How Does Pruning Change Decisions in Large Language Models?

Jiang Li[1] **Tian Lan**[2] **Chenxi Zhou**[3] **Zehua Duo**[1] **Xianpeng Shang**[1]
Guanglai Gao[1] **Xiangdong Su**[1]*

[1]College of Computer Science, Inner Mongolia University
[2]Kyoto University
[3]Institute of Automation, Chinese Academy of Sciences

lijiangimu@gmail.com, cssxd@imu.edu.cn

Abstract

Post-training pruning reduces the memory and computation costs of large language models (LLMs), and N:M semi-structured sparsity provides a hardware-friendly pruning pattern for efficient inference. However, pruned models are often evaluated mainly by perplexity and average downstream accuracy, which are useful aggregate metrics but do not fully explain how pruning changes individual candidate-level decisions. We evaluate magnitude pruning, Wanda, SparseGPT, Wanda++, and our proposed MR-GOBS across multiple model families and sparsity settings. We further conduct a paired decision-level evaluation that measures true-label confidence, answer margin, prediction flips, candidate-level distribution shift, and near-boundary vulnerability. Across the evaluated configurations, pruning generally weakens decision reliability by lowering confidence in the correct answer, shrinking answer margins, and increasing correct-to-wrong flips, especially on near-boundary examples. To examine whether better reconstruction reliably improves decision reliability, we revisit SparseGPT through its local least-squares reconstruction formulation and refine its N:M mask-selection step. MR-GOBS uses GroupOBS, a set-level OBS reconstruction cost for jointly pruning a group of weights, to score each legal local N:M mask. It then selects masks using a minimax-regret criterion over multiple calibration views. Although MR-GOBS often improves average perplexity over SparseGPT, these gains do not always translate into better downstream accuracy or decision-level metrics. Our findings suggest that reliable evaluation of pruned LLMs requires both aggregate performance metrics and decision-level diagnostics[2].

1 Introduction

Large language models (LLMs) have become a general-purpose foundation for language understanding, reasoning, and generation [1, 2, 3, 4, 5, 6]. As model size and deployment cost continue to grow, post-training compression has become important for reducing memory and computation without full retraining. Post-training pruning is a practical approach in this direction. Existing methods, including magnitude pruning, Wanda, SparseGPT, and their variants, show that pretrained LLMs can tolerate substantial sparsity when pruning decisions use weight magnitude, activation statistics, or local reconstruction objectives [7, 8, 9, 10, 11, 12]. The quality of a pruned model is often summarized by perplexity, which is easy to compute and closely related to next-token prediction.

*Corresponding author
[2]Code is available at

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

However, perplexity does not fully characterize downstream decision behavior. Recent studies on compressed LLMs show that small perplexity changes can hide larger changes in downstream capabilities, especially when evaluation moves beyond average language modeling loss [13, 14, 15]. Many downstream tasks require the model to choose the correct answer among competing candidates. In such settings, reliability depends on average next-token likelihood, the score assigned to the correct answer, and its separation from plausible wrong answers. As shown in Figure 1, lower perplexity generally aligns with higher downstream accuracy, but models with similar perplexity can still differ in task performance. This motivates a decision-level analysis of pruning.

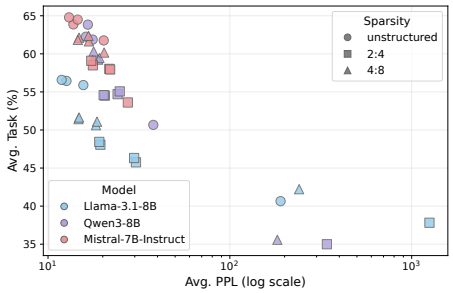


Figure 1: Perplexity is informative, but it does not fully characterize downstream behavior.

What does pruning change at the decision level? A pruned model may retain reasonable perplexity while lowering confidence in the correct answer, narrowing the margin to competing answers, or flipping a previously correct prediction into an error. Related analyses of compressed models also show that compression can affect confidence, calibration, and distributional behavior [16, 13]. We therefore analyze pruning through both aggregate performance and individual answer-level decisions.

We evaluate magnitude pruning [8], Wanda [10], SparseGPT [9], Wanda++ [17], and our proposed MR-GOBS on Llama-3.1-8B, Qwen3-8B, and Mistral-7B-Instruct-v0.3. Our experiments cover both unstructured pruning and semi-structured N:M pruning. Besides standard aggregate metrics, including perplexity and downstream Avg. Task, we conduct a paired decision-level evaluation that measures true-label confidence, answer margin, prediction flips, candidate-level distribution shift, and near-boundary vulnerability. This paired protocol compares each pruned model with its dense counterpart on the same examples.

Across the evaluated models, pruning methods, and sparsity settings, pruning generally weakens decision reliability by reducing true-label confidence, shrinking answer margins, and producing asymmetric prediction flips. Correct-to-wrong flips are more frequent than wrong-to-correct flips, and near-boundary examples are especially fragile because their dense-model margins are already small. These findings reveal pruning-induced degradation that can be hidden by aggregate metrics alone.

We further test whether improving the reconstruction objective used for pruning is sufficient to improve downstream decisions. To this end, we introduce MR-GOBS, a minimax-regret GroupOBS method for semi-structured N:M pruning. GroupOBS denotes a set-level OBS reconstruction cost and local selector for jointly pruning a group of weights. MR-GOBS applies this selector across multiple calibration views and chooses the local N:M mask with the smallest worst-case normalized reconstruction regret.

MR-GOBS serves as a diagnostic probe based on reconstruction. It strengthens local N:M mask selection under a second-order reconstruction criterion that does not use task labels, allowing us to test whether reconstruction improvements translate into downstream reliability. Our experiments show that MR-GOBS often improves average perplexity over SparseGPT, but these gains do not always lead to higher downstream accuracy or better decision-level metrics. Our contributions are threefold. First, we introduce a paired candidate-level diagnostic framework for evaluating how post-training pruning changes individual decisions beyond perplexity and average task accuracy. Second, across three LLM families and multiple sparsity patterns, we show that pruning generally reduces true-label confidence, shrinks answer margins, and increases harmful C$\rightarrow$W flips, with higher C$\rightarrow$W flip rates on examples with small absolute dense-model margins. Third, we introduce MR-GOBS, a minimax-regret GroupOBS selector for N:M pruning, and use it to test whether improved local reconstruction and lower perplexity imply improved downstream decision reliability.

2 Methodology

We study post-training pruning, where a pretrained LLM is compressed without full retraining. Our methodology consists of a paired decision-level diagnostic framework and MR-GOBS for semi-

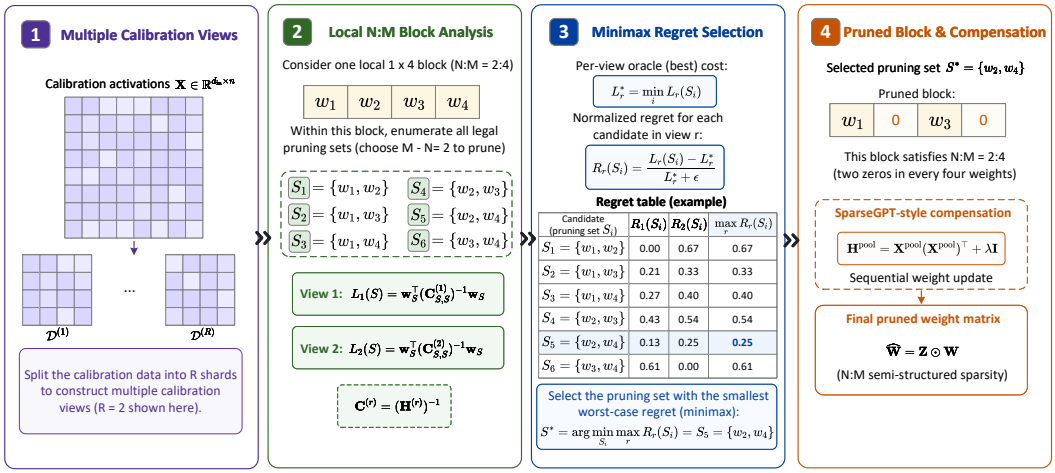


Figure 2: Overview of MR-GOBS. MR-GOBS splits calibration data into multiple views, evaluates legal local N:M pruning sets using the GroupOBS cost, selects the set with minimum worst-case normalized regret, and applies the compensation procedure used by SparseGPT with pooled calibration statistics.

structured N:M pruning. The diagnostic framework compares each pruned model with its dense counterpart on the same examples. MR-GOBS tests whether changes to local mask selection under a reconstruction objective translate into improved downstream decision reliability.

2.1 Decision-Level Diagnostic Framework

We compare each pruned model with its dense counterpart on the same examples. Let f denote the dense model and $\tilde{f}$ denote the corresponding pruned model. For an example x, let $\mathcal{Y}(x) = \{y_1, \ldots, y_K\}$ be the candidate answer set and let $y^\star$ be the ground-truth answer. A model $g \in \{f, \tilde{f}\}$ assigns a length-normalized candidate-token log-likelihood $s_g(y \mid x)$ to each candidate answer under the same prompt. Its prediction $\hat{y}_g(x)$ is the highest-scoring candidate, and $p_g(y \mid x)$ is the softmax probability over the candidate set. The corresponding formulas are given in Appendix B.

Decision accuracy, confidence, and margin. Under the paired candidate-scoring protocol, $\mathrm{DecAcc}(g)$ is the fraction of correct predictions and $\Delta\mathrm{DecAcc}$ is its change from the dense to the pruned model. This metric provides the paired diagnostic measurement, while LM-Evaluation-Harness provides the standard downstream Avg. Task.

We define true-label confidence as

$$\mathrm{TrueConf}_g(x) = p_g(y^\star \mid x), \tag{1}$$

with pruning-induced change

$$\Delta\mathrm{TrueConf}(x) = \mathrm{TrueConf}_{\tilde{f}}(x) - \mathrm{TrueConf}_f(x). \tag{2}$$

The score-space answer margin is

$$m_g(x) = s_g(y^\star \mid x) - \max_{y \in \mathcal{Y}(x),\, y \neq y^\star} s_g(y \mid x), \tag{3}$$

and its pruning-induced change is

$$\Delta m(x) = m_{\tilde{f}}(x) - m_f(x). \tag{4}$$

The probability-space margin is $m_g^{\mathrm{prob}}(x) = p_g(y^\star \mid x) - \max_{y \in \mathcal{Y}(x),\, y \neq y^\star} p_g(y \mid x)$, with pruning-induced change $\Delta m^{\mathrm{prob}}(x) = m_{\tilde{f}}^{\mathrm{prob}}(x) - m_f^{\mathrm{prob}}(x)$.

Prediction flips. We define harmful correct-to-wrong flips and beneficial wrong-to-correct flips as

$$\mathrm{C}\to\mathrm{W}(x) = \mathbb{I}\left[\hat{y}_f(x) = y^\star \ \wedge\ \hat{y}_{\tilde{f}}(x) \neq y^\star\right], \tag{5}$$

and

$$\mathrm{W}\to\mathrm{C}(x) = \mathbb{I}\left[\hat{y}_f(x) \neq y^\star \ \wedge\ \hat{y}_{\tilde{f}}(x) = y^\star\right]. \tag{6}$$

We report their aggregate rates by averaging these indicators over $\mathcal{D}_{\text{eval}}$.

Candidate-level distribution shift. We measure the distributional change between dense and pruned candidate distributions using Jensen–Shannon divergence. Its standard expression is given in Appendix B.

Near-boundary vulnerability. We define near-boundary examples using the absolute dense-model score-space margin. Let $Q_q^{|\text{margin}|}$ be the q-th quantile of $|m_f(x)|$ over the evaluation set:

$$\mathcal{B}_q = \left\{x \in \mathcal{D}_{\text{eval}} : |m_f(x)| \leq Q_q^{|\text{margin}|}\right\}. \tag{7}$$

We use $q = 0.3$ in our experiments. The near-boundary correct-to-wrong flip rate is

$$\text{NB C}\to\text{W} = \frac{1}{|\mathcal{B}_q|}\sum_{x\in\mathcal{B}_q} \mathrm{C}\to\mathrm{W}(x). \tag{8}$$

Together, these diagnostics complement perplexity and Avg. Task by measuring how pruning changes confidence, margins, output distributions, and prediction flips on individual examples.

2.2 MR-GOBS: Testing Whether Reconstruction Gains Improve Decisions

MR-GOBS is designed for N:M semi-structured pruning, where each group of M consecutive weights retains exactly N nonzero weights. It follows the local least-squares reconstruction framework of SparseGPT [9] and changes the mask-selection rule within each block. We use GroupOBS to denote a set-level OBS reconstruction cost and the corresponding local selector for jointly pruning a group of weights. MR-GOBS extends this selector with a minimax-regret criterion over multiple calibration views.

Consider one linear layer with dense weight matrix $\mathbf{W} \in \mathbb{R}^{d_{\text{out}}\times d_{\text{in}}}$ and calibration activations $\mathbf{X} \in \mathbb{R}^{d_{\text{in}}\times n}$. Pruning with SparseGPT can be viewed as minimizing the local reconstruction objective

$$\min_{\widehat{\mathbf{W}}} \left\|\mathbf{W}\mathbf{X} - \widehat{\mathbf{W}}\mathbf{X}\right\|_F^2. \tag{9}$$

For the derivation and implementation, we use the standard damped Hessian approximation

$$\mathbf{H} = \mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I}, \qquad \mathbf{C} = \mathbf{H}^{-1}, \tag{10}$$

where $\lambda > 0$ is a damping coefficient. The damping term makes $\mathbf{H}$ positive definite and ensures that the inverse is well defined. Equivalently, for a single output row, this corresponds to a damped local quadratic reconstruction objective. Since the objective decomposes over output rows, we analyze one row at a time. For notational convenience, we treat each output row as a column vector $\mathbf{w} \in \mathbb{R}^{d_{\text{in}}}$.

Proposition 1 (GroupOBS set-level cost)**.** *For a fixed pruning set S, the minimum damped second-order reconstruction cost under the OBS compensation solution is*

$$L(S) = \mathbf{w}_S^\top \left(\mathbf{C}_{S,S}\right)^{-1}\mathbf{w}_S. \tag{11}$$

When $S = \{i\}$ contains a single weight, this cost reduces to the standard single-weight OBS saliency:

$$L(\{i\}) = \frac{w_i^2}{C_{ii}}. \tag{12}$$

The proof is provided in Appendix C.1. Proposition 1 gives a set-level pruning cost for evaluating a group of weights jointly. We apply it inside each local N:M block. Let $B \subseteq \{1,\ldots,d_{\text{in}}\}$ be a block

of M consecutive input dimensions, and let $k = M - N$ be the number of weights pruned in the block. The legal pruning sets are

$$\Omega_{B,k} = \{S \subseteq B : |S| = k\}. \tag{13}$$

A single-view GroupOBS selector chooses

$$S^\star_{\mathrm{GOBS}} = \arg\min_{S \in \Omega_{B,k}} L(S). \tag{14}$$

For 2:4 pruning, this enumerates $\binom{4}{2} = 6$ candidates per block; for 4:8 pruning, it enumerates $\binom{8}{4} = 70$ candidates per block.

MR-GOBS makes this local selector robust to calibration variation. We split the calibration set into R disjoint views. For each view r, we construct

$$\mathbf{H}^{(r)} = \mathbf{X}^{(r)}\mathbf{X}^{(r)\top} + \lambda \mathbf{I}, \qquad \mathbf{C}^{(r)} = \left(\mathbf{H}^{(r)}\right)^{-1}. \tag{15}$$

The GroupOBS cost of pruning set S under view r is

$$L_r(S) = \mathbf{w}_S^\top \left(\mathbf{C}^{(r)}_{S,S}\right)^{-1} \mathbf{w}_S. \tag{16}$$

Let each view's best legal candidate be

$$L_r^\star = \min_{T \in \Omega_{B,k}} L_r(T). \tag{17}$$

We define the normalized regret of choosing S under view r as

$$R_r(S) = \frac{L_r(S) - L_r^\star}{L_r^\star + \epsilon}, \tag{18}$$

where $\epsilon > 0$ is used for numerical stability. The worst-case normalized regret is

$$J(S) = \max_{1 \le r \le R} R_r(S). \tag{19}$$

MR-GOBS selects the legal pruning set with the smallest worst-case regret:

$$S^\star = \arg\min_{S \in \Omega_{B,k}} J(S) = \arg\min_{S \in \Omega_{B,k}} \max_{1 \le r \le R} R_r(S). \tag{20}$$

If multiple minimizers exist, $S^\star$ denotes any one of them.

Proposition 2 (Local minimax-regret optimality)**.** *For a fixed layer, output row, N:M block, candidate set $\Omega_{B,k}$, and Hessian estimates for each calibration view, MR-GOBS is locally optimal under the minimax objective based on normalized regret:*

$$J(S^\star) \le J(S), \qquad \forall S \in \Omega_{B,k}. \tag{21}$$

In particular, this inequality holds for any legal pruning set selected by a baseline N:M mask-selection rule inside the same block.

The proof is provided in Appendix C.2. This guarantee establishes optimality under the defined minimax objective for reconstruction regret within each N:M block. Perplexity, downstream Avg. Task, and decision-level metrics remain empirical outcomes of the full model governed by nonlinear error propagation. After selecting $S^\star$, MR-GOBS uses the sequential compensation procedure of SparseGPT with pooled calibration statistics. MR-GOBS therefore implements robust local N:M mask selection within the SparseGPT framework. Its empirical behavior tests whether mask selection that better preserves local reconstruction improves perplexity, downstream Avg. Task, and decision-level reliability.

3 Experimental Setup

Models and pruning settings. We evaluate three open-weight LLMs: Llama-3.1-8B, Qwen3-8B, and Mistral-7B-Instruct-v0.3 [6, 18, 19]. We compare magnitude pruning [8], Wanda [10], SparseGPT [9], and Wanda++ [17] under unstructured 50% pruning, 2:4 pruning, and 4:8 pruning. MR-GOBS is evaluated under the two N:M patterns.

Pruning protocol. All calibration-based methods use the same calibration data and layer-wise pruning protocol. For MR-GOBS, calibration samples are split into deterministic views for minimax-regret mask selection, while compensation uses pooled calibration statistics. MR-GOBS uses the same damping convention as SparseGPT, a fixed numerical stability constant ϵ, and $R = 2$ calibration views.

Aggregate evaluation. We report perplexity on WikiText2 [20], PTB [21], and C4 [22], and their average as Avg. PPL. The definition of perplexity is given in Appendix B. We evaluate downstream accuracy using LM-Evaluation-Harness[3] on ARC-Challenge, ARC-Easy [23], BoolQ [24], HellaSwag [25], OpenBookQA [26], RTE [27], and WinoGrande [28], reporting their mean accuracy as Avg. Task. In addition, we evaluate the pruned models on GSM8K [29] with 5-shot chain-of-thought prompting, HumanEval [30] with 0-shot prompting, MBPP [31] with 3-shot prompting, and TruthfulQA-Generation [32] with its fixed question-answer prompt. We report flexible and strict exact match for GSM8K, execution-based pass@1 for HumanEval and MBPP, and BLEU accuracy for TruthfulQA-Generation. We follow the task-specific prompting configurations of LM-Evaluation-Harness and keep the prompts, few-shot examples, and decoding settings fixed across pruning methods within each benchmark.

Paired decision-level evaluation. We separately evaluate the diagnostics in Section 2.1. For each example, the dense model and the corresponding pruned model score the same candidate answers under the same prompt. We use length-normalized candidate-token log-likelihoods as scores and apply a softmax over candidates to compute true-label confidence, score-space and probability-space margins, prediction flips, and JSD. The near-boundary subset contains the 30% of examples for which the dense model assigns the closest scores to the correct answer and the highest-scoring incorrect answer. The paired decision accuracy DecAcc provides the diagnostic measurement, while LM-Evaluation-Harness Avg. Task provides the aggregate benchmark score. Reported confidence changes, margin changes, and JSD are sample means; overbars are omitted in tables and captions.

4 Results and Analysis

We organize our analysis around five questions. First, we examine whether perplexity explains downstream behavior after pruning. Second, we analyze how pruning changes decision-level reliability, including confidence, margins, and prediction flips. Third, we study which diagnostics best track degradation in paired decision accuracy. Fourth, we use MR-GOBS to test whether reductions in PPL translate into better downstream decisions. Finally, we test whether the mismatch between PPL and generation performance also appears on open-ended generation benchmarks.

Table 1: Performance under N:M pruning. PPL is averaged over WikiText2, PTB, and C4; Avg. Task is computed using LM-Evaluation-Harness.

Model	Method	2:4		4:8	
		PPL ↓	Task ↑	PPL ↓	Task ↑
Llama-3.1-8B	Magnitude	1252.41	37.8	240.45	42.2
	Wanda	30.45	45.8	18.36	50.6
	SparseGPT	19.40	48.1	**14.71**	51.4
	Wanda++	29.75	46.3	18.59	51.1
	MR-GOBS	**19.14**	**48.4**	14.81	**51.6**
Qwen3-8B	Magnitude	341.89	35.0	182.50	35.6
	Wanda	24.14	54.7	18.85	59.2
	SparseGPT	20.56	54.5	17.82	**60.3**
	Wanda++	24.79	**55.1**	19.19	59.5
	MR-GOBS	**20.23**	54.6	**17.62**	59.2
Mistral-7B-Inst.	Magnitude	27.48	53.6	20.35	60.2
	Wanda	21.73	58.0	16.78	61.6
	SparseGPT	17.65	58.5	14.81	62.1
	Wanda++	21.96	57.9	16.70	**62.4**
	MR-GOBS	**17.29**	**59.1**	**14.58**	61.8

4.1 Perplexity Is Informative but Incomplete

We first compare average perplexity with standard downstream Avg. Task. Table 1 reports aggregate results under semi-structured N:M pruning. As expected, methods with extremely high perplexity usually show poor downstream accuracy. For example, magnitude pruning under N:M sparsity causes severe PPL degradation and substantially lower Avg. Task, confirming that perplexity remains a useful coarse indicator of pruning quality.

[3] https://github.com/EleutherAI/lm-evaluation-harness

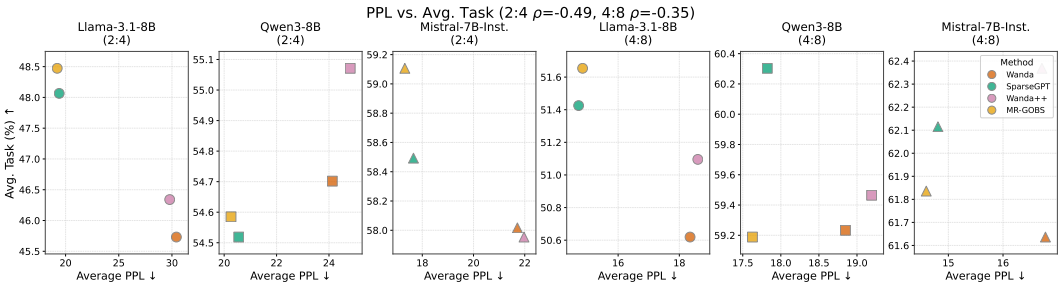


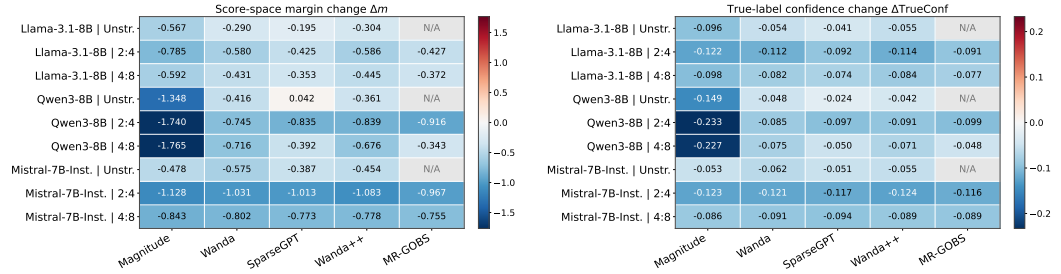
Figure 3: Relationship between perplexity and task performance under semi-structured N:M pruning. Perplexity remains informative as a coarse indicator, but it does not fully determine standard downstream Avg. Task.

However, perplexity does not fully determine downstream behavior. Figure 3 shows the relationship between PPL and task performance under 2:4 and 4:8 pruning. Among the 12 configurations excluding magnitude pruning for each sparsity pattern, the correlation is only moderate, with Spearman correlations of -0.49 for 2:4 and -0.35 for 4:8. Several settings show mismatches between PPL and task accuracy. For example, in Qwen3-8B 2:4 pruning, MR-GOBS achieves the best average PPL, while Wanda++ has the highest Avg. Task. In Mistral-7B-Inst. 4:8 pruning, MR-GOBS achieves the best average PPL, while Wanda++ retains the highest Avg. Task.

Finding 1. Perplexity remains a useful coarse indicator of pruning quality, but it is not a sufficient proxy for downstream decision reliability.

4.2 Pruning Changes Decision-Level Reliability

We next analyze how pruning changes individual decisions. Figure 4 shows that pruning generally reduces true-label confidence and score-space margin change Δm across models, pruning methods, and sparsity settings. This degradation can occur even when the final prediction remains correct, indicating that pruning weakens the stability of correct decisions before an error appears.

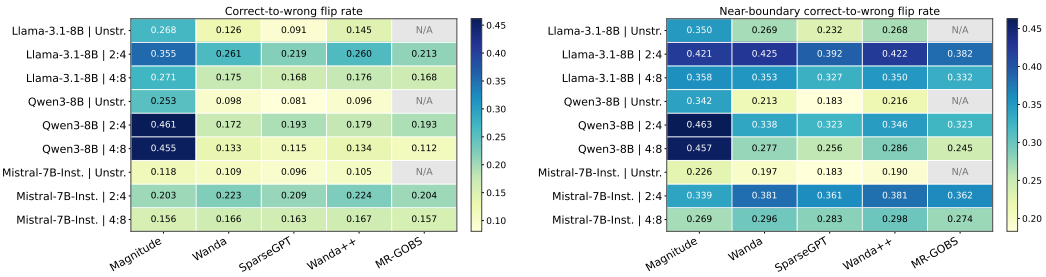


(a) Score-space margin change

(b) True-label confidence change

Figure 4: Confidence and margin changes under pruning. Most settings show negative score-space margin change Δm and negative true-label confidence change $\Delta\mathrm{TrueConf}$, indicating weakened support for the correct answer.

The degradation is not limited to soft confidence scores. Figure 5 shows that pruning also produces nontrivial C→W flips. Under 2:4 pruning, Llama-3.1-8B has a C→W rate of 21.9% with SparseGPT and 21.3% with MR-GOBS. Qwen3-8B has a C→W rate of 19.3% with both SparseGPT and MR-GOBS. Even under the milder 4:8 setting, harmful flips remain; for Qwen3-8B 4:8, SparseGPT has an 11.5% C→W rate, while MR-GOBS has an 11.2% rate. Near-boundary examples are especially vulnerable. Figure 5 shows that near-boundary C→W rates are consistently higher than overall C→W rates. For Qwen3-8B under 4:8 pruning, MR-GOBS has an overall C→W rate of 11.2%, while the near-boundary rate rises to 24.5%. For Mistral-7B-Inst. under 4:8 pruning, the corresponding rates are 15.7% and 27.4%. These results indicate that pruning-induced errors concentrate around decisions with small absolute dense-model score-space margins. Full decision-level metrics across all



(a) Overall C→W flip rate (b) Near-boundary C→W flip rate

Figure 5: Correct-to-wrong flips under pruning. Near-boundary examples, defined by small absolute dense-model score-space margins, show higher C→W rates than the overall sample set, indicating concentrated failures around fragile decisions.

sparsity settings are reported in Appendix Table D.6. Additional near-boundary amplification ratios and analyses of the asymmetry between flips are provided in Appendix Figures 7 and 8.

Finding 2. Across the evaluated settings, pruning generally weakens decision reliability by reducing true-label confidence, shrinking answer margins, and producing harmful C→W flips. Near-boundary examples are especially vulnerable.

4.3 Decision-Level Metrics Track Paired Accuracy Changes

We next ask which metrics are most associated with changes in paired decision accuracy under the diagnostic protocol. Figure 6 compares aggregate metrics and decision-level diagnostics descriptively over 294 records covering the three main models, pruning methods, sparsity settings, and tasks. The change in true-label confidence has the strongest positive correlation with the change in paired decision accuracy, followed by the change in probability-space margin. Metrics derived from prediction flips are strongly negatively correlated with the change in paired decision accuracy, with C→W and near-boundary C→W showing correlations of -0.84 and -0.92, respectively. This sign is expected: a larger harmful flip rate corresponds to larger accuracy degradation. Average PPL and JSD are also correlated with the change in paired decision accuracy, but their correlations are weaker than those of the most direct decision-level metrics. These results show that metrics tied to the correct answer better track degradation in paired decision accuracy than coarse distributional shift alone. Detailed scatter plots for margin and true-label confidence are provided in Appendix Figure 9.

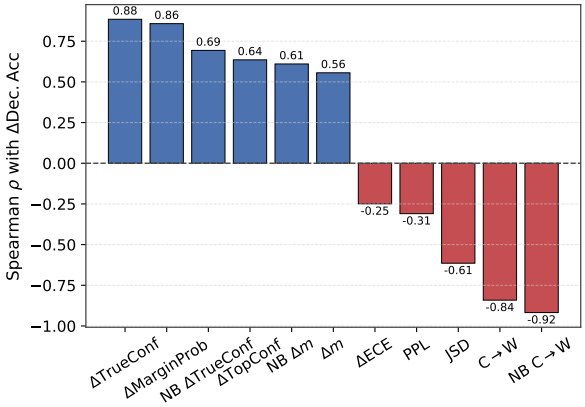


Figure 6: Correlation between diagnostics and the change in paired decision accuracy. The change in true-label confidence and near-boundary C→W rate are more strongly associated with ΔDec. Acc. than average PPL or JSD.

Finding 3. Decision-level changes, especially $\Delta\mathrm{TrueConf}$ and near-boundary C→W, closely track changes in paired decision accuracy and complement aggregate metrics such as PPL and Avg. Task.

4.4 MR-GOBS Tests Whether Reconstruction Gains Transfer to Downstream Tasks

We next use MR-GOBS to test whether improving the reconstruction objective used for pruning is sufficient to improve downstream decisions. Table 2 compares MR-GOBS with SparseGPT under N:M pruning. Differences are computed as MR-GOBS minus SparseGPT using unrounded values. MR-GOBS improves average PPL in five out of six N:M settings, with the only exception being Llama-3.1-8B 4:8. However, downstream changes are mixed. MR-GOBS improves Avg. Task in

Table 2: MR-GOBS compared with SparseGPT under N:M pruning. All differences are computed as MR-GOBS minus SparseGPT using unrounded values. Lower ΔAvg. PPL, higher ΔAvg. Task and $\Delta_{\mathrm{MR-SG}}\Delta m$, and lower changes in flip rates indicate improvement.

Model	Sparsity	ΔAvg. PPL ↓	ΔAvg. Task (pp) ↑	$\Delta_{\mathrm{MR-SG}}\Delta m$ ↑	ΔC→W (pp) ↓	ΔNB C→W (pp) ↓
Llama-3.1-8B	2:4	-0.261	0.37	-0.002	-0.67	-0.90
	4:8	0.105	0.21	-0.018	0.05	0.41
Qwen3-8B	2:4	-0.326	0.06	-0.081	0.04	-0.04
	4:8	-0.205	-1.13	0.049	-0.30	-1.05
Mistral-7B-Inst.	2:4	-0.367	0.60	0.045	-0.46	0.10
	4:8	-0.228	-0.29	0.017	-0.57	-0.94

four settings, but reduces Avg. Task for Qwen3-8B 4:8 and Mistral-7B-Inst. 4:8. For Qwen3-8B 4:8, MR-GOBS improves average PPL and reduces C→W, yet its Avg. Task decreases by 1.13 percentage points. For Mistral-7B-Inst. 4:8, MR-GOBS improves PPL and reduces C→W, but its Avg. Task decreases by 0.29 points. These results show that even when a mask selector based on reconstruction improves PPL or some decision-level indicators, it does not guarantee better downstream task performance. The corresponding visualization is provided in Appendix Figure 10.

Finding 4. MR-GOBS often improves PPL over SparseGPT, but downstream accuracy and decision-level metrics do not always improve together. This reveals a gap between improvements in reconstruction and decision-level reliability.

Implications. These findings suggest that aggregate pruning quality and decision reliability should be treated as related but distinct evaluation targets. Perplexity remains useful for measuring next-token prediction quality, but it does not fully capture confidence loss, margin shrinkage, or harmful C→W flips. MR-GOBS further shows that mask selection that better preserves reconstruction does not necessarily improve downstream reliability, motivating future pruning methods with criteria based on decision reliability. Additional Qwen3 scale validation in Appendix D.5 suggests that larger models can be more robust to pruning, while near-boundary vulnerability persists across scales.

4.5 Open-Ended Generation

Table 3 reports average PPL together with GSM8K, HumanEval, MBPP, and TruthfulQA-Generation for all five pruning methods under 2:4 and 4:8 sparsity; Appendix D.8 provides the corresponding unstructured results. These endpoint metrics are separated from the Decision analysis with a fixed set of candidates because free-form generation does not provide a common candidate set for paired margin or JSD calculations. The rankings remain task dependent. For Llama, MR-GOBS often improves PPL and TruthfulQA BLEU accuracy but does not lead GSM8K, HumanEval, or MBPP. For Qwen3-8B, MR-GOBS gives the lowest PPL and the highest TruthfulQA BLEU accuracy under both N:M patterns, whereas Wanda or Wanda++ performs better on GSM8K, HumanEval, and MBPP. For Mistral, MR-GOBS leads PPL and GSM8K, while its HumanEval score is zero and its MBPP score trails other methods. Thus, lower PPL does not imply uniformly better open-ended generation.

5 Conclusion

We studied how post-training pruning changes candidate-level decisions in large language models. While perplexity remains a useful coarse indicator, it does not fully capture pruning-induced changes in confidence, margins, and prediction flips. Our analysis shows that pruning often reduces true-label confidence, shrinks answer margins, and causes correct-to-wrong flips, especially for examples near the dense model's decision boundary. We introduced MR-GOBS for local N:M mask selection. Although it often improves average perplexity over SparseGPT, these gains do not consistently translate into better downstream accuracy or decision-level reliability. The mixed GSM8K, HumanEval, MBPP, and TruthfulQA-Generation results show that this mismatch also extends to the evaluated open-ended generation benchmarks. These findings suggest that reliable evaluation of pruned LLMs requires both aggregate metrics and paired decision-level diagnostics.

Table 3: Average PPL and open-ended generation under N:M pruning. Avg. PPL averages WikiText2, PTB, and C4. The best value for each combination of model and sparsity pattern is bolded.

Model	Sparsity	Method	Avg. PPL ↓	GSM8K Flex. ↑	GSM8K Strict ↑	HumanEval ↑	MBPP ↑	TruthfulQA ↑
Llama-3.1-8B	2:4	Magnitude	1252.41	1.5	0.0	0.0	0.0	25.1
		Wanda	30.45	2.4	1.6	2.4	**1.8**	4.5
		SparseGPT	19.40	**4.4**	**3.9**	0.6	1.4	26.3
		Wanda++	29.75	2.4	2.0	**3.0**	0.4	8.2
		MR-GOBS	**19.14**	4.2	3.8	0.6	0.6	**30.8**
	4:8	Magnitude	240.45	0.5	0.0	1.8	0.0	13.7
		Wanda	18.36	7.1	6.7	**11.0**	9.2	22.4
		SparseGPT	**14.71**	**11.9**	**11.8**	2.4	8.4	26.8
		Wanda++	18.59	6.7	6.6	9.8	**9.8**	25.7
		MR-GOBS	14.81	7.7	7.4	3.0	5.2	**30.1**
Qwen3-8B	2:4	Magnitude	341.89	1.4	0.0	0.0	0.0	4.9
		Wanda	24.14	48.5	48.3	5.5	23.0	35.3
		SparseGPT	20.56	47.2	46.4	0.0	4.2	37.7
		Wanda++	24.79	**48.7**	**48.7**	**8.5**	**23.4**	33.7
		MR-GOBS	**20.23**	47.2	46.6	0.0	5.8	**39.7**
	4:8	Magnitude	182.50	1.4	0.0	0.0	0.0	33.7
		Wanda	18.85	68.4	68.2	**16.5**	32.0	38.3
		SparseGPT	17.82	64.6	64.6	0.0	18.4	48.6
		Wanda++	19.19	**68.6**	**68.4**	12.8	**33.8**	41.0
		MR-GOBS	**17.62**	61.6	60.8	1.2	15.2	**53.1**
Mistral-7B-Inst.	2:4	Magnitude	27.48	2.2	0.1	0.0	5.6	**43.0**
		Wanda	21.73	8.1	8.0	**0.6**	4.0	35.9
		SparseGPT	17.65	14.3	14.0	0.0	2.2	34.4
		Wanda++	21.96	9.4	9.2	**0.6**	**6.4**	34.1
		MR-GOBS	**17.29**	**14.6**	**14.4**	0.0	4.0	34.6
	4:8	Magnitude	20.35	8.6	7.6	**1.2**	9.8	**40.3**
		Wanda	16.78	16.5	16.5	0.0	12.6	36.5
		SparseGPT	14.81	21.6	21.5	0.0	10.4	36.7
		Wanda++	16.70	16.8	16.7	0.0	**14.0**	36.5
		MR-GOBS	**14.58**	**22.7**	**22.5**	0.0	9.4	39.8

Limitations

This work focuses on post-training pruning and decision-level evaluation for open-weight LLMs. Our main experiments cover three model families, three sparsity settings, and several representative pruning methods, with additional scale validation on the Qwen3 family. Quantization, distillation, and mixed compression pipelines may introduce additional forms of decision shift that require separate investigation. MR-GOBS targets local N:M mask selection under a reconstruction objective. Its effects on downstream task accuracy, true-label confidence, answer margins, and prediction flips are evaluated empirically. Future work can build on this analysis by developing pruning objectives that explicitly preserve decision margins or reduce harmful correct-to-wrong flips, while avoiding overfitting to a small set of downstream tasks.

Acknowledgments

This work was funded by the National Natural Science Foundation of China (Grant No. 62366036), Outstanding Youth Fund Project of Inner Mongolia Autonomous Region (Grant No. 2025JQ010), Program for Young Talents of Science and Technology in Universities of Inner Mongolia Autonomous Region (Grant No. NJYT24033), Major Science and Technology Projects of Inner Mongolia Autonomous Region (Grant No. 2025ZDSF0029), Key R&D and Achievement Transformation Program of Inner Mongolia Autonomous Region (Grant Nos. 2025YFDZ0011, 2025YFDZ0026, 2025YFSH0021, 2025YFHH0073, 2026YFDZ0158), and Hohhot R&D Investment Incentive Project (Grant No. 2025-gui-yan-1-23).

References

[1] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.

[2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[3] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. Casas, L. A. Hendricks, J. Welbl, A. Clark *et al.*, "Training compute-optimal large language models," *arXiv preprint arXiv:2203.15556*, vol. 10, 2022.

[4] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman *et al.*, "Gpt-4 technical report," 2024. [Online]. Available: https://arxiv.org/abs/2303.08774

[5] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[6] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan *et al.*, "The llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, 2024.

[7] Y. LeCun, J. Denker, and S. Solla, "Optimal brain damage," *Advances in neural information processing systems*, vol. 2, 1989.

[8] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," *Advances in neural information processing systems*, vol. 28, 2015.

[9] 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.

[10] M. Sun, Z. Liu, A. Bair, and J. Z. Kolter, "A simple and effective pruning approach for large language models," in *12th International Conference on Learning Representations, ICLR 2024*, 2024.

[11] Y. Ji, Y. Xiang, J. Li, Q. Xia, P. Li, X. Duan, Z. Wang, and M. Zhang, "Beware of calibration data for pruning large language models," in *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. [Online]. Available: https://openreview.net/forum?id=x83w6yGIWb

[12] P. Zhao, H. Hu, P. Li, Y. Zheng, Z. Wang, and X. Yuan, "Fistapruner: Layer-wise post-training pruning for large language models," in *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025, pp. 29 457–29 475.

[13] A. K. JAISWAL, Z. Gan, X. Du, B. Zhang, Z. Wang, and Y. Yang, "Compressing LLMs: The truth is rarely pure and never simple," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=B9klVS7Ddk

[14] M. Williams and N. Aletras, "On the impact of calibration data in post-training quantization and pruning," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024, pp. 10 100–10 118.

[15] B. Khanal and J. M. Capone, "Evaluating the impact of compression techniques on task-specific performance of large language models," *arXiv preprint arXiv:2409.11233*, 2024.

[16] I. Proskurina, L. Brun, G. Metzler, and J. Velcin, "When quantization affects confidence of large language models?" in *Findings of the Association for Computational Linguistics: NAACL 2024*, 2024, pp. 1918–1928.

[17] Y. Yang, K. Zhen, B. Ganesh, A. Galstyan, G. Huybrechts, M. Müller, J. M. Kübler, R. V. Swaminathan, A. Mouchtaris, S. B. Bodapati *et al.*, "Wanda++: Pruning large language models via regional gradients," in *Findings of the Association for Computational Linguistics: ACL 2025*, 2025, pp. 4321–4333.

[18] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv *et al.*, "Qwen3 technical report," *arXiv preprint arXiv:2505.09388*, 2025.

[19] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed, "Mistral 7b," 2023. [Online]. Available: https://arxiv.org/abs/2310.06825

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

[21] 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.

[22] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of machine learning research*, vol. 21, no. 140, pp. 1–67, 2020.

[23] 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.

[24] C. Clark, K. Lee, M.-W. Chang, T. Kwiatkowski, M. Collins, and K. Toutanova, "Boolq: Exploring the surprising difficulty of natural yes/no questions," in *Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics: Human language technologies, volume 1 (long and short papers)*, 2019, pp. 2924–2936.

[25] R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, and Y. Choi, "Hellaswag: Can a machine really finish your sentence?" in *Proceedings of the 57th annual meeting of the association for computational linguistics*, 2019, pp. 4791–4800.

[26] T. Mihaylov, P. Clark, T. Khot, and A. Sabharwal, "Can a suit of armor conduct electricity? a new dataset for open book question answering," in *Proceedings of the 2018 conference on empirical methods in natural language processing*, 2018, pp. 2381–2391.

[27] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding," in *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP*, 2018, pp. 353–355.

[28] K. Sakaguchi, R. L. Bras, C. Bhagavatula, and Y. Choi, "Winogrande: An adversarial winograd schema challenge at scale," *Communications of the ACM*, vol. 64, no. 9, pp. 99–106, 2021.

[29] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano *et al.*, "Training verifiers to solve math word problems," *arXiv preprint arXiv:2110.14168*, 2021.

[30] M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. D. O. Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman *et al.*, "Evaluating large language models trained on code," *arXiv preprint arXiv:2107.03374*, 2021.

[31] J. Austin, A. Odena, M. Nye, M. Bosma, H. Michalewski, D. Dohan, E. Jiang, C. Cai, M. Terry, Q. Le *et al.*, "Program synthesis with large language models," *arXiv preprint arXiv:2108.07732*, 2021.

[32] S. Lin, J. Hilton, and O. Evans, "Truthfulqa: Measuring how models mimic human falsehoods," in *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: long papers)*, 2022, pp. 3214–3252.

[33] B. Hassibi and D. Stork, "Second order derivatives for network pruning: Optimal brain surgeon," *Advances in neural information processing systems*, vol. 5, 1992.

[34] Y. Zhang, H. Bai, H. Lin, J. Zhao, L. Hou, and C. V. Cannistraci, "Plug-and-play: An efficient post-training pruning method for large language models," in *The Twelfth International Conference on Learning Representations*, 2024.

[35] A. Sengupta, S. Chaudhary, and T. Chakraborty, "You only prune once: Designing calibration-free model compression with policy learning," *arXiv preprint arXiv:2501.15296*, 2025.

[36] J. Li, P. Cao, C. Zhou, T. Lan, X. Su, K. Liu, J. Zhao, and G. Gao, "Lightweight haar wavelet subband pruning for llms," in *Findings of the Association for Computational Linguistics: ACL 2026*, 2026, pp. 16 242–16 259.

[37] 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.

[38] M. Zhang, H. Chen, C. Shen, Z. Yang, L. Ou, X. Yu, and B. Zhuang, "Loraprune: Structured pruning meets low-rank parameter-efficient fine-tuning," in *Findings of the Association for Computational Linguistics: ACL 2024*, 2024, pp. 3013–3026.

[39] Z. Xu, A. Gupta, T. Li, O. Bentham, and V. Srikumar, "Beyond perplexity: Multi-dimensional safety evaluation of llm compression," in *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024, pp. 15 359–15 396.

[40] C. Zhou, P. Cao, J. Li, B. Yu, J. Ye, J. Zhao, and K. Liu, "From signal degradation to computation collapse: Uncovering the two failure modes of llm quantization," in *Findings of the Association for Computational Linguistics: ACL 2026*, 2026, pp. 23 204–23 222.

[41] C. Zhou, P. Cao, J. Li, B. Yu, J. Ye, J. Zhao, and K. Liu, "Task-stratified knowledge scaling laws for post-training quantized large language models," in *Findings of the Association for Computational Linguistics: ACL 2026*, 2026, pp. 23 268–23 285.

[42] C. Zhou, P. Cao, J. Ye, B. Yu, H. Yu, J. Li, J. Zhao, and K. Liu, "Quantization degradation in large language models: A signal-noise perspective," *arXiv preprint arXiv:2608.08188*, 2026.

[43] E. Frantar and D. Alistarh, "Optimal brain compression: A framework for accurate post-training quantization and pruning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 4475–4488, 2022.

[44] S. Shin, W. Park, J. Lee, and N. Lee, "Rethinking pruning large language models: Benefits and pitfalls of reconstruction error minimization," in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024, pp. 1182–1191.

[45] A. Dutta, S. Krishnan, N. Kwatra, and R. Ramjee, "Accuracy is not all you need," *Advances in Neural Information Processing Systems*, vol. 37, pp. 124 347–124 390, 2024.

[46] A. Ghriss, "S^3: Structured sparsity specification," 2026. [Online]. Available: https://arxiv.org/abs/2604.11315

A Related Work

Post-training pruning for large language models. Pruning has long been studied as a way to reduce model size and inference cost. Early work explored magnitude-based pruning and iterative retraining for efficient neural networks [8], while second-order pruning methods such as Optimal Brain Damage and Optimal Brain Surgeon use curvature information to estimate the effect of removing parameters [7, 33]. Recent LLM pruning methods focus on the post-training setting, where the model is compressed without full retraining. SparseGPT formulates one-shot LLM pruning as a local reconstruction problem with second-order error compensation and supports both unstructured and N:M sparsity [9]. Wanda provides a simpler activation-aware pruning rule by combining weight magnitude with input activation statistics and does not update the remaining weights [10]. Other recent methods improve post-training pruning through efficient layer-wise optimization, calibration-free pruning, or regional pruning signals [34, 35, 12, 17, 36]. Structured pruning methods further remove larger units such as channels, heads, or groups, and can require dependency-aware pruning strategies [37, 38]. Our study analyzes how these pruning mechanisms change answer-level decisions after compression.

Evaluating compressed LLMs beyond perplexity. Perplexity is widely used in post-training compression because it is stable, easy to compute, and directly reflects average next-token prediction quality. However, recent studies show that perplexity alone may not fully capture the downstream behavior of compressed LLMs. Jaiswal et al. introduce LLM-KICK and show that compressed models with small perplexity degradation can still suffer capability loss on knowledge-intensive and reasoning tasks [13]. Khanal and Capone [15] evaluate magnitude pruning, SparseGPT, and Wanda on LLaMA-2-7B and report that methods preserving perplexity at 50% sparsity can still degrade downstream task performance, motivating broader evaluation metrics such as Jensen–Shannon divergence. Xu et al. [39] further show that compression can affect safety-related and representational behavior, covering unstructured pruning, semi-structured pruning, and quantization. Related quantization studies distinguish signal degradation from computation collapse [40] and characterize different scaling behaviors across memorization, application, and reasoning [41]. A signal–noise analysis further explains quantization degradation through error generation within modules and propagation across layers [42]. Our work follows this line but focuses on paired answer-level changes, including true-label confidence, answer margins, correct-to-wrong flips, and near-boundary vulnerability.

Confidence, calibration, and decision reliability. Confidence and calibration analyses provide a useful lens for understanding compressed model behavior. Prior work on quantized LLMs shows that compression can affect confidence, calibration error, entropy, and distributional similarity between dense and compressed models [16]. Related calibration studies also show that compressed or quantized models can remain accurate while changing their probability estimates, which makes confidence-based diagnostics important for reliability analysis. Our study brings this decision-level perspective to pruning. We compare dense and pruned models on the same examples and decompose pruning-induced changes into true-label confidence shifts, margin shifts, harmful and beneficial prediction flips, and near-boundary failure rates. This paired design allows us to identify when pruning weakens a correct decision before it becomes an error.

Reconstruction objectives, compression based on OBS, and structured sparsity. Many post-training pruning methods rely on local reconstruction objectives estimated from calibration activations. Optimal Brain Compression extends ideas from OBS to modern post-training pruning and quantization, providing an efficient framework for using inverse-Hessian information in compression [43]. SparseGPT then scales this view to large language models through efficient layer-wise reconstruction and sequential error compensation [9]. Recent work also shows that reconstruction error minimization and calibration data selection can strongly affect pruning quality. Shin et al. [44] analyze the benefits and pitfalls of reconstruction error minimization for LLM pruning, while Ji et al. [11] show that calibration data can be crucial, especially under high sparsity. Dutta et al. [45] show that aggregate accuracy can hide answer flips. Recent structured sparsity work also studies how to specify or optimize legal sparsity patterns, including N:M patterns and broader structured masks [46]. MR-GOBS is related to this family of methods based on reconstruction, especially Optimal Brain Compression [43], which provides an efficient realization of the classical OBS framework for post-training pruning and quantization. We study semi-structured N:M pruning of LLMs and refine the local mask-selection step under the SparseGPT least-squares reconstruction view. MR-GOBS evaluates each legal N:M pruning set with a set-level reconstruction cost derived from OBS and selects masks using a minimax-regret criterion across calibration views, followed by the compensation procedure used by SparseGPT. MR-GOBS tests whether stronger local reconstruction is sufficient for downstream decision reliability.

Position of this work. Our work contributes a decision-level analysis of pruning that connects aggregate perplexity and accuracy with paired confidence loss, margin shrinkage, correct-to-wrong flips, wrong-to-correct flips, and near-boundary vulnerability. MR-GOBS tests whether stronger local reconstruction objectives improve downstream reliability. Together, these elements complement existing pruning algorithms and benchmarks for compressed models.

B Standard Evaluation Definitions

This section gives the standard definitions used in Sections 2.1 and 3.

Perplexity. For a token sequence $x_{1:T}$, perplexity is

$$\mathrm{PPL} = \exp\left(-\frac{1}{T}\sum_{t=1}^{T}\log p(x_t \mid x_{<t})\right). \tag{22}$$

Candidate predictions and probabilities. Using the notation in Section 2.1, a model g predicts the highest-scoring candidate:

$$\hat{y}_g(x) = \arg\max_{y\in\mathcal{Y}(x)} s_g(y \mid x). \tag{23}$$

We convert scores into candidate-level probabilities by applying a softmax over the candidate set:

$$p_g(y \mid x) = \frac{\exp(s_g(y \mid x))}{\sum_{y'\in\mathcal{Y}(x)}\exp(s_g(y' \mid x))}. \tag{24}$$

Paired decision accuracy. Under the paired candidate-scoring protocol, decision accuracy is

$$\mathrm{DecAcc}(g) = \frac{1}{|\mathcal{D}_{\mathrm{eval}}|}\sum_{x\in\mathcal{D}_{\mathrm{eval}}}\mathbb{I}\left[\hat{y}_g(x) = y^\star\right], \tag{25}$$

and the pruning-induced accuracy change is

$$\Delta\mathrm{DecAcc} = \mathrm{DecAcc}(\tilde{f}) - \mathrm{DecAcc}(f). \tag{26}$$

Candidate-level distribution shift. We measure the distributional change between dense and pruned candidate distributions using Jensen–Shannon divergence. Let

$$p_{\mathrm{mix}}(\cdot \mid x) = \frac{1}{2}p_f(\cdot \mid x) + \frac{1}{2}p_{\tilde{f}}(\cdot \mid x). \tag{27}$$

Then

$$\mathrm{JSD}(x) = \frac{1}{2}D_{\mathrm{KL}}\left(p_f(\cdot \mid x) \| p_{\mathrm{mix}}(\cdot \mid x)\right) + \frac{1}{2}D_{\mathrm{KL}}\left(p_{\tilde{f}}(\cdot \mid x) \middle\| p_{\mathrm{mix}}(\cdot \mid x)\right). \tag{28}$$

C Proofs for MR-GOBS

C.1 Proof of the GroupOBS Set-Level Cost

Proof of Proposition 1. We derive the set-level OBS cost used by GroupOBS. For a single output row, we treat the dense row as a column vector $\mathbf{w} \in \mathbb{R}^{d_{\mathrm{in}}}$ and denote the pruned and compensated row by $\widehat{\mathbf{w}}$. Let

$$\Delta = \widehat{\mathbf{w}} - \mathbf{w}. \tag{29}$$

For a fixed pruning set S, the entries in S are constrained to be zero after pruning. Therefore,

$$\widehat{\mathbf{w}}_S = \mathbf{0}, \qquad \Delta_S = -\mathbf{w}_S. \tag{30}$$

The damped local quadratic reconstruction problem is

$$\Delta^\star(S) = \arg\min_{\Delta} \Delta^\top \mathbf{H}\Delta \quad \text{s.t.} \quad \Delta_S = -\mathbf{w}_S, \tag{31}$$

where $\mathbf{H}$ is positive definite and $\mathbf{C} = \mathbf{H}^{-1}$.

Let $\mathbf{E}_S$ be the matrix that selects columns and extracts the coordinates in S. Then

$$\Delta_S = \mathbf{E}_S^\top \Delta. \tag{32}$$

The constrained problem can therefore be written as

$$\min_{\Delta} \Delta^\top \mathbf{H}\Delta \quad \text{s.t.} \quad \mathbf{E}_S^\top \Delta = -\mathbf{w}_S. \tag{33}$$

The Lagrangian is

$$\mathcal{L}(\Delta, \boldsymbol{\nu}) = \Delta^\top \mathbf{H}\Delta + 2\boldsymbol{\nu}^\top \left(\mathbf{E}_S^\top \Delta + \mathbf{w}_S\right). \tag{34}$$

The stationarity condition with respect to Δ gives

$$\mathbf{H}\Delta + \mathbf{E}_S\boldsymbol{\nu} = \mathbf{0}. \tag{35}$$

Thus,

$$\Delta = -\mathbf{H}^{-1}\mathbf{E}_S\boldsymbol{\nu} = -\mathbf{C}_{:,S}\boldsymbol{\nu}. \tag{36}$$

Substituting this expression into the constraint gives

$$-\mathbf{E}_S^\top \mathbf{C}_{:,S}\boldsymbol{\nu} = -\mathbf{w}_S. \tag{37}$$

Since

$$\mathbf{E}_S^\top \mathbf{C}_{:,S} = \mathbf{C}_{S,S}, \tag{38}$$

we obtain

$$\boldsymbol{\nu} = \left(\mathbf{C}_{S,S}\right)^{-1} \mathbf{w}_S. \tag{39}$$

Therefore, the optimal OBS compensation perturbation is

$$\Delta^\star(S) = -\mathbf{C}_{:,S}\left(\mathbf{C}_{S,S}\right)^{-1} \mathbf{w}_S. \tag{40}$$

Substituting $\Delta^\star(S)$ into the quadratic objective gives

$$L(S) = \Delta^\star(S)^\top \mathbf{H}\Delta^\star(S) = \mathbf{w}_S^\top \left(\mathbf{C}_{S,S}\right)^{-1} \mathbf{w}_S. \tag{41}$$

Because $\mathbf{H}$ is positive definite, $\mathbf{C}$ is also positive definite. Therefore, the principal submatrix $\mathbf{C}_{S,S}$ is positive definite and invertible.

When $S = \{i\}$, we have

$$\mathbf{C}_{S,S} = C_{ii}, \qquad \mathbf{w}_S = w_i. \tag{42}$$

Thus,

$$L(\{i\}) = \frac{w_i^2}{C_{ii}}, \tag{43}$$

which recovers the standard single-weight OBS saliency under the same convention for the quadratic objective. □

C.2 Proof of Local Minimax-Regret Optimality

Proof of Proposition 2. For a fixed layer, output row, N:M block, candidate set $\Omega_{B,k}$, and Hessian estimates for each calibration view, the costs $L_r(S)$, regrets $R_r(S)$, and worst-case regret $J(S)$ are fixed for every legal candidate $S \in \Omega_{B,k}$. MR-GOBS selects

$$S^\star = \arg\min_{S \in \Omega_{B,k}} J(S). \tag{44}$$

By the definition of an argmin over the finite candidate set $\Omega_{B,k}$, any selected minimizer $S^\star$ satisfies

$$J(S^\star) \le J(S), \qquad \forall S \in \Omega_{B,k}. \tag{45}$$

Any baseline N:M mask-selection rule considered inside the same block also selects a legal pruning set in $\Omega_{B,k}$. Therefore, the same inequality holds for the legal pruning set selected by that baseline rule. This proves the local minimax-regret optimality claim. □

C.3 Single-View Reduction

If there is only one calibration view, then $L_1^\star + \epsilon$ is constant with respect to S. Therefore,

$$\arg\min_{S \in \Omega_{B,k}} R_1(S) = \arg\min_{S \in \Omega_{B,k}} L_1(S). \tag{46}$$

Thus, when $R = 1$, MR-GOBS reduces to the single-view GroupOBS selector.

D Additional Experimental Results

D.1 Computational Cost of MR-GOBS

MR-GOBS replaces the local N:M mask-selection rule within each N:M block of a row while retaining the Hessian construction and the sequential compensation procedure used by SparseGPT. Therefore, the additional cost comes from evaluating the minimax-regret objective over legal local N:M pruning sets. For a block of size M with N retained weights, let $k = M - N$ be the number of pruned weights. A standard N:M SparseGPT selector ranks or selects weights within the block using scalar saliency scores, which costs $O(M)$ or $O(M \log M)$ for each N:M block in a row depending on the implementation. MR-GOBS evaluates all legal pruning sets, giving

$$|\Omega_{B,k}| = \binom{M}{k}. \tag{47}$$

For each calibration view r, it evaluates the GroupOBS cost

$$L_r(S) = \mathbf{w}_S^\top \left(\mathbf{C}^{(r)}_{S,S}\right)^{-1} \mathbf{w}_S. \tag{48}$$

A direct implementation therefore has the following complexity for mask selection in each block

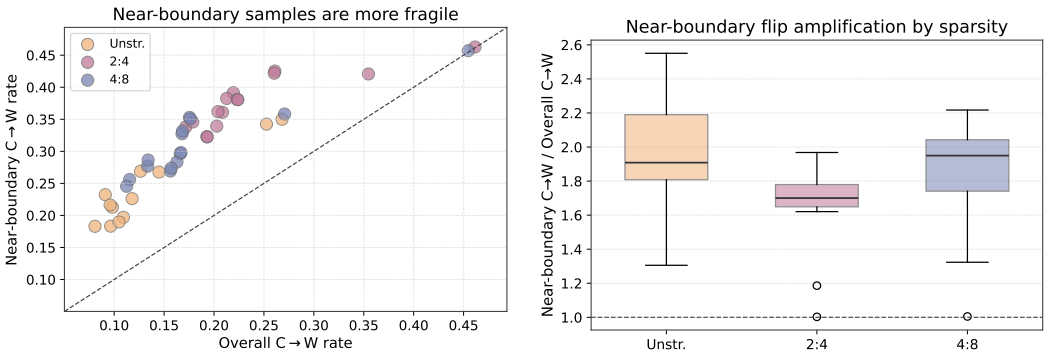
$$O\left(R\binom{M}{k}k^3\right), \tag{49}$$

where R is the number of calibration views and the k^3 term comes from solving a small $k \times k$ system. This cost is constant for fixed N:M patterns. For 2:4 pruning, $k = 2$ and only $\binom{4}{2} = 6$ candidates are evaluated per view. For 4:8 pruning, $k = 4$ and $\binom{8}{4} = 70$ candidates are evaluated per view. In our setting with $R = 2$, this corresponds to 12 small 2×2 costs for each N:M block in a row for 2:4 and 140 small 4×4 costs for each N:M block in a row for 4:8.

This enumeration increases local selection cost over scalar top-k selection, with a small fixed candidate set for common semi-structured patterns. It uses the existing forward passes and calibration signals without retraining or task-specific supervision. The resulting overhead is a local constant-factor increase in mask selection within the same pruning pipeline.

Table 4: Local mask-selection cost of MR-GOBS. The additional cost is restricted to the N:M mask-selection step; Hessian construction and the compensation procedure used by SparseGPT are unchanged.

Pattern	M	$k = M - N$	Candidates $\binom{M}{k}$	Costs per block with $R = 2$
2:4	4	2	6	12 small 2×2 costs
4:8	8	4	70	140 small 4×4 costs



(a) Overall vs. near-boundary C→W

(b) Near-boundary amplification ratio

Figure 7: Near-boundary amplification ratios under pruning. (a) Near-boundary C→W rates are generally higher than overall C→W rates. (b) The amplification ratio, defined as NB C→W divided by overall C→W, shows that examples with small absolute dense-model score-space margins are disproportionately affected by pruning.

D.2 Near-Boundary Amplification Ratios and Flip Asymmetry

Figure 7 provides an expanded view of near-boundary vulnerability. Across sparsity settings, near-boundary C→W rates are generally higher than overall C→W rates, and the amplification ratio is above one in most cases. This supports the conclusion in the main text that pruning errors concentrate around decisions with small margins.

Figure 8 further compares C→W and W→C flips. Most points lie above the symmetric flip line, indicating that pruning more often breaks originally correct predictions than repairs originally wrong ones.

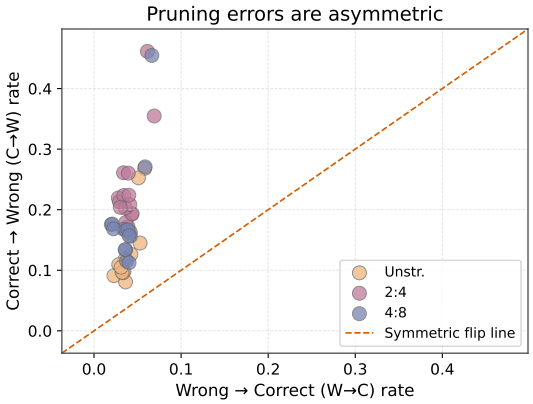
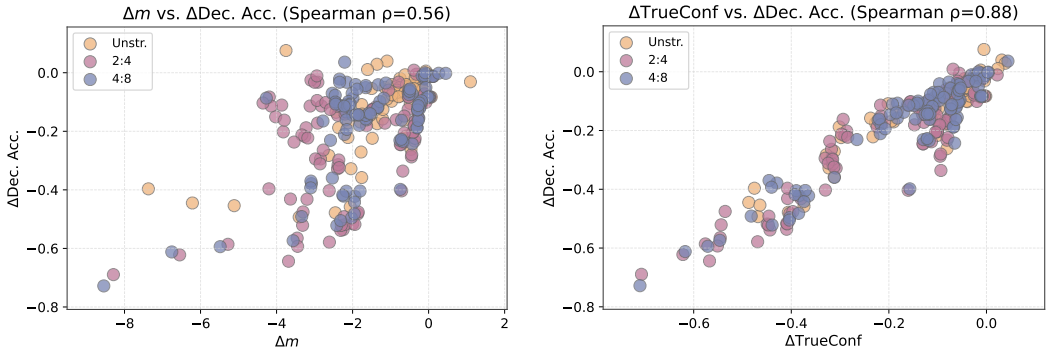


Figure 8: Correct-to-wrong and wrong-to-correct flip asymmetry. Most points lie above the symmetric flip line, showing that pruning creates more harmful C→W flips than beneficial W→C flips.

D.3 Detailed Scatter Plots of Metrics and Accuracy

Figure 9 expands the correlation analysis in Section 4.3. The score-space margin change Δm shows a moderate association with the change in paired decision accuracy, while $\Delta\mathrm{TrueConf}$ is more strongly aligned with paired accuracy degradation. These scatter plots provide the detailed evidence behind the summary correlation figure in the main text.



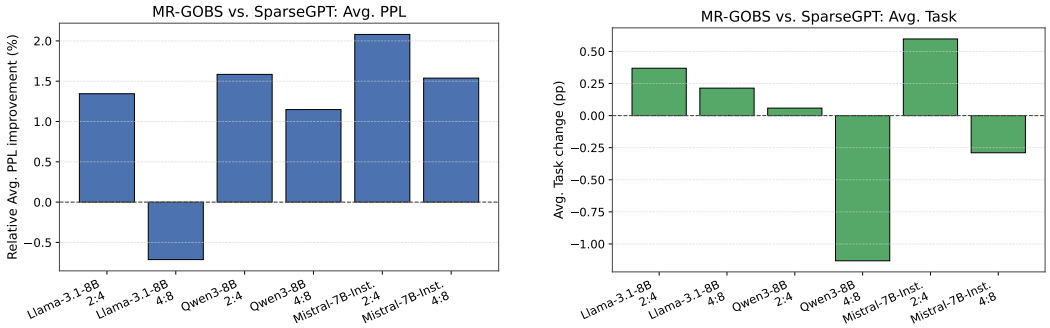
(a) Answer-margin degradation

(b) True-label confidence degradation

Figure 9: Detailed scatter plots of metrics and accuracy. Score-space margin change Δm shows a moderate association with the change in paired decision accuracy, while true-label confidence change $\Delta\mathrm{TrueConf}$ is more strongly aligned with the change in paired decision accuracy.

D.4 Additional MR-GOBS Visualization

Figure 10 visualizes the differences between MR-GOBS and SparseGPT reported in Table 2. MR-GOBS often improves Avg. PPL, but the corresponding Avg. Task changes are mixed. This visualization supports the conclusion in the main text that improvements in PPL obtained through reconstruction do not necessarily translate into better downstream task performance.



(a) Relative PPL improvement

(b) Change in task score

Figure 10: MR-GOBS compared with SparseGPT under N:M pruning. MR-GOBS often improves average perplexity, but the corresponding standard Avg. Task changes are mixed. This shows that improvements in PPL do not necessarily translate into better downstream performance.

D.5 Scale Validation on Qwen3 Models

We further test whether the observed decision-level degradation is restricted to a single 8B-scale model. We conduct scale validation within the Qwen3 family on Qwen3-1.7B, Qwen3-4B, Qwen3-8B, and Qwen3-14B, covering unstructured, 2:4, and 4:8 pruning. Table 5 reports an aggregate summary over representative post-training pruners, excluding magnitude pruning to avoid dominance by extremely degraded outlier cases.

The results show two consistent patterns. First, larger Qwen3 models are generally more robust to pruning in aggregate. Average task accuracy improves with scale, while overall C→W and near-boundary C→W rates generally decrease. Second, decision-level degradation does not disappear at larger scales. Across model sizes and sparsity settings, pruning still reduces true-label confidence and answer margins in most cases, and near-boundary examples remain more vulnerable than the overall sample set.

Figure 11 visualizes this pattern. Near-boundary C→W rates are consistently higher than overall C→W rates across Qwen3 scales and sparsity settings. Figure 12 shows that pruning generally reduces true-label confidence and score-space margins across Qwen3 scales, with smaller reductions for larger models in several settings. Figure 14 further shows that the near-boundary amplification ratio remains above one, indicating that pruning errors continue to concentrate around low-margin examples.

This scale validation further evaluates MR-GOBS across Qwen3 scales. As shown in Table 5 and Figure 13, MR-GOBS reduces PPL over SparseGPT in the evaluated Qwen3 N:M settings. However, these reductions in PPL translate into task gains only in part of the settings, and changes in flip rates are not always aligned. This confirms, within a single model family across scales, that improvements in PPL are not interchangeable with downstream decision reliability.

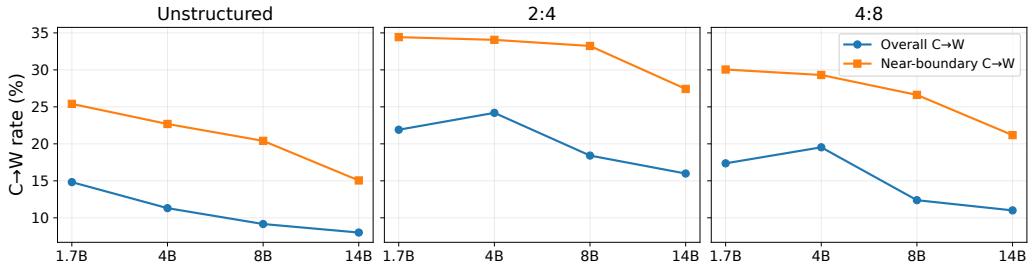


Figure 11: Overall and near-boundary C→W flip rates across Qwen3 scales. Values are averaged over representative post-training pruners other than magnitude pruning to avoid dominance by severely degraded outliers produced by magnitude pruning. Near-boundary examples consistently have higher C→W rates than the overall sample set, indicating that low-margin decisions remain more vulnerable across scales.

Table 5: Aggregate Qwen3 scale validation across 1.7B, 4B, 8B, and 14B models. Values are averaged over representative post-training pruners other than magnitude pruning. For N:M pruning, the average includes Wanda, SparseGPT, Wanda++, and MR-GOBS; for unstructured pruning, the average includes Wanda, SparseGPT, and Wanda++. NB denotes the near-boundary subset defined by the lowest 30% absolute dense-model score-space margins, and NB/Overall denotes the ratio between NB C→W and overall C→W rates.

Model	Sparsity	Avg. PPL ↓	Avg. Task (%) ↑	ΔTrueConf ↑	Δm ↑	JSD ↓	C→W (%) ↓	W→C (%) ↑	NB C→W (%) ↓	NB/Overall ↓
Qwen3-1.7B	Unstr.	33.80	49.2	-0.040	-0.873	0.064	14.8	8.3	25.4	1.71
	2:4	73.72	41.0	-0.093	-1.165	0.057	21.9	5.4	34.4	1.57
	4:8	40.58	44.5	-0.076	-1.105	0.053	17.4	5.4	30.0	1.73
Qwen3-4B	Unstr.	25.06	56.9	-0.041	-0.221	0.030	11.3	4.4	22.7	2.01
	2:4	37.35	50.5	-0.120	-0.967	0.069	24.2	5.1	34.1	1.41
	4:8	29.79	54.8	-0.090	-0.651	0.062	19.5	5.4	29.3	1.50
Qwen3-8B	Unstr.	16.78	62.7	-0.038	-0.245	0.020	9.2	3.5	20.4	2.23
	2:4	22.43	54.7	-0.093	-0.834	0.039	18.4	4.1	33.2	1.80
	4:8	18.37	59.5	-0.061	-0.532	0.026	12.4	3.7	26.6	2.15
Qwen3-14B	Unstr.	14.94	65.5	-0.023	0.103	0.030	8.0	4.4	15.1	1.88
	2:4	18.08	60.9	-0.072	-0.411	0.048	16.0	4.8	27.4	1.72
	4:8	15.81	63.7	-0.043	-0.111	0.034	11.0	4.4	21.2	1.92

D.6 Sensitivity to the Near-Boundary Threshold

Our main decision-level analysis defines near-boundary examples using the absolute dense-model score-space margin. Specifically, examples with the smallest margins are considered closer to

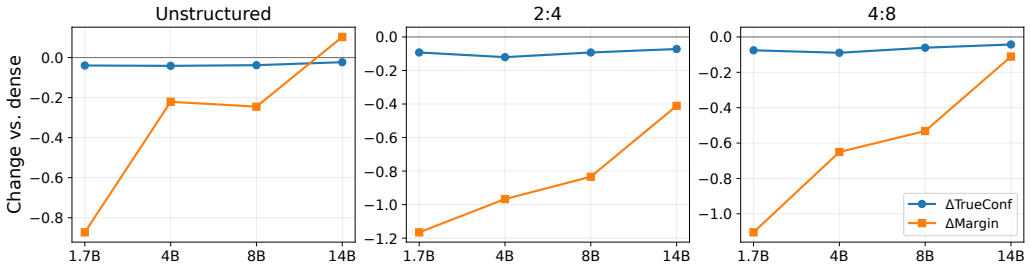


Figure 12: True-label confidence and score-space margin changes across Qwen3 scales. Pruning generally yields negative $\Delta\mathrm{TrueConf}$ and negative Δm across model sizes and sparsity settings, although larger models show smaller degradation in several settings.

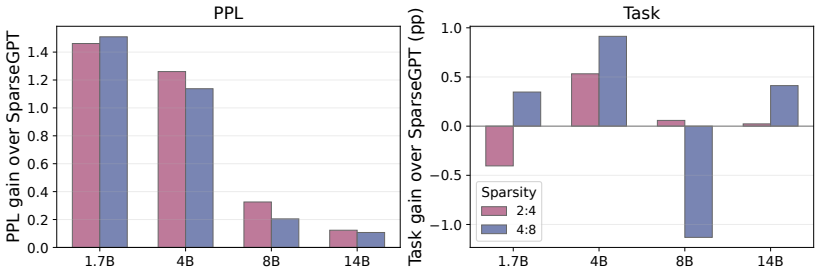


Figure 13: MR-GOBS compared with SparseGPT across Qwen3 scales. MR-GOBS reduces average PPL over SparseGPT in the evaluated Qwen3 N:M settings, but the corresponding Avg. Task changes are mixed. This further supports the mismatch between improvements in PPL and downstream decision reliability.

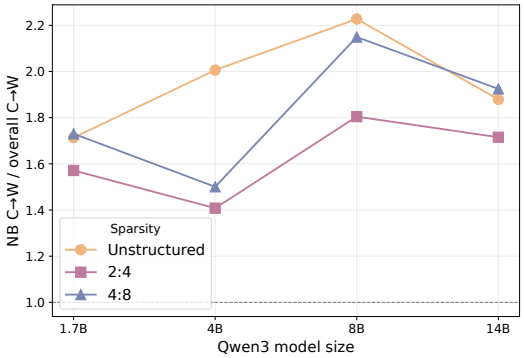


Figure 14: NB Amplification Ratio. The ratio of near-boundary to overall C→W remains > 1.0 across Qwen3 scales, showing that low-margin samples are consistently more susceptible to pruning noise.

the dense model's decision boundary. To examine whether the default threshold $q = 0.3$ affects our conclusion, we vary the near-boundary quantile from $q = 0.1$ to $q = 0.5$ and compute the near-boundary C→W rate difference:

$$\Delta_{\mathrm{NB}}(q) = \mathrm{NB\ C{\to}W}(q) - \mathrm{Overall\ C{\to}W}. \tag{50}$$

A positive value indicates that examples with small absolute dense-model margins have higher C→W rates than the overall evaluation set.

Figures 15–17 show that the near-boundary C→W rate difference is generally positive across the three main model families and both N:M sparsity patterns. It is positive in 146 of the 150 plotted combinations of models, pruning methods, N:M sparsity patterns, and thresholds; the four exceptions are Qwen3-8B pruned with magnitude pruning at $q \in \{0.1, 0.2\}$ under 2:4 and 4:8 sparsity. The rate difference is typically larger for smaller q, because smaller thresholds select examples closer to the

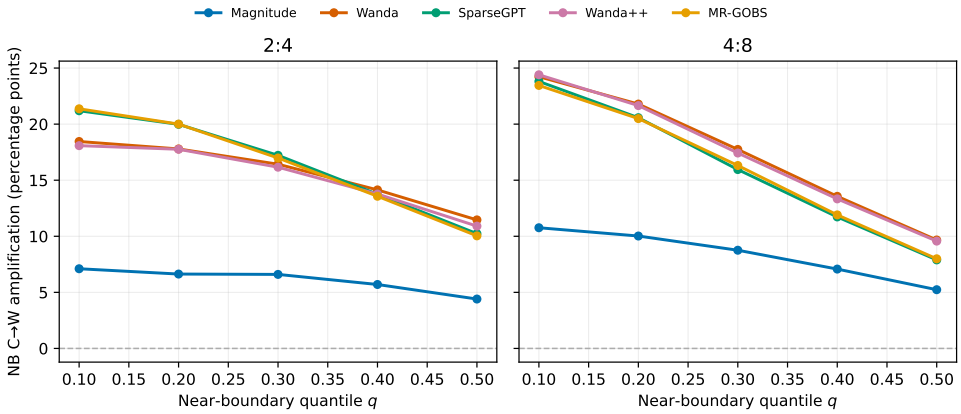


Figure 15: Near-boundary threshold sensitivity on Llama-3.1-8B. The y-axis reports the difference between near-boundary and overall C→W rates, in percentage points. Positive values indicate higher vulnerability among examples with small absolute dense-model score-space margins.

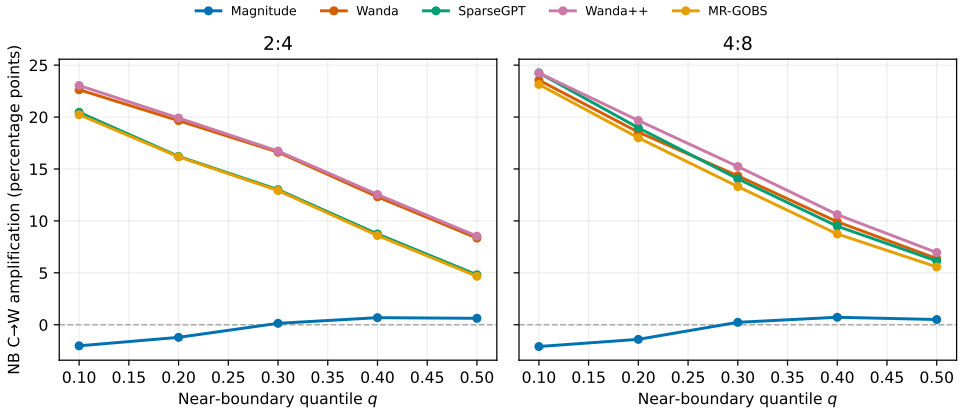


Figure 16: Near-boundary threshold sensitivity on Qwen3-8B. The y-axis reports the difference between near-boundary and overall C→W rates, in percentage points. Positive values indicate higher vulnerability among examples with small absolute dense-model score-space margins.

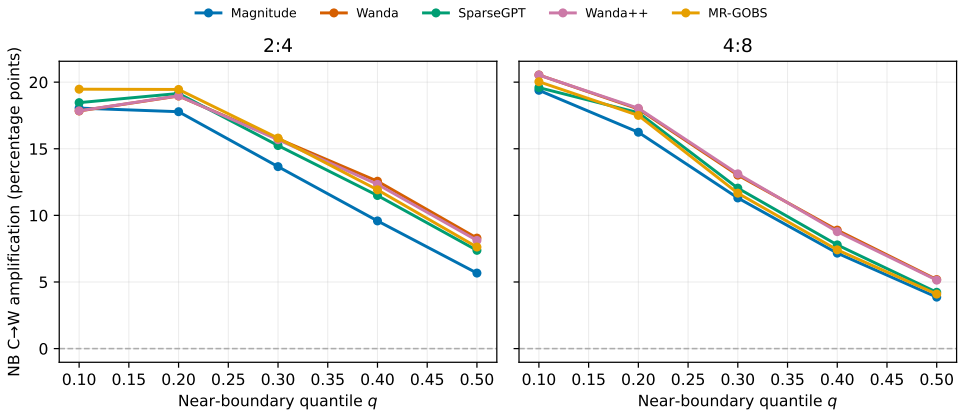


Figure 17: Near-boundary threshold sensitivity on Mistral-7B-Instruct-v0.3. The y-axis reports the difference between near-boundary and overall C→W rates, in percentage points. Positive values indicate higher vulnerability among examples with small absolute dense-model score-space margins.

dense model's decision boundary. As q increases, the near-boundary set becomes less selective and includes examples with larger margins, so this difference tends to decrease.

These results support the main conclusion that near-boundary vulnerability is not an artifact of choosing the default threshold $q = 0.3$. Instead, the dominant qualitative pattern persists across a range of thresholds from $q = 0.1$ to $q = 0.5$. We therefore use $q = 0.3$ in the main experiments as a moderate operating point that captures low-margin examples while retaining enough samples for stable aggregate estimates.

Table 6: Decision-level degradation under all pruning settings. ΔDec. Acc denotes the change in decision-level accuracy measured by the paired candidate-scoring protocol. C→W denotes correct-to-wrong flips, W→C denotes wrong-to-correct flips, and NB denotes the near-boundary subset defined by the lowest 30% absolute dense-model score-space margins. The unstructured rows report magnitude pruning, Wanda, SparseGPT, and Wanda++, while the N:M rows additionally include MR-GOBS.

Model	Sparsity	Method	ΔDec. Acc (pp) ↑	$\Delta\mathrm{TrueConf}$ ↑	Δm ↑	JSD ↓	C→W (%) ↓	W→C (%) ↑	NB C→W (%) ↓
Llama-3.1-8B	Unstr.	Magnitude	-21.0	-0.096	-0.567	0.044	26.8	5.8	35.0
		Wanda	-8.4	-0.054	-0.290	0.017	12.6	4.2	26.9
		SparseGPT	-6.8	-0.041	-0.195	0.014	9.1	2.3	23.2
		Wanda++	-9.2	-0.055	-0.304	0.025	14.5	5.3	26.8
	2:4	Magnitude	-28.6	-0.122	-0.785	0.055	35.5	6.9	42.1
		Wanda	-22.7	-0.112	-0.580	0.031	26.1	3.4	42.5
		SparseGPT	-19.1	-0.092	-0.425	0.032	21.9	2.8	39.2
		Wanda++	-22.1	-0.114	-0.586	0.032	26.0	3.9	42.2
		MR-GOBS	-18.3	-0.091	-0.427	0.031	21.3	2.9	38.2
	4:8	Magnitude	-21.2	-0.098	-0.592	0.042	27.1	5.9	35.8
		Wanda	-15.6	-0.082	-0.431	0.022	17.5	2.0	35.3
		SparseGPT	-13.4	-0.074	-0.353	0.025	16.8	3.4	32.7
		Wanda++	-15.5	-0.084	-0.445	0.022	17.6	2.1	35.0
		MR-GOBS	-14.6	-0.077	-0.372	0.027	16.8	2.2	33.2
Qwen3-8B	Unstr.	Magnitude	-20.2	-0.149	-1.348	0.068	25.3	5.1	34.2
		Wanda	-6.4	-0.048	-0.416	0.021	9.8	3.4	21.3
		SparseGPT	-4.4	-0.024	0.042	0.019	8.1	3.6	18.3
		Wanda++	-6.2	-0.042	-0.361	0.020	9.6	3.4	21.6
	2:4	Magnitude	-40.0	-0.233	-1.740	0.106	46.1	6.1	46.3
		Wanda	-13.3	-0.085	-0.745	0.033	17.2	3.9	33.8
		SparseGPT	-15.0	-0.097	-0.835	0.045	19.3	4.3	32.3
		Wanda++	-14.2	-0.091	-0.839	0.035	17.9	3.6	34.6
		MR-GOBS	-15.0	-0.099	-0.916	0.044	19.3	4.4	32.3
	4:8	Magnitude	-38.8	-0.227	-1.765	0.112	45.5	6.6	45.7
		Wanda	-9.7	-0.075	-0.716	0.028	13.3	3.6	27.7
		SparseGPT	-7.8	-0.050	-0.392	0.024	11.5	3.7	25.6
		Wanda++	-9.8	-0.071	-0.676	0.027	13.4	3.6	28.6
		MR-GOBS	-7.2	-0.048	-0.343	0.025	11.2	4.0	24.5
Mistral-7B-Inst.	Unstr.	Magnitude	-8.2	-0.053	-0.478	0.028	11.8	3.6	22.6
		Wanda	-8.1	-0.062	-0.575	0.024	10.9	2.8	19.7
		SparseGPT	-6.4	-0.051	-0.387	0.021	9.6	3.2	18.3
		Wanda++	-7.4	-0.055	-0.454	0.024	10.5	3.1	19.0
	2:4	Magnitude	-16.7	-0.123	-1.128	0.051	20.3	3.6	33.9
		Wanda	-18.9	-0.121	-1.031	0.043	22.3	3.4	38.1
		SparseGPT	-16.7	-0.117	-1.013	0.042	20.9	4.1	36.1
		Wanda++	-18.4	-0.124	-1.083	0.047	22.4	4.0	38.1
		MR-GOBS	-17.4	-0.116	-0.967	0.039	20.4	3.0	36.2
	4:8	Magnitude	-11.5	-0.086	-0.843	0.039	15.6	4.2	26.9
		Wanda	-13.0	-0.091	-0.802	0.035	16.6	3.6	29.6
		SparseGPT	-12.1	-0.094	-0.773	0.040	16.3	4.2	28.3
		Wanda++	-12.8	-0.089	-0.778	0.036	16.7	3.9	29.8
		MR-GOBS	-11.7	-0.089	-0.755	0.038	15.7	4.0	27.4

D.7 Complete Perplexity and Task Results

Tables 7 through 10 report the comprehensive PPL and downstream task results.

Table 7: Complete perplexity results for the main models. Avg. PPL is the average over WikiText2, PTB, and C4. The unstructured rows include magnitude pruning, Wanda, SparseGPT, and Wanda++, while the N:M rows additionally include MR-GOBS.

Model	Sparsity	Method	WikiText2 ↓	PTB ↓	C4 ↓	Avg. PPL ↓
Llama-3.1-8B	Unstr.	Magnitude	134.08	161.10	275.11	190.10
		Wanda	9.29	13.48	15.13	12.63
		SparseGPT	**8.26**	**12.87**	**14.55**	**11.89**
		Wanda++	11.51	16.65	18.76	15.64
	2:4	Magnitude	792.97	540.76	2423.50	1252.41
		Wanda	20.44	35.53	35.39	30.45
		SparseGPT	12.11	21.09	25.00	19.40
		Wanda++	20.32	34.10	34.84	29.75
		MR-GOBS	**12.02**	**20.89**	**24.51**	**19.14**
	4:8	Magnitude	212.59	210.12	298.63	240.45
		Wanda	13.02	20.52	21.55	18.36
		SparseGPT	**9.75**	**15.98**	**18.40**	**14.71**
		Wanda++	13.25	20.72	21.80	18.59
		MR-GOBS	9.83	16.08	18.54	14.81
Qwen3-8B	Unstr.	Magnitude	28.27	45.26	40.26	37.93
		Wanda	11.29	**18.32**	**18.87**	**16.16**
		SparseGPT	**11.24**	19.32	19.13	16.56
		Wanda++	12.04	20.24	20.52	17.60
	2:4	Magnitude	294.00	310.53	421.14	341.89
		Wanda	15.49	28.83	28.10	24.14
		SparseGPT	12.97	23.81	24.90	20.56
		Wanda++	15.92	29.92	28.55	24.79
		MR-GOBS	**12.89**	**23.53**	**24.28**	**20.23**
	4:8	Magnitude	115.58	218.49	213.42	182.50
		Wanda	12.87	21.59	22.10	18.85
		SparseGPT	**11.81**	20.40	21.26	17.82
		Wanda++	13.10	22.06	22.41	19.19
		MR-GOBS	11.88	**20.01**	**20.97**	**17.62**
Mistral-7B-Inst.	Unstr.	Magnitude	8.12	39.87	12.77	20.25
		Wanda	6.39	24.90	10.06	13.78
		SparseGPT	**6.09**	**23.07**	**10.00**	**13.05**
		Wanda++	6.48	26.92	10.28	14.56
	2:4	Magnitude	12.52	53.86	16.08	27.48
		Wanda	9.36	40.57	15.25	21.73
		SparseGPT	7.62	31.29	14.05	17.65
		Wanda++	9.52	41.21	15.13	21.96
		MR-GOBS	**7.59**	**30.33**	**13.94**	**17.29**
	4:8	Magnitude	8.95	39.71	12.40	20.35
		Wanda	7.46	30.99	11.90	16.78
		SparseGPT	6.67	26.25	11.51	14.81
		Wanda++	7.49	30.75	11.88	16.70
		MR-GOBS	**6.65**	**25.74**	**11.36**	**14.58**

Table 8: Complete downstream task results for the main models. Scores are accuracies in percent, and Avg. Task is averaged over all listed tasks. The unstructured rows include magnitude pruning, Wanda, SparseGPT, and Wanda++, while the N:M rows additionally include MR-GOBS.

Model	Sparsity	Method	ARC-C ↑	ARC-E ↑	BoolQ ↑	HellaSwag ↑	OBQA ↑	RTE ↑	WinoGrande ↑	Avg. Task ↑
Llama-3.1-8B	Unstr.	Magnitude	26.4	47.6	46.3	35.0	22.0	52.3	54.9	40.6
		Wanda	41.3	71.3	**79.3**	50.7	27.2	55.2	**70.1**	56.4
		SparseGPT	**41.6**	**73.7**	72.2	**52.4**	**31.0**	55.6	69.5	**56.6**
		Wanda++	39.5	68.1	77.9	50.8	28.6	**56.7**	69.7	55.9
	2:4	Magnitude	25.7	45.4	38.1	33.4	15.6	52.7	53.8	37.8
		Wanda	27.0	57.4	66.6	37.4	19.6	52.7	59.5	45.8
		SparseGPT	30.9	62.8	63.2	40.9	21.8	53.4	**63.5**	48.1
		Wanda++	26.9	56.9	**68.0**	37.5	20.8	**56.3**	57.9	46.3
		MR-GOBS	**31.1**	**63.0**	63.9	**41.4**	**22.6**	53.8	63.4	**48.4**
	4:8	Magnitude	27.6	52.9	49.3	37.5	20.2	53.4	54.7	42.2
		Wanda	32.4	64.9	68.7	44.4	24.2	53.8	66.0	50.6
		SparseGPT	**34.4**	**68.0**	65.6	**46.6**	24.4	53.4	67.6	51.4
		Wanda++	33.3	65.6	**69.6**	44.2	24.2	53.8	67.0	51.1
		MR-GOBS	33.3	65.6	67.2	46.6	**26.2**	**54.9**	**67.7**	**51.6**
Qwen3-8B	Unstr.	Magnitude	34.2	66.1	69.3	42.1	22.4	62.1	58.4	50.7
		Wanda	50.6	80.0	**85.2**	49.2	30.2	71.8	68.7	62.3
		SparseGPT	**53.2**	**81.3**	84.8	**51.2**	**30.8**	**75.8**	**69.8**	**63.8**
		Wanda++	48.7	79.3	85.0	49.1	29.4	75.5	66.2	61.9
	2:4	Magnitude	18.6	36.7	43.4	27.0	14.6	52.7	52.0	35.0
		Wanda	38.7	**74.2**	**81.1**	41.3	23.6	62.8	61.3	54.7
		SparseGPT	40.8	71.9	73.5	42.7	26.2	61.7	64.8	54.5
		Wanda++	38.3	73.2	80.9	41.0	23.6	**65.7**	62.7	**55.1**
		MR-GOBS	**41.8**	72.6	71.5	**43.7**	**27.2**	58.5	**66.8**	54.6
	4:8	Magnitude	19.4	40.6	41.4	28.1	15.4	52.3	51.9	35.6
		Wanda	44.5	76.8	84.8	45.6	27.2	**71.1**	64.6	59.2
		SparseGPT	45.7	77.7	84.4	46.7	**29.6**	**71.1**	**66.9**	**60.3**
		Wanda++	**46.1**	**77.8**	**85.2**	45.2	26.0	70.8	65.2	59.5
		MR-GOBS	45.6	77.4	82.6	**46.9**	28.2	67.1	66.4	59.2
Mistral-7B-Inst.	Unstr.	Magnitude	48.0	77.1	77.6	59.3	**35.8**	64.6	69.9	61.8
		Wanda	**51.7**	**80.6**	**86.4**	58.7	33.6	63.9	72.2	63.9
		SparseGPT	50.5	80.5	86.1	59.4	32.8	**70.8**	**73.5**	**64.8**
		Wanda++	51.1	79.9	86.1	**60.2**	34.6	66.4	73.1	64.5
	2:4	Magnitude	40.1	71.5	64.1	**51.9**	27.4	57.0	63.2	53.6
		Wanda	40.6	73.9	80.3	48.0	26.8	70.0	66.5	58.0
		SparseGPT	42.7	74.8	78.8	49.3	27.2	67.5	**69.2**	58.5
		Wanda++	39.9	73.0	**82.6**	47.9	24.6	**71.1**	66.4	57.9
		MR-GOBS	**43.3**	**75.2**	81.0	49.6	**28.6**	66.8	69.1	**59.1**
	4:8	Magnitude	46.6	76.8	77.4	**56.5**	**32.0**	64.6	67.3	60.2
		Wanda	47.7	77.4	84.5	53.8	29.6	69.3	69.1	61.6
		SparseGPT	47.8	**78.2**	84.6	54.1	30.6	67.9	**71.7**	62.1
		Wanda++	**47.9**	77.9	**84.9**	53.9	31.0	**71.5**	69.5	**62.4**
		MR-GOBS	47.8	77.8	84.7	54.6	30.6	66.1	71.2	61.8

D.8 Open-Ended Generation under Unstructured Pruning

Table 11 reports unstructured pruning results for the three main models, complementing the N:M generation results in Table 3.

Table 9: Complete perplexity results for additional models used for Qwen3 scale validation. Avg. PPL is the average over WikiText2, PTB, and C4. The unstructured rows include magnitude pruning, Wanda, SparseGPT, and Wanda++, while the N:M rows additionally include MR-GOBS.

Model	Sparsity	Method	WikiText2 ↓	PTB ↓	C4 ↓	Avg. PPL ↓
Qwen3-1.7B	Unstr.	Magnitude	320.39	480.96	397.17	399.51
		Wanda	**20.05**	**37.97**	**31.60**	**29.87**
		SparseGPT	21.68	43.02	33.83	32.84
		Wanda++	24.47	51.49	40.10	38.69
	2:4	Magnitude	1808.04	1614.40	1282.09	1568.18
		Wanda	59.03	126.94	111.59	99.19
		SparseGPT	**27.19**	**70.11**	57.67	51.65
		Wanda++	55.89	118.79	106.90	93.86
		MR-GOBS	27.29	71.49	**51.79**	**50.19**
	4:8	Magnitude	615.16	665.14	654.83	645.04
		Wanda	28.49	55.89	48.47	44.28
		SparseGPT	22.36	51.09	**38.94**	37.47
		Wanda++	28.72	56.55	48.56	44.61
		MR-GOBS	**21.85**	**44.65**	41.37	**35.96**
Qwen3-4B	Unstr.	Magnitude	64.33	86.23	91.79	80.79
		Wanda	**16.01**	**27.29**	**25.69**	**23.00**
		SparseGPT	17.18	31.72	28.88	25.93
		Wanda++	18.69	31.05	29.00	26.24
	2:4	Magnitude	1970.29	986.86	2423.50	1793.55
		Wanda	28.77	50.20	45.44	41.47
		SparseGPT	20.05	40.81	39.71	33.52
		Wanda++	28.55	52.20	45.71	42.15
		MR-GOBS	**19.43**	**38.34**	**39.02**	**32.26**
	4:8	Magnitude	150.16	163.64	197.39	170.40
		Wanda	23.85	39.17	38.19	33.74
		SparseGPT	18.22	30.57	31.29	26.69
		Wanda++	23.57	37.97	37.97	33.17
		MR-GOBS	**17.35**	**29.98**	**29.34**	**25.55**
Qwen3-14B	Unstr.	Magnitude	21.51	34.98	34.91	30.46
		Wanda	**9.87**	**16.30**	**16.68**	**14.28**
		SparseGPT	9.92	17.45	17.01	14.80
		Wanda++	10.61	18.98	17.66	15.75
	2:4	Magnitude	38.56	118.79	61.63	72.99
		Wanda	12.45	22.19	21.63	18.76
		SparseGPT	**11.47**	20.28	20.64	17.46
		Wanda++	12.45	22.28	21.55	18.76
		MR-GOBS	11.53	**20.09**	**20.40**	**17.34**
	4:8	Magnitude	21.17	42.19	34.03	32.47
		Wanda	10.88	18.36	18.40	15.88
		SparseGPT	**10.65**	18.47	18.25	15.79
		Wanda++	10.92	18.32	18.43	15.89
		MR-GOBS	**10.65**	**18.29**	**18.11**	**15.68**

Table 10: Complete downstream task results for additional models used for Qwen3 scale validation. Scores are accuracies in percent, and Avg. Task is averaged over all listed tasks. The unstructured rows include magnitude pruning, Wanda, SparseGPT, and Wanda++, while the N:M rows additionally include MR-GOBS.

Model	Sparsity	Method	ARC-C ↑	ARC-E ↑	BoolQ ↑	HellaSwag ↑	OBQA ↑	RTE ↑	WinoGrande ↑	Avg. Task ↑
Qwen3-1.7B	Unstr.	Magnitude	19.1	34.6	56.9	27.7	14.4	53.1	49.0	36.4
		Wanda	30.4	**63.0**	71.8	37.8	20.8	**67.9**	56.5	49.7
		SparseGPT	29.9	60.2	68.2	**39.5**	**25.0**	53.1	**59.8**	48.0
		Wanda++	**31.4**	61.0	**75.8**	37.2	23.6	62.1	57.3	**49.8**
	2:4	Magnitude	20.4	29.1	50.5	26.5	13.6	49.1	52.2	34.5
		Wanda	20.4	48.9	63.4	29.5	13.2	52.3	49.9	39.6
		SparseGPT	**21.8**	**49.7**	62.8	33.3	**18.0**	**55.2**	**56.2**	**42.4**
		Wanda++	19.8	48.2	**63.5**	29.4	13.6	52.7	53.0	40.0
		MR-GOBS	21.2	48.8	62.4	**33.3**	17.8	**55.2**	55.3	42.0
	4:8	Magnitude	21.0	30.2	51.1	26.9	14.4	51.3	50.7	35.1
		Wanda	24.0	57.2	64.1	33.6	17.6	52.7	53.4	43.2
		SparseGPT	27.6	**60.6**	**66.2**	36.2	20.6	54.9	53.7	45.7
		Wanda++	23.8	57.1	64.3	33.5	16.6	52.7	52.6	42.9
		MR-GOBS	**28.1**	59.3	63.6	**36.4**	**23.0**	**55.6**	**56.4**	**46.0**
Qwen3-4B	Unstr.	Magnitude	25.3	54.8	39.8	33.3	17.2	53.1	52.1	39.4
		Wanda	39.5	72.1	**82.8**	44.5	27.0	69.7	61.0	56.7
		SparseGPT	**44.1**	**73.7**	78.0	**46.7**	**29.8**	**71.5**	**62.6**	**58.0**
		Wanda++	39.4	69.7	81.5	44.8	27.2	67.5	61.9	56.0
	2:4	Magnitude	22.4	33.5	47.1	27.0	13.0	47.7	51.2	34.6
		Wanda	31.9	58.5	71.3	36.5	20.8	56.7	56.2	47.4
		SparseGPT	34.2	**64.2**	79.6	38.7	23.2	**74.0**	59.4	53.3
		Wanda++	31.8	58.8	72.6	36.5	20.4	56.3	56.7	47.6
		MR-GOBS	**34.9**	63.5	**80.1**	**39.0**	**25.4**	**74.0**	**60.1**	**53.9**
	4:8	Magnitude	23.5	42.6	42.1	30.4	14.4	52.3	51.4	36.7
		Wanda	36.3	64.3	70.8	40.2	22.6	**75.8**	61.0	53.0
		SparseGPT	39.9	**71.4**	76.9	41.8	26.8	71.1	**62.5**	55.8
		Wanda++	38.1	67.1	71.8	39.9	24.2	74.4	60.5	53.7
		MR-GOBS	**41.1**	71.3	**80.9**	**42.3**	**27.4**	73.3	60.7	**56.7**
Qwen3-14B	Unstr.	Magnitude	43.2	69.3	74.6	46.9	30.8	75.1	63.7	57.7
		Wanda	**55.4**	81.2	84.3	54.3	33.6	79.1	71.3	65.6
		SparseGPT	53.8	**81.3**	**87.4**	**55.3**	**33.8**	80.1	70.5	**66.0**
		Wanda++	50.5	79.0	83.6	55.2	31.0	**83.0**	**71.7**	64.9
	2:4	Magnitude	37.1	63.2	65.7	42.2	26.2	48.7	60.9	49.2
		Wanda	43.3	73.9	83.3	46.9	29.0	72.2	66.4	59.3
		SparseGPT	**47.4**	**77.7**	**86.9**	48.4	**31.4**	81.6	68.0	**63.1**
		Wanda++	43.3	71.1	79.0	46.6	29.6	72.2	66.3	58.3
		MR-GOBS	46.9	77.4	86.4	**48.6**	31.2	**82.3**	**68.7**	**63.1**
	4:8	Magnitude	44.5	72.4	68.6	45.7	28.2	54.9	64.3	54.1
		Wanda	51.2	78.2	83.1	50.7	30.0	75.8	69.2	62.6
		SparseGPT	49.7	78.9	**88.7**	51.2	33.2	**80.9**	69.8	64.6
		Wanda++	50.1	77.7	83.1	50.7	31.2	78.3	67.7	62.7
		MR-GOBS	**52.2**	**79.9**	87.7	**51.8**	**33.8**	79.4	**70.3**	**65.0**

Table 11: Open-ended generation results under unstructured pruning for the three main models. The best value for each metric and model is bolded.

Model	Sparsity	Method	GSM8K Flex. ↑	GSM8K Strict ↑	HumanEval ↑	MBPP ↑	TruthfulQA ↑
Llama-3.1-8B	Unstr.	Magnitude	1.2	0.0	0.0	0.0	21.9
		Wanda	15.2	15.0	**15.9**	**20.4**	15.5
		SparseGPT	**20.5**	**19.9**	9.8	19.2	**33.3**
		Wanda++	8.3	7.9	13.4	19.0	13.0
Qwen3-8B	Unstr.	Magnitude	3.3	2.1	5.5	2.8	**70.7**
		Wanda	**79.1**	**78.9**	25.0	45.0	50.2
		SparseGPT	78.0	77.9	14.0	39.8	48.0
		Wanda++	77.9	77.4	**29.3**	**47.0**	45.8
Mistral-7B-Inst.	Unstr.	Magnitude	15.0	13.9	**5.5**	14.4	**62.7**
		Wanda	31.9	31.8	0.6	25.6	46.0
		SparseGPT	**34.0**	**33.8**	0.6	25.4	47.5
		Wanda++	30.9	30.9	0.0	**29.8**	57.9

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: See abstract and introduction. The main claims made in the abstract and introduction accurately reflect the paper's contributions and scope, providing a clear overview of the research objectives and findings.

   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: See Appendix 5. Our paper discusses the limitations of the work performed by the authors.

   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: [Yes]

Justification: All theoretical results are accompanied by clearly stated assumptions and formal proofs. The main theorems are proved in full in the Appendix, with each condition explicitly defined.

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: See Experiment. Our paper fully discloses all the information needed to reproduce the main experimental results, including detailed descriptions of the experimental setup, parameters, and methodologies used, which are essential for verifying the main claims and conclusions.

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: Our paper provides open access to the data and code, along with sufficient instructions in the supplemental material to enable others to faithfully reproduce the main experimental results.

   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: See Experiment.

   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: [N/A]

   Justification: The experiments evaluate fixed pretrained large language models on standard, deterministic benchmarks (e.g., LM-Evaluation-Harness) using fixed calibration splits. Therefore, the evaluation results are deterministic, and reporting variance or error bars is not applicable.

   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: See Experiment.

   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 conforms to the NeurIPS Code of Ethics. It uses public models and benchmarks and does not involve human subjects or sensitive personal data.

   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: [Yes]

    Justification: Only technical reports.

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: No high risk for misuse.

    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: We cite all pretrained models, datasets, and evaluation tools used in the paper and follow their corresponding licenses and terms of use.

    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 implementation code with documentation and reproduction instructions.

    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: [No]

    Justification: The paper does not involve crowdsourcing or human-subject experiments.

    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-subject experiments, so IRB approval is not applicable.

    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: [Yes]

    Justification: LLMs were used solely for grammar correction and formatting purposes.

    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.