Bilinear Query–Key Scoring in Transformer Attention: A Strict-Generation Stress Test on Small Language Models

Anonymous ACL submission

Abstract

Generation-based multiple-choice evaluation can collapse option preference and answer formatting into one score. We use Multi-Branch Bilinear Attention (MBBA) as a localized scoring-path intervention to stress-test that collapse in small language models. MBBA replaces the fixed identity query–key scorer $q_i^\top k_j$ with an identity-initialized full-rank bilinear scorer: the key-side scoring path expands into K branch logits, then contracts before a single softmax. In SmolLM2-360M training logs, a frozen-backbone MBBA-only ablation changes strict EvalScope GEN-ACC in a sharply non-uniform way: at the $\eta{=}1\times 10^{-3}$, $K{=}2$ headline row, Winogrande and HellaSwag move by $+18.2$ and $+3.2$ absolute points, while MMLU moves by -7.4. These deltas are not task gains. Several scores are far below random-choice references, and a post-hoc LL-MCQ check shows high empty-output rates for the headline checkpoint. The measurement finding is simple: a narrow attention-scoring intervention can produce large strict-generation score shifts while format-validity evidence blocks an option-competence interpretation.

1 Introduction

Multiple-choice benchmarks are often evaluated by free-form generation and a parser. For small base models and lightly continued-trained models, that protocol can test answer formatting as much as option preference. We use an attention-scoring intervention to make that failure mode visible. Standard scaled dot-product attention (Vaswani et al., 2017) uses a fixed identity scorer $q_i^\top k_j$ to assign query–key compatibility; we replace it with an identity-initialized, full-rank, learnable bilinear scorer under low-resource continued post-training and track how strict generation scores respond.

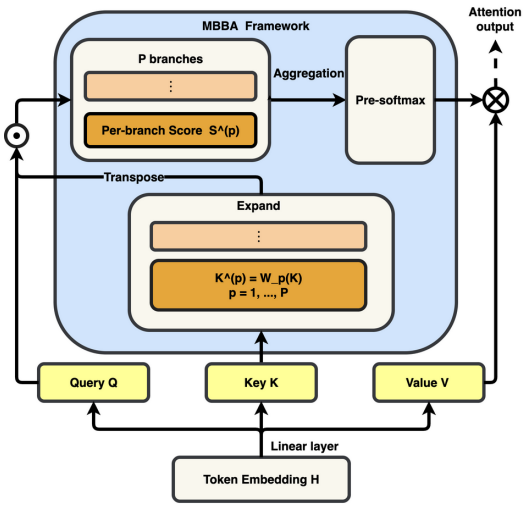

Figure 1: MBBA framework: branch scores are aggregated before a single softmax. The headline configuration uses $K{=}2$ branches; larger K changes the branch-expanded scoring parameterization rather than reasoning depth.

Related bilinear mechanisms include multimodal low-rank bilinear pooling (Kim et al., 2017), bilinear attention networks (Kim et al., 2018), and Hadamard-product perception attention (Jiang and Hu, 2023). These prior mechanisms mainly address multimodal pooling or task-specific attention with factorized Hadamard parameterizations. Here they serve as mechanism background for bilinear compatibility; the empirical object is the full-rank transformer query–key scorer used here as an evaluation stressor during continued language-model post-training.

This scoring-function view fixes the experimental question. MBBA is not a new general-purpose attention claim. It is a targeted perturbation: if strict generation scores move while format-validity checks fail, the movement belongs to the measurement pipeline before it belongs to task competence.

The evidence comes from low-resource continued post-training of small language models from existing checkpoints on the original language-modeling corpus. In this regime, optimization, output formatting, and knowledge retention in-

teract. A scoring-path change that raises some strict generation scores, lowers others, and fails format-validity checks should be analyzed as a measurement-sensitive effect rather than converted into a blanket accuracy claim.

Mechanism accounting. We use a single algebraic identity (Proposition 1) showing how Hadamard interaction implements a factorized bilinear scorer and how standard dot-product attention is the fixed identity case. This identity is not a novelty claim. It prevents a category error: the reported implementation is a full-rank key-side scorer, not the low-rank Hadamard mechanism used in several related models. Branch count therefore changes parameterization and optimization, not a formal low-rank bound.

Empirical finding. We evaluate the formulation in a low-resource continued post-training study on SmolLM2-360M that holds the data slice, optimizer schedule, checkpointing, prompting, decoding, parser, and benchmark suite fixed. The headline run is a confirmed frozen-backbone MBBA-only ablation: it keeps the pretrained backbone fixed and updates only the newly introduced MBBA scoring path. Across same-learning-rate rows for this seed, MBBA yields positive strict-score deltas with a stable sign pattern: Winogrande and HellaSwag move upward, while MMLU moves downward. The claim is the signed movement plus its failure to imply option-ranking competence.

Evaluation protocol. We evaluate generation-style EvalScope accuracy under a strict `ANSWER: X` output format. This metric combines option selection with exact answer formatting, so we pair GEN-ACC with post-hoc LL-MCQ and invalid-output checks. Format validity is therefore part of the reported result rather than an implicit confound in the aggregate score.

2 Related Work

Bilinear and Hadamard scoring. Bilinear forms $q^\top W k$ for compatibility have a long history, including the "general" alignment score in neural machine translation (Luong et al., 2015). Related work also uses low-rank Hadamard factorizations in multimodal pooling and bilinear attention networks (Kim et al., 2017, 2018), Hadamard-perceptron-style attention in captioning and physics simulation (Jiang and Hu, 2023; Xu and Li, 2024), and element-wise interaction scoring in CTR prediction (Zhou et al., 2018). Table 1 uses these lines as mechanism background by separating score form, matrix parameterization, and where scores or outputs are merged. This separation matters because the reported implementation tests an identity-initialized full-rank key-side scorer, whereas MLB/BAN-style models use explicit low-rank Hadamard factorizations.

Attention scoring vs. attention efficiency. Linear attention (Katharopoulos et al., 2020), Performer (Choromanski et al., 2021), and FlashAttention (Dao et al., 2022) change *how* attention is computed without changing the $q{\cdot}k$ scoring rule. The intervention here changes the scoring rule and keeps quadratic attention. Other attention variants also modify scoring or logit combination rather than primarily targeting IO efficiency. Talking-Heads Attention (Shazeer et al., 2020) linearly mixes attention logits across heads before and after softmax, while Synthesizer (Tay et al., 2021) learns attention weights without standard token-token dot-product interactions. MBBA differs in using an identity-initialized full-rank bilinear query–key scorer inside a head and in evaluating the resulting strict-generation score pattern.

Multi-head vs. multi-branch. Multi-head attention (Vaswani et al., 2017) aggregates after value projection by concatenating head outputs. Multi-branch bilinear attention aggregates *before* softmax with a branch-wise linear contraction over branch logits. This gives a different logit geometry from simply adding heads; matched multi-head or wider-head baselines are attribution controls in Section 5.2.

3 Method: Multi-Branch Bilinear Attention

3.1 Preliminaries

For an input sequence $X \in \mathbb{R}^{L\times d}$, standard self-attention (Vaswani et al., 2017) first computes

$$\begin{aligned} Q &= XW_Q, \\ K &= XW_K, \\ V &= XW_V, \\ \mathrm{Attn}(Q,K,V) &= \mathrm{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V. \end{aligned} \tag{1}$$

For each query–key pair, the dot-product logit is $s_{ij}^{\mathrm{dot}}{=}q_i^\top k_j/\sqrt{d_k}$: a uniformly weighted sum of

Table 1: Mechanism table for query–key scoring. "Combination" denotes where multiple scores or outputs are merged; the final row is the implementation evaluated here, with full-rank key-side branch logits contracted before one softmax.

Mechanism / reference	Scorer form	Matrix form	Combination	Role here
Dot-product attention (Vaswani et al., 2017)	$q^\top k$	fixed identity, $M{=}I$	single output	Standard attention baseline
Multi-head attention (Vaswani et al., 2017)	$q_h^\top k_h$ per head	fixed identity per head	post-softmax concat.	Aggregation contrast
MLB (Kim et al., 2017)	$q^\top A^\top \operatorname{diag}(w) B k$	low-rank bilinear, $\operatorname{rank}(M){\le}d_e$	single output	Hadamard identity background
BAN (Kim et al., 2018)	K bilinear glimpses	low-rank bilinear matrices	post-softmax	VQA bilinear-attention precedent
HPPA (Jiang and Hu, 2023)	Hadamard-perceptron scorer	low-rank / element-wise route	single output	Captioning precedent
MPT (Xu and Li, 2024)	Hadamard-perceptron scorer	low-rank / element-wise route	single output	Physics-simulation precedent
DIN (Zhou et al., 2018)	element-wise interaction scorer	learned feature interaction	single output	CTR precedent
MBBA full-rank branch-expanded scorer (ours)	$A_{\text{branch}}(\tilde{\boldsymbol{\ell}})$ before softmax	K identity-init full-rank $D_{\text{block}} \to D_{\text{block}}$ maps	pre-softmax branch-linear	Reported LM implementation

dimension-wise products $q_{i,m}k_{j,m}$. Throughout the paper, d_k denotes the per-head key dimension, equivalently d_{head} in the implementation. Figure 2 situates MBBA against this standard pipeline: the Q, K, V projections, rotary position embeddings (RoPE), key–value (KV) cache, softmax, value matmul, and the output projection W_O remain unchanged, and the only modification is on the key-side scoring path defined in Section 3.3. Section 3.2 gives the related bilinear and low-rank Hadamard accounting.

3.2 Bilinear scoring and Hadamard accounting

Bilinear query–key scoring is the umbrella family we study:

$$s_{ij}(M) = \frac{q_i^\top M k_j}{\sqrt{d_k}}, \qquad M \in \mathbb{R}^{d_k \times d_k}. \quad (2)$$

Standard dot-product attention is the fixed identity case $M{=}I$; in the rest of the paper, M denotes a scorer matrix, not an attention mask. Hadamard interaction is one factorized route into this family and gives a compact way to locate related mechanisms.

Proposition 1 (Hadamard scorer = factorized bilinear)**.** *For any* $u, v, w \in \mathbb{R}^d$, $w^\top(u \odot v){=}u^\top \operatorname{diag}(w) v$. *Therefore, for projections* $A, B \in \mathbb{R}^{d_e \times d_k}$ *and reduction* $\mu \in \mathbb{R}^{d_e}$,

$$\begin{aligned} M &= A^\top \operatorname{diag}(\mu) B, \\ \mu^\top((Aq_i) \odot (Bk_j)) &= q_i^\top M k_j, \\ \operatorname{rank}(M) &\le d_e. \end{aligned}$$

Scorer family	Status in this paper
Standard dot product: $M = I$	Fixed baseline scorer; also the identity initialization point for MBBA.
Low-rank Hadamard bilinear: $M = A^\top \operatorname{diag}(w) B$, $\operatorname{rank}(M) \le d_e$	Related bilinear subfamily used to position the tested scorer.
Reported MBBA scorer: full-rank key-side branch expansion	Tested mechanism; branch logits are contracted before one softmax, so K changes parameterization rather than a Hadamard low-rank bound.

Table 2: Scoring table used to separate the implemented MBBA scorer from related low-rank Hadamard mechanisms.

The rank statement is useful only as mechanism accounting. The reported experiments use a learnable full-rank key-side branch-expansion map, so branch count changes the parameterization and optimization path of a full-rank scorer instead of increasing a formal low-rank bound. Figure 1 visualises where branch scores are aggregated relative to the softmax; Table 2 separates that implementation from related low-rank Hadamard mechanisms.

3.3 Reported full-rank key-side instantiation

The reported MBBA experiments instantiate the same bilinear scoring umbrella with an identity-initialized full-rank key-side branch-expansion map. For each enabled layer, the implementation uses K parallel bias-free full-rank key-side linear maps over the head-concatenated block width; we write their stacked action as one map W_{branch}. In

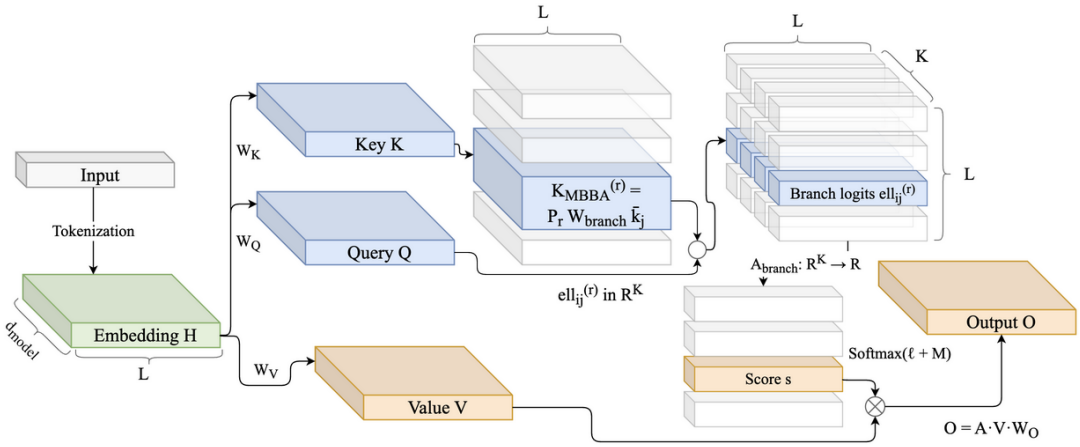

Figure 2: MBBA attention layer. The key-side scoring path (blue) replaces standard QK scoring; Q, V, RoPE, softmax, and W_O are unchanged. The attention mask is denoted $\mathcal{M}$ and the post-softmax attention matrix is denoted $\mathcal{A}$ to avoid conflict with the bilinear scorer matrix M. Branch logits are aggregated before a single softmax.

notation,

$$D_{\text{block}} = N_{\text{heads}} d_{\text{head}},$$
$$W_{\text{branch}} : \mathbb{R}^{D_{\text{block}}} \to \mathbb{R}^{K \times D_{\text{block}}},$$

initialized as K identity blocks along the branch axis. This map is applied to the standard key states after RoPE, cache update, and key–value head replication; values are not branch-expanded. Let $\bar{k}_j = [k_j^{(1)}; \ldots; k_j^{(N_{\text{heads}})}]$ be the head-concatenated key vector for token j. Let $P_r \in \{0,1\}^{d_{\text{head}} \times D_{\text{block}}}$ select the r-th contiguous d_{head} slice; when applied to $W_{\text{branch}} \bar{k}_j \in \mathbb{R}^{K \times D_{\text{block}}}$, it is applied branch-wise. The branch-expanded key features and branch-logit vector used by head r are

$$\begin{aligned} \mathbf{k}_{j,\text{MBBA}}^{(r)} &= P_r W_{\text{branch}} \bar{k}_j \in \mathbb{R}^{K \times d_{\text{head}}}, \\ \tilde{\boldsymbol{\ell}}_{ij}^{(r)} &= \frac{\mathbf{k}_{j,\text{MBBA}}^{(r)} q_i^{(r)}}{\sqrt{d_{\text{head}}}} \in \mathbb{R}^K. \end{aligned} \tag{3}$$

These branch logits are contracted before one softmax by a learned branch-wise linear map:

$$\begin{aligned} \ell_{ij}^{(r)} &= A_{\text{branch}}\left(\tilde{\boldsymbol{\ell}}_{ij}^{(r)}\right), \\ A_{\text{branch}} &: \mathbb{R}^K \to \mathbb{R}. \end{aligned} \tag{4}$$

In the reported implementation, A_{branch} is a bias-free linear contraction, shared across heads within an enabled layer and not shared across layers. Thus the experiment tests full-rank multi-branch bilinear query–key scoring. Since W_{branch} expands the key-side scoring path into a K-dimensional branch-logit vector that is then contracted by a learned branch-wise linear map, increasing K should be read as adding branch-expanded full-rank parameterization rather than as increasing a low-rank bound.

The pre-softmax placement is part of the method definition and is the SDPA implementation used for the reported runs. A variant that applies a separate softmax to each branch and only then mixes attention outputs would define a different mechanism and is outside the reported experiment set.

3.4 Interpretation and scope

The bilinear view is mainly a scoping device. In the low-rank Hadamard subfamily, increasing d_e or K can raise the rank bound. In the reported full-rank implementation, K instead changes branch-expanded parameterization and optimization while the formal rank bound remains D_{block}. The branch-count sweep is therefore not an ablation of reasoning depth or low-rank expressivity; it is a sweep over full-rank scoring-path parameterization.

MBBA can alter pairwise attention logits, but the value vectors, feed-forward blocks, residual stream, and token embeddings are unchanged. It therefore isolates a narrow adaptation channel: the scorer can change which tokens are emphasized without adding a new factual memory store. We interpret task differences through that channel under the continued post-training and generation-evaluation pipeline.

The mechanism is also distinct from simply adding heads: MBBA branches are contracted before a single head softmax, whereas multi-head outputs are mixed after value aggregation. The tested implementation adds full-rank key-side projections plus branch aggregation while keeping quadratic

attention. Parameter-, FLOP-, and architecture-matched controls remain attribution tests, as discussed in Section 5.2.

4 Experimental Protocol

Models. We use the SmolLM2-360M (Ben Allal et al., 2025) architecture as our primary backbone because its logs contain identity dot-product baselines and MBBA variants under the same data slice, learning-rate grid, checkpointing schedule, prompting, parser, and benchmark suite. The key contrast is trainable-parameter set: the identity baseline continues the original model end-to-end, whereas the selected headline MBBA row is a confirmed frozen-backbone MBBA-only ablation that updates only the newly introduced scoring path. Other model families provide boundary context in Section 5.2.

Data. Starting from existing SmolLM2 checkpoints, models are trained for one additional epoch on a fixed 5% subsample of the original HuggingFaceTB/smollm-corpus dataset (SmolLM-Corpus) (Ben Allal et al., 2024); tokenizer, context length, and packing follow the SmolLM2 setup.

Artifact use. We use public research artifacts, including SmolLM2 checkpoints, SmolLM-Corpus, EvalScope, and standard NLP benchmarks, only for research-oriented continued post-training and evaluation. We do not use them for deployment, user profiling, or decision-making applications.

Optimization. All runs use AdamW (Loshchilov and Hutter, 2019) for 1 epoch with the same batch size, sequence length, warmup, and decay schedule (Table 3). The identity baseline continues training the original SmolLM2-360M model end-to-end. The selected headline MBBA run keeps the pretrained backbone fixed and updates only the newly introduced MBBA scoring path, where identity $q^\top k$ scoring is replaced by an identity-initialized full-rank key-side branch-expansion map whose branch logits are contracted before softmax. The 6×10^{-3} setting is excluded from the main table because some identity/variant rows lack complete five-task same-pipeline checkpoints.

Evaluation protocol. We report EvalScope (Alibaba ModelScope Team, 2024) free-form generation scores, denoted GEN-ACC. The prompt, parser, decoding temperature, and evaluator are fixed across compared rows. Because exact-format failures count as task failures, GEN-ACC measures generation under a strict answer format as well as option selection. The aggregate logs preserve the prompts and commands used for evaluation, while the package commit hash is missing; we therefore report GEN-ACC together with post-hoc format checks in Section 5.1. Native aggregate logs do not include option-ranking scores.

Component	Setting
Backbone / data	SmolLM2-360M (Ben Allal et al., 2025); fixed 5% subsample of HuggingFaceTB/smollm-corpus (SmolLM-Corpus) (Ben Allal et al., 2024).
Training grid	AdamW, 1 epoch, 10% warmup, cosine decay; batch size, sequence length, and weight decay use the same settings across compared rows, but exact trainer arguments are not preserved in the aggregate logs. Attempted $\eta \in \{6\times10^{-4}, 1\times10^{-3}, 3\times10^{-3}, 6\times10^{-3}\}$ with complete same-pipeline rows used in the main tables.
Comparison	End-to-end identity $q^\top k$ continued-training baseline versus confirmed frozen-backbone MBBA-only scoring-path adaptation with full-rank key-side branch expansion before softmax.
Evaluation	EvalScope GEN-ACC with fixed free-form prompt, parser, decoding, and benchmark set across compared rows.

Table 3: Main experimental protocol. All rows use the same SmolLM2-360M starting family, 5% SmolLM-Corpus slice, optimizer schedule, checkpointing, decoding setup, prompt format, parser, and EvalScope benchmark suite. The identity baseline is an end-to-end continued-training run. The selected headline MBBA row is a frozen-backbone MBBA-only ablation with added scoring-path parameters.

Parameter and inference cost. The evaluated full-rank MBBA variants add one $D_{\text{block}} \times D_{\text{block}}$ key-side projection per branch and enabled layer. On SmolLM2-360M, MBBA-2 increases the total parameter count from 361.8M to 420.8M (+59.0M, +16.3%), while MBBA-4 reaches 479.8M (+118.0M, +32.6%). In A40/A100 generation-speed measurements over the same batch and prompt grid, MBBA-2 runs at about 0.68× baseline throughput and MBBA-4 at about 0.58–0.59×, with larger prefill slowdowns at long context lengths. The selected MBBA-2 headline run is therefore best read as a frozen-backbone scoring-path ablation with explicit parameter and latency costs.

Benchmarks. ARC-Easy (Clark et al., 2018), HellaSwag (Zellers et al., 2019), MMLU 5-shot (Hendrycks et al., 2021), PIQA (Bisk et al., 2020), and Winogrande (Sakaguchi et al., 2020). Random-chance references: 25, 25, 25, 50, 50. Generation-protocol scoring follows EvalScope (Alibaba ModelScope Team, 2024). The main experimental setup is summarized in Table 3.

Log inventory and selection rule. We use the available aggregate EvalScope analysis as the quantitative source for this study. The evidence includes 222 evaluation targets from 46 experiments, with 209 complete targets across SmolLM2-135M, 360M, and 1.7B. For the main analysis, we use only the SmolLM2-360M family because it has the densest same-learning-rate coverage between identity baselines and full-rank multi-branch bilinear runs. We first compare best checkpoints at shared learning rates, then use the $\eta{=}1\times10^{-3}$ MBBA-2 final checkpoint from the second logged variant (run B) as the task-decomposition headline. The run A/B labels are bookkeeping labels for two logged MBBA-2 variants. Run B is used because the recovered configuration and trainable-parameter audit identify it as the frozen-backbone MBBA-only ablation; it is also among the highest final-checkpoint rows and is stable across the last checkpoints.

This selection rule keeps the main comparison on a single backbone and pipeline. Qwen3 public checkpoints score far above the local continued post-training architecture-swap runs, which would conflate method, pretraining, and checkpoint quality. TinyLlama and SmolLM2-1.7B provide scope checks, but the SmolLM2-360M family provides the cleanest coverage for the main analysis.

5 Main Results

Shared learning-rate comparison. Across the SmolLM2-360M shared learning rates reported here, the strict five-task GEN-ACC average shifts by $+5.93$ at $6{\times}10^{-4}$, $+3.67$ at $1{\times}10^{-3}$, and $+2.36$ at 3×10^{-3} (Table 7). These are strict-score perturbations, not clean performance estimates: higher learning rates make several MBBA rows peak early and lose final-checkpoint advantage. We use the $1{\times}10^{-3}$ selected MBBA-2 final checkpoint only as the task-decomposition row because it is stable late in training. Its deltas in Figure 3 are sharply non-uniform: Winogrande and HellaSwag are higher, while MMLU is lower. The sweep supports the stress-test claim: a narrow scoring-path update can move strict generated-answer scores, and the meaning of that movement is inseparable from task format and output validity.

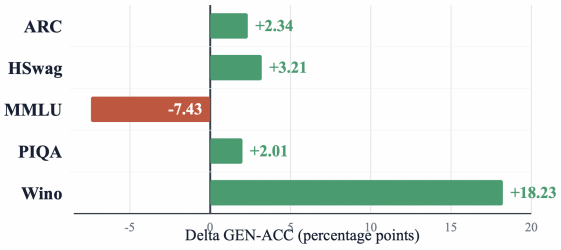

Figure 3: Task-level GEN-ACC deltas for the $\eta{=}1\times10^{-3}$ selected MBBA-2 run B final checkpoint relative to the same-LR identity-baseline best checkpoint. The plot shows the single-seed, mixed-checkpoint task decomposition used for the headline result.

5.1 Frozen-backbone MBBA-only ablation

This ablation tests whether updating only the MBBA scoring path, while keeping the pretrained backbone fixed, perturbs strict GEN-ACC scores. Table 4 reports both logged SmolLM2-360M $\eta{=}1\times10^{-3}$ MBBA-2 variants with a public calibration row and the same-LR identity baseline. The identity row is the best logged baseline checkpoint, while both MBBA rows are final frozen-backbone MBBA-only checkpoints. Run B is the selected task-decomposition headline, and the table therefore reports mixed-checkpoint contrasts, with the surrounding trajectory given in Table 5.

Non-uniform deltas and metric anatomy. Table 4 and Fig. 3 show that the selected run B has a non-uniform $+3.7$ strict-score delta: Winogrande ($+18.2$) and HellaSwag ($+3.2$) account for most of the upward movement, MMLU drops by 7.4, and ARC/PIQA move only slightly. GEN-ACC combines answer selection with output-format validity, and the absolute scores expose that interaction: several values lie far below random chance (e.g., PIQA 3.8 vs. chance 50). These numbers are strict generated-answer scores, not evidence of option-ranking accuracy. Wilson intervals in Figure 4 summarize benchmark sampling uncertainty; seed variance remains a separate quantity.

A post-hoc LL-MCQ check on 600 stratified examples at a 256-token budget makes the format effect explicit. No MBBA output exactly matched the `ANSWER: X` surface form (0% strict-format rate vs. 24.3% for identity); under a lenient extractor only 36.8% of MBBA outputs are parseable versus 93.5% for identity, and 50.8% of MBBA outputs are empty or whitespace. The direction survives parser choice: relaxing strict to lenient parsing lifts identity accuracy from 9.0% to 29.2% but moves MBBA only from 9.5% to 11.3%. Under lenient parsing MBBA scores *below* identity on every benchmark—including Winogrande (37% vs. 10%) and HellaSwag (17% vs. 7%), the two tasks that drove the apparent strict-GEN-ACC gains. The ob-

Table 4: Frozen-backbone MBBA-only GEN-ACC ablation on SmolLM2-360M. The public row calibrates the EvalScope protocol; the identity row is the best same-LR baseline checkpoint; the two MBBA rows are final frozen-backbone checkpoints. Run B is the selected task-decomposition headline. Diff. reports the strict-score contrast against the identity row.

Model	Params	ARC	HSwag	MMLU	PIQA	Wino.	Avg	Diff.
Public SmolLM2-360M (calib.)	361.8M	4.68	3.88	17.73	7.07	5.13	7.70	—
Identity dot-product baseline	361.8M	4.03	6.65	**24.24**	1.80	17.68	10.88	—
Full-rank MBBA-2, MBBA-only (run A)	420.8M	**6.63**	9.57	16.88	3.65	35.04	14.35	+3.47
Full-rank MBBA-2, MBBA-only (run B)	420.8M	6.37	**9.86**	16.81	**3.81**	**35.91**	**14.55**	**+3.67**
Random chance reference: ARC≈25, HSwag≈25, MMLU≈25, PIQA≈50, Wino.≈50.								

served GEN-ACC deltas therefore reflect output-format compliance rather than option-ranking competence.

The broader sweep makes the pattern less dependent on the headline row: across best checkpoints, ARC, PIQA, HellaSwag, and Winogrande have positive strict-score deltas while MMLU is lower in every same-LR comparison (Table 6). Final checkpoints remain learning-rate sensitive, especially at the higher LR; detailed checkpoint and LR traces are reported in the appendix.

5.2 Attribution scope

The headline row isolates one trainable-parameter regime: a frozen backbone with newly introduced scoring-path parameters. It does not separate bilinear scoring from added capacity. A dot-product model with the same extra budget placed in the FFN or in non-multiplicative projections is the next control needed for that attribution. Boundary runs narrow the scope. SmolLM2-135M contains both high-scoring rows and clear collapse cases, SmolLM2-1.7B is sparse and partly incomplete, and earlier TinyLlama/SmolTalk-style logs have weaker same-pipeline coverage. Public Qwen checkpoints remain much stronger than local low-resource continued post-training runs, and separate Qwen3-0.6B Banking77/TREC logs contain no comparable attention-scorer controls (Section A.3). The current result is an optimization-sensitive strict-generation failure mode for SmolLM2-360M, not an architecture recommendation.

6 Conclusion

We use bilinear query–key scoring as a localized intervention for stress-testing strict generation-based multiple-choice evaluation. The implemented intervention is identity-initialized full-rank multi-branch key-side scoring, separated from related low-rank Hadamard formulations. On SmolLM2-360M, the confirmed MBBA-only frozen-backbone headline row produces a consistent non-uniform strict GEN-ACC profile: Winogrande and HellaSwag are higher while MMLU is lower. The format-validity evidence fixes the interpretation. Several absolute GEN-ACC values fall far below random-choice levels, and the headline checkpoint often fails to emit a parseable answer. The result is a stress-test failure case for strict generated-answer scoring, not an option-ranking accuracy improvement. The next controls are straightforward: format-robust option scoring, multi-seed reruns, frozen-backbone non-MBBA adapters, and parameter-matched capacity baselines.

7 Limitations

This study uses a single training seed and available experiment logs. A preregistered multi-seed sweep would estimate variance around the strict GEN-ACC profile reported here. The current logs also omit four controls: parameter-matched FFN capacity, a single-full-rank bilinear scorer, a frozen-backbone non-MBBA adapter, and an explicit low-rank Hadamard variant. Without those rows, scoring geometry remains entangled with trainable-parameter set, added capacity, checkpoint choice, and answer-format compliance.

The reported metric is EvalScope generation accuracy under the strict `ANSWER: X` output format. Several absolute scores fall below random chance, and the post-hoc LL-MCQ check shows that answer format validity is a dominant factor. A clean multiple-choice version of the study should add option-ranking evaluation and then rerun the matched capacity and multi-seed controls above.

References

Alibaba ModelScope Team. 2024. Evalscope: A comprehensive evaluation framework for large language models. https://github.com/modelscope/evalscope.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martin Blazquez, Guilherme Penedo, Lewis Tunstall, Andres Marafioti, Hynek Kydlicek, Agustin Piqueres Lajarin, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clementine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, and 3 others. 2025. SmolLM2: When smol goes big – data-centric training of a small language model. *Preprint*, arXiv:2502.02737.

Loubna Ben Allal, Anton Lozhkov, Guilherme Penedo, Thomas Wolf, and Leandro von Werra. 2024. SmolLM-Corpus. Hugging Face dataset.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of AAAI*.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamás Sarlós, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. 2021. Rethinking attention with performers. In *International Conference on Learning Representations*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. In *Advances in Neural Information Processing Systems*, volume 35, pages 16344–16359.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Weitao Jiang and Haifeng Hu. 2023. Hadamard product perceptron attention for image captioning. *Neural Processing Letters*, 55(3):2707–2724.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. 2020. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, pages 5156–5165.

Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. 2018. Bilinear attention networks. In *Advances in Neural Information Processing Systems*, volume 31.

Jin-Hwa Kim, Kyoung-Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. 2017. Hadamard product for low-rank bilinear pooling. In *International Conference on Learning Representations*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of EMNLP*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. WinoGrande: An adversarial winograd schema challenge at scale. In *Proceedings of AAAI*.

Noam Shazeer, Zhenzhong Lan, Youlong Cheng, Nan Ding, and Le Hou. 2020. Talking-heads attention. *Preprint*, arXiv:2003.02436.

Yi Tay, Dara Bahri, Donald Metzler, Da-Cheng Juan, Zhe Zhao, and Che Zheng. 2021. Synthesizer: Rethinking self-attention for transformer models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10183–10192.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.

Zeyi Xu and Yifei Li. 2024. Learning physical simulation with message passing transformer. *arXiv preprint arXiv:2406.06060*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of ACL*.

Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of ACM SIGKDD*.

A Appendix

A.1 Wilson interval check

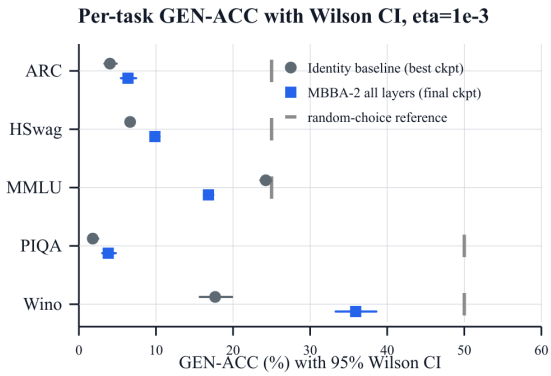

Figure 4: Per-task GEN-ACC with 95% Wilson (binomial) confidence intervals for the $\eta{=}10^{-3}$ MBBA-only ablation. Grey circles show the identity baseline at its best logged checkpoint; blue squares show full-rank MBBA-2 run B at the final checkpoint. Ticks mark random-chance references (25 for ARC/HSwag/MMLU, 50 for PIQA/Wino). Wilson intervals use per-task sample counts and summarize benchmark sampling uncertainty; they do not estimate training-seed variance or parser-induced format effects.

A.2 Checkpoint stability

Table 5 checks whether the headline final row is a checkpoint artifact for full-rank MBBA-2 at $\eta{=}10^{-3}$. The identity baseline peaks at the midpoint and then declines by final checkpoint. In contrast, both MBBA-2 variants remain in a narrow band after the first checkpoint, and run B reaches its best average at the final checkpoint. This supports the headline row as a stable late-run point for the selected setting; other branch counts and learning rates are covered by the LR sweep.

Table 5: Five-task GEN-ACC average across checkpoints at $\eta{=}1{\times}10^{-3}$. Fractions are approximate training progress from the run directory's checkpoint schedule. Δ_{final} compares each final checkpoint against the identity final checkpoint.

Config	.25	.50	.75	final	Δ_{final}
Identity baseline	7.33	10.88	10.42	8.05	—
full-rank MBBA-2 run A	13.38	**14.43**	14.33	14.35	+6.30
full-rank MBBA-2 run B	**13.49**	14.22	**14.44**	**14.55**	**+6.50**

A.3 Learning-rate and branch-count sensitivity

Table 7 reports the five-task average for the SmolLM2-360M family across selected learning rates and branch counts. Within this SmolLM-corpus setting, best-checkpoint strict-score deltas are positive across the shared learning rates reported here for this seed. Branch count is not monotone, and final checkpoints become less stable at higher learning rates. MBBA-4 can be strong at conservative learning rates, but its cost and final-checkpoint degradation make MBBA-2 the cleaner headline setting.

The task-level sign pattern repeats across the best full-rank MBBA configuration at each shared LR. Winogrande is higher by +20.2, +18.2, and +10.5 at $6{\times}10^{-4}$, $1{\times}10^{-3}$, and $3{\times}10^{-3}$ respectively; HellaSwag is higher by +9.1, +3.2, and +3.8; MMLU decreases by −6.9, −7.4, and −7.5. This repeated sign pattern is the empirical basis for the non-uniform interpretation; format compliance is measured separately through the post-hoc checks.

Table 7: Five-task average (GEN-ACC) across selected SmolLM2-360M learning rates. Δ_b compares best checkpoints against the same-LR identity best checkpoint; $\Delta_{f|b}$ compares final checkpoints against the identity best checkpoint; $\Delta_{f|f}$ compares final checkpoints against the identity final checkpoint. The $6{\times}10^{-4}$ MBBA-4 run A lacks a complete five-task final checkpoint because final MMLU is missing. Run A and run B preserve the two logged variants; they are bookkeeping labels, not a clean layer placement ablation.

Config	LR	Best	Final	Δ_b	$\Delta_{f\|b}$	$\Delta_{f\|f}$
Identity baseline	$6{\times}10^{-4}$	8.34	7.41	—	—	—
MBBA-2 run A	$6{\times}10^{-4}$	13.63	13.63	+5.29	+5.29	+6.22
MBBA-2 run B	$6{\times}10^{-4}$	13.99	13.99	+5.65	+5.65	+6.58
MBBA-4 run A	$6{\times}10^{-4}$	13.51	—	+5.17	—	—
MBBA-4 run B	$6{\times}10^{-4}$	**14.26**	**14.18**	**+5.93**	**+5.84**	**+6.77**
Identity baseline	10^{-3}	10.88	8.05	—	—	—
MBBA-2 run A	10^{-3}	14.43	14.35	+3.55	+3.47	+6.30
MBBA-2 run B	10^{-3}	**14.55**	**14.55**	**+3.67**	**+3.67**	**+6.50**
MBBA-4 run A	10^{-3}	13.47	11.63	+2.59	+0.75	+3.58
MBBA-4 run B	10^{-3}	13.56	11.96	+2.68	+1.08	+3.91
Identity baseline	$3{\times}10^{-3}$	11.52	11.52	—	—	—
MBBA-2 run A	$3{\times}10^{-3}$	**13.88**	**12.22**	**+2.36**	**+0.70**	**+0.70**
MBBA-2 run B	$3{\times}10^{-3}$	13.80	9.91	+2.28	−1.61	−1.61
MBBA-4 run A	$3{\times}10^{-3}$	13.17	10.10	+1.65	−1.42	−1.42
MBBA-4 run B	$3{\times}10^{-3}$	12.87	8.96	+1.35	−2.56	−2.56

Table 6: Sign consistency of task deltas across complete SmolLM2-360M same-LR comparisons. "+" counts tasks where MBBA is higher than the same-LR identity baseline. Best-checkpoint deltas use the same-LR identity best checkpoint; final-checkpoint deltas use the same-LR identity final checkpoint. The 6×10^{-4} MBBA-4 run A has a complete best checkpoint but no complete five-task final checkpoint, so $n{=}12$ for best and $n{=}11$ for final.

Set	n	Avg Δ	ARC+	PIQA+	HSwag+	Wino+	MMLU lower
Best ckpt.	12	+3.51	12	12	12	12	12
Final ckpt.	11	+3.18	10	11	7	11	11

The LR sweep also explains why the main text reports both best-checkpoint and final-checkpoint averages. Several configurations have high intermediate checkpoints and weaker final checkpoints, especially at $\eta{=}3\times10^{-3}$. At that LR, the best MBBA-2 rows are above the same-LR baseline at early checkpoints, but MBBA-2 run B's final checkpoint is below the best identity baseline. A stronger protocol would select checkpoints by held-out validation perplexity; those values are absent from the aggregate logs.