SWIS: An Efficient Structured Pruning Framework for LLMs with Spectral Geometric Alignment

1st Lan Xu
School of Computer Science and Technology
Changsha University of Science and Technology
Changsha, China
lanx56218@gmail.com

2nd Ping Li
School of Computer Science and Technology
Changsha University of Science and Technology
Changsha, China
lping9188@163.com

***Abstract*—Network pruning is a promising approach to alleviate the computational burden of deploying and serving large language models (LLMs). Training-free paradigms are particularly appealing in this context due to the prohibitive cost of retraining. However, most existing training-free methods implicitly rely on magnitude-based criteria that effectively ignore the anisotropic geometry of LLM representations, overlooking the directional structure of information flow and struggling to adapt to modern architectures. In this paper, we propose SWIS (Spectral-Weighted Importance Scoring), a training-free structured pruning framework that offers a geometry-aware perspective on LLM compression. SWIS employs Eigenvalue Decomposition (EVD) on activation covariance matrices to identify geometric directions governing dominant information flow. By integrating weight magnitudes with geometric alignment to high-energy principal subspaces, SWIS quantifies true contribution of channels, distinguishing critical signals from orthogonal noise. Furthermore, to optimize layer-wise sparsity allocation, SWIS-Pruner uses Feature Angular Distance to quantify intensity of geometric transformations across layers. Extensive evaluations on language benchmarks demonstrate that, without retraining, SWIS outperforms state-of-the-art baselines, including LLM-Pruner and structured extensions of Wanda.**

***Index Terms*—Large Language Models, Structured Pruning, Spectral Analysis**

I. Introduction

In recent years, the large memory footprint and inference latency of Large Language Models (LLMs) [1] have posed major challenges for deployment on resource-constrained devices. To alleviate these bottlenecks, various model compression techniques have been proposed, including quantization [2] [3], knowledge distillation [4] [5], and pruning [6]. Among them, pruning has drawn particular attention for its ability to directly remove redundant parameters from over-parameterized models.

The effectiveness of pruning critically depends on its granularity. Existing methods can be broadly categorized into unstructured, semi-structured, and structured pruning, each offering different trade-offs between accuracy and hardware efficiency. Unstructured pruning [7] [8] achieves high accuracy retention but produces irregular sparsity patterns that are difficult to exploit on general-purpose hardware [9]. Semi-structured pruning [10] improves hardware compatibility through predefined patterns, yet suffers from limited flexibility and fragility under fine-tuning [11]. In contrast, structured pruning [12] [13] removes entire components such as neurons, channels, or attention heads, yielding regular sparsity that directly translates into memory reduction and inference acceleration, and is therefore generally considered the most practical paradigm for efficient large-scale deployment.

Despite the strong hardware efficiency of structured pruning, existing methods suffer from fundamental limitations in both importance evaluation and sparsity allocation. First, most structured pruning approaches rely on magnitude-based criteria [8], which implicitly assume an isotropic feature space where each channel independently contributes information proportional to its scalar magnitude. However, spectral analyses of Transformer activations [14] reveal pronounced anisotropy: token representations are concentrated along a few dominant directions rather than being uniformly distributed across dimensions. Crucially, these semantic directions typically span multiple channels instead of aligning with any single physical axis, implying that semantic importance arises from coordinated projections across channels rather than individual magnitudes. As a result, magnitude-only criteria fail to reliably distinguish informative channels from orthogonal noise. Second, existing unified or global sparsity allocation strategies [8] [15] neglect the heterogeneous redundancy across layers. Prior work such as ShortGPT [16] observes that many deep Transformer layers induce only minor geometric rotations and behave close to identity mappings. Nevertheless, directly removing such layers often leads to severe performance degradation [17], suggesting that they still preserve essential semantic continuity. The absence of a fine-grained metric to characterize this geometric redundancy prevents existing methods from balancing structural preservation and redundancy removal, leading to suboptimal compression.

To address these challenges, we propose SWIS-Pruner, a training-free structured pruning framework that integrates spectral geometric analysis with dynamic sparsity allocation. Specifically, we introduce Spectral-Weighted Importance Scoring (SWIS) to evaluate channel saliency by jointly considering weight magnitude and geometric alignment with the principal directions of the activation space, identified via eigenvalue decomposition of the empirical covariance matrix. In addition, we propose a Feature Angular Distance metric to quantify layer-wise geometric transformation strength, enabling adap-

tive and non-uniform sparsity allocation across layers.

Our contributions are summarized as follows:

- We propose SWIS, a geometry-aware channel importance metric that accounts for feature space anisotropy and provides more accurate saliency estimation than magnitude-based criteria.
- We introduce Feature Angular Distance to measure layer-wise redundancy, enabling dynamic and adaptive sparsity allocation across Transformer layers.
- Extensive experiments on mainstream LLMs demonstrate that SWIS-Pruner consistently outperforms state-of-the-art structured pruning methods under equal sparsity budgets, and achieves superior performance in high-compression regimes.

II. RELATED WORK

A. Channel Importance Scoring

Structured pruning compresses models by removing structural components based on importance metrics. Early approaches relied on static weight magnitude [18] or gradient-based Taylor approximations, such as LLM-Pruner [12], which either provide limited pruning fidelity or incur substantial computational overhead. Hessian-based methods, including SparseGPT [7], further improve pruning quality but remain resource-intensive for large-scale models. Consequently, recent studies have shifted toward efficient activation-driven criteria, such as Wanda [8], FLAP [15], and E-Sparse [19], which assess importance via activation magnitude, variability, or information-theoretic measures. Despite their effectiveness, these methods predominantly rely on scalar statistics and do not explicitly capture the geometric or directional structure of feature transformations in the representation space.

B. Layer-wise Sparsity Allocation

Early works like Wanda [8] adopted uniform pruning ratios, ignoring the inherent variance in layer-wise redundancy and limiting performance. Recent studies address this through non-uniform allocation strategies. OWL [20] adjusts layer-wise ratios based on the uneven distribution of activation outliers to preserve information-dense layers. Alternatively, ShortGPT [16] utilizes the Block Influence (BI) metric—measuring input-output cosine similarity—to identify and remove redundant layers that function as approximate identity mappings. Furthermore, SlimGPT [21] accounts for error propagation by designing a depth-dependent incremental allocation rule, prioritizing parameter retention in shallower layers to mitigate cumulative errors.

C. Low-Rank Decomposition

Low-rank decomposition reduces complexity by factorizing weight matrices into smaller components [22]. Methods like TensorGPT [23]and ASVD [24] use tensor decomposition or SVD to numerically reconstruct weights via optimal low-rank approximations. Although our work also employs eigen-decomposition, we utilize it distinctly as a tool for geometric manifold analysis rather than low-rank reconstruction. By identifying principal semantic directions in high-dimensional space, we quantify channel contribution based on geometric alignment. This allows us to selectively preserve critical physical channels in the original network, thereby circumventing the reconstruction errors inherent in approximation-based decomposition methods.

III. METHOD

Problem Definition. Given a pre-trained network $f(\mathbf{x}; \mathcal{W})$ and a target sparsity $\bar{p}$, we aim to find a binary mask $\mathcal{M} \in \{0,1\}^{|\mathcal{W}|}$ to prune weights. We denote the set of structurally constrained masks (e.g., channel-wise) as $\mathbb{S}$. The optimization problem is formulated as:

$$\begin{aligned} \min_{\mathcal{M}} \quad & \mathcal{L}\left(f(\mathbf{x}; \mathcal{W}), f(\mathbf{x}; \mathcal{W} \odot \mathcal{M})\right) \\ \text{s.t.} \quad & \frac{\|\mathcal{M}\|_0}{\|\mathcal{W}\|_0} \leq 1 - \bar{p}, \quad \mathcal{M} \in \mathbb{S}, \end{aligned} \tag{1}$$

where the objective minimizes the output discrepancy between the pruned and original networks, subject to the compression constraint. Due to the NP-hard nature of this problem, we propose the SWIS metric to approximate the optimal mask $\mathcal{M}$ under a dynamic layer-wise sparsity budget.

Method Overview. We propose SWIS-Pruner, with its overall workflow illustrated in Fig. 1. First, we employ eigenvalue decomposition to identify the dominant directions within the activation space. Subsequently, we compute a weighted accumulation based on channel weight magnitudes and their projection magnitudes onto these high-energy principal directions. Then, in the unified structured pruning stage, we utilize Feature Angular Distance to dynamically allocate layer-wise sparsity and aggregate channel scores into structured metrics, enabling the removal of redundant MLP neurons and GQA attention heads.

A. Spectral Weighted Importance Scoring

Motivation and Intuition. We first analyze the geometric structure of the activation space in large language models. Specifically, we collect channel-wise activations from each transformer layer on a held-out calibration set and compute the corresponding activation covariance matrices, whose eigenvalue spectra are shown in Fig. 2(Left).Across layers, the eigenvalue spectra exhibit a pronounced long-tailed distribution, where a small number of dominant components account for the majority of the activation variance. This spectral anisotropy becomes increasingly severe in deeper layers, indicating that information is progressively concentrated into a low-dimensional principal subspace.

Figure 2(Right) further visualizes the last-layer activations projected onto the first two principal components. We observe that some channels with relatively small L_2 norms are strongly aligned with the dominant principal directions, whereas certain large-magnitude channels lie in low-variance or nearly orthogonal subspaces. These observations suggest that magnitude-only pruning criteria fail to accurately capture the true importance of channels.

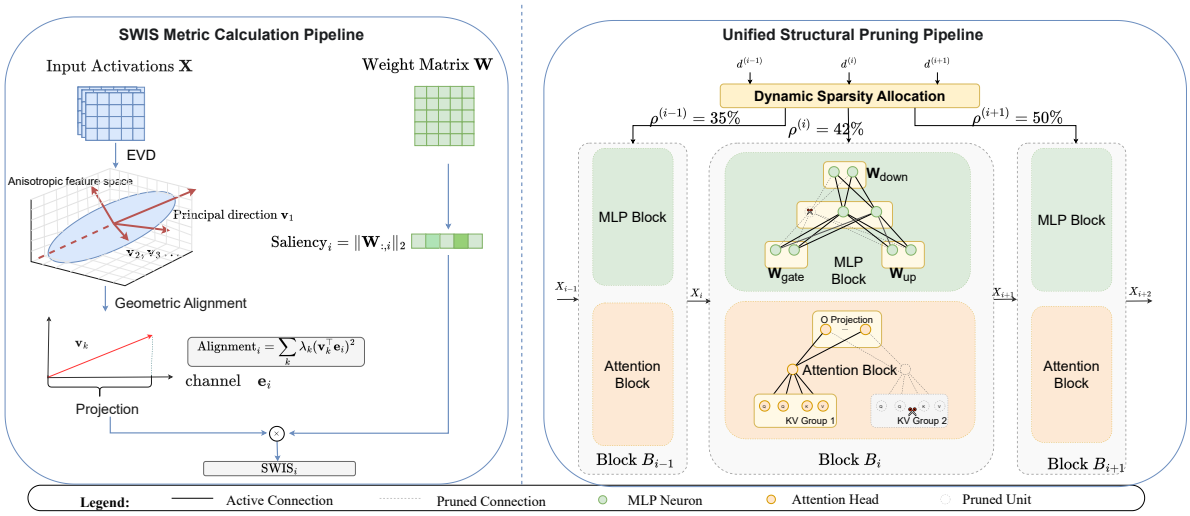

Fig. 1. The overall framework of SWIS-Pruner. (Left) SWIS Metric Calculation: We evaluate channel importance by fusing weight saliency with geometric alignment, where geometric alignment measures the projection of channels onto the principal directions of the anisotropic feature space derived from input activations. (Right) Unified Structural Pruning:Based on aggregated SWIS scores and dynamic sparsity allocation, we perform unified structural pruning on MLP neurons and GQA attention heads (KV groups), removing redundant structures while preserving active connections.

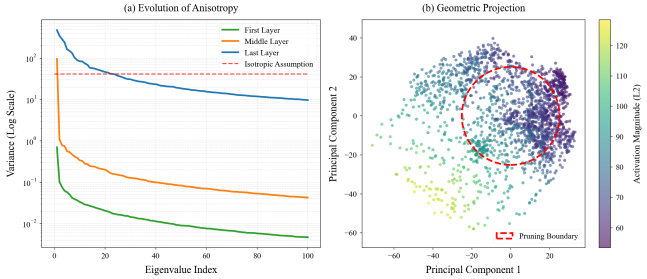

Fig. 2. Activation anisotropy analysis of Llama-3.1-8B. Left: Spectral analysis across layers. The sharp eigenvalue decay shows that information concentrates in few dominant directions, especially in deeper layers. Right: PCA projection of the last layer. The misalignment between the anisotropic distribution and the magnitude pruning boundary (red circle) illustrates how magnitude-based metrics fail to preserve low-magnitude features aligned with principal information flows.

Motivated by this insight, we reevaluate channel importance from a geometry-aware perspective by measuring the effectiveness of channel projections onto high-energy principal subspaces. Intuitively, only channels that exhibit both substantial magnitude and strong alignment with the dominant information directions are considered critical.

Formulation. To decode the anisotropic geometric structure of the activation space, given input activations $\mathbf{X} \in \mathbb{R}^{N \times d_{in}}$, we perform Eigenvalue Decomposition (EVD) on the empirical covariance matrix Σ:

$$\Sigma = \frac{1}{N}\mathbf{X}^\top \mathbf{X} = \sum_{k=1}^{d_{in}} \lambda_k \mathbf{v}_k \mathbf{v}_k^\top, \tag{2}$$

where $\mathbf{v}_k$ and λ_k denote the k-th principal direction and its corresponding spectral energy, respectively.

Having identified the principal directions, we quantify the relationship between each physical channel and these directions. To this end, we define the importance of the i-th input channel (represented by the standard basis vector $\mathbf{e}_i$). To capture the effective information flow, we measure its *geometric alignment* $(\mathbf{v}_k^\top \mathbf{e}_i)^2$ with respect to all principal directions. These terms are weighted by normalized spectral energies and combined with the corresponding parameter magnitude to derive the final SWIS score:

$$\text{SWIS}_i = \underbrace{\|\mathbf{W}_{:,i}\|_2}_{\text{Parameter Saliency}} \cdot \underbrace{\sum_{k=1}^{d_{in}} \left(\frac{\lambda_k}{\sum_j \lambda_j + \epsilon} \right) (\mathbf{v}_k^\top \mathbf{e}_i)^2}_{\text{Spectral Geometric Alignment}}, \tag{3}$$

where $\|\mathbf{W}_{:,i}\|_2$ denotes the L_2 norm of the weight column corresponding to the i-th channel, and ϵ is a smoothing term. This metric quantifies the degree of geometric alignment between the channel and the principal subspace, thereby adaptively preserving dimensions that possess significant projection components along critical semantic directions.

B. Structured Importance Score Aggregation

After obtaining the fine-grained channel-level scores SWIS_i, we need to aggregate them into unified scores for MLP neurons and attention heads based on the Transformer topology to achieve hardware-friendly structured pruning.

MLP Neuron Aggregation. For the MLP block, each column of the down-projection matrix $\mathbf{W}_{\text{down}}$ directly corresponds to an activated neuron. Therefore, the importance $S_{\text{MLP}}^{(u)}$ of the u-th neuron is equivalent to the SWIS score of its corresponding column:

$$S_{\text{MLP}}^{(u)} = \text{SWIS}_i, \tag{4}$$

where the i-th column corresponds to neuron u.

GQA Mean-Pooling Aggregation. regarding the attention modules, we model the Grouped Query Attention (GQA) mechanism widely adopted in modern LLMs. Given input activations $\mathbf{X} \in \mathbb{R}^{N\times C}$, GQA partitions a total of H_Q query heads into H_{KV} Key-Value (KV) groups. Each group $g \in \{1,\dots,H_{KV}\}$ contains $G = H_Q/H_{KV}$ query heads and shares the same pair of key-value projection matrices $\mathbf{W}_K^g$ and $\mathbf{W}_V^g$.

Let $h \in \{1,\dots,G\}$ denote the index of the query head within a group, and $\mathrm{Attn}(\cdot)$ represent the standard scaled dot-product attention operation. Let $\mathbf{W}_Q^{gh}$ and $\mathbf{W}_O^{gh}$ denote the input and output projection weights, respectively, for the h-th query head in the g-th group. The computation process of this mechanism is formulated as follows:

$$\mathrm{GQA}(\mathbf{X}) = \sum_{g=1}^{H_{KV}} \sum_{h=1}^{G} \mathrm{Attn}(\mathbf{X}\mathbf{W}_Q^{gh}, \mathbf{X}\mathbf{W}_K^{g}, \mathbf{X}\mathbf{W}_V^{g})\mathbf{W}_O^{gh} \quad (5)$$

In the context of structured pruning, when the g-th KV group $(\mathbf{W}_K^g, \mathbf{W}_V^g)$ is removed, the output contributions of all query heads within this group, $\sum_{h=1}^{G} \mathrm{head}_{gh}\mathbf{W}_O^{gh}$, will be simultaneously eliminated.

To evaluate the overall importance of the g-th KV group, we must collectively consider the contributions of all query heads it supports. We employ a mean-pooling aggregation strategy, defining the group importance $S_{\text{Group}}^{(g)}$ as the average of the SWIS scores of the output projection matrix columns corresponding to all query heads within that group:

$$S_{\text{Group}}^{(g)} = \frac{1}{G}\sum_{h=1}^{G}\left(\sum_{j\in \mathrm{cols}(gh)} \mathrm{SWIS}_j\right), \quad (6)$$

C. Dynamic Sparsity Allocation

Uniform layer-wise pruning ratios often yield suboptimal performance as they overlook the functional heterogeneity in information processing across different transformer layers. Recent studies have demonstrated that parameter redundancy in LLMs is non-uniformly distributed, with certain layers exhibiting significantly higher compressibility [17]. This phenomenon is further corroborated by our analysis of geometric feature evolution, as illustrated in Fig. 2. Specifically, measurements based on feature angular distance reveal that the input, output, and initial shallow layers typically induce substantial feature rotations, signifying their critical roles in executing complex semantic transformations. In contrast, deeper layers—particularly those in the range of layers 20 to 30—exhibit minimal geometric rotation and behave similarly to identity mappings. This observation implies that these layers offer limited marginal semantic contributions and possess high structural redundancy. Motivated by these geometric insights, we leverage Feature Angular Distance to quantify layer-wise redundancy and adaptively determine sparsity targets, thereby optimizing the allocation of computational resources.

Formally, we define the angular distance $d^{(l)}$ to measure the extent to which the l-th layer alters the direction of the feature manifold. Given the input activations $\mathbf{x}_{\text{in}}^{(l)}$ and output activations $\mathbf{x}_{\text{out}}^{(l)}$ computed from calibration data, the average angular distance for the layer is formulated as:

$$d^{(l)} = \mathbb{E}_{\mathbf{X}}\left[\frac{1}{\pi}\arccos\left(\frac{\mathbf{x}_{\text{in}}\cdot\mathbf{x}_{\text{out}}}{\|\mathbf{x}_{\text{in}}\|_2\|\mathbf{x}_{\text{out}}\|_2}\right)\right] \quad (7)$$

A smaller $d^{(l)}$ indicates that the layer behaves close to an identity mapping and thus exhibits higher redundancy, while a larger value implies stronger semantic transformation.

To adaptively allocate sparsity, we map the angular distance vector $\mathbf{d}$ to layer-wise pruning ratios via a Softmax function with a negative temperature:

$$\rho^{(l)} = r_0 \cdot \mathrm{Softmax}(-\alpha\cdot\mathbf{d})_l = r_0 \cdot \frac{\exp(-\alpha\cdot d^{(l)})}{\sum_{k=1}^{L}\exp(-\alpha\cdot d^{(k)})} \quad (8)$$

where α controls the sensitivity to inter-layer differences, and $r_0 = L\cdot\bar{\rho}$ ensures that the expected sparsity matches the global budget $\bar{\rho}$.

As shown in Fig. 3, the resulting pruning ratios exhibit a clear inverse correlation with angular distance: layers with weaker feature transformations are pruned more aggressively, enabling precise compression of redundant layers while preserving critical ones.

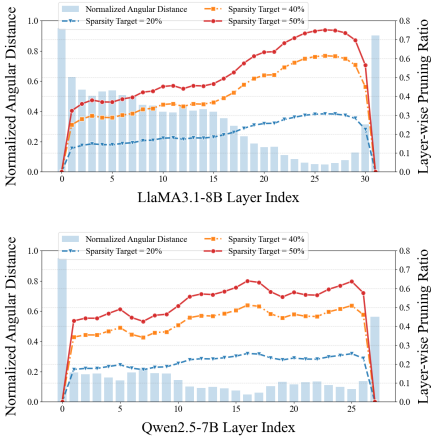

Fig. 3. Visualization of layer-wise angular distance and pruning ratios on LLaMA3.1-8B and Qwen2.5-7B.

IV. EXPERIENTS

A. Experimental Setup

Baselines. To evaluate the efficacy of our proposed method, we benchmark it against representative state-of-the-art structural pruning baselines, including the pioneering task-agnostic method LLM-Pruner [12], as well as recent advanced approaches like SlimLLM [25] and FLAP [15].

Evaluation Tasks and Datasets. Following the standard protocol of LLM-Pruner [12], we evaluate zero-shot perplexity on WikiText2 [26], PTB [27] and Lambada [28]for language

TABLE I

ZERO-SHOT PERFORMANCE OF THE COMPRESSED LLAMA-3.1-8B. THE AVERAGE SCORE IS COMPUTED ON THE COMMONSENSE REASONING DATASETS. THE "**BOLDED**" REPRESENTS THE BEST RESULT UNDER THE SAME PRUNING RATIO.

Ratio	Method	Language Modeling (↓)			Zero-shot Common-sense Reasoning (↑)							Average
		Wiki	PTB	LambD	ARC-e	ARC-c	PIQA	BoolQ	Wino	OBQA	Hella	
Ratio=0%	Dense	8.50	14.02	20.09	81.10	53.50	81.28	82.11	73.88	44.80	78.92	70.80
Ratio=20% w/o tune	SlimLLM	15.79	25.84	27.72	63.97	38.91	73.43	70.03	67.40	**39.40**	62.14	59.33
	LLM-Pruner	17.48	27.83	29.57	**69.57**	34.22	72.65	67.25	66.06	38.20	65.89	59.12
	FLAP	20.88	31.35	31.72	55.26	32.08	72.78	68.23	65.19	35.40	58.87	55.40
	SWIS(ours)	**15.43**	**23.75**	**27.13**	66.37	**41.72**	**74.59**	**70.57**	**69.22**	37.20	**67.14**	**60.97**
Ratio=20% w/ tune	SlimLLM	12.83	22.06	**24.23**	**74.33**	43.06	76.35	72.15	68.19	**41.80**	74.80	64.38
	LLM-Pruner	12.89	21.04	26.61	70.37	41.38	76.66	69.02	63.22	39.80	70.71	61.59
	FLAP	16.14	25.00	26.40	65.15	40.10	75.73	72.75	66.34	38.20	69.46	61.10
	SWIS(ours)	**12.79**	**20.80**	25.24	72.85	**44.20**	**77.97**	**73.79**	**70.32**	40.80	**75.71**	**65.09**
Ratio=40% w/o tune	SlimLLM	**33.57**	64.73	41.78	**46.96**	26.44	**68.66**	63.03	55.72	**33.00**	44.16	48.28
	LLM-Pruner	49.32	83.53	55.24	40.11	24.91	62.51	52.66	52.25	28.80	35.93	42.45
	FLAP	37.67	65.80	49.23	42.47	25.43	64.80	59.72	57.38	30.80	43.60	46.31
	SWIS(ours)	34.16	**52.40**	**39.24**	45.37	**27.99**	68.23	**64.34**	**58.17**	32.60	**44.86**	**48.79**
Ratio=40% w/ tune	SlimLLM	**19.78**	33.72	35.66	58.10	36.59	73.56	60.06	60.54	35.20	57.43	54.50
	LLM-Pruner	22.28	36.72	38.26	56.48	32.34	71.11	58.90	55.64	32.80	52.33	51.37
	FLAP	20.02	33.97	34.32	56.36	32.17	70.78	66.33	59.47	33.60	56.45	53.59
	SWIS(ours)	20.40	**31.53**	**33.44**	**61.19**	**37.68**	**74.38**	**66.39**	**61.40**	**36.80**	**58.25**	**56.58**
Ratio=50% w/o tune	SlimLLM	67.94	80.69	66.37	39.04	**26.37**	**63.38**	46.54	**58.64**	29.40	36.83	42.89
	LLM-Pruner	74.04	116.49	73.46	37.29	21.84	59.30	48.41	50.67	28.40	33.25	39.88
	FLAP	68.21	81.64	61.87	36.20	23.04	61.32	60.89	52.01	29.60	38.08	43.02
	SWIS(ours)	**47.99**	**66.89**	**48.37**	**40.57**	24.40	62.51	**61.65**	56.83	**30.04**	**39.39**	**45.06**
Ratio=50% w/ tune	SlimLLM	30.12	41.23	39.42	56.10	**32.25**	69.04	62.97	57.77	**32.80**	51.70	51.80
	LLM-Pruner	27.88	44.99	43.27	52.40	28.41	68.77	53.52	53.83	29.80	45.59	47.47
	FLAP	26.40	40.10	40.96	50.76	28.44	68.66	62.00	57.01	30.00	46.57	49.06
	SWIS(ours)	**25.94**	**38.19**	**37.74**	**57.34**	30.00	**69.41**	**64.07**	**58.56**	31.78	**52.27**	**51.92**

modeling. For commonsense reasoning, we employ a diverse benchmark suite including ARC [29], PIQA [30], BoolQ [31], Winogrande [32], OpenBookQA [33], and HellaSwag [34].

Implementation Details. To ensure a fair comparison, all methods utilize a unified calibration setting for importance score calculation. Specifically, we use 32 samples randomly drawn from the BookCorpus dataset, with the sequence length truncated to 128 tokens. During the fine-tuning stage, we perform recovery training on the cleaned version of the Alpaca dataset [35]. All experiments are fine-tuned on a single GPU for 2 epochs, with the learning rate set to 1e-4.

B. Main Results

We compare SWIS-Pruner with the aforementioned baselines on LLaMA-3.1-8B and Qwen2.5-7B. Table I and Table II report the zero-shot performance under varying structural sparsity levels for LLaMA-3.1-8B and Qwen2.5-7B, respectively.

Language Modeling. As shown in Table I, SWIS-Pruner consistently outperforms the baselines in most settings without fine-tuning, with particularly notable gains under high compression (50% sparsity). On WikiText2, SlimLLM yields a perplexity of 67.94, whereas SWIS-Pruner reduces it to 47.99, significantly mitigating the performance degradation induced by aggressive pruning. Similarly, our method achieves the lowest perplexities on PTB and Lambda, at 66.89 and 48.37, respectively, indicating more accurate preservation of language-critical weights. After fine-tuning, SWIS-Pruner remains the best or highly competitive on WikiText2 and PTB, demonstrating the superior recoverability of the resulting sparse structures.

TABLE II

ZERO-SHOT PERFORMANCE OF THE COMPRESSED QWEN2.5-7B. WE REPORT THE PERPLEXITY (PPL) FOR LANGUAGE MODELING AND THE AVERAGE ACCURACY FOR ZERO-SHOT COMMON-SENSE REASONING TASKS.

Ratio	Method	Language Modeling (↓)			Zero-shot (↑)
		Wiki	PTB	LambD	Avg.
Ratio=0%	Dense	8.96	15.72	22.42	70.31
Ratio=20%	SlimLLM	16.50	33.54	**25.14**	59.26
	LLM-Pruner	16.11	28.09	27.94	59.45
	FLAP	17.48	29.94	29.44	56.21
	SWIS(ours)	**15.79**	**26.03**	26.34	**60.23**
Ratio=40%	SlimLLM	35.60	75.20	42.36	48.53
	LLM-Pruner	46.45	77.03	45.72	47.31
	FLAP	29.91	73.91	39.76	49.19
	SWIS(ours)	**26.48**	**65.86**	**37.89**	**49.21**
Ratio=50%	SlimLLM	125.45	210.55	140.23	40.07
	LLM-Pruner	112.67	188.61	151.32	41.63
	FLAP	143.42	430.80	129.34	37.70
	SWIS(ours)	**81.32**	**165.67**	**107.58**	**42.42**

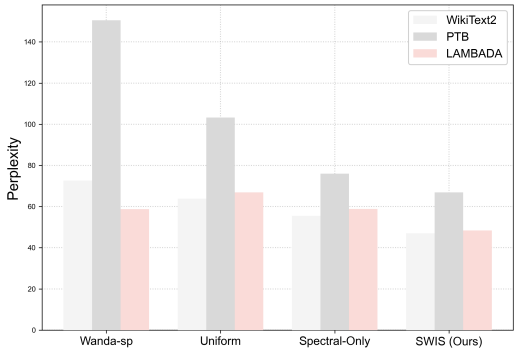

Fig. 4. Comparison of different pruning metrics across multiple datasets at 50% sparsity.

Zero-shot Tasks. As summarized in Table I, our method surpasses SlimLLM, LLM-Pruner, and FLAP in average accuracy across all sparsity levels, under both pruning-only and fine-tuning recovery settings. At the aggressive 50% sparsity level, SWIS-Pruner achieves an average accuracy of 45.05% without fine-tuning, significantly outperforming the runner-up and demonstrating stronger knowledge retention under severe compression. After fine-tuning, particularly at 40% sparsity, our model reaches a peak average accuracy of 56.58%, substantially exceeding SlimLLM (54.49%). These results indicate that the proposed approach effectively removes redundant parameters while maximally preserving the core capabilities required for complex reasoning tasks. Consistent performance trends are also observed on the Qwen2.5-7B model, as reported in Table II.

C. Ablation Study

We systematically examine three fundamental components of the SWIS-Pruner framework: the pruning metric, the global compression structure, and the dynamic allocation strategy. All ablation experiments are conducted on the LLaMA-3.1-8B model at 50% structural sparsity and evaluated on zero-shot benchmarks.

Pruning Metrics. By introducing spectral geometric alignment, our method aims to overcome the limitations of traditional magnitude-based pruning metrics. To evaluate its effectiveness, we investigate three structural pruning metrics: 1) **Spectral-Only**: A geometric metric that evaluates channel importance solely based on spectral projection scores, disregarding parameter magnitudes; 2) **Wanda-sp**: A structural variant of Wanda [8] that estimates group importance by aggregating the product of weight magnitudes and input norms; and 3) **SWIS**: The metric utilized in our method, which fuses parameter saliency with spectral geometric alignment to assess the recovery potential of output feature maps.

As shown in figure 4, SWIS consistently achieves the highest average accuracy. At 50% sparsity, our method attains an average accuracy of 44.06%, outperforming Wanda-sp (39.88%) and Spectral-Only. This performance gap indicates that by capturing the alignment with principal components, SWIS effectively distinguishes between signal and noise, thereby determining channel importance more accurately than scalar-based metrics.

Layer-wise Sparsity Allocation. To validate the effectiveness of our dynamic allocation strategy, we first benchmarked it against a uniform pruning baseline, i.e., applying a fixed sparsity across all layers. As shown in figure 4, at 50% sparsity, uniform pruning resulted in severe performance degradation, with its PTB perplexity surging to 103.2. In contrast, our transformation-aware strategy, which adaptively adjusts sparsity based on Feature Angular Distance, achieved a significantly lower perplexity of 66.89, underscoring the importance of non-uniform resource allocation.

Furthermore, we evaluated the impact of the temperature coefficient α with values of 3, 4, and 5 under different sparsity levels. Results indicate that the optimal α value stabilizes between 4 and 5, suggesting that a moderately aggressive allocation strategy—biasing towards redundant layers—is optimal. We also observed that as the global pruning rate decreases, the performance gaps caused by different α values diminish significantly. This is because, at low compression rates, the retained parameter budget is sufficient, reducing the model's sensitivity to the allocation strategy; whereas at high compression rates, precise resource allocation via an appropriate α is crucial to prevent performance collapse.

TABLE III
ABLATION STUDY OF TEMPERATURE COEFFICIENT α UNDER DIFFERENT SPARSITY LEVELS.

Sparsity	α	Perplexity (↓)		
		WikiText2	PTB	LAMBADA
50%	3	51.67	72.97	50.67
	4	**47.99**	**66.89**	**48.37**
	5	48.37	67.63	57.21
40%	3	35.98	55.70	44.47
	4	36.41	53.04	43.52
	5	**34.16**	**52.40**	**39.24**
20%	3	17.32	26.54	28.87
	4	15.98	24.67	**26.04**
	5	**15.43**	**23.75**	27.13

D. Efficiency Analysis

To evaluate deployment efficiency, we benchmarked the proposed method against LLM-Pruner, FLAP, and SlimLLM on a single NVIDIA L20 GPU across varying sparsity levels (Table IV). Our method significantly reduces computational costs while achieving superior inference performance. At 20% and 40% sparsity, it attains the lowest latencies of 0.54s and 0.43s and the highest throughputs of 7411.99 tokens/s and 9319.78 tokens/s, consistently outperforming baselines like FLAP and SlimLLM. The sparse structures induced by our approach are highly hardware-friendly, delivering the best balance of speed and throughput while maintaining competitive compression rates and memory efficiency comparable to state-of-the-art methods.

V. CONCLUSION

In this paper, we presented SWIS-Pruner, a training-free structured pruning framework for LLMs. To address feature

TABLE IV
POST-PRUNING STATISTICS DURING INFERENCE, TESTED ON 20 ROUNDS OF WIKITEXT (BATCH SIZE 32).

Sparsity	Method	Param (B)	FLOPs (G)	Latency (s)	Throughput (tok/s)	Mem (GB)
0%	Baseline	8.03	6.98	0.63	6426.15	20.89
20%	LLM-Pruner	6.71	5.66	0.57	7125.16	18.27
	FLAP	6.63	5.58	0.55	7366.44	18.31
	SlimLLM	6.58	5.56	0.59	6929.91	17.30
	Ours	6.62	5.58	**0.54**	**7411.99**	18.29
40%	LLM-Pruner	5.27	4.33	0.47	8709.75	12.79
	FLAP	5.24	4.19	0.44	9298.92	15.71
	SlimLLM	5.26	4.23	0.45	9169.50	14.85
	Ours	5.24	4.19	**0.43**	**9319.78**	15.69
50%	LLM-Pruner	4.62	3.58	0.37	**11045.22**	11.21
	FLAP	4.54	3.49	0.39	10453.36	14.34
	SlimLLM	4.57	3.54	0.39	10443.21	13.55
	Ours	4.54	3.49	0.38	10730.18	14.30

space anisotropy, we introduced the Spectral Weighted Importance Score (SWIS), which evaluates channel importance from an information-theoretic perspective. By quantifying the geometric alignment between channels and the dominant directions of the activation space, SWIS effectively preserves dimensions critical for global information representation. Furthermore, we developed a dynamic sparsity strategy based on Feature Angular Distance, which models layer-wise importance as the intensity of geometric transformations. This enables adaptive pruning of redundant layers approximating identity mappings. Extensive evaluations on LLaMA-3.1 and Qwen2.5 demonstrate that SWIS-Pruner significantly outperforms existing state-of-the-art methods in zero-shot language modeling and common-sense reasoning tasks.

REFERENCES

[1] R. Pope, S. Douglas, A. Chowdhery, J. Devlin, J. Bradbury, J. Heek, K. Xiao, S. Agrawal, and J. Dean, "Efficiently scaling transformer inference," *Proceedings of machine learning and systems*, vol. 5, pp. 606–624, 2023.

[2] T. Dettmers, M. Lewis, Y. Belkada, and L. Zettlemoyer, "Gpt3. int8 (): 8-bit matrix multiplication for transformers at scale," *Advances in neural information processing systems*, vol. 35, pp. 30 318–30 332, 2022.

[3] E. Frantar, S. Ashkboos, T. Hoefler, and D. Alistarh, "Gptq: Accurate post-training quantization for generative pre-trained transformers," *arXiv preprint arXiv:2210.17323*, 2022.

[4] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[5] S. Saha, P. Hase, and M. Bansal, "Can language models teach? teacher explanations improve student performance via personalization," *Advances in Neural Information Processing Systems*, vol. 36, pp. 62 869–62 891, 2023.

[6] T. Chen, T. Ding, B. Yadav, I. Zharkov, and L. Liang, "Lorashear: Efficient large language model structured pruning and knowledge recovery," *arXiv preprint arXiv:2310.18356*, 2023.

[7] E. Frantar and D. Alistarh, "Sparsegpt: Massive language models can be accurately pruned in one-shot," in *International conference on machine learning*. PMLR, 2023, pp. 10 323–10 337.

[8] M. Sun, Z. Liu, A. Bair, and J. Z. Kolter, "A simple and effective pruning approach for large language models," *arXiv preprint arXiv:2306.11695*, 2023.

[9] H. Cheng, M. Zhang, and J. Q. Shi, "A survey on deep neural network pruning-taxonomy, comparison," *Analysis, and Recommendations*, 2023.

[10] H. Zheng, X. Bai, X. Liu, Z. M. Mao, B. Chen, F. Lai, and A. Prakash, "Learn to be efficient: Build structured sparsity in large language models, 2024," *URL https://arxiv. org/abs/2402.06126*.

[11] A. Zhou, Y. Ma, J. Zhu, J. Liu, Z. Zhang, K. Yuan, W. Sun, and H. Li, "Learning n: m fine-grained structured sparse neural networks from scratch," *arXiv preprint arXiv:2102.04010*, 2021.

[12] X. Ma, G. Fang, and X. Wang, "Llm-pruner: On the structural pruning of large language models," *Advances in neural information processing systems*, vol. 36, pp. 21 702–21 720, 2023.

[13] M. Xia, T. Gao, Z. Zeng, and D. Chen, "Sheared llama: Accelerating language model pre-training via structured pruning," *arXiv preprint arXiv:2310.06694*, 2023.

[14] K. Ethayarajh, "How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings," *arXiv preprint arXiv:1909.00512*, 2019.

[15] Y. An, X. Zhao, T. Yu, M. Tang, and J. Wang, "Fluctuation-based adaptive structured pruning for large language models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 10, 2024, pp. 10 865–10 873.

[16] X. Men, M. Xu, Q. Zhang, Q. Yuan, B. Wang, H. Lin, Y. Lu, X. Han, and W. Chen, "Shortgpt: Layers in large language models are more redundant than you expect," in *Findings of the Association for Computational Linguistics: ACL 2025*, 2025, pp. 20 192–20 204.

[17] A. Gromov, K. Tirumala, H. Shapourian, P. Glorioso, and D. A. Roberts, "The unreasonable ineffectiveness of the deeper layers," *arXiv preprint arXiv:2403.17887*, 2024.

[18] D. Filters'Importance, "Pruning filters for efficient convnets," *arXiv preprint arXiv:1608.08710*, 2016.

[19] Y. Li, L. Niu, X. Zhang, K. Liu, J. Zhu, and Z. Kang, "E-sparse: Boosting the large language model inference through entropy-based n: M sparsity," *arXiv preprint arXiv:2310.15929*, 2023.

[20] L. Yin, Y. Wu, Z. Zhang, C.-Y. Hsieh, Y. Wang, Y. Jia, G. Li, A. Jaiswal, M. Pechenizkiy, Y. Liang *et al.*, "Outlier weighed layerwise sparsity (owl): A missing secret sauce for pruning llms to high sparsity," *arXiv preprint arXiv:2310.05175*, 2023.

[21] G. Ling, Z. Wang, and Q. Liu, "Slimgpt: Layer-wise structured pruning for large language models," *Advances in Neural Information Processing Systems*, vol. 37, pp. 107 112–107 137, 2024.

[22] Y.-C. Hsu, T. Hua, S. Chang, Q. Lou, Y. Shen, and H. Jin, "Language model compression with weighted low-rank factorization," *arXiv preprint arXiv:2207.00112*, 2022.

[23] M. Xu, Y. L. Xu, and D. P. Mandic, "Tensorgpt: Efficient compression of the embedding layer in llms based on the tensor-train decomposition," *arXiv e-prints*, pp. arXiv–2307, 2023.

[24] Z. Yuan, Y. Shang, Y. Song, D. Yang, Q. Wu, Y. Yan, and G. Sun, "Asvd: Activation-aware singular value decomposition for compressing large language models," *arXiv preprint arXiv:2312.05821*, 2023.

[25] J. Guo, X. Chen, Y. Tang, and Y. Wang, "Slimllm: Accurate structured pruning for large language models," *arXiv preprint arXiv:2505.22689*, 2025.

[26] S. Merity, C. Xiong, J. Bradbury, and R. Socher, "Pointer sentinel mixture models," *arXiv preprint arXiv:1609.07843*, 2016.

[27] M. Marcus, B. Santorini, and M. A. Marcinkiewicz, "Building a large annotated corpus of english: The penn treebank," *Computational linguistics*, vol. 19, no. 2, pp. 313–330, 1993.

[28] D. Paperno, G. Kruszewski, A. Lazaridou, N.-Q. Pham, R. Bernardi, S. Pezzelle, M. Baroni, G. Boleda, and R. Fernández, "The lambada dataset: Word prediction requiring a broad discourse context," in *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: Long papers)*, 2016, pp. 1525–1534.

[29] P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord, "Think you have solved question answering? try arc, the ai2 reasoning challenge," *arXiv preprint arXiv:1803.05457*, 2018.

[30] Y. Bisk, R. Zellers, R. Le Bras, J. Gao, and Y. Choi, "Reasoning about physical commonsense in natural language," 2019.

[31] C. Clark, K. Lee, M.-W. Chang, T. Kwiatkowski, M. Collins, and K. Toutanova, "Boolq: Exploring the surprising difficulty of natural yes/no questions," *arXiv preprint arXiv:1905.10044*, 2019.

[32] K. Sakaguchi, R. Le Bras, C. Bhagavatula, and Y. Choi, "An adversarial winograd schema challenge at scale," *arXiv preprint arXiv:1907.10641*, vol. 6, 2019.

[33] T. Mihaylov, P. Clark, T. Khot, and A. Sabharwal, "Can a suit of armor conduct electricity? a new dataset for open book question answering," *arXiv preprint arXiv:1809.02789*, 2018.

[34] R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, and Y. Choi, "Hellaswag: Can a machine really finish your sentence?" *arXiv preprint arXiv:1905.07830*, 2019.

[35] R. Taori, I. Gulrajani, T. Zhang, Y. Dubois, X. Li, C. Guestrin, P. Liang, and T. B. Hashimoto, "Stanford alpaca: An instruction-following llama model," 2023.