta = 0.7$	$\beta = 1.0$
AGNews	83.01	83.20	85.94	85.94	**86.72**	86.52	81.05
LAP14	74.51	75.59	74.30	**77.32**	76.03	75.16	75.16
REST14	76.17	75.59	76.95	77.73	**77.93**	76.76	75.59
EMOC	60.16	63.87	62.30	**68.36**	67.77	66.41	66.80
Avg.	73.46	74.56	74.87	**77.34**	77.11	76.21	74.65

Table 26: Summary of SAE layer sensitivity for PULSE-Retriever on Gemma2-2B in the pool-scale retrieval setting. Best Avg. is the highest average accuracy across layers. Layer-12 Avg. denotes the default layer used in the main experiments. Gap is the difference between Best Avg. and Layer-12 Avg.; Layer Range is the difference between the best and worst layer averages. All numeric entries except layer indices are reported in percentages or percentage points.

Shot	Best Layer	Best Avg. (%)	Layer-12 Avg. (%)	Gap (pp)	Layer Range (pp)
1-shot	12	**70.64**	70.64	0.00	4.86
2-shot	21	**75.02**	74.31	0.71	2.57
4-shot	5	**78.48**	77.34	1.14	3.22
8-shot	24	**80.10**	79.61	0.49	2.40

These results suggest that SAE width and feature retention should be considered jointly: a wider dictionary does not necessarily improve retrieval when the retained feature budget remains fixed. The default 16K configuration remains competitive across tasks, while proportional retention allows wider dictionaries to improve performance on some datasets.

J Compute Resources

Experiments were run as local single-GPU jobs on NVIDIA GPUs. The primary workstation used for the reported runs contains 8*NVIDIA GeForce RTX 5090 (32GB) and 8*NVIDIA GeForce RTX 3090 (24GB); each run was assigned to one GPU at a time. We use frozen backbone language models and pretrained SAEs, without fine-tuning the language models or training new SAEs. The compute cost is dominated by forward passes for discovery-time paired-utility estimation and downstream ICL evaluation. SAE activations and candidate-level intermediate results are cached and reused across methods where possible. In our Gemma2-2B classification logs, constructing the static SAE cache for a pool of 1.85k–2k examples took about 141–277 seconds per dataset, while the discovery forward phase with 64 discovery queries and 32 candidate discovery sets per discovery query took about 448–884 seconds per dataset. As a hardware-independent scale estimate, a dense forward pass costs roughly $2P$ floating-point operations per token for a model with P parameters, corresponding to approximately 5.2 GFLOPs/token for Gemma2-2B and 16 GFLOPs/token for Llama3.1-8B, plus the SAE hook cost. Using the observed Gemma2-2B prompt lengths in the classification setting, this corresponds to roughly 0.25–0.6 PFLOPs for constructing one static pool cache and roughly 7–16 PFLOPs for one discovery pass. We report these as approximate ranges rather than exact total FLOPs because the total depends on prompt length, label-word batching, generation length, and cache reuse.

K Seed Stability of PULSE-Retriever

To assess the stability of PULSE-Retriever across random seeds, we run the 4-shot pool-scale retrieval experiment with three independent seeds. Each seed controls the sampling of the discovery set $\mathcal{D}_{\text{disc}}$ from the training split and the sampling of the N candidate demonstration sets per discovery query, while the test split and evaluation protocol are held fixed. Table 32 reports the per-seed accuracy together with the mean and standard deviation across seeds for both backbones.

Table 27: Full SAE layer sweep for PULSE-Retriever on Gemma2-2B in the 1-shot pool-scale retrieval setting. Each entry reports downstream accuracy (%). Avg. denotes the average across AGNews, LAP14, REST14, and EMOC. † denotes the default layer used in the main experiments. Bold indicates the best value in each row.

Layer	0	1	2	3	4	5	6	7	8	9	10	11	12†
AGNews	73.24	79.49	77.54	76.37	74.80	73.63	75.00	77.54	73.63	79.49	78.91	72.66	**81.25**
LAP14	67.39	65.44	69.55	67.60	67.39	70.41	68.68	68.25	66.95	68.68	71.92	69.33	**72.79**
REST14	64.26	60.94	65.43	65.82	**67.58**	65.82	66.80	65.62	66.02	65.62	64.65	66.60	64.65
EMOC	62.11	62.50	63.28	62.70	65.23	64.06	60.74	63.28	**65.43**	58.40	61.52	62.30	63.87
Avg.	66.75	67.09	68.95	68.12	68.75	68.48	67.81	68.67	68.01	68.05	69.25	67.72	**70.64**
Layer	13	14	15	16	17	18	19	20	21	22	23	24	25
AGNews	71.68	76.95	72.27	71.68	75.00	77.93	77.54	79.49	75.20	75.00	80.86	74.41	75.98
LAP14	68.03	66.74	67.82	66.95	69.55	71.27	70.41	68.90	68.47	68.90	68.03	67.60	66.74
REST14	65.43	65.82	64.45	64.84	67.19	65.04	**67.58**	67.19	66.80	64.06	63.48	66.60	65.82
EMOC	64.65	57.23	58.59	59.77	57.62	64.65	59.96	62.70	60.74	64.06	58.79	59.77	56.45
Avg.	67.45	66.68	65.78	65.81	67.34	69.72	68.87	69.57	67.80	68.01	67.79	67.10	66.25

Table 28: Full SAE layer sweep for PULSE-Retriever on Gemma2-2B in the 2-shot pool-scale retrieval setting. Each entry reports downstream accuracy (%). Avg. denotes the average across AGNews, LAP14, REST14, and EMOC. † denotes the default layer used in the main experiments. Bold indicates the best value in each row.

Layer	0	1	2	3	4	5	6	7	8	9	10	11	12†
AGNews	80.86	76.95	79.49	79.30	82.23	80.66	80.86	83.20	83.98	81.64	81.84	83.01	82.62
LAP14	75.59	74.30	74.95	74.51	74.51	75.16	72.57	73.65	73.43	73.43	76.46	75.38	**77.11**
REST14	73.05	68.95	69.53	72.07	73.44	73.44	74.22	70.70	73.24	73.83	71.29	**75.00**	71.68
EMOC	69.14	**72.07**	69.34	70.51	65.82	68.36	68.95	65.82	68.36	67.38	67.77	66.02	65.82
Avg.	74.66	73.07	73.33	74.10	74.00	74.41	74.15	73.34	74.76	74.07	74.34	74.85	74.31
Layer	13	14	15	16	17	18	19	20	21	22	23	24	25
AGNews	83.20	83.20	82.42	85.55	84.38	84.57	**86.13**	83.01	84.38	83.79	85.35	82.62	84.96
LAP14	75.81	74.30	75.38	74.08	75.81	74.08	74.51	73.00	76.24	74.08	75.81	76.03	73.00
REST14	73.63	73.05	70.90	72.66	70.31	70.51	70.31	72.66	72.27	69.14	70.70	71.48	71.48
EMOC	66.99	62.70	67.77	67.19	66.02	70.51	64.26	63.67	67.19	67.19	63.67	66.41	60.35
Avg.	74.91	73.31	74.12	74.87	74.13	74.92	73.80	73.08	**75.02**	73.55	73.88	74.13	72.45

The results show two consistent patterns. First, PULSE-Retriever is substantially more stable on Llama3.1-8B than on Gemma2-2B: standard deviations on Llama3.1-8B range from 0.53 to 1.81 across all six benchmarks, whereas on Gemma2-2B they range from 0.62 to 2.60. We attribute this gap primarily to the smaller capacity of Gemma2-2B, which makes the identified utility-localization vector more sensitive to the particular discovery-set composition sampled by each seed. Second, on Llama3.1-8B the per-seed values for every benchmark fall within roughly one standard deviation of the mean, and the GSM8K result in particular is highly stable (std 0.91 over a $+3.16$-point reported gain), supporting the robustness of the reasoning-task improvements reported in Section 4.2.

Table 32 summarizes the variation across the three discovery seeds. On the four classification benchmarks, the standard deviations range from 0.69 to 1.81 percentage points for Llama3.1-8B and from 1.07 to 2.60 percentage points for Gemma2-2B. On CommonGen, the corresponding standard deviations are 0.53 and 0.62 BLEU-4, respectively. On GSM8K, Llama3.1-8B achieves a mean exact-match accuracy of 55.20%, with a standard deviation of 0.91 percentage points and individual scores ranging from 54.66% to 56.25%.

L Limitations

PULSE relies on reliable SAEs for the target backbone and a small labeled discovery split, although it is gradient-free and does not train a retriever. It also adds offline compute and storage cost, since feature discovery requires full-prompt backbone and SAE forward passes over multiple candidate

Table 29: Full SAE layer sweep for PULSE-Retriever on Gemma2-2B in the 4-shot pool-scale retrieval setting. Each entry reports downstream accuracy (%). Avg. denotes the average across AGNews, LAP14, REST14, and EMOC. † denotes the default layer used in the main experiments. Bold indicates the best value in each row.

Layer	0	1	2	3	4	5	6	7	8	9	10	11	12†
AGNews	84.77	84.18	83.20	85.55	83.40	86.52	86.33	85.16	86.33	88.67	84.77	85.55	85.94
LAP14	75.59	77.97	77.32	75.81	77.54	77.97	79.70	77.75	77.75	77.54	77.54	77.32	77.32
REST14	75.59	73.63	75.78	75.39	73.44	**77.93**	74.80	75.39	75.98	75.39	74.41	75.00	77.73
EMOC	70.51	71.29	69.34	71.09	71.48	71.48	67.58	69.53	**72.07**	68.95	69.92	70.12	68.36
Avg.	76.61	76.77	76.41	76.96	76.46	**78.48**	77.10	76.96	78.03	77.64	76.66	77.00	77.34
Layer	13	14	15	16	17	18	19	20	21	22	23	24	25
AGNews	85.94	86.52	85.16	86.52	84.96	86.91	88.09	**89.06**	86.13	86.91	88.87	87.70	84.57
LAP14	78.19	77.97	79.27	78.83	77.75	77.32	78.62	**80.13**	76.24	77.54	77.32	76.89	78.19
REST14	75.59	**77.93**	75.78	75.59	75.98	75.59	74.61	73.63	75.39	75.39	75.20	73.83	76.37
EMOC	68.95	66.99	66.99	69.14	69.73	70.12	68.75	66.80	67.97	65.43	68.36	66.60	61.91
Avg.	77.16	77.35	76.80	77.52	77.10	77.48	77.52	77.41	76.43	76.32	77.44	76.25	75.26

Table 30: Full SAE layer sweep for PULSE-Retriever on Gemma2-2B in the 8-shot pool-scale retrieval setting. Each entry reports downstream accuracy (%). Avg. denotes the average across AGNews, LAP14, REST14, and EMOC. † denotes the default layer used in the main experiments. Bold indicates the best value in each row.

Layer	0	1	2	3	4	5	6	7	8	9	10	11	12†
AGNews	86.52	85.74	85.55	83.59	**90.04**	86.13	85.55	88.28	88.67	88.48	86.91	87.50	88.09
LAP14	78.40	77.97	78.62	79.91	79.70	79.27	79.48	80.35	79.27	80.78	79.48	79.48	80.35
REST14	78.32	78.91	76.95	77.34	76.37	77.15	77.73	77.73	78.32	77.73	76.17	**79.49**	77.73
EMOC	73.83	73.44	74.80	73.83	73.05	72.85	**75.00**	73.44	72.85	70.31	71.68	70.31	72.27
Avg.	79.27	79.01	78.98	78.67	79.79	78.85	79.44	79.95	79.78	79.33	78.56	79.20	79.61
Layer	13	14	15	16	17	18	19	20	21	22	23	24	25
AGNews	87.50	87.30	88.28	89.06	87.30	88.87	88.28	87.89	88.87	89.26	89.65	88.48	88.09
LAP14	79.05	80.13	**80.99**	79.70	77.97	78.19	77.97	80.56	79.27	79.27	79.48	80.56	80.13
REST14	77.34	78.71	78.91	78.71	76.76	77.15	77.93	76.95	77.73	77.54	77.73	79.30	77.34
EMOC	69.14	68.95	70.51	70.90	70.12	69.73	70.12	69.92	68.75	70.70	71.88	72.07	65.23
Avg.	78.26	78.77	79.67	79.59	78.04	78.48	78.57	78.83	78.65	79.19	79.68	**80.10**	77.70

sets, and PULSE-Retriever caches zero-shot SAE encodings for the support pool. These costs avoid exhaustive scoring over all k-shot subsets, but remain higher than lightweight embedding retrievers.

PULSE-Retriever is a scalable proxy rather than an exact set-utility decomposition: the utility-localization vector is estimated from set-level activation shifts but used as a per-demonstration magnitude mask for retrieval. Moreover, mean-pooled SAE activations may be affected by prompt-length differences, suggesting future extensions such as query-segment pooling, length-matched sets, or length-residualized activation differences.

For GSM8K-style reasoning, our discovery utility uses teacher-forced likelihood of supervised solution sequences, which assumes access to reasoning traces. Settings with only final answers may require alternative proxies such as final-answer likelihood, execution-based rewards, or generated rationales. Finally, our feature analyses are post-hoc and associational; causal interventions on identified features remain future work.

Table 31: Sensitivity to SAE dictionary width and feature-retention policy on Gemma2-2B with 4-shot retrieval. Classification datasets use accuracy (%), and CommonGen uses BLEU-4. The reported feature budget applies separately to the positive and negative tails. The 16K configuration is shared by both policies. Bold marks the best result for each dataset within each policy, including ties.

			Classification (Acc.)				Generation
Budget policy	SAE width	$K^+ = K^-$	AGNews	REST14	LAP14	EMOC	CommonGen
Fixed	16K	512	**85.94**	**77.73**	77.32	**68.36**	**11.09**
	32K	512	84.96	75.20	**80.13**	67.38	10.84
	65K	512	83.98	75.78	79.91	**68.36**	10.56
Proportional	16K	512	**85.94**	**77.73**	77.32	68.36	**11.09**
	32K	1,024	84.96	76.56	81.21	**68.55**	10.92
	65K	2,048	84.76	76.76	**81.43**	67.97	10.64

Table 32: Seed stability of 4-shot PULSE-Retriever. We report mean $\pm$ standard deviation and individual results over three seeds. Accuracy and exact match are percentages; CommonGen uses BLEU-4.

	Gemma2-2B		Llama3.1-8B	
Dataset	Mean $\pm$ Std	Per-seed values	Mean $\pm$ Std	Per-seed values
AGNews	87.76 ± 1.62	89.06, 88.28, 85.94	87.44 ± 0.69	88.09, 86.72, 87.50
REST14	76.43 ± 2.60	77.73, 73.44, 78.12	79.89 ± 1.69	81.84, 78.91, 78.91
LAP14	79.41 ± 2.43	82.07, 77.32, 78.83	81.28 ± 0.76	82.07, 80.56, 81.21
EMOC	69.46 ± 1.07	68.36, 70.51, 69.53	72.13 ± 1.81	71.09, 74.22, 71.09
CommonGen	10.47 ± 0.62	11.09, 10.47, 9.85	11.07 ± 0.53	10.90, 10.65, 11.67
GSM8K	–	–	55.20 ± 0.91	54.66, 56.25, 54.69

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 and introduction state that PULSE identifies utility-related internal features through SAE activations and uses them for controlled set-level ranking and scalable demonstration retrieval. These claims are supported by the experiments in Sections 4.2 and 5 and are scoped to the evaluated models, datasets, and available SAEs.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: The paper discusses limitations in Appendix L, including the dependence on reliable public SAEs, the non-causal interpretation of the identified features, the cost of the discovery procedure, and the limited range of evaluated model families and benchmarks.

   Guidelines:

   - The answer [N/A] 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: [N/A]

Justification: The paper does not present formal theoretical results, theorems, or proof-based claims. Appendix B provides an informal derivation and explicitly states the assumptions behind interpreting PULSE-Retriever as a scalable proxy, but this derivation is not used as a formal proof of the method.

Guidelines:

- The answer [N/A] 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: Section 4 specifies all models, SAEs, datasets, metrics, and the discovery protocol. Appendix A reports exact backbone identifiers, SAE checkpoints, layer choices, dictionary sizes, decoding settings, dataset splits, and all hyperparameter defaults. Appendix C describes the greedy selection procedure. Appendix I provides sensitivity analyses. Together, these details are sufficient to reproduce the main results.

Guidelines:

- The answer [N/A] 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: [Yes]

   Justification: We provide our code at `https://github.com/aohenuo/PULSE`, including core code and the datasets used in this work are all publicly available.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4 specifies models, datasets, metrics, discovery protocol, feature budgets, and baselines. Appendix C details the greedy selection procedure. Appendix I provides hyperparameter sensitivity analyses for feature count K, discovery pool size, blend weight beta, and SAE layer choice.

   Guidelines:

   - The answer [N/A] 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: Appendix K reports means and standard deviations over three independent random seeds for PULSE-Retriever in the 4-shot setting. The evaluation covers the four classification benchmarks and CommonGen on both backbones, and GSM8K on Llama3.1-8B. The seed sweep varies discovery-set and candidate-set sampling, while keeping the evaluation queries fixed. These results characterize the variability of PULSE-Retriever with respect to discovery-time sampling. A fully matched multi-seed comparison across all baselines is left to future work.

Guidelines:

- The answer [N/A] 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: [Yes]

   Justification: Appendix J reports the compute environment used for the experiments.

   Guidelines:

   - The answer [N/A] 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 research uses only publicly available models and datasets for academic benchmarking. No human subjects, private data, or sensitive content is involved. The work conforms with the NeurIPS Code of Ethics.

   Guidelines:

- The answer [N/A] 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: [N/A]

    Justification: The research uses publicly available models and datasets for academic benchmarking. No human subjects, private data, or sensitive content is involved. The work conforms with the NeurIPS Code of Ethics.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release new pretrained language models, image generators, scraped datasets, or other assets with high misuse risk. Therefore, no additional release safeguards are applicable.

    Guidelines:

    - The answer [N/A] 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: All datasets and models are cited with their original publications (References [13, 12, 37, 20, 32, 4, 7, 48, 21]). The backbone models (Gemma2-2B, Llama3.1-8B) and SAEs (Gemma Scope) are publicly released under their respective licenses.

    Guidelines:

    - The answer [N/A] 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: [Yes]

    Justification: We release the code for the PULSE framework at `https://github.com/aohenuo/PULSE`, with documentation and instructions provided in the repository. The paper does not introduce new datasets or models.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: The paper does not involve human subjects research.

    Guidelines:

    - The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs (Gemma2-2B, Llama3.1-8B) are used as standard backbone models for evaluation, which is a conventional use in ICL research rather than a novel or non-standard methodological component. No LLMs were used for generating core methodological ideas or experimental designs.

    Guidelines:

    - The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.