STABILITY-AWARE STRUCTURED PRUNING OF LARGE LANGUAGE MODELS WITH DYNAMIC SEARCH SPACES

Anonymous authors
Paper under double-blind review

ABSTRACT

Structured pruning reduces the memory and computational costs of large language models (LLMs) by removing complete attention heads and multilayer perceptron (MLP) neurons. However, importance estimates derived from limited calibration data can vary across samples, and fixed search boundaries restrict adaptation to optimization feedback. We propose DSASP, a structured pruning framework that combines cross-sample stability assessment with dynamic search spaces. DSASP augments first-order Taylor and empirical-Fisher importance estimates with two complementary statistics: Importance Variance, which measures fluctuations in score magnitude, and Cross-sample Ranking Consistency, which measures stability in relative ordering. A Tree-structured Parzen Estimator optimizes the four scoring weights using proxy perplexity within an adaptive search region. Improvements shift and contract this region, whereas sustained stagnation expands it. Across LLaMA-7B and Vicuna-7B at pruning ratios of 20% and 40%, DSASP achieves the lowest perplexity among the evaluated pruning methods on both WikiText-2 and PTB and the highest mean zero-shot accuracy in all four settings. Ablation studies support the contribution of both the combined stability signals and the dynamic search mechanism. These results establish cross-sample stability and adaptive configuration search as complementary tools for preserving model performance under structured pruning.

1 INTRODUCTION

Structured pruning provides a practical approach to reducing the memory and computational costs of large language models (LLMs). By removing complete computational units, such as attention heads and feed-forward network neurons, it produces smaller models while retaining hardware-friendly dense computation. SliceGPT, for example, reduces model dimensions by removing rows and columns from transformed weight matrices (Ashkboos et al., 2024). Nevertheless, determining which structures can be removed remains challenging: pruning must reduce model capacity while preserving the components required for language modeling and downstream tasks, typically using only a limited calibration set.

A central difficulty is that structural importance is estimated from input-dependent evidence. Aggregating sample-wise estimates into a single score conceals whether a structure contributes consistently across calibration samples. Two structures with similar average importance may behave differently: one may remain consistently important across inputs, whereas the other may derive its average from a few unusually large contributions. Their relative positions in the importance ranking may also vary across samples. Such differences are important because pruning decisions depend on comparisons among structures rather than isolated score magnitudes. Moreover, optimizing an objective on limited calibration data does not necessarily improve the resulting model; minimizing reconstruction error may overfit the calibration set and degrade downstream performance (Shin et al., 2024). These observations motivate evaluating the reliability of the evidence used for structural selection.

We characterize this reliability from two complementary perspectives: variation in importance magnitude and variation in relative rank. These properties capture different forms of instability.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Importance scores may fluctuate substantially while preserving their relative ordering, whereas small changes between closely scored structures may reverse their rankings. Magnitude variation alone therefore cannot characterize ranking stability, while ranking consistency does not retain information about absolute score fluctuations. Both signals provide supplementary evidence for structural evaluation, with stability complementing rather than replacing the estimated contribution of each structure.

Combining these criteria introduces a second source of uncertainty: selecting their fusion weights and determining which region of the configuration space should be explored. Adaptive pruning methods have demonstrated that Bayesian optimization can identify effective calibration and importance-estimation configurations (Kong et al., 2025). However, existing Bayesian search settings optimize candidate configurations within predefined global bounds; although the candidate distribution evolves, the admissible search region itself remains fixed. Consequently, the search cannot concentrate its boundaries around promising regions or broaden them when optimization stagnates.

The two sources of uncertainty are coupled. The reliability of sample-wise importance evidence affects which fusion weights should be preferred, while the selected weights reshape structural rankings and determine the feedback received by the optimizer. Unstable evidence may therefore favor calibration-specific configurations, whereas a fixed search region may limit the optimizer's ability to refine or revise these configurations efficiently. Addressing structural stability and weight-space exploration jointly is thus necessary for reliable pruning decisions.

We propose DSASP, a structured pruning framework that combines cross-sample stability assessment with a dynamically adjusted search space. DSASP retains sample-wise importance estimates for attention heads and MLP neurons. It combines first-order Taylor sensitivity with second-order Hessian curvature information, for which the empirical Fisher diagonal serves as a tractable gradient-based surrogate. DSASP further introduces Importance Variance to measure cross-sample dispersion in importance magnitude and Cross-sample Ranking Consistency to quantify agreement in relative ordering. The resulting structural score balances estimated sensitivity with the reliability of the underlying sample-wise evidence.

Specifically, the main contributions can be summarized as:

- **Stability-aware structural scoring.** We combine first-order Taylor sensitivity and an empirical-Fisher curvature surrogate with complementary measures of cross-sample score dispersion and ranking consistency.
- **Dynamic configuration search.** We introduce feedback-driven search boundaries for TPE-based weight optimization, contracting the admissible region around promising configurations and expanding it when progress stalls.
- **Empirical validation.** We evaluate two 7B models at two pruning ratios and use ablations to assess the combined stability signals and the dynamic search mechanism.

2 Related Work

Recent studies on structured pruning explore importance estimation, automated compression, and stability-aware methods, emphasizing calibration data and configuration search to guide model-structure selection.

2.1 Structured pruning and redundancy in Transformers.

Structured pruning removes parameter groups to produce executable networks with smaller computational dimensions. For example, Sheared LLaMA learns a compact target architecture through pruning and continued pretraining (Xia et al., 2024); SliceGPT transforms representations before deleting rows and columns, preserving dense matrix operations (Ashkboos et al., 2024); and FLAP uses activation fluctuations to estimate redundancy and adapt pruning ratios without retraining (An et al., 2024). These methods enable hardware-friendly compression through architecture selection, invariance-based slicing, and fluctuation-aware allocation.

Other algorithms address dependencies and pruning allocation: Coupled Components Elimination groups interdependent Transformer components to preserve the computation graph (Zhang et al., 2024); GAP allocates sparsity globally according to model capability (Ban et al., 2025); From Local to Global coordinates structured decisions across the model (Wang et al., 2026); and TransAct reduces transitional activations to obtain an intra-module low-rank architecture (Shen et al., 2024).

2.2 Structured pruning and redundancy in Transformers.

Pruning criteria use weights, activations, gradients, and curvature to estimate removal effects. Wanda combines weight magnitude with input-activation statistics without retraining (Sun et al., 2024); FISTAPruner develops layer-wise post-training scores for depth-dependent sensitivity (Zhao et al., 2025a); HyWIA combines fine- and coarse-grained importance evidence (Liu et al., 2025); and Less is KEN introduces a non-parametric criterion without an additional importance model (Mastromattei & Zanzotto, 2024).

Their effectiveness also depends on calibration data: Rethinking Pruning shows that aggressively minimizing reconstruction error on a small calibration set can impair downstream generalization (Shin et al., 2024); calibration language affects multilingual pruning performance (Kurz et al., 2026); and AdaPruner addresses calibration sensitivity by jointly searching samples and importance-estimation configurations (Kong et al., 2025). These findings motivate retaining sample-wise evidence when evaluating structural importance.

2.3 Automated compression and adaptive search.

Automated compression adapts architectures, policies, or compact-model construction to resource constraints. Probe Pruning uses lightweight probing for dynamic pruning (Le et al., 2025); pruning with knowledge distillation uses teacher supervision to recover smaller-model capabilities (Muralidharan et al., 2024); AlphaPruning uses heavy-tailed self-regularization for layer-wise sparsity allocation (Lu et al., 2024); and customized expert pruning specializes a general LLM without conventional post-training (Zhao et al., 2025b). These approaches adapt compression to model and task characteristics.

Recent methods also adapt pruning evidence or recovery data: Wanda++ incorporates regional gradients into importance estimation (Yang et al., 2025), while DRPruning adjusts the training distribution to reduce uneven cross-domain degradation (Deng et al., 2025).

Because candidate evaluation requires pruning and inference, Bayesian optimization is useful. AdaPruner applies TPE-based search to calibration data and importance metrics (Kong et al., 2025).

2.4 Stability and robustness in pruning.

Stability and robustness concern distinct sources of variation. Truthfulness-aware pruning examines downstream behavior beyond aggregate accuracy (Fu et al., 2025), while MoreauPruner uses a Moreau-envelope formulation to address weight-perturbation sensitivity (Wang et al., 2024). Both highlight the need to specify the variation a criterion must withstand.

The LLM Surgeon uses scalable Kronecker-factored curvature approximations for structural removal and retained-weight updates (van der Ouderaa et al., 2024); LaCo reduces depth through layer collapse and parameter merging (Yang et al., 2024); and Dual Activation-Weight Sparsity combines activation and weight sparsity through training-free magnitude routing (Sun et al., 2026).

3 Method

3.1 Framework Overview

This paper focuses on the two methodological innovations of DSASP: multidimensional stability-aware importance evaluation and dynamic search-space adaptation.

Let $M(\theta)$ denote the pretrained Transformer, where θ is the set of model parameters, and let $\mathcal{G} = \{g_1, \ldots, g_N\}$ denote its N prunable structures, comprising attention heads and MLP neurons.

Given a calibration set $\mathcal{D} = \{x_1, \ldots, x_m\}$ of m samples, DSASP retains sample-wise Taylor and empirical-Fisher scores for each structure. These scores support computing Importance Variance and Cross-sample Ranking Consistency (RC), yielding four components for structural scoring.

As illustrated in Figure 1, DSASP consists of stability-aware scoring and dynamic configuration search. The scoring module combines structural sensitivity with cross-sample magnitude dispersion and ranking consistency. The search module treats the four scoring weights as a configuration vector and adjusts the admissible region using proxy performance. The best-evaluated configuration then constructs the final pruning mask.

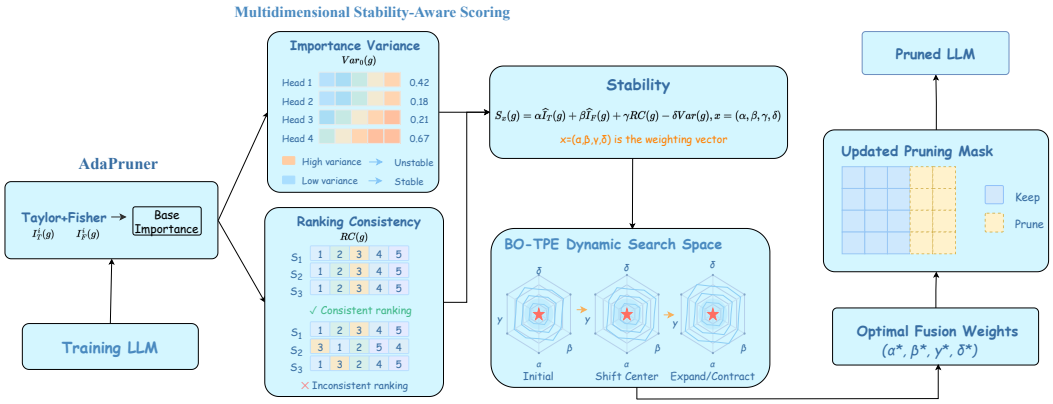

Figure 1: Overview of DSASP. Sample-wise importance estimates support stability-aware structural scoring, and proxy performance guides dynamic TPE-based search over the scoring weights.

3.2 Multidimensional Stability-Aware Importance Evaluation

Let g denote a prunable attention head or MLP neuron, and let $I^i(g)$ be its sample-wise importance on the ith of m calibration samples. The Taylor and empirical-Fisher components are normalized separately as $\widehat{I}_T(g)$ and $\widehat{I}_F(g)$; their exact definitions and normalization are given in Appendix A. Mean importance alone does not reveal how consistently a structure contributes across calibration inputs. DSASP therefore evaluates two complementary properties: dispersion in score magnitude and consistency in relative rank.

3.2.1 Importance Variance

For structure g, the mean importance over m calibration samples is

$$\mu(g) = \frac{1}{m} \sum_{i=1}^{m} I^i(g) \tag{1}$$

Here, g denotes the prunable attention head or MLP neuron under evaluation, m is the number of calibration samples, i indexes a calibration sample, $I^i(g)$ is the sample-wise importance of structure g on sample i, and $\mu(g)$ is its mean importance across the m samples.

We measure cross-sample fluctuation as

$$\mathrm{Var}_0(g) = \sqrt{\frac{1}{m-1} \sum_{i=1}^{m} \big(I^i(g) - \mu(g)\big)^2} \tag{2}$$

where $\mathrm{Var}_0(g)$ quantifies absolute dispersion. To account for scale differences between projections, we standardize this statistic within each projection:

$$\mathrm{Var}(g) = \frac{\mathrm{Var}_0(g) - \mu_V}{\sigma_V} \tag{3}$$

Where μ_V and σ_V are the mean and standard deviation of Var_0 across structures in the same projection. Thus, $\mathrm{Var}(g) > 0$ indicates above-average dispersion, and $\mathrm{Var}(g) < 0$ indicates below-average dispersion. The term $-\delta\,\mathrm{Var}(g)$ in Equation (8) penalizes the former and rewards the latter.

3.2.2 Cross-sample Ranking Consistency

Magnitude dispersion does not fully characterize ranking stability: a structure can have fluctuating scores but a stable relative rank. We therefore introduce Cross-sample Ranking Consistency (RC) to measure agreement in structural ordering across samples.

For calibration sample i, we rank candidate structures by their sample-wise importance in descending order. Let $R_i(g)$ denote the rank of structure g. Its cross-sample rank dispersion is

$$\sigma_R(g) = \mathrm{Std}[R_1(g), R_2(g), \ldots, R_m(g)] \tag{4}$$

where $\mathrm{Std}[\cdot]$ denotes the standard deviation over the m sample-wise ranks.

For N prunable structures in the current layer, we use the standard deviation of a continuous uniform distribution over a rank interval of width N as a reference scale:

$$\sigma_R^{\max} = \frac{N}{\sqrt{12}} \tag{5}$$

The unthresholded ranking-consistency score is then

$$\mathrm{RC}_0(g) = \mathrm{clip}\left(1 - \frac{\sigma_R(g)}{\sigma_R^{\max}}, 0, 1\right) \tag{6}$$

The operator $\mathrm{clip}(z, 0, 1)$ truncates z to $[0, 1]$. Values closer to 1 indicate more consistent ranks across calibration samples. We apply a one-sided threshold so that the stability reward is relative to other structures in the same layer. Let τ denote the within-layer median of the unthresholded scores. The final reward is

$$\mathrm{RC}(g) = \max\big(\mathrm{RC}_0(g) - \tau, 0\big) \tag{7}$$

Only structures with ranking consistency above the layer median receive a positive reward.

3.2.3 Stability-Aware Structural Scoring

We combine importance and stability into the structural score

$$S_x(g) = \alpha \widehat{I}_T(g) + \beta \widehat{I}_F(g) + \gamma\,\mathrm{RC}(g) - \delta\,\mathrm{Var}(g), \quad x = (\alpha, \beta, \gamma, \delta) \tag{8}$$

where $x = (\alpha, \beta, \gamma, \delta)$ is the scoring-weight configuration: α controls the normalized first-order Taylor contribution, β controls the normalized empirical-Fisher contribution, γ controls the reward for cross-sample ranking consistency, and δ controls the penalty for cross-sample magnitude dispersion. Thus, $\widehat{I}_T(g)$ and $\widehat{I}_F(g)$ are the projection-wise normalized Taylor and empirical-Fisher scores, respectively; $\mathrm{RC}(g) \in [0, 1]$ is the thresholded ranking-consistency reward; and $\mathrm{Var}(g)$ is the standardized magnitude-dispersion statistic. The configuration vector x is distinct from a calibration input x_i. TPE optimizes these four nonnegative weights within $[0, 1]^4$.

This score combines evidence of structural contribution with evidence of its consistency across calibration inputs.

3.3 Dynamic Search Space and TPE Optimization

For each weight configuration, DSASP constructs the corresponding pruning mask and uses proxy perplexity as the black-box objective. TPE proposes configurations from historical evaluations, while the dynamic search mechanism adapts the admissible region from which those configurations can be drawn. The standard black-box objective and TPE density formulation are provided in Appendix B.

The search variable is $x = (\alpha, \beta, \gamma, \delta)$, with initial center c_0 and radius r_0. At iteration t, the admissible region is

$$\Omega_t = [c_t - r_t, c_t + r_t] \cap [0, 1]^4 \tag{9}$$

When a candidate x_t improves on the best objective value observed so far, we move the center toward that candidate and contract the radius:

$$c_{t+1} = (1 - \eta)c_t + \eta x_t \tag{10}$$

$$r_{t+1} = \rho_r r_t \tag{11}$$

If no improvement is observed for K consecutive iterations, we expand the radius:

$$r_{t+1} = \min(r_{\max}, \rho_e r_t) \tag{12}$$

where c_t and c_{t+1} are the current and updated four-dimensional region centers, while r_t and r_{t+1} are the current and updated radii applied to each coordinate. The coefficient $\eta \in (0, 1]$ controls the center update, $\rho_r < 1$ is the contraction factor, $\rho_e > 1$ is the expansion factor, K is the number of consecutive non-improving iterations that triggers expansion, and $r_{\max}$ is the maximum permitted radius. The intersection with $[0, 1]^4$ in Equation (9) enforces the global bounds on all four weights.

These updates concentrate the search around improving configurations and restore broader exploration after sustained stagnation. The best evaluated configuration x^* determines the final pruning mask, whose construction is described in Appendix C.

4 Experiments

4.1 Experimental Setup

We evaluate LLaMA-7B and Vicuna-7B at pruning ratios of 20% and 40%. Language-modeling performance is measured by perplexity (PPL) on WikiText-2 and the Penn Treebank (PTB). Zero-shot accuracy is evaluated on PIQA, BoolQ, HellaSwag, WinoGrande, ARC-e, ARC-c, and OpenBookQA (OBQA). All experiments use a single NVIDIA RTX 6000D GPU with PyTorch and Hugging Face Transformers. Table 1 summarizes the experimental configuration.

TPE optimizes the four structural scoring weights. The dynamic region is initialized with center $c_0 = (0.5, 0.5, 0.5, 0.5)$ and radius $r_0 = 0.5$. The main experiments and ablations use the respective search budgets reported in Table 1.

Table 1: Experimental configuration.

Item	**Setting**
Model	LLaMA-7B; Vicuna-7B
Pruning ratio	20%; 40%
Pruning layers	Layers 4–29
Calibration data	BookCorpus; 300 candidates, top 10 selected
Sequence length	128
Main experiment search	TPE; 150 trials $\times$ 4 candidates; 16 proxy samples
Ablation search	TPE; 25 trials; 8 proxy samples
Final PPL evaluation	WikiText-2 / PTB, 128 samples each
Zero-shot tasks	BoolQ / PIQA / HellaSwag / WinoGrande / ARC-e / ARC-c / OBQA

4.2 BASELINES

We compare DSASP with Vector (Kong et al., 2025), Magnitude (Sun et al., 2024), MoreauPruner (Wang et al., 2024), AdaPruner (Kong et al., 2025), and LLM-Pruner-E1/E2 (Ma et al., 2023). These baselines represent magnitude-based selection, Taylor-based scoring, perturbation-aware estimation, sample-aware adaptation, and first- or second-order Taylor criteria, respectively.

Specifically, Magnitude ranks candidate structures according to weight magnitude, whereas Vector uses a first-order Taylor score. MoreauPruner incorporates perturbation-aware importance estimation through a Moreau-envelope formulation, and AdaPruner jointly searches calibration samples and importance-estimation configurations. LLM-Pruner-E1 and LLM-Pruner-E2 use first- and second-order Taylor criteria, respectively. These comparisons evaluate whether the cross-sample stability signals and dynamic configuration search in DSASP provide complementary benefits over representative existing pruning criteria.

4.3 RESULTS ON LLAMA-7B

Tables 2 and 3 report results at 20% and 40% pruning. At 20%, DSASP achieves perplexities of 16.63 on WikiText-2 and 167.77 on PTB, the lowest among the evaluated pruning methods. It also achieves the highest accuracy on five of seven zero-shot tasks and the highest mean accuracy, 59.65%. At 40%, DSASP again yields the lowest perplexity, at 28.80 and 226.62, respectively. It achieves the highest accuracy on four tasks, ties for the highest on ARC-e, and attains a mean accuracy of 47.81%.

Table 2: LLaMA-7B results at 20% pruning. Lower perplexity and higher accuracy (%) are better. Best results among pruning methods are bolded. WT2 denotes WikiText-2, HellaS. denotes HellaSwag, and WinoG. denotes WinoGrande.

Method	**WT2** ↓	**PTB** ↓	**PIQA** ↑	**BoolQ** ↑	**HellaS.** ↑	**WinoG.** ↑	**ARC-e** ↑	**ARC-c** ↑	**OBQA** ↑
Magnitude	129.31	712.95	63.33	54.74	48.50	58.09	44.99	30.29	30.40
Vector	18.09	168.67	76.12	62.11	65.19	60.69	62.37	35.92	36.80
LLM-Pruner-E1	17.12	168.59	75.42	63.21	67.07	62.43	64.02	37.40	40.60
LLM-Pruner-E2	32.08	224.05	70.51	54.43	51.46	54.46	50.13	30.12	34.60
MoreauPruner	20.90	190.84	65.29	65.41	53.73	56.67	48.02	31.83	31.80
AdaPruner	17.50	171.78	75.62	**66.65**	66.03	62.5	**65.03**	37.50	39.50
DSASP	**16.63**	**167.77**	**76.17**	65.17	**67.66**	**64.72**	64.81	**38.05**	**41.00**

Table 3: LLaMA-7B results at 40% pruning. Lower perplexity and higher accuracy (%) are better.

Method	**WT2** ↓	**PTB** ↓	**PIQA** ↑	**BoolQ** ↑	**HellaS.** ↑	**WinoG.** ↑	**ARC-e** ↑	**ARC-c** ↑	**OBQA** ↑
Magnitude	347.64	1260.55	57.51	49.65	36.09	52.01	33.80	25.43	26.80
Vector	33.71	235.16	66.70	48.68	50.10	53.59	47.31	28.46	31.40
LLM-Pruner-E1	32.15	229.88	67.18	48.77	48.63	**54.62**	46.52	**30.72**	32.12
LLM-Pruner-E2	39.56	254.04	68.39	50.92	45.74	53.59	46.25	27.73	34.40
MoreauPruner	50.30	292.58	56.80	50.55	36.17	51.54	36.53	24.91	27.20
AdaPruner	31.34	227.14	67.22	50.14	49.38	**54.62**	**47.94**	29.44	34
DSASP	**28.8**	**226.62**	**68.77**	**51.13**	**50.49**	53.12	**47.94**	27.99	**35.20**

Figure 2(a) summarizes mean zero-shot accuracy. DSASP provides the strongest mean accuracy at 20% pruning and remains competitive at 40%, alongside consistently lower perplexity. These results show that its benefits extend across both evaluated compression levels, with the clearest joint advantage in language modeling and mean task accuracy at 20% pruning.

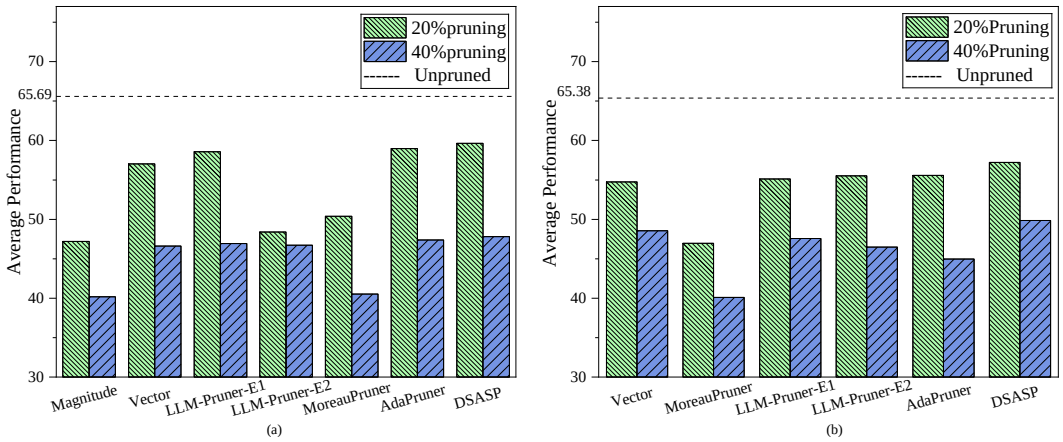

Figure 2: Mean accuracy across seven zero-shot tasks for LLaMA-7B (a) and Vicuna-7B (b). Dashed lines indicate the unpruned models; bars show results at 20% and 40% pruning.

4.4 Results on Vicuna-7B

Tables 4 and 5 report results for Vicuna-7B. At 20% pruning, DSASP achieves perplexities of 19.96 on WikiText-2 and 195.30 on PTB, with mean zero-shot accuracy of 57.24%. At 40%, the corresponding values are 35.28, 262.80, and 49.86%. DSASP attains the lowest perplexity on both datasets and the highest mean zero-shot accuracy among the evaluated pruning methods at both ratios.

Figure 2(b) shows a particularly clear mean-accuracy advantage at 40% pruning. Together with the LLaMA-7B results, these findings support the effectiveness of DSASP on both base and instruction-tuned checkpoints within the same architectural family.

Table 4: Vicuna-7B results at 20% pruning. Lower perplexity and higher accuracy (%) are better.

Method	**WT2** ↓	**PTB** ↓	**PIQA** ↑	**BoolQ** ↑	**HellaS.** ↑	**WinoG.** ↑	**ARC-e** ↑	**ARC-c** ↑	**OBQA** ↑
Vector	22.49	207.07	74.48	58.44	60.75	61.75	60.05	35.44	32.40
MoreauPruner	23.98	197.40	60.99	**64.62**	44.08	58.09	42.42	29.44	29.00
LLM-Pruner-E1	20.68	198.48	71.06	56.25	63.38	63.85	61.17	36.33	33.80
LLM-Pruner-E2	20.34	198.03	71.54	56.98	63.59	63.09	62.49	37.03	33.90
AdaPruner	21.08	202.26	72.03	55.66	63.88	**64.77**	62.40	36.08	34.2
DSASP	**19.96**	**195.30**	**74.86**	54.65	**65.14**	64.01	**64.77**	**37.63**	**39.60**

Table 5: Vicuna-7B results at 40% pruning. Lower perplexity and higher accuracy (%) are better.

Method	**WT2** ↓	**PTB** ↓	**PIQA** ↑	**BoolQ** ↑	**HellaS.** ↑	**WinoG.** ↑	**ARC-e** ↑	**ARC-c** ↑	**OBQA** ↑
Vector	43.58	286.18	68.50	**50.34**	50.17	54.93	53.16	29.78	33.00
MoreauPruner	67.83	360.29	53.21	50.06	34.65	53.51	34.89	26.71	27.60
LLM-Pruner-E1	39.99	274.77	67.30	41.28	48.83	59.19	52.61	29.61	34.20
LLM-Pruner-E2	36.58	272.63	67.03	43.79	46.52	58.33	48.61	28.24	32.80
AdaPruner	35.77	266.48	67.90	46.64	38.69	58.96	**55.09**	25.51	22.00
DSASP	**35.28**	**262.80**	**69.53**	48.87	**50.66**	**60.14**	51.68	**31.74**	**36.40**

4.5 Ablation Studies

We assess the two main components using the variants defined in Table 6. The full variant includes Importance Variance, RC, and dynamic search. The norcv variant removes both stability signals while retaining dynamic search. The nodyn variant retains stability-aware scoring but uses fixed search boundaries. Within each ablation setting, the model, pruning ratio, search budget, and evaluation procedure are held constant across variants. The ablation search budget differs from that of the main experiments, as specified in Table 1. Results are reported in Tables 7 and 8.

Table 6: Ablation variants and the components they assess.

Variant	Components	Comparison
full	RC + Var + dynamic search	Complete DSASP
norcv	Dynamic search	Contribution of the combined RC and Var signals
nodyn	RC + Var + fixed search	Effect of adapting the search boundaries

Table 7: Ablation results on LLaMA-7B. Accuracy is reported as a percentage; Average is the mean across seven tasks. Best values are bolded.

Method	PIQA ↑	BoolQ ↑	HellaS. ↑	WinoG. ↑	ARC-e ↑	ARC-c ↑	OBQA ↑	Average ↑
full	**69.59**	53.09	**53.33**	**56.04**	**49.54**	**30.46**	35.8	**49.69**
norcv	69.37	**54.19**	52.76	53.83	47.9	28.07	33.2	48.47
nodyn	69.42	51.56	52.05	55.01	47.22	29.61	**36**	48.69

Table 8: Ablation results on Vicuna-7B. Accuracy is reported as a percentage; Average is the mean across seven tasks. Best values are bolded.

Method	PIQA ↑	BoolQ ↑	HellaS. ↑	WinoG. ↑	ARC-e ↑	ARC-c ↑	OBQA ↑	Average ↑
full	75.63	**60.64**	**67.32**	64.8	**66.79**	39.25	**41.6**	**59.43**
norcv	74.86	59.97	66.99	**65.67**	66.58	**40.36**	40.8	59.32
nodyn	**76.28**	59.88	66.89	65.11	64.73	36.52	39	58.34

The full variant achieves the highest mean accuracy on both checkpoints. On LLaMA-7B, it reaches 49.69%, compared with 48.47% for norcv and 48.69% for nodyn. On Vicuna-7B, the corresponding values are 59.43%, 59.32%, and 58.34%. The comparison with norcv evaluates the combined contribution of the two stability signals, while the comparison with nodyn isolates the effect of adapting the search boundaries.

These results support the joint use of cross-sample stability information and dynamic configuration search. The ablation design evaluates Importance Variance and RC as a combined module.

5 Conclusion

We introduced DSASP, a structured pruning framework that combines cross-sample stability assessment with dynamic configuration search. DSASP augments Taylor and empirical-Fisher importance estimates with magnitude dispersion and ranking consistency, then uses TPE with feedback-driven search boundaries to optimize their weights. Across LLaMA-7B and Vicuna-7B at 20% and 40% pruning, DSASP achieves the lowest perplexity among the evaluated pruning methods on both language-modeling datasets in all four settings and the highest mean zero-shot accuracy in three. Ablations support the contribution of both the combined stability signals and dynamic search. The results show that accounting for the consistency of calibration evidence, together with adapting configuration search, can help preserve LLM performance under structured pruning. Future work will further evaluate broader model families and scales and investigate more robust calibration strategies.

AI USE STATEMENT

Generative AI tools were used solely for language polishing after the manuscript had been completed by the authors.

REFERENCES

Yongqi An, Xu Zhao, Tao Yu, Ming Tang, and Jinqiao Wang. Fluctuation-Based Adaptive Structured Pruning for Large Language Models. In *AAAI*, pp. 10865–10873, 2024.

Saleh Ashkboos, Maximilian Croci, Marcelo Gennari do Nascimento, Torsten Hoefler, and James Hensman. SliceGPT: Compress Large Language Models by Deleting Rows and Columns. In *ICLR*, 2024.

Zhihua Ban, Haotian Ma, Siheng Zhang, Shengyu Liu, Xichen Chen, and Ming Yang. GAP: a Global Adaptive Pruning Method for Large Language Models. In *EMNLP*, pp. 20898–20903, 2025.

Hexuan Deng, Wenxiang Jiao, Xuebo Liu, Jing Li, Min Zhang, and Zhaopeng Tu. DRPruning: Efficient Large Language Model Pruning through Distributionally Robust Optimization. In *ACL*, pp. 29152–29173, 2025.

Yao Fu, Runchao Li, Xianxuan Long, Haotian Yu, Xiaotian Han, Yu Yin, and Pan Li. Pruning Weights but Not Truth: Safeguarding Truthfulness While Pruning LLMs. In *Findings of EMNLP*, pp. 20750–20768, 2025.

Jun Kong, Xinge Ma, Jin Wang, and Xuejie Zhang. Sample-aware Adaptive Structured Pruning for Large Language Models. In *AAAI*, pp. 17938–17946, 2025.

Simon Kurz, Jian-Jia Chen, Lucie Flek, and Zhixue Zhao. On the Limitations of Language-targeted Pruning: Investigating the Calibration Language Impact in Multilingual LLM Pruning. *Transactions of the Association for Computational Linguistics*, pp. 167–192, 2026.

Qi Le, Enmao Diao, Ziyan Wang, Xinran Wang, Jie Ding, Li Yang, and Ali Anwar. Probe Pruning: Accelerating LLMs through Dynamic Pruning via Model-Probing. In *ICLR*, 2025.

Jun Liu, Zhenglun Kong, Pu Zhao, Changdi Yang, Xuan Shen, Hao Tang, Geng Yuan, Wei Niu, Wenbin Zhang, Xue Lin, Dong Huang, and Yanzhi Wang. Toward Adaptive Large Language Models Structured Pruning via Hybrid-grained Weight Importance Assessment. In *AAAI*, pp. 18879–18887, 2025.

Haiquan Lu, Yefan Zhou, Shiwei Liu, Zhangyang Wang, Michael W. Mahoney, and Yaoqing Yang. AlphaPruning: Using Heavy-Tailed Self Regularization Theory for Improved Layer-wise Pruning of Large Language Models. In *NeurIPS*, 2024.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. LLM-Pruner: On the Structural Pruning of Large Language Models. In *NeurIPS*, pp. 21702–21720, 2023.

Michele Mastromattei and Fabio Massimo Zanzotto. Less is KEN: a Universal and Simple Non-Parametric Pruning Algorithm for Large Language Models. In *Findings of ACL*, pp. 11361–11374, 2024.

Saurav Muralidharan, Sharath Turuvekere Sreenivas, Raviraj Joshi, Marcin Chochowski, Mostofa Patwary, Mohammad Shoeybi, Bryan Catanzaro, Jan Kautz, and Pavlo Molchanov. Compact Language Models via Pruning and Knowledge Distillation. In *NeurIPS*, pp. 41076–41102, 2024.

Bowen Shen, Zheng Lin, Daren Zha, Wei Liu, Jian Luan, Bin Wang, and Weiping Wang. Pruning Large Language Models to Intra-module Low-rank Architecture with Transitional Activations. In *Findings of ACL*, pp. 9781–9793, 2024.

Sungbin Shin, Wonpyo Park, Jaeho Lee, and Namhoon Lee. Rethinking Pruning Large Language Models: Benefits and Pitfalls of Reconstruction Error Minimization. In *EMNLP*, pp. 1182–1191, 2024.

Luoyang Sun, Guangyan Li, Cheng Deng, Haifeng Zhang, Jian Zhao, Yongqiang Tang, Wensheng Zhang, and Jun Wang. Dual Activation-Weight Sparsity: A Training-Free Framework for Efficient Large Language Model Compression. In *ACL*, pp. 8350–8366, 2026.

Mingjie Sun, Zhuang Liu, Anna Bair, and Zico Kolter. A Simple and Effective Pruning Approach for Large Language Models. In *ICLR*, 2024.

Tycho F. A. van der Ouderaa, Markus Nagel, Mart van Baalen, Yuki M. Asano, and Tijmen Blankevoort. The LLM Surgeon. In *ICLR*, 2024.

Zixiao Wang, Jingwei Zhang, Wenqian Zhao, Farzan Farnia, and Bei Yu. MoreauPruner: Robust Pruning of Large Language Models against Weight Perturbations. arXiv preprint arXiv:2406.07017, 2024.

Ziyan Wang, Enmao Diao, Qi Le, Pu Wang, Minwoo Lee, Shu ping Yeh, Evgeny Stupachenko, Hao Feng, and Li Yang. From Local to Global: Revisiting Structured Pruning Paradigms for Large Language Models. In *ACL*, pp. 35720–35739, 2026.

Mengzhou Xia, Tianyu Gao, Zhiyuan Zeng, and Danqi Chen. Sheared LLaMA: Accelerating Language Model Pre-training via Structured Pruning. In *ICLR*, 2024.

Yifan Yang, Kai Zhen, Bhavana Ganesh, Aram Galstyan, Goeric Huybrechts, Markus Müller, Jonas M. Kübler, Rupak Vignesh Swaminathan, Athanasios Mouchtaris, Sravan Babu Bodapati, Nathan Susanj, Zheng Zhang, Jack FitzGerald, and Abhishek Kumar. Wanda++: Pruning Large Language Models via Regional Gradients. In *Findings of ACL*, pp. 4321–4333, 2025.

Yifei Yang, Zouying Cao, and Hai Zhao. LaCo: Large Language Model Pruning via Layer Collapse. In *Findings of EMNLP*, pp. 6401–6417, 2024.

Honghe Zhang, Xiaolong Shi, Jingwei Sun, and Guangzhong Sun. Structured Pruning for Large Language Models Using Coupled Components Elimination and Minor Fine-tuning. In *Findings of NAACL*, pp. 1–12, 2024.

Pengxiang Zhao, Hanyu Hu, Ping Li, Yi Zheng, Zhefeng Wang, and Xiaoming Yuan. FISTAPruner: Layer-wise Post-training Pruning for Large Language Models. In *EMNLP*, pp. 29469–29487, 2025a.

Yiran Zhao, Guizhen Chen, Kenji Kawaguchi, Lidong Bing, and Wenxuan Zhang. Pruning General Large Language Models into Customized Expert Models. In *Findings of ACL*, pp. 23377–23391, 2025b.

A Base Importance Estimation

A.1 Basic Structural Importance Estimation

For calibration sample i, let $\mathcal{L}_i$ denote the model loss. A second-order Taylor expansion involves the Hessian, which is impractical to form or store for an LLM. For negative log-likelihood objectives, the Fisher information equals the expected Hessian under standard regularity conditions and a model-distributed expectation. DSASP uses squared sample gradients as an empirical-Fisher surrogate for diagonal curvature; this surrogate need not equal the finite-sample Hessian.

The sample-wise importance of a structure combines first-order Taylor sensitivity with this empirical-Fisher surrogate:

$$I^i(g) = I^i_T(g) + I^i_F(g) \tag{13}$$

where the two components are

$$I^i_T(g) = \sum_{\theta_j \in g} \left| \theta_j \frac{\partial \mathcal{L}_i}{\partial \theta_j} \right|, \qquad I^i_F(g) = \sum_{\theta_j \in g} \frac{1}{2} \theta_j^2 \left(\frac{\partial \mathcal{L}_i}{\partial \theta_j} \right)^2 \tag{14}$$

Here, i indexes the calibration sample, g is a prunable attention head or MLP neuron, θ_j is the jth model parameter belonging to g, and $\partial \mathcal{L}_i / \partial \theta_j$ is the sample gradient. The quantities $I^i_T(g)$ and $I^i_F(g)$ denote the first-order Taylor and empirical-Fisher terms for structure g on sample i, respectively, and their sum $I^i(g)$ supplies the sample-wise evidence used to compute the stability statistics.

Before score fusion, the Taylor and empirical-Fisher components are normalized separately using robust min–max scaling:

$$\widehat{I}(g) = \mathrm{clip}\left(\frac{I(g) - P_2}{P_{98} - P_2}, 0, 1 \right) \tag{15}$$

where $I(g)$ is the unnormalized Taylor or empirical-Fisher component for structure g, $\widehat{I}(g)$ is its normalized value, and P_2 and P_{98} are the 2nd and 98th percentiles within the corresponding projection. The operator $\mathrm{clip}(z, 0, 1)$ truncates z to the interval $[0, 1]$.

B Black-Box Objective and TPE Density Formulation

For a scoring-weight configuration x, we compute Equation (8) and construct the corresponding pruning mask. We evaluate the pruned model on a small proxy set to limit search cost:

$$y_t = f(x_t) = \mathrm{PPL}(M_{x_t}; \mathcal{D}_{\mathrm{proxy}}) \tag{16}$$

Here, t indexes the search iteration, x_t is the candidate weight configuration, M_{x_t} is the model pruned under that configuration, $\mathcal{D}_{\mathrm{proxy}}$ is the proxy evaluation set, and $y_t = f(x_t)$ is the resulting perplexity objective. The search seeks a configuration that minimizes proxy perplexity within the admissible region:

$$x^* = \arg\min_{x \in \Omega_t} f(x) \tag{17}$$

Here, x^* denotes the best configuration sought within the iteration-dependent admissible region Ω_t. Evaluating $f(x)$ involves discrete structural ranking, mask construction, and model inference. We therefore use TPE to optimize this black-box objective and retain the best evaluated configuration across the search history.

Given the evaluation history $\mathcal{H}_t = \{(x_j, y_j)\}$ at iteration t, where x_j is the jth evaluated weight configuration and $y_j = f(x_j)$ is its proxy perplexity, we use a prescribed quantile y^* of the observed objective values to partition configurations into lower- and higher-perplexity groups:

$$\mathcal{G}_t = \{x_j \mid y_j < y^*\}, \qquad \mathcal{B}_t = \{x_j \mid y_j \geq y^*\} \tag{18}$$

Here, $\mathcal{G}_t$ is the lower-perplexity (good) group and $\mathcal{B}_t$ is the higher-perplexity (bad) group. We fit Parzen density estimates to the two groups:

$$H_{\text{good}}(x) \approx p(x \mid y < y^*), \qquad H_{\text{bad}}(x) \approx p(x \mid y \geq y^*) \tag{19}$$

Candidate preference is expressed as a log-density ratio:

$$A_t(x) = \log \frac{H_{\text{good}}(x) + \epsilon}{H_{\text{bad}}(x) + \epsilon} \tag{20}$$

Here, $H_{\text{good}}(x)$ and $H_{\text{bad}}(x)$ approximate the conditional densities of configurations in the two groups, $\epsilon > 0$ is a numerical stabilizer, and $A_t(x)$ is the preference assigned to configuration x at iteration t. A larger $A_t(x)$ indicates greater relative density among historically better configurations. TPE uses this preference to propose candidates within the current search boundaries.

C Pruning Mask Construction

This appendix preserves the complete content of the original Section 3.5.

After search, we use the best evaluated configuration x^* to recompute Equation (8) and rank candidate structures. For a target pruning ratio ρ and N candidate structures, the number removed is

$$N_p = \rho N \tag{21}$$

We place the N_p lowest-scoring structures in the pruning set $\mathcal{G}_p$ and define the binary mask as

$$m_g = \begin{cases} 0, & g \in \mathcal{G}_p, \\ 1, & g \notin \mathcal{G}_p. \end{cases} \tag{22}$$

Here, ρ is the prescribed pruning ratio, N is the number of candidate structures, N_p is the number selected for removal, $\mathcal{G}_p$ is the set of the N_p lowest-scoring structures, and m_g is the binary mask assigned to structure g. Attention heads and MLP neurons are scored separately and subject to separate pruning-ratio constraints. A minimum retention constraint is imposed in each layer to prevent excessive local pruning. Masks are applied consistently across the corresponding coupled projections to preserve structural compatibility.