ASK SELF, ASK OTHERS: RELATION IS ALL YOU NEED

Anonymous authors
Paper under double-blind review

ABSTRACT

Attention dominates token mixing, but it collapses relation formation and flow allocation into a single score-to-flow step. We introduce Relation, which separates them by first organizing pairwise evidence into explicit Self and Exchange relations and deriving information flow afterward. Relation first decides whether a token should rely on itself or draw from its history, and if it draws from history, where to look. This relational organization gives rise to Full Relation, FlashRelation, Linear Relation, and Hybrid Relation. Across matched decoder-only models, Full Relation achieves lower mean final-validation NLL than MHA and reaches the paired MHA final training loss with 4.5–7.3% fewer tokens. Structural diagnostics further show that Relation learns a distinct depth organization: the first layer acts as a current-token anchor and a high-rank router, while later layers shift strongly toward history. In a fixed-context reference benchmark, FlashRelation is 4.17–$5.28\times$ faster than the materialized Full Relation implementation. Across scale-matched production workloads, it reaches 89.7–92.9% of PyTorch FlashAttention throughput while executing the exact Full Relation operator. Hybrid Relation demonstrates that Full and Linear Relation layers can be composed within a single decoder. These results support a relation-first view of token mixing: ask Self, ask Others, then let Flow follow Relation.

Keywords: Relation, Self–Exchange Relation, Token Mixing, Language Modeling, Efficient Sequence Modeling

1 INTRODUCTION

In scaled dot-product Attention, each head maps the interaction between two projected token states to a scalar score (Vaswani et al., 2017),

$$U_{ij} = \frac{q_i^\top k_j}{\sqrt{d_h}}, \qquad F_i = \mathrm{Softmax}(U_i).$$

Token interactions in language are heterogeneous, but Attention reduces each pairwise interaction to a single scalar. The scalar carries too much: the same U_{ij} must encode whatever relational structure matters and directly determine information flow. Relation formation and flow allocation are therefore compressed into a single $U \to F$ step.

Attention itself was not always primary. It first appeared as an auxiliary mechanism within recurrent encoder–decoder models before becoming the organizing primitive of the Transformer (Bahdanau et al., 2015; Vaswani et al., 2017). Relation-aware Attention places relation in a similarly subordinate position. Shaw et al. (2018) augment self-attention with pairwise relation representations, but relation is still inside Attention. Attention remains primary: it incorporates relation into its own computation and still determines the flow. We reverse this hierarchy: Flow follows Relation.

We begin by partitioning each causal Relation row into two minimal structural roles: Self and Exchange. Self describes a token's relation with itself, while Exchange describes its relation with other tokens. Pairwise evidence is mapped into these two roles before normalization, and the resulting entries are assembled into a causal Relation matrix R. Normalized information flow F is derived only afterward. This defines the basic Self–Exchange Relation (SER) operator, which extends naturally to Multi-Head Relation (MHR). This explicit organization also exposes further forms of Relation

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

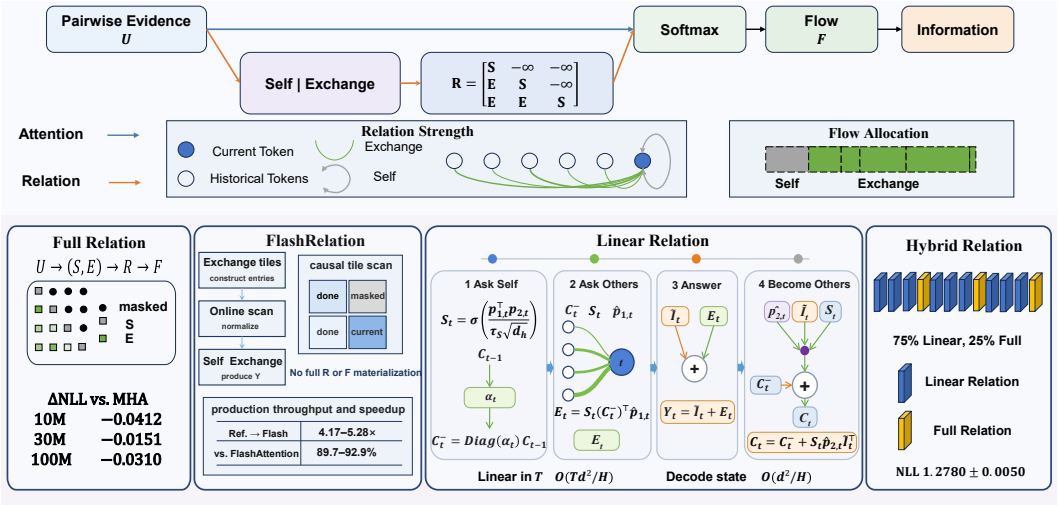

Figure 1: Overview of Relation.

in practice: Full Relation retains explicit token-wise history, FlashRelation factorizes it for tiled execution, Linear Relation compresses historical Others into a recurrent state, and Hybrid Relation combines Full and Linear layers.

We evaluate Relation in matched decoder-only models at approximately 10M, 30M, and 100M parameters. Full Relation improves mean final-validation NLL at all three scales. Structural ablations examine its behavior, while FlashRelation, Linear Relation, and Hybrid Relation demonstrate practical implementations of the framework (Figure 1). Our main contributions are:

- We introduce Self–Exchange Relation (SER) and Multi-Head Relation (MHR) as Relation operators for token mixing.
- We characterize Self–Exchange flow and derive its exact factorization into Self mass, aggregate Exchange mass, and allocation within historical Exchange. Matched structural diagnostics further show that Relation learns an internal organization distinct from Attention.
- Building on Relation's distinct relational organization, we develop Relation-specific practical forms, including FlashRelation, Linear Relation, and Hybrid Relation, to make the framework practical across exact fused, recurrent, and hybrid execution.

2 Background

Scaled dot-product Attention maps pairwise compatibility scores directly to normalized token-to-token flow (Vaswani et al., 2017). Several lines modify this pipeline without changing its organizing primitive. Relative-position methods add pairwise representations or structured biases to Attention scores (Shaw et al., 2018; Raffel et al., 2020; Press et al., 2022; Chi et al., 2022), Sigmoid self-attention changes normalization (Ramapuram et al., 2025), Talking-Heads Attention mixes across the head dimension around normalization (Shazeer et al., 2020), and GAU combines attention with gating (Hua et al., 2022). These methods alter how Attention scores or flows are formed, but Attention remains the organizing primitive. Relation instead makes pairwise relation formation an explicit stage before normalization.

Beyond Attention, alternative token-mixing operators explore different ways to organize information exchange across a sequence. FNet uses Fourier mixing (Lee-Thorp et al., 2022), PoNet uses multi-granularity pooling (Tan et al., 2022), and HyperMixer constructs token-mixing MLPs (Mai et al., 2023). These approaches organize token interaction through different mixing operators, whereas Relation organizes token mixing around explicit pairwise relations.

A third line changes how token mixing is executed or how history is represented. FlashAttention and FlashAttention-2 provide tiled, IO-aware execution of exact softmax Attention (Dao et al., 2022; Dao, 2024). Linear and recurrent methods replace explicit quadratic token-token interaction with associative or recurrent states (Katharopoulos et al., 2020; Sun et al., 2023; Yang et al., 2024; 2025). KDA further develops recurrent-state token mixing, and Kimi Linear combines recurrent KDA with full-attention MLA layers in a hybrid architecture (Kimi Team et al., 2025). Relation likewise admits efficient and recurrent realizations, while preserving the same principle: Flow follows Relation.

3 RELATION

We first introduce Self–Exchange Relation (SER), the basic Relation operator. We then extend SER to Multi-Head Relation (MHR), as illustrated in Figure 2.

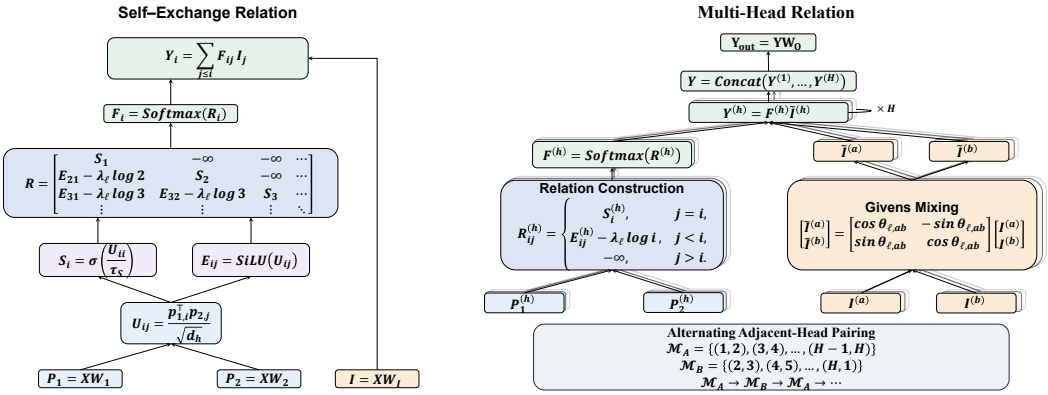

Figure 2: Self–Exchange Relation and Multi-Head Relation.

3.1 SELF–EXCHANGE RELATION

Given an input sequence X, we project it into two relation spaces and an information space,

$$P_1 = XW_1, \qquad P_2 = XW_2, \qquad I = XW_I.$$

$$U_{ij} = \frac{p_{1,i}^{\top} p_{2,j}}{\sqrt{d_h}}. \tag{1}$$

The head-wise P_1 and P_2 entering U use RoPE (Su et al., 2024), while I remains unrotated. We use two basic relational roles: Self (*ask self*), which measures the current token itself, and Exchange (*ask others*), which measures other tokens. We map pairwise evidence into these roles as

$$S_i = \sigma\left(\frac{U_{ii}}{\tau_S}\right), \qquad E_{ij} = \mathrm{SiLU}\left(U_{ij}\right),$$

$$R_{ij} = \begin{cases} S_i, & j = i, \\ E_{ij} - \lambda_\ell \log i, & j < i, \\ -\infty, & j > i. \end{cases} \qquad R = \begin{bmatrix} S_1 & -\infty & -\infty & \cdots \\ E_{21} - \lambda_\ell \log 2 & S_2 & -\infty & \cdots \\ E_{31} - \lambda_\ell \log 3 & E_{32} - \lambda_\ell \log 3 & S_3 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{2}$$

The Self temperature $\tau_S > 0$ controls the scale and smoothness of the bounded Self mapping. We set $\tau_S = 2$ in all formal experiments.

Aggregate Exchange evidence grows with the number of historical candidates, even without stronger individual Exchange relations. We therefore apply a learnable layer-wise correction $-\lambda_\ell \log i$ to all

historical Exchange entries, following the broader use of logarithmic length corrections in attention mechanisms (Ramapuram et al., 2025).

Normalized information flow is derived only after the Relation has been constructed and is applied to the information states,

$$F_i = \mathrm{Softmax}\left(R_i\right), \qquad Y_i = \sum_{j \le i} F_{ij} I_j, \qquad X_{\ell+1} = X_\ell + Y W_O. \tag{3}$$

3.2 Multi-Head Relation

We extend Relation to H heads, each of width $d_h = d/H$. Each head independently constructs its own relation matrix $R^{(h)}$ and normalized flow $F^{(h)}$.

However, the projected information state I is simply split into H head-wise slices $I^{(h)}$. Because each $R^{(h)}$ is formed independently while $I^{(h)}$ comes from direct slicing, their head-wise correspondence is imposed. We therefore mix adjacent information slices using learnable Givens rotations (Jing et al., 2017). The pairing pattern alternates between successive layers,

$$\begin{aligned} \begin{bmatrix} \widetilde{I}^{(a)} \\ \widetilde{I}^{(b)} \end{bmatrix} &= \begin{bmatrix} \cos\theta_{\ell,ab} & -\sin\theta_{\ell,ab} \\ \sin\theta_{\ell,ab} & \cos\theta_{\ell,ab} \end{bmatrix} \begin{bmatrix} I^{(a)} \\ I^{(b)} \end{bmatrix}, \\ Y^{(h)} = F^{(h)} \widetilde{I}^{(h)}, &\qquad X_{\ell+1} = X_\ell + \mathrm{Concat}_{h=1}^{H}\left(Y^{(h)}\right) W_O. \end{aligned} \tag{4}$$

4 Why Relation

We now examine what follows from placing Relation before Flow: how Self and Exchange organize Flow, and what determines the resulting allocation.

4.1 Relation Is Constructed Before Flow

Attention normalizes pairwise compatibility scores directly into token-to-token flow. Relation does not. Pairwise evidence is first organized into Self and Exchange and assembled into a role-structured matrix R. Flow is derived only afterward. Relation therefore separates relation construction from flow allocation.

Making R explicit in the operator does not require materializing the full matrix in memory. Its entries can be constructed and consumed on demand.

4.2 How Relation Forms Self–Exchange Flow

Because Self and Exchange are explicit in R, Flow inherits an exact two-level structure. For token $i > 1$, define the historical Exchange log-normalizer L_i^E, its count-corrected aggregate A_i, and the full row normalizer Z_i as

$$\begin{aligned} \mathrm{LSE}_{j<i}(a_j) \equiv \log \sum_{j<i} e^{a_j}, &\qquad L_i^E = \mathrm{LSE}_{j<i}(E_{ij}), \\ A_i = L_i^E - \lambda_\ell \log i = \mathrm{LSE}_{j<i}(R_{ij}), &\qquad Z_i = e^{S_i} + e^{A_i}. \end{aligned} \tag{5}$$

The first level divides Flow between Self and aggregate Exchange,

$$
\begin{aligned}
F_{ii} &= \frac{e^{S_i}}{Z_i} = \frac{e^{S_i}}{e^{S_i} + e^{A_i}} = \sigma(S_i - A_i), \qquad F_{ij} = \frac{e^{R_{ij}}}{Z_i}, \quad j < i, \\
g_i &\equiv \sum_{j<i} F_{ij} = \frac{\sum_{j<i} e^{R_{ij}}}{Z_i} = \frac{e^{A_i}}{Z_i} = \sigma(A_i - S_i), \qquad F_{ii} + g_i = 1, \\
Y_i &= F_{ii}\widetilde{I}_i + \sum_{j<i} F_{ij}\widetilde{I}_j = (1 - g_i)\widetilde{I}_i + \sum_{j<i} F_{ij}\widetilde{I}_j.
\end{aligned} \tag{6}
$$

Thus, F_{ii} and g_i form an exact Self–Exchange split. Conditional on Exchange, the second level allocates the Exchange mass across historical tokens according to

$$
\begin{aligned}
\pi^E_{ij} &\equiv \frac{F_{ij}}{g_i} = \frac{e^{R_{ij}}/Z_i}{\left(\sum_{k<i} e^{R_{ik}}\right)/Z_i} = \frac{e^{R_{ij}}}{\sum_{k<i} e^{R_{ik}}} \\
&= \frac{e^{E_{ij} - \lambda_\ell \log i}}{\sum_{k<i} e^{E_{ik} - \lambda_\ell \log i}} = \frac{e^{E_{ij}}}{\sum_{k<i} e^{E_{ik}}}, \qquad j < i, \\
\sum_{j<i} \pi^E_{ij} &= 1, \qquad F_{ij} = g_i \pi^E_{ij}, \qquad Y_i = (1 - g_i)\widetilde{I}_i + g_i \sum_{j<i} \pi^E_{ij}\widetilde{I}_j.
\end{aligned} \tag{7}
$$

The same two-level structure has a direct interpretation in information space. Define the Exchange aggregate and its displacement from Self as

$$
\begin{aligned}
\bar{I}^E_i &\equiv \sum_{j<i} \pi^E_{ij}\widetilde{I}_j, \qquad \Delta^E_i \equiv \bar{I}^E_i - \widetilde{I}_i, \\
Y_i &= (1 - g_i)\widetilde{I}_i + g_i \bar{I}^E_i = \widetilde{I}_i + g_i \Delta^E_i.
\end{aligned} \tag{8}
$$

Here, Δ^E_i points from the Self state toward aggregate Exchange, while g_i determines how far the output moves in that direction. As g_i increases from 0 to 1, Y_i moves continuously from $\widetilde{I}_i$ toward $\bar{I}^E_i$. Since $0 \le g_i \le 1$, Y_i lies exactly on the segment between them.

The normalized flow therefore has two levels. It first determines how much mass remains with Self and how much enters Exchange. Conditional on Exchange, it then determines how that mass is distributed across historical tokens. Intuitively, Relation first asks whether a token should rely on itself or draw from its history, and if it draws from history, it then decides where to look.

4.3 Inside Relation: How a Token's Flow Is Determined

For an individual token i, the first level is determined by the gap between aggregate Exchange and Self:

$$
\log \frac{g_i}{F_{ii}} = A_i - S_i = \mathrm{LSE}_{j<i}\left(E_{ij}\right) - S_i - \lambda_\ell \log i. \tag{9}
$$

A larger $A_i - S_i$ shifts Flow toward Exchange, while a smaller $A_i - S_i$ shifts it toward Self.

Conditional on Exchange, the second level determines allocation within history. For any two historical positions $j, k < i$,

$$
R_{ij} - R_{ik} = E_{ij} - E_{ik}. \tag{10}
$$

The row-wise correction $-\lambda_\ell \log i$ cancels in these pairwise differences. It therefore changes the Self–Exchange split without changing allocation within Exchange. That allocation is determined entirely by the Exchange relations E_{ij}.

The same two-level Relation structure also appears in backward credit assignment: gradients first act on the Self–Exchange balance and then on allocation among historical Others. Appendix A.1.7 gives the full derivation.

5 RELATION IN PRACTICE

The same Self–Exchange organization gives rise to a family of practical Relation forms. We call the explicit token-to-token operator defined in Sec. 3 Full Relation. FlashRelation is its exact fused execution, while Linear Relation represents historical Relation through a recurrent state. Hybrid Relation composes Full and Linear layers within a single decoder.

5.1 FACTORIZED SELF–EXCHANGE: FLASHRELATION

The Relation factorization derived in Sec. 4.2 separates historical Exchange aggregation from the final allocation between Self and Exchange. This factorization leads us to develop FlashRelation, a tiled realization of Full Relation. For token i, historical Exchange entries are processed through a causal tiled scan. For each tile $\mathcal{B}$, we maintain a running maximum m, normalizer l, and information accumulator z:

$$\begin{aligned} m' &= \max\left(m, \max_{j\in\mathcal{B}} E_{ij}\right), \\ l' = e^{m-m'}l + \sum_{j\in\mathcal{B}} e^{E_{ij}-m'}, &\qquad z' = e^{m-m'}z + \sum_{j\in\mathcal{B}} e^{E_{ij}-m'}\widetilde{I}_j. \end{aligned} \tag{11}$$

After each tile, $(m, l, z) \leftarrow (m', l', z')$. Let (m_i, l_i, z_i) denote the terminal accumulators after the historical Exchange scan. They yield

$$L_i^E = m_i + \log l_i, \qquad \bar{I}_i^E = \frac{z_i}{l_i}, \tag{12}$$

where L_i^E is the aggregate historical Exchange log-normalizer and $\bar{I}_i^E$ is the normalized historical information summary. The row is then completed by

$$A_i = L_i^E - \lambda_\ell \log i, \qquad g_i = \sigma(A_i - S_i), \qquad Y_i = (1-g_i)\widetilde{I}_i + g_i\bar{I}_i^E. \tag{13}$$

Following the online-softmax tiling principle of FlashAttention (Dao et al., 2022), the historical Exchange scan is evaluated tile by tile without materializing the full R or F.

5.2 CHANGE EXCHANGE: LINEAR AND HYBRID RELATION

Linear Relation preserves Self and changes Exchange. Self remains tied to the current token, while Exchange changes from explicit interaction with individual historical tokens to a recurrent relational state. Historical Others are compressed into

$$C_t^{(h)} \in \mathbb{R}^{d_h \times d_h}.$$

The Self relation keeps the construction of Sec. 3.1. Using the same projections, Linear Relation computes S_t together with two normalized relation coordinates as

$$\begin{aligned} S_t &= \sigma\left(\frac{p_{1,t}^\top p_{2,t}}{\tau_S\sqrt{d_h}}\right), \\ \hat{p}_{1,t} = \frac{p_{1,t}}{\|p_{1,t}\|_2}, &\qquad \hat{p}_{2,t} = \frac{p_{2,t}}{\|p_{2,t}\|_2}. \end{aligned} \tag{14}$$

The recurrent state carries historical Others through an input-dependent channel-wise retention. The same RoPE used in Sec. 3.1 is applied to $p_{1,t}$ and $p_{2,t}$ before S_t and the normalized coordinates are formed. We parameterize this retention following KDA (Kimi Team et al., 2025) as

$$r_t = W_\alpha^\uparrow W_\alpha^\downarrow \bar{x}_t, \qquad g_t = -\exp(A_{\log}) \odot \mathrm{softplus}(r_t + dt_{\mathrm{bias}}), \qquad \alpha_t = \exp(g_t) \in (0,1]^{d_h}. \tag{15}$$

With historical Others compressed into a recurrent state, Self remains explicit in both reading history and writing the current token into that state. Linear Relation therefore follows

$$\boxed{\text{Ask Self} \rightarrow \text{Ask Others} \rightarrow \text{Answer} \rightarrow \text{Become Others}}$$

$$\begin{aligned} \text{Ask Self:} \quad & S_t = \sigma\left(\frac{p_{1,t}^\top p_{2,t}}{\tau_S \sqrt{d_h}}\right), \\ \text{Ask Others:} \quad & E_t = S_t \left(\mathrm{Diag}(\alpha_t) C_{t-1}\right)^\top \hat{p}_{1,t}, \\ \text{Answer:} \quad & Y_t = \widetilde{I}_t + E_t, \\ \text{Become Others:} \quad & C_t = \mathrm{Diag}(\alpha_t) C_{t-1} + S_t \hat{p}_{2,t} \widetilde{I}_t^\top. \end{aligned} \tag{16}$$

Unrolling the recurrent state exposes how each historical Other contributes to the current Exchange:

$$E_t = \sum_{j<t} \underbrace{S_t}_{\text{read by Self}} \underbrace{S_j}_{\text{written by Self}} \left[\hat{p}_{2,j}^\top \underbrace{\left(\prod_{u=j+1}^{t} D_u \right)}_{\text{retention}} \hat{p}_{1,t} \right] \underbrace{\widetilde{I}_j}_{\text{historical Other}}, \qquad D_u = \mathrm{Diag}(\alpha_u).$$

Self therefore remains active on both sides of recurrent Exchange: S_j when a token becomes historical, and S_t when history is read. Linear Relation changes how Exchange is represented while preserving the Self–Exchange organization. The full derivation is given in Appendix A.4.

Exchange remains strictly historical because the retained state is read before the current token is written into it.

Changing Exchange to a recurrent state also changes its computational regime. Table 1 summarizes the resulting token-mixing complexity and decode-state requirements.

Table 1: Complexity refers to the token-mixing core. Linear sequential operations refer to the direct recurrence.

Layer Type	Token-Mixing Complexity	Sequential Ops.	Decode History State
MHA / FlashAttention	$O(T^2 d)$	$O(1)$	$O(Td)$
Full / FlashRelation	$O(T^2 d)$	$O(1)$	$O(Td)$
Linear Relation	$\mathbf{O(Td^2/H)}$	$O(T)$	$\mathbf{O(d^2/H)}$

Full and Linear Relation occupy complementary execution regimes. Linear Relation scales linearly with sequence length and uses a fixed-size decode state, while Full Relation retains $O(1)$ sequential operations. Hybrid Relation combines both within a single decoder by interleaving Linear and Full Relation layers. Following the hybrid layerwise design used in Kimi Linear (Kimi Team et al., 2025), our evaluated configuration uses nine Linear Relation layers and three Full Relation layers.

6 Experiments

We compare MHA and Full Relation under matched decoder geometry, data order, and training budget. Experiments cover approximately 10M, 30M, and 100M parameters and use paired seeds 42, 43, and 44. Scale-specific architectures, datasets, context lengths, and token budgets are summarized in Table 2. Our primary metric is final-checkpoint NLL on the full validation set, and structural ablations use the 10M setting with the same three seeds. We additionally report matched-train-loss token reduction, measuring how many fewer training tokens Relation requires to reach the paired MHA run's final smoothed training loss. All models use full-head RoPE with base 10^4 and no positional scaling. Systems benchmarks are conducted separately on an NVIDIA RTX 5090 in BF16 and report tuned steady-state throughput. Full optimization, evaluation, and per-seed details are provided in Appendices B and C.

6.1 Language Modeling and Structural Ablations

Full Relation improves mean final-validation NLL at all three model scales while also reaching the paired MHA final training loss with fewer tokens (Table 2). The corresponding token reductions are 7.3%, 5.7%, and 4.5% at 10M, 30M, and 100M.

Table 2: Main validation results. NLL is mean ± sample standard deviation over three seeds. Backbone reports layers / d_{model} / heads $\times d_h$ / d_{ff}. Token reduction reports fewer Relation training tokens required to reach the paired MHA run's final smoothed train loss.

Scale	Backbone	Data/Vocab	Ctx	Tokens	MHA NLL ↓	Relation NLL ↓	Δ	Token reduction ↑
10M	6L/384/8×48/768	TinyStories/4K	1024	150M	1.6853 ± 0.0042	$\mathbf{1.6441 \pm 0.0124}$	-0.0412	$7.3 \pm 3.4\%$
30M	10L/512/8×64/1024	TinyStories/4K	2048	450M	1.3001 ± 0.0044	$\mathbf{1.2850 \pm 0.0136}$	-0.0151	$5.7 \pm 4.9\%$
100M	20L/640/8×80/1280	SmolLM/32K	4096	1.071B	2.9373 ± 0.0093	$\mathbf{2.9063 \pm 0.0061}$	-0.0310	$4.5 \pm 3.4\%$

The Full Relation structure matters. Exchange-only transport and Raw-X communication increase NLL by 0.0320 and 0.0366, while removing count calibration or collapsing Relation to a single head produces larger degradations of about 0.051 (Table 3). The exact ablation definitions and per-seed results are provided in Appendix D.

Table 3: Structural ablations on the 10M setting. NLL is mean ± sample standard deviation over three seeds.

Metric	Full Relation	Exchange-only transport $Y_i = \sum_{j<i} F_{ij} \widetilde{I}_j$	Raw-X communication $I = X$	No count calibration $\lambda_\ell = 0$	Single head $H = 1$
NLL ↓	$\mathbf{1.6441 \pm 0.0124}$	1.6761 ± 0.0097	1.6807 ± 0.0074	1.6947 ± 0.0203	1.6948 ± 0.0221
Δ vs Full	—	$+0.0320$	$+0.0366$	$+0.0506$	$+0.0507$

6.2 Is Relation Attention?

Both Relation and Attention ultimately produce normalized token-to-token flow. Their learned organization, however, is not the same. We compare matched checkpoints at 10M, 30M, and 100M across three paired seeds, focusing on two questions: how flow is organized across depth and how historical flow is routed across tokens (Table 4).

Table 4: Internal organization of matched MHA and Relation checkpoints. (a) Depth-wise organization of diagonal/Self flow and routing rank. (b) Conditional historical routing across distance and first-token sink. Values are three-seed means. Metric definitions and sampling protocol are given in Appendix B.4.

Arrows denote MHA → Relation.

(a) Depth organization

Scale	L0 diag. / Self flow	L0 Flow EffRank	L1 diag. / Self flow	Rel. L2+ Self / Ex.
10M	$6.62\% \rightarrow \mathbf{28.57}\%$	$173.3 \rightarrow 405.4$ $(\mathbf{2.34}\times)$	$7.30\% \rightarrow 1.49\%$	$2.87\%/\mathbf{97.13}\%$
30M	$5.54\% \rightarrow \mathbf{23.80}\%$	$155.4 \rightarrow 406.9$ $(\mathbf{2.62}\times)$	$6.53\% \rightarrow 0.37\%$	$1.73\%/\mathbf{98.27}\%$
100M	$13.94\% \rightarrow \mathbf{43.81}\%$	$176.6 \rightarrow 470.8$ $(\mathbf{2.67}\times)$	$1.86\% \rightarrow 0.39\%$	$1.01\%/\mathbf{98.99}\%$

(b) Historical routing

Scale	$d > 128$	$d > 256$	$d > 512$	First-token sink
10M	$18.12\% \rightarrow 25.21\%$	$10.43\% \rightarrow 15.63\%$	$4.17\% \rightarrow 6.09\%$	$0.566\% \rightarrow 0.483\%$
30M	$18.17\% \rightarrow 23.94\%$	$9.40\% \rightarrow 14.00\%$	$3.18\% \rightarrow 5.23\%$	$0.772\% \rightarrow 0.579\%$
100M	$37.99\% \rightarrow 39.58\%$	$22.59\% \rightarrow 23.91\%$	$8.03\% \rightarrow 8.51\%$	$10.17\% \rightarrow 8.43\%$

Relation uses relation. At L0, Relation combines substantially higher Self flow than MHA's diagonal flow with a 2.34–2.67× higher flow effective rank. At L1, Self flow falls below 1.5% at every scale.

From L2 onward, 97.13–98.99% of Relation's normalized flow is assigned to Exchange. Unlike MHA, Relation uses L0 as a Self anchor and a high-rank router, then rapidly shifts into an Exchange-dominant regime. Historical routing also shifts outward. Conditional historical mass beyond 128, 256, and 512 tokens is higher at every scale, while the first-token sink is lower at every scale.

Relation is not Attention. Across all three scales, it learns a distinct depth organization. In Attention, the normalized token-to-token flow is the Attention itself. In Relation, Flow is only the consequence of a prior Relation. Attention ends at Flow. Relation begins before Flow.

6.3 Efficient Implementations

In a fixed-context $T = 1024$ reference benchmark, FlashRelation is 4.17–$5.28\times$ faster than the materialized Full Relation reference implementation. Across the three scale-matched production workloads, FlashRelation reaches 89.7–92.9% of PyTorch FlashAttention throughput while executing the exact Full Relation operator (Table 5). Linear Relation throughput results are reported in Appendix H.

Table 5: FlashRelation systems results on an RTX 5090 in BF16. (a) Fixed-context ($T = 1024$) comparison with the materialized Full Relation reference. (b) Scale-matched production throughput against PyTorch FlashAttention.

(a) Fixed-context $T = 1024$

Scale	Ref. Full	FlashRelation	Speedup
10M	128,515	679,121	$\mathbf{5.28}\times$
30M	74,376	365,766	$\mathbf{4.92}\times$
100M	30,631	127,715	$\mathbf{4.17}\times$

(b) Scale-matched production

Scale	T	FlashAttn.	FlashRelation	FR / FA
10M	1024	841,834	755,088	$\mathbf{0.897}\times$
30M	2048	361,793	334,028	$\mathbf{0.923}\times$
100M	4096	107,783	100,167	$\mathbf{0.929}\times$

Hybrid Relation combines nine Linear and three Full Relation layers in a $(LLLF)^3$ layout. Under the 30M-class training setting, the 31.97M-parameter model achieves 1.2780 ± 0.0050 final-validation NLL across three seeds. This demonstrates that Linear and Full Relation layers can be composed in a decoder in which nine of twelve token-mixing layers use Linear Relation.

7 Limitations and Conclusion

Limitations. Our experiments are limited to decoder-only language models up to approximately 100M parameters and 1.071B training tokens. Full Relation is evaluated systematically across scales, while Linear and Hybrid Relation are studied only in the configurations reported here. We also fix $\tau_S = 2$, sigmoid for Self, and SiLU for Exchange. Other role-specific mappings may behave differently. Larger models, multimodal settings, and post-training remain unexplored.

Conclusion. We introduced Relation, an alternative token-mixing primitive that organizes pairwise evidence into explicit Self and Exchange relations before deriving information flow. Full Relation retains explicit token-wise history. FlashRelation provides exact factorized execution. Linear Relation compresses historical Others into a recurrent state, and Hybrid Relation combines Full and Linear layers.

Structural diagnostics show a repeatable organization distinct from matched MHA. Relation uses L0 as a Self anchor and a high-rank router, then shifts rapidly into an Exchange-dominant regime. Across three model scales, Full Relation achieves lower final-validation NLL than matched MHA. FlashRelation provides practical execution, while Hybrid Relation retains strong language-modeling quality with 75% Linear Relation layers.

Together, these results support a broader view of token mixing. Token mixing need not be organized directly around information flow: it can begin with Relation.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1409.0473.

Ta-Chung Chi, Ting-Han Fan, Peter J. Ramadge, and Alexander Rudnicky. Kerple: Kernelized relative positional embedding for length extrapolation. In *Advances in Neural Information Processing Systems*, volume 35, pp. 8386–8399, 2022. doi: 10.52202/068431-0610. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/37a413841a614b5414b333585e7613b8-Abstract-Conference.html.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, volume 2024, pp. 35549–35562, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/98ed250b203d1ac6b24bbcf263e3d4a7-Paper-Conference.pdf.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. In *Advances in Neural Information Processing Systems*, volume 35, pp. 16344–16359, 2022. doi: 10.52202/068431-1189. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/67d57c32e20fd0a7a302cb81d36e40d5-Abstract-Conference.html.

Weizhe Hua, Zihang Dai, Hanxiao Liu, and Quoc Le. Transformer quality in linear time. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 9099–9117. PMLR, 2022. URL https://proceedings.mlr.press/v162/hua22a.html.

Li Jing, Yichen Shen, Tena Dubcek, John Peurifoy, Scott Skirlo, Yann LeCun, Max Tegmark, and Marin Soljačić. Tunable efficient unitary neural networks (EUNN) and their application to RNNs. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1733–1741. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/jing17a.html.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 5156–5165. PMLR, 2020. URL https://proceedings.mlr.press/v119/katharopoulos20a.html.

Kimi Team, Yu Zhang, Zongyu Lin, Xingcheng Yao, Jiaxi Hu, Fanqing Meng, Chengyin Liu, Xin Men, Songlin Yang, Zhiyuan Li, Wentao Li, Enzhe Lu, Weizhou Liu, Yanru Chen, Weixin Xu, Longhui Yu, Yejie Wang, Yu Fan, Longguang Zhong, Enming Yuan, Dehao Zhang, Yizhi Zhang, T. Y. Liu, Haiming Wang, Shengjun Fang, Weiran He, Shaowei Liu, Yiwei Li, Jianlin Su, Jiezhong Qiu, Bo Pang, Junjie Yan, Zhejun Jiang, Weixiao Huang, Bohong Yin, Jiacheng You, Chu Wei, Zhengtao Wang, Chao Hong, Yutian Chen, Guanduo Chen, Yucheng Wang, Huabin Zheng, Feng Wang, Yibo Liu, Mengnan Dong, Zheng Zhang, Siyuan Pan, Wenhao Wu, Yuhao Wu, Longyu Guan, Jiawen Tao, Guohong Fu, Xinran Xu, Yuzhi Wang, Guokun Lai, Yuxin Wu, Xinyu Zhou, Zhilin Yang, and Yulun Du. Kimi linear: An expressive, efficient attention architecture. *arXiv preprint arXiv:2510.26692*, 2025. URL https://arxiv.org/abs/2510.26692.

James Lee-Thorp, Joshua Ainslie, Ilya Eckstein, and Santiago Ontañón. Fnet: Mixing tokens with fourier transforms. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4296–4313. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.naacl-main.319. URL https://aclanthology.org/2022.naacl-main.319/.

Florian Mai, Arnaud Pannatier, Fabio Fehr, Haolin Chen, Francois Marelli, Francois Fleuret, and James Henderson. Hypermixer: An mlp-based low cost alternative to transformers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15632–15654. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.acl-long.871. URL https://aclanthology.org/2023.acl-long.871/.

Ofir Press, Noah Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=R8sQPpGCv0.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL https://jmlr.org/papers/v21/20-074.html.

Jason Ramapuram, Federico Danieli, Eeshan Gunesh Dhekane, Floris Weers, Dan Busbridge, Pierre Ablin, Tatiana Likhomanenko, Jagrit Digani, Zijin Gu, Amitis Shidani, and Russell Webb. Theory, analysis, and best practices for sigmoid self-attention. In *International Conference on Learning Representations*, volume 2025, pp. 65464–65519, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/a43b459e9fab3a703148ba0c83b8a442-Paper-Conference.pdf.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 464–468, 2018. doi: 10.18653/v1/N18-2074. URL https://aclanthology.org/N18-2074/.

Noam Shazeer, Zhenzhong Lan, Youlong Cheng, Nan Ding, and Le Hou. Talking-heads attention. *arXiv preprint arXiv:2003.02436*, 2020. URL https://arxiv.org/abs/2003.02436.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. doi: 10.1016/j.neucom.2023.127063. URL https://doi.org/10.1016/j.neucom.2023.127063.

Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models. *arXiv preprint arXiv:2307.08621*, 2023. URL https://arxiv.org/abs/2307.08621.

Chao-Hong Tan, Qian Chen, Wen Wang, Qinglin Zhang, Siqi Zheng, and Zhen-Hua Ling. Ponet: Pooling network for efficient token mixing in long sequences. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=9jInD9JjicF.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pp. 5998–6008, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 56501–56523. PMLR, 2024. URL https://proceedings.mlr.press/v235/yang24ab.html.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving mamba2 with delta rule. In *International Conference on Learning Representations*, pp. 29687–29707, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/4904fad153f6434a7bcf04465d4be2cc-Abstract-Conference.html.

AI USE STATEMENT

In this work, we used generative AI tools to assist with implementing methods through experimental software code and debugging. We have not used generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in writing proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, assist with translation, clean or reformat datasets, support qualitative or thematic data analysis, or interpret results; the remaining required-disclosure tasks are not applicable to this work. Additionally, we used generative AI tools to edit the research paper to improve readability. We have reviewed all AI-assisted work. AI-assisted code incorporated into this work was reviewed, tested, and validated by the author, and all references and citations were independently verified against their original or public sources. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

A Detailed Operator Derivations

This appendix records the operator definitions used throughout the paper. Token indices in the count correction are one-based. A head has width d_h, and the information state after input projection and head mixing is denoted by $\widetilde{I}$.

A.1 Self–Exchange Relation

A.1.1 Relation construction

We first write the construction for one head. If the model-wide projection has width d, the following quantities are understood after selecting one head subspace of width $d_h = d/H$. Thus

$$P_1, P_2, I \in \mathbb{R}^{T\times d_h}, \qquad P_1 = XW_1, \quad P_2 = XW_2, \quad I = XW_I.$$

The head-wise P_1 and P_2 entering U use full-head RoPE, whereas I remains unrotated. Writing $p_{1,i}$ and $p_{2,j}$ for rows of these projected states,

$$U_{ij} = \frac{p_{1,i}^\top p_{2,j}}{\sqrt{d_h}}.$$

The Self and Exchange entries are

$$S_i = \sigma\left(\frac{U_{ii}}{\tau_S}\right), \qquad E_{ij} = \mathrm{SiLU}(U_{ij}), \qquad \tau_S > 0, \quad \tau_S = 2.$$

Here τ_S is the Self temperature used to set the scale and smoothness of the bounded Self mapping. The count-calibration parameter λ_ℓ is one unconstrained FP32 scalar per layer, shared across heads and initialized to 0.5. Causality is imposed when these role-specific entries are assembled:

$$R_{ij} = \begin{cases} S_i, & j = i, \\ E_{ij} - \lambda_\ell \log i, & j < i, \\ -\infty, & j > i. \end{cases}$$

A.1.2 Exact row normalizer

The normalized row is $F_i = \mathrm{Softmax}(R_i)$. For $i > 1$, its normalizer is

$$\begin{aligned} Z_i &= \sum_{k\le i} e^{R_{ik}} \\ &= e^{S_i} + \sum_{j<i} e^{E_{ij} - \lambda_\ell \log i} \\ &= e^{S_i} + e^{-\lambda_\ell \log i} \sum_{j<i} e^{E_{ij}} \\ &= e^{S_i} + i^{-\lambda_\ell} \sum_{j<i} e^{E_{ij}}. \end{aligned}$$

Define the historical Exchange log-normalizer and its count-corrected version by

$$L_i^E = \mathrm{LSE}_{j<i}(E_{ij}) = \log \sum_{j<i} e^{E_{ij}}, \qquad A_i = L_i^E - \lambda_\ell \log i.$$

Then

$$e^{A_i} = e^{-\lambda_\ell \log i} \sum_{j<i} e^{E_{ij}}, \qquad \boxed{Z_i = e^{S_i} + e^{A_i}}.$$

A.1.3 SELF MASS AND AGGREGATE EXCHANGE MASS

The diagonal Self probability is

$$F_{ii} = \frac{e^{S_i}}{Z_i} = \frac{e^{S_i}}{e^{S_i} + e^{A_i}} = \frac{1}{1 + e^{A_i - S_i}} = \boxed{\sigma(S_i - A_i)}.$$

The total historical Exchange mass is

$$g_i = \sum_{j<i} F_{ij} = \frac{e^{A_i}}{e^{S_i} + e^{A_i}} = \frac{1}{1 + e^{S_i - A_i}} = \boxed{\sigma(A_i - S_i)}.$$

Consequently,

$$\boxed{F_{ii} + g_i = 1}.$$

The Self–Exchange odds are

$$\frac{g_i}{F_{ii}} = \frac{e^{A_i}}{e^{S_i}} = e^{A_i - S_i},$$

and therefore

$$\boxed{\log \frac{g_i}{F_{ii}} = A_i - S_i = L_i^E - S_i - \lambda_\ell \log i.}$$

A.1.4 CONDITIONAL EXCHANGE ALLOCATION

For a historical position $j < i$, define its conditional Exchange allocation as

$$\pi_{ij}^E = \frac{F_{ij}}{g_i}.$$

Substituting the normalized row and the aggregate Exchange mass gives

$$\pi_{ij}^E = \frac{e^{E_{ij} - \lambda_\ell \log i}/Z_i}{e^{A_i}/Z_i} = \frac{e^{E_{ij} - \lambda_\ell \log i}}{\sum_{k<i} e^{E_{ik} - \lambda_\ell \log i}} = \frac{e^{E_{ij}}}{\sum_{k<i} e^{E_{ik}}}.$$

The row-level count correction cancels because it is identical for every historical position. Hence

$$\boxed{\pi_{ij}^E = \mathrm{Softmax}_{j<i}(E_{ij})}, \qquad \sum_{j<i} \pi_{ij}^E = 1,$$

and the original historical flow entries factor as

$$\boxed{F_{ij} = g_i \pi_{ij}^E}.$$

A.1.5 COUNT CORRECTION

For any two historical positions $j, k < i$,

$$R_{ij} - R_{ik} = (E_{ij} - \lambda_\ell \log i) - (E_{ik} - \lambda_\ell \log i) = \boxed{E_{ij} - E_{ik}}.$$

Equivalently,

$$\frac{\pi_{ij}^E}{\pi_{ik}^E} = e^{E_{ij} - E_{ik}}.$$

Thus $-\lambda_\ell \log i$ is a row-level Exchange translation. It changes the Self–Exchange mass competition through $A_i - S_i$, but it does not change conditional Exchange allocation.

A.1.6 EXACT INFORMATION TRANSPORT

Define the normalized historical information summary

$$\bar{I}_i^E = \sum_{j<i} \pi_{ij}^E \widetilde{I}_j.$$

The original Full Relation transport is

$$\begin{aligned} Y_i &= \sum_{j \le i} F_{ij} \widetilde{I}_j \\ &= F_{ii} \widetilde{I}_i + \sum_{j<i} F_{ij} \widetilde{I}_j \\ &= F_{ii} \widetilde{I}_i + g_i \sum_{j<i} \pi_{ij}^E \widetilde{I}_j \\ &= \boxed{(1-g_i)\widetilde{I}_i + g_i \bar{I}_i^E}. \end{aligned}$$

Define the Self-to-Exchange displacement by

$$\Delta_i^E \equiv \bar{I}_i^E - \widetilde{I}_i.$$

Therefore,

$$Y_i = \widetilde{I}_i + g_i \Delta_i^E.$$

Since $0 \le g_i \le 1$, this is an exact interpolation between the current Self information state and the conditional Exchange aggregate. Equivalently, Δ_i^E specifies the Self-to-Exchange displacement and g_i specifies its magnitude.

A.1.7 BACKWARD CREDIT DECOMPOSITION

Define:

$$G_i \equiv \frac{\partial \mathcal{L}}{\partial Y_i}.$$

From

$$Y_i = \sum_{j \le i} F_{ij} \widetilde{I}_j$$

we obtain

$$\frac{\partial \mathcal{L}}{\partial \widetilde{I}_j} = \sum_{i \ge j} F_{ij} G_i.$$

Using the Self–Exchange factorization,

$$\frac{\partial \mathcal{L}}{\partial \widetilde{I}_j} = (1-g_j) G_j + \sum_{i>j} g_i \pi_{ij}^E G_i.$$

Using

$$Y_i = \widetilde{I}_i + g_i \Delta_i^E$$

define

$$G_i \equiv \frac{\partial \mathcal{L}}{\partial Y_i}, \qquad c_i \equiv G_i^\top \Delta_i^E.$$

$$\frac{\partial \mathcal{L}}{\partial g_i} = \left(\frac{\partial \mathcal{L}}{\partial Y_i}\right)^\top \frac{\partial Y_i}{\partial g_i} = G_i^\top \Delta_i^E = c_i, \qquad \frac{\partial Y_i}{\partial g_i} = \frac{\partial}{\partial g_i}\left(\widetilde{I}_i + g_i \Delta_i^E\right) = \Delta_i^E.$$

$$g_i = \sigma(A_i - S_i),$$

$$\frac{\partial g_i}{\partial (A_i - S_i)} = \sigma(A_i - S_i)\left[1 - \sigma(A_i - S_i)\right] = g_i(1-g_i).$$

$$\frac{\partial \mathcal{L}}{\partial (A_i - S_i)} = \frac{\partial \mathcal{L}}{\partial g_i} \frac{\partial g_i}{\partial (A_i - S_i)} = g_i(1 - g_i)c_i.$$

$$\frac{\partial \mathcal{L}}{\partial A_i} = g_i(1 - g_i)c_i, \qquad \frac{\partial \mathcal{L}}{\partial S_i} = -g_i(1 - g_i)c_i.$$

Since

$$A_i = \mathrm{LSE}_{j<i}(E_{ij}) - \lambda_\ell \log i,$$
$$\frac{\partial A_i}{\partial E_{ij}} = \pi^E_{ij}.$$

Moreover,

$$\bar{I}^E_i = \sum_{k<i} \pi^E_{ik} \widetilde{I}_k$$

satisfies

$$\frac{\partial \bar{I}^E_i}{\partial E_{ij}} = \pi^E_{ij} \left(\widetilde{I}_j - \bar{I}^E_i \right).$$

Therefore,

$$\frac{\partial \mathcal{L}}{\partial E_{ij}} = g_i(1 - g_i)c_i \pi^E_{ij} + g_i \pi^E_{ij} G_i^\top \left(\widetilde{I}_j - \bar{I}^E_i \right),$$

which can be written as

$$\frac{\partial \mathcal{L}}{\partial E_{ij}} = g_i \pi^E_{ij} \left[(1 - g_i)c_i + G_i^\top \left(\widetilde{I}_j - \bar{I}^E_i \right) \right].$$

The first term propagates credit through the Self–Exchange amount g_i, while the second propagates credit through the conditional Exchange allocation π^E_{ij}.

Backward propagation mirrors the same two-stage Relation semantics as the forward pass: first how much flow remains with Self versus enters Exchange, then how that Exchange flow is allocated among historical Others.

For the general Relation transport, let

$$H_{ij} \equiv G_i^\top \widetilde{I}_j.$$

Then

$$\frac{\partial \mathcal{L}}{\partial F_{ij}} = H_{ij}.$$

Because $F_i = \mathrm{Softmax}(R_i)$,

$$\frac{\partial \mathcal{L}}{\partial R_{ij}} = F_{ij} \left(H_{ij} - \sum_{k \le i} F_{ik} H_{ik} \right).$$

For the role mappings, Self gives

$$\frac{\partial S_i}{\partial U_{ii}} = \frac{1}{\tau_S} S_i(1 - S_i).$$

If

$$E_{ij} = \mathrm{SiLU}(U_{ij}),$$

then

$$\frac{\partial E_{ij}}{\partial U_{ij}} = \sigma(U_{ij}) + U_{ij}\sigma(U_{ij})\big(1 - \sigma(U_{ij})\big).$$

For

$$U = \frac{P_1 P_2^\top}{\sqrt{d_h}},$$

$$\nabla_{P_1} \mathcal{L} = \frac{\nabla_U \mathcal{L}\, P_2}{\sqrt{d_h}}, \qquad \nabla_{P_2} \mathcal{L} = \frac{\nabla_U^\top \mathcal{L}\, P_1}{\sqrt{d_h}}.$$

Finally,

$$\frac{\partial \mathcal{L}}{\partial \lambda_\ell} = -\sum_i g_i(1 - g_i)c_i \log i,$$

where the sum ranges over the valid $i > 1$ rows in the layer.

A.1.8 BOUNDARY CASE

For the first token, the historical set is empty. Define

$$\mathrm{LSE}_{j<1} = -\infty, \qquad A_1 = -\infty.$$

It follows directly that

$$\boxed{g_1 = 0, \qquad F_{11} = 1, \qquad Y_1 = \widetilde{I}_1}.$$

A.2 MULTI–HEAD RELATION

A.2.1 HEADWISE CONSTRUCTION

Let $d_h = d/H$. After projection and head splitting,

$$P_1^{(h)}, P_2^{(h)}, I^{(h)} \in \mathbb{R}^{T\times d_h}, \qquad h = 1, \dots, H.$$

Each head independently applies the Self–Exchange construction of Appendix A.1, yielding

$$R^{(h)}, \qquad F^{(h)}, \qquad Y^{(h)}.$$

Relation construction is head-local and cross-head interaction is applied only to the information branch.

A.2.2 GIVENS BLOCK

For an adjacent head pair (a, b), define

$$G(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.$$

The information states are mixed by

$$\begin{bmatrix} \widetilde{I}^{(a)} \\ \widetilde{I}^{(b)} \end{bmatrix} = G(\theta_{\ell,ab}) \begin{bmatrix} I^{(a)} \\ I^{(b)} \end{bmatrix}.$$

Since

$$G(\theta)^\top G(\theta) = I_2,$$

the transformation is orthogonal. Applied to every information channel of the pair,

$$\left\|\widetilde{I}^{(a)}\right\|_2^2 + \left\|\widetilde{I}^{(b)}\right\|_2^2 = \left\|I^{(a)}\right\|_2^2 + \left\|I^{(b)}\right\|_2^2.$$

The same scalar angle $\theta_{\ell,ab}$ acts on the d_h information channels associated with the pair.

A.2.3 ALTERNATING PAIRING

The formal construction uses an even number of heads, and in all formal experiments, $H = 8$. With one-based head indices, the two pairings are

$$\mathcal{M}_A = \{(1,2), (3,4), \dots, (H-1, H)\}, \qquad \mathcal{M}_B = \{(2,3), (4,5), \dots, (H, 1)\}.$$

They alternate across layers:

$$\mathcal{M}_A \longrightarrow \mathcal{M}_B \longrightarrow \mathcal{M}_A \longrightarrow \cdots.$$

For $H = 8$,

$$\mathcal{M}_A = \{(1,2), (3,4), (5,6), (7,8)\}, \qquad \mathcal{M}_B = \{(2,3), (4,5), (6,7), (8,1)\}.$$

There are $H/2 = 4$ learnable angles per layer. They are stored in FP32 and initialized to zero.

A.2.4 MULTI-HEAD TRANSPORT

After the role-specific flow is formed for each head,

$$Y^{(h)} = F^{(h)}\widetilde{I}^{(h)}.$$

The head outputs are concatenated and projected:

$$Y = \mathrm{Concat}\left(Y^{(1)}, \ldots, Y^{(H)}\right), \qquad X_{\ell+1} = X_\ell + YW_O.$$

Thus the Relation construction remains head-local, Givens rotations act on the information branch, and W_O combines the head outputs.

A.3 FLASHRELATION

A.3.1 HISTORICAL EXCHANGE INVARIANT

FlashRelation evaluates the exact factorization of Appendix A.1 without materializing the full $T \times T$ matrices. For a fixed query row i, let $\mathcal{J} \subseteq \{1, \ldots, i-1\}$ be the historical indices already scanned. The historical Exchange state is defined by the invariant

$$m = \max_{j \in \mathcal{J}} E_{ij}, \qquad l = \sum_{j \in \mathcal{J}} e^{E_{ij} - m}, \qquad z = \sum_{j \in \mathcal{J}} e^{E_{ij} - m}\widetilde{I}_j.$$

This scan contains Exchange entries only and Self is not included in (m, l, z). For the empty set,

$$m = -\infty, \qquad l = 0, \qquad z = 0.$$

A.3.2 ADDING ONE TILE

Let $\mathcal{B}$ be the next causal tile and set

$$m_{\mathcal{B}} = \max_{j \in \mathcal{B}} E_{ij}, \qquad m' = \max(m, m_{\mathcal{B}}).$$

The previously scanned terms are rescaled into the new maximum:

$$\begin{aligned}\sum_{j \in \mathcal{J}} e^{E_{ij} - m'} &= e^{m - m'} \sum_{j \in \mathcal{J}} e^{E_{ij} - m} \\ &= e^{m - m'} l.\end{aligned}$$

Consequently, the updated normalizer is

$$\boxed{l' = e^{m - m'} l + \sum_{j \in \mathcal{B}} e^{E_{ij} - m'}.}$$

The same rescaling gives the information accumulator

$$\boxed{z' = e^{m - m'} z + \sum_{j \in \mathcal{B}} e^{E_{ij} - m'}\widetilde{I}_j.}$$

Together with $m' = \max(m, m_{\mathcal{B}})$, these updates preserve the invariant with $\mathcal{J} \leftarrow \mathcal{J} \cup \mathcal{B}$.

A.3.3 TERMINAL ACCUMULATORS

After all historical positions $j < i$ have been scanned,

$$m_i = \max_{j < i} E_{ij}, \qquad l_i = \sum_{j < i} e^{E_{ij} - m_i}, \qquad z_i = \sum_{j < i} e^{E_{ij} - m_i}\widetilde{I}_j.$$

The terminal log-normalizer is

$$\begin{aligned} m_i + \log l_i &= m_i + \log \sum_{j<i} e^{E_{ij} - m_i} \\ &= \log \left(e^{m_i} \sum_{j<i} e^{E_{ij} - m_i} \right) \\ &= \log \sum_{j<i} e^{E_{ij}} \\ &= \boxed{L_i^E}. \end{aligned}$$

A.3.4 Historical information summary

The terminal accumulator gives

$$\begin{aligned} \frac{z_i}{l_i} &= \frac{\sum_{j<i} e^{E_{ij} - m_i} \widetilde{I}_j}{\sum_{k<i} e^{E_{ik} - m_i}} \\ &= \frac{\sum_{j<i} e^{E_{ij}} \widetilde{I}_j}{\sum_{k<i} e^{E_{ik}}} \\ &= \sum_{j<i} \pi_{ij}^E \widetilde{I}_j. \end{aligned}$$

The common factor e^{-m_i} cancels. Therefore

$$\boxed{\frac{z_i}{l_i} = \bar{I}_i^E.}$$

A.3.5 Recovering the Full Relation row

The row-level count correction is applied after the historical scan:

$$A_i = m_i + \log l_i - \lambda_\ell \log i, \qquad g_i = \sigma(A_i - S_i).$$

The final transport is

$$Y_i = (1 - g_i)\widetilde{I}_i + g_i \frac{z_i}{l_i}.$$

Using the identities in Appendix A.1,

$$\boxed{Y_i^{\text{FlashRelation}} = Y_i^{\text{FullRelation}}.}$$

So FlashRelation changes the execution of the exact operator.

A.3.6 Materialization and boundary

FlashRelation does not materialize full $T \times T$ U, R, or F matrices. The required pairwise evidence and Exchange entries are formed and consumed tile by tile. The historical state is maintained as running statistics.

A.4 Linear Relation

A.4.1 Relation coordinates and retention

In the equations below, $p_{1,t}$ and $p_{2,t}$ denote the projected relation coordinates after the formal RoPE step. The positional order is

$$P_1/P_2 \longrightarrow \mathrm{RoPE} \longrightarrow S_t \longrightarrow \mathrm{L2Norm} \longrightarrow \text{recurrence}.$$

Thus

$$S_t = \sigma\left(\frac{p_{1,t}^\top p_{2,t}}{\tau_S\sqrt{d_h}}\right), \qquad \hat{p}_{1,t} = \frac{p_{1,t}}{\|p_{1,t}\|_2}, \qquad \hat{p}_{2,t} = \frac{p_{2,t}}{\|p_{2,t}\|_2}.$$

The retention parameterization is

$$r_t = W_\alpha^\uparrow W_\alpha^\downarrow \bar{x}_t, \qquad g_t = -\exp(A_{\log}) \odot \mathrm{softplus}(r_t + dt_{\mathrm{bias}}), \qquad \alpha_t = \exp(g_t).$$

Because $\exp(A_{\log}) > 0$ elementwise and $\mathrm{softplus}(\cdot) > 0$,

$$g_t < 0, \qquad \boxed{0 < \alpha_{t,c} \leq 1}.$$

Here S_t is the Self Relation, whereas α_t is channel-wise historical retention.

A.4.2 Recurrent step

Let

$$D_t = \mathrm{Diag}(\alpha_t), \qquad \boxed{C_0 = 0}.$$

The recurrent step is

$$\begin{aligned} C_t^- &= D_t C_{t-1}, \\ E_t &= S_t (C_t^-)^\top \hat{p}_{1,t}, \\ Y_t &= \widetilde{I}_t + E_t, \\ C_t &= C_t^- + S_t \hat{p}_{2,t} \widetilde{I}_t^\top. \end{aligned}$$

The execution order is

$$\boxed{\text{Ask Self} \longrightarrow \text{Ask Others} \longrightarrow \text{Answer} \longrightarrow \text{Become Others.}}$$

A.4.3 Strict historical support

The read uses $C_t^- = D_t C_{t-1}$, while the current write

$$S_t \hat{p}_{2,t} \widetilde{I}_t^\top$$

is added only after E_t has been formed. Hence E_t depends only on positions $1, \ldots, t-1$, and the current token cannot enter its own historical read through the current write.

A.4.4 Unrolling the recurrent state

Starting from

$$C_t = D_t C_{t-1} + S_t \hat{p}_{2,t} \widetilde{I}_t^\top,$$

one expansion gives

$$\begin{aligned} C_t &= D_t \left(D_{t-1} C_{t-2} + S_{t-1} \hat{p}_{2,t-1} \widetilde{I}_{t-1}^\top \right) + S_t \hat{p}_{2,t} \widetilde{I}_t^\top \\ &= D_t D_{t-1} C_{t-2} + D_t S_{t-1} \hat{p}_{2,t-1} \widetilde{I}_{t-1}^\top + S_t \hat{p}_{2,t} \widetilde{I}_t^\top. \end{aligned}$$

With the ordered product

$$\prod_{u=j+1}^{t} D_u \;\equiv\; D_t D_{t-1} \cdots D_{j+1},$$

and the empty product equal to the identity,

$$\boxed{C_t = \sum_{j \leq t} \left(\prod_{u=j+1}^{t} D_u \right) S_j \hat{p}_{2,j} \widetilde{I}_j^\top.}$$

Thus each token contributes $S_j \hat{p}_{2,j} \widetilde{I}_j^\top$ and is subsequently transformed by $D_{j+1}, \ldots, D_t$.

A.4.5 EXPANDED HISTORICAL EXCHANGE

The pre-write state is

$$C_t^- = \sum_{j<t} \left(\prod_{u=j+1}^{t} D_u \right) S_j \hat{p}_{2,j} \widetilde{I}_j^\top .$$

Substituting this into the read gives

$$\begin{aligned} E_t &= S_t (C_t^-)^\top \hat{p}_{1,t} \\ &= \boxed{\sum_{j<t} S_t S_j \left[\hat{p}_{2,j}^\top \left(\prod_{u=j+1}^{t} D_u \right) \hat{p}_{1,t} \right] \widetilde{I}_j} . \end{aligned}$$

Because each D_u is diagonal, the transpose leaves the ordered product unchanged.

A.4.6 INFORMATION WRITE AND COMPLEXITY

The output is

$$Y_t = \widetilde{I}_t + E_t,$$

but the state writes

$$S_t \hat{p}_{2,t} \widetilde{I}_t^\top ,$$

not $S_t \hat{p}_{2,t} Y_t^\top$. Each new state contribution therefore corresponds to the current token's own information.

For one head, $C_t^{(h)} \in \mathbb{R}^{d_h \times d_h}$. The principal recurrent operations per token are

$$D_t C_{t-1}, \qquad (C_t^-)^\top \hat{p}_{1,t}, \qquad \hat{p}_{2,t} \widetilde{I}_t^\top ,$$

with leading cost $O(d_h^2)$ per token per head. Across H heads,

$$O(THd_h^2) = O\left(TH \left(\frac{d}{H} \right)^2 \right) = \boxed{O\left(\frac{Td^2}{H} \right)} .$$

The recurrent state size is

$$Hd_h^2 = H \left(\frac{d}{H} \right)^2 = \frac{d^2}{H}, \qquad \boxed{O(d^2/H)}$$

for the decode state. These complexities refer to the recurrent Relation token-mixing core and exclude common FFN and embedding costs.

B EXPERIMENTAL SETUP AND EVALUATION

B.1 MODEL CONFIGURATIONS

All models use full-head RoPE with base 10^4. The three main scales use paired seeds $42, 43, 44$.

Table 6: Formal decoder-only model configurations. Global tokens per optimizer update are 131,072 for all three scales.

Scale	Params MHA/Rel.	L/d/H × d_h/d_{ff}	Vocab	Data	T	Train	Micro/GA
10M	10,425,216/10,425,246	6/384/8×48/768	4K	TinyStories	1024	150M	32/4
30M	28,322,304/28,322,354	10/512/8×64/1024	4K	TinyStories	2048	450M	16/4
100M	102,917,760/102,917,860	20/640/8×80/1280	32K	SmolLM corpus	4096	1.071B	MHA 2/16; Rel. 4/8

B.2 OPTIMIZATION AND CHECKPOINTS

Training uses BF16 and AdamW with betas $(0.9, 0.95)$, weight decay 0.1, gradient clipping at 1.0, and a WSD80 schedule. Learning rates are 10^{-3}, 8×10^{-4}, and 6×10^{-4} for 10M, 30M, and 100M. Warmup tokens are 1.5M, 4.5M, and 10.7114M. Formal NLL is evaluated on the full validation set at the final checkpoint.

B.3 Position Encoding

RoPE covers the full head dimension. The base is 10^4. For MHA, RoPE is applied to Q and K after projection and head splitting. For Full Relation and FlashRelation, RoPE is applied to P_1 and P_2 before pairwise evidence. The information projection and Givens output remain unrotated. Linear Relation applies RoPE to its two relation coordinates before Self and normalized coordinate formation. Hybrid layers use the corresponding Full or Linear path.

B.4 Evaluation Protocol

The primary metric is final-checkpoint NLL. Systems measurements report complete optimizer-step throughput. One-time initialization and JIT, warmup, and terminal partial steps are excluded when present.

For matched-train-loss token efficiency, training loss is smoothed with a trailing token window equal to 1% of the formal training budget. For each paired seed, the target is the MHA run's final smoothed training loss. We linearly interpolate the earliest Relation crossing between adjacent logged points and report the token reduction as $1 - N_{\mathrm{Rel}}/N_{\mathrm{budget}}$.

Structural diagnostic definitions. MHA→Rel. reports matched MHA and Relation means. *diag. / Self flow* compares the MHA diagonal normalized-flow entry with the Relation Self-flow entry; the MHA diagonal is not an explicit Self relation, whereas the Relation entry is induced by the explicit Self relation. L2+ Self / Exchange is reported for Relation only and averages normalized Self and Exchange flow over all layers from L2 onward. Historical-distance and first-token-sink statistics are conditioned on historical flow, with first-token sink denoting the conditional mass assigned to the first sequence token. Flow statistics use the same four fixed sequences of length 1024 across matched checkpoints; flow effective rank uses the same four fixed sequences of length 512.

For a per-head normalized flow matrix M, with $M = A$ for MHA and $M = F$ for Relation, let $\sigma_k(M)$ denote singular values computed in float32. The flow effective rank is

$$\mathrm{EffRank}(M) = \exp\left(-\sum_k \widetilde{p}_k \log \widetilde{p}_k\right),$$
$$p_k = \frac{\sigma_k(M)}{\max(\sum_r \sigma_r(M), 10^{-20})}, \qquad \widetilde{p}_k = \max(p_k, 10^{-20}).$$

For each layer and head, the metric pools the four fixed length-512 rank samples. Within each checkpoint, heads are count-weighted at each layer, and scale-level summaries count-weight layers. We report the mean and sample standard deviation across the three seeds.

The remaining flow-spectrum quantities use squared singular values. With $\sigma_1(M) \geq \sigma_2(M) \geq \cdots$ from the float32 SVD, define

$$E(M) \equiv \max\left(\sum_k \sigma_k^2(M), 10^{-20}\right), \qquad \mathrm{StableRank}(M) \equiv \frac{E(M)}{\max(\sigma_1^2(M), 10^{-20})}.$$

The energy dimensions and top singular energy are

$$r_\eta(M) \equiv \min\left\{r : \frac{\sum_{k=1}^r \sigma_k^2(M)}{E(M)} \geq \eta\right\}, \qquad \eta \in \{0.90, 0.95\},$$
$$\mathrm{TopSingularEnergy}(M) \equiv \frac{\sigma_1^2(M)}{E(M)}.$$

The 90% and 95% dimensions are $r_{0.90}(M)$ and $r_{0.95}(M)$, respectively. These spectrum statistics use the same per-head flow matrices, fixed rank samples, sample pooling, head aggregation, checkpoint/layer aggregation, and seed aggregation as Effective Rank.

For conditional sink metrics, we sum π_{ij}^E over zero-indexed source positions $j < k$ for each query row. Rows without historical Exchange mass, including the first token, contribute zero to the reported mean.

B.5 Structural Diagnostic Results

The formal structural diagnostics cover all 18 final checkpoints: three scales, three seeds, and matched MHA/Relation pairs. Flow, distance, and sink statistics use the same four fixed sequences of length 1024 for every checkpoint, while spectrum statistics use the same four fixed sequences of length 512. For MHA, the normalized flow is reconstructed from the checkpoint's actual Q and K projections with RoPE, causal masking, and Softmax; Relation uses its actual normalized flow F.

For Relation, the aggregate normalized Self/Exchange flow from L2 onward is $2.87\%/97.13\%$ at 10M, $1.73\%/98.27\%$ at 30M, and $1.01\%/98.99\%$ at 100M (three-seed means).

Table 7: Layerwise normalized-flow organization in the formal structural diagnostics. MHA diag. is the diagonal normalized-flow entry. Rel. Self is Relation's explicit Self flow. EffRank is the per-head normalized-flow spectrum metric defined in Appendix B.4. Values are three-seed means.

Scale	Layer	MHA diag.	Rel. Self	MHA EffRank	Rel. EffRank	Rel./MHA
10M	L0	6.62%	28.57%	173.3	405.4	2.34
10M	L1	7.30%	1.49%	244.2	216.3	0.89
10M	L2	4.29%	1.87%	192.2	224.5	1.17
10M	L3	5.36%	3.10%	189.0	211.1	1.12
10M	L4	5.76%	3.39%	173.1	211.2	1.22
10M	L5	6.59%	3.13%	203.7	201.2	0.99
30M	L0	5.54%	23.80%	155.4	406.9	2.62
30M	L1	6.53%	0.37%	242.6	240.7	0.99
30M	L2	4.83%	1.34%	226.3	200.2	0.88
30M	L3	4.43%	1.22%	197.3	200.6	1.02
30M	L4	4.14%	1.27%	181.7	180.8	1.00
30M	L5	4.11%	1.69%	179.2	181.5	1.01
30M	L6	3.85%	2.19%	169.8	201.1	1.18
30M	L7	4.98%	1.65%	186.3	182.5	0.98
30M	L8	5.04%	2.34%	184.3	190.4	1.03
30M	L9	5.21%	2.17%	189.7	218.6	1.15
100M	L0	13.94%	43.81%	176.6	470.8	2.67
100M	L1	1.86%	0.39%	129.7	185.0	1.43
100M	L2	3.23%	1.19%	159.9	202.8	1.27
100M	L3	3.01%	1.44%	166.2	192.4	1.16
100M	L4	3.76%	1.36%	174.3	187.0	1.07
100M	L5	4.27%	1.65%	171.9	198.7	1.16
100M	L6	3.29%	2.03%	175.0	195.8	1.12
100M	L7	3.08%	1.48%	136.6	179.7	1.32
100M	L8	2.56%	1.12%	140.0	168.0	1.20
100M	L9	2.84%	2.13%	147.8	159.7	1.08
100M	L10	3.52%	1.22%	152.1	155.6	1.02
100M	L11	2.54%	1.02%	141.7	145.6	1.03
100M	L12	2.97%	0.39%	124.6	152.4	1.22
100M	L13	3.43%	0.23%	164.3	146.5	0.89
100M	L14	3.39%	0.33%	157.0	158.4	1.01
100M	L15	3.80%	0.17%	145.3	142.9	0.98
100M	L16	4.37%	0.27%	165.3	141.3	0.85
100M	L17	4.97%	0.69%	141.2	154.8	1.10
100M	L18	7.41%	0.62%	172.1	152.7	0.89
100M	L19	6.06%	0.82%	153.4	148.4	0.97

Table 8: Scale-level conditional historical routing in the formal structural diagnostics. Each entry is shown as MHA $\rightarrow$ Relation; distance mass and first-token sink are conditioned on historical flow. Values are three-seed means.

Scale	$d > 128$	$d > 256$	$d > 512$	First-token sink
10M	$18.12\% \rightarrow 25.21\%$	$10.43\% \rightarrow 15.63\%$	$4.17\% \rightarrow 6.09\%$	$0.566\% \rightarrow 0.483\%$
30M	$18.17\% \rightarrow 23.94\%$	$9.40\% \rightarrow 14.00\%$	$3.18\% \rightarrow 5.23\%$	$0.772\% \rightarrow 0.579\%$
100M	$37.99\% \rightarrow 39.58\%$	$22.59\% \rightarrow 23.91\%$	$8.03\% \rightarrow 8.51\%$	$10.17\% \rightarrow 8.43\%$

Table 9: Conditional early-token sink profile in the formal structural diagnostics. All sink metrics are conditioned on historical flow. *First* k denotes the conditional historical-flow mass assigned to the first k sequence positions. Arrows denote MHA $\rightarrow$ Relation; values are three-seed means.

(a) First 1 / 2 / 4

Scale	First 1	First 2	First 4
10M	$0.566\% \rightarrow 0.483\%$	$0.990\% \rightarrow 0.908\%$	$1.552\% \rightarrow 1.534\%$
30M	$0.772\% \rightarrow 0.579\%$	$1.217\% \rightarrow 1.055\%$	$1.763\% \rightarrow 1.688\%$
100M	$10.173\% \rightarrow 8.430\%$	$10.427\% \rightarrow 8.730\%$	$10.917\% \rightarrow 9.257\%$

(b) First 8 / 16 / 32

Scale	First 8	First 16	First 32
10M	$2.563\% \rightarrow 2.682\%$	$4.315\% \rightarrow 4.721\%$	$7.157\% \rightarrow 8.037\%$
30M	$2.855\% \rightarrow 2.885\%$	$4.616\% \rightarrow 4.872\%$	$7.325\% \rightarrow 8.077\%$
100M	$11.839\% \rightarrow 10.270\%$	$13.576\% \rightarrow 12.209\%$	$17.636\% \rightarrow 16.506\%$

The full sink profile shows that Relation's lower first-token sink does not imply uniformly lower mass over the entire early context; instead, early historical mass is redistributed across a broader prefix.

Table 10: Scale-level aggregate flow-spectrum statistics in the formal structural diagnostics. Each entry is shown as MHA $\rightarrow$ Relation; values are computed from the normalized flow spectra and are three-seed means.

Scale	Stable rank	Effective rank	90% dims	95% dims	Top singular energy
10M	$23.447 \rightarrow 24.160$	$195.924 \rightarrow 244.945$	$77.07 \rightarrow 115.67$	$101.70 \rightarrow 147.85$	$0.07446 \rightarrow 0.06399$
30M	$18.982 \rightarrow 19.879$	$191.268 \rightarrow 220.353$	$70.64 \rightarrow 93.32$	$95.52 \rightarrow 123.35$	$0.08551 \rightarrow 0.08239$
100M	$8.981 \rightarrow 11.055$	$154.758 \rightarrow 181.916$	$49.83 \rightarrow 66.55$	$69.38 \rightarrow 89.11$	$0.41186 \rightarrow 0.37475$

Taken together, the diagnostics show a consistent depth pattern. Relation uses L0 as a Self-emphasizing, high-rank routing layer. Self flow then drops sharply, and later layers become Exchange-dominant. Historical routing also shifts farther back in the sequence, especially at 10M and 30M, while the first-token sink is lower. These results show that Relation and MHA organize normalized flow differently across depth and history.

C Per-Seed Language-Model Results

C.1 Final Validation NLL

Table 11 reports final validation NLL for all paired seeds. Aggregate rows use the sample standard deviation over three seeds.

Table 11: Final validation NLL for the three paired seeds.

Scale	Seed	MHA NLL	Full Relation NLL	Full–MHA
10M	42	1.688823	1.631738	-0.057085
10M	43	1.686405	1.643978	-0.042427
10M	44	1.680626	1.656633	-0.023993
10M	Mean $\pm$ SD	1.685285 ± 0.004212	1.644117 ± 0.012448	-0.041168
30M	42	1.295263	1.300730	+0.005467
30M	43	1.303885	1.277177	-0.026708
30M	44	1.301160	1.277225	-0.023935
30M	Mean $\pm$ SD	1.300103 ± 0.004407	1.285044 ± 0.013584	-0.015059
100M	42	2.945050	2.912234	-0.032816
100M	43	2.939826	2.899965	-0.039861
100M	44	2.926899	2.906657	-0.020242
100M	Mean $\pm$ SD	2.937258 ± 0.009344	2.906285 ± 0.006143	-0.030973

Full Relation wins all three paired comparisons at 10M, two of three at 30M, and all three at 100M, for eight wins out of nine paired comparisons.

C.2 Matched-Train-Loss Token Efficiency

Table 12: Per-seed matched-train-loss token efficiency. The MHA target is the paired run's final smoothed training loss.

Scale	Seed	MHA target	Relation crossing tokens	Token reduction
10M	42	1.645060	143.62M	4.25%
10M	43	1.696446	133.67M	10.88%
10M	44	1.680547	139.89M	6.74%
30M	42	1.302310	419.67M	6.74%
30M	43	1.256313	448.42M	0.35%
30M	44	1.303514	405.50M	9.89%
100M	42	2.914568	1.00168B	6.48%
100M	43	2.903508	1.00316B	6.35%
100M	44	2.870894	1.06466B	0.60%

Across the three paired seeds, the mean token reductions are $7.3 \pm 3.4\%$, $5.7 \pm 4.9\%$, and $4.5 \pm 3.4\%$ at 10M, 30M, and 100M, respectively.

D Structural Ablation Details

D.1 Ablation Definitions

All structural ablations use the 10M setting, the same prepared data, and seeds 42, 43, and 44. Each ablation changes only the named component of Full Relation.

- Exchange-only transport computes the complete Full Relation flow and transports only historical entries: $Y_i = \sum_{j<i} F_{ij} \widetilde{I}_j$. Historical weights are not renormalized.
- Raw-X communication replaces the independent information projection by $I = X$. Head splitting, Givens mixing, Flow, and W_O remain unchanged.
- No count calibration sets $\lambda_\ell = 0$.
- No Givens mixing bypasses adjacent-head rotation, so $\widetilde{I} = I$, while the Relation and transport path remains unchanged.
- Single head uses $H = 1$.

D.2 Per-Seed Ablation Results

Table 13: Per-seed final validation NLL for the structural ablations.

Ablation	Seed 42	Seed 43	Seed 44	Mean $\pm$ SD	Δ vs Full
Full Relation	1.631738	1.643978	1.656633	1.644117 $\pm$ 0.012448	0.000000
Exchange-only transport	1.665658	1.677815	1.684822	1.676099 $\pm$ 0.009697	+0.031982
Raw-X communication	1.672332	1.686600	1.683090	1.680674 $\pm$ 0.007434	+0.036557
No count calibration	1.671373	1.704531	1.708256	1.694720 $\pm$ 0.020305	+0.050604
No Givens mixing	1.630321	1.646350	1.665386	1.647352 $\pm$ 0.017554	+0.003236
Single head	1.669349	1.707688	1.707422	1.694820 $\pm$ 0.022059	+0.050703

E Hybrid Relation

E.1 Configuration

Hybrid Relation uses a 12-layer, 480-dimensional decoder with eight heads of width 60 and $d_{\text{ff}} = 960$. Its layout is $(LLLF)^3$, with nine Linear Relation layers and three Full Relation layers. The vocabulary has 4,096 entries, context length is 2,048, and training budget is 450M tokens. The model has 31,967,571 parameters and uses micro/GA $16/4$, BF16, AdamW, and 131,072 global tokens per update. This is a 30M-class configuration and is not geometry-identical to the 30M Full Relation model.

E.2 Per-Seed Results

Table 14: Per-seed Hybrid Relation results.

Seed	Final NLL
42	1.274777
43	1.283791
44	1.275435
Mean ± SD	1.278001 ± 0.005025

F Systems Benchmark Protocol

F.1 Hardware and Software

Systems measurements use an NVIDIA RTX 5090 GPU in BF16. The recorded software environment is CUDA 13.0, PyTorch 2.12.1+cu130, and Triton 3.7.1. Each campaign uses its formal prepared data and frozen model geometry.

F.2 Timing and Accounting

One-time initialization and JIT compilation are outside the reported interval. Warmup optimizer steps are excluded. Throughput uses complete optimizer steps, with the terminal partial token step excluded where the campaign reaches a non-integral final update. The denominator is the number of tokens consumed by the complete measured optimizer steps.

F.3 Campaign Settings

The fixed-context Reference–FlashRelation campaign uses $T = 1024$ for all scales. The scale-matched FlashRelation–FlashAttention campaign uses $T = 1024, 2048, 4096$ for 10M, 30M, and 100M. Linear long-context scaling uses micro $= 1$, 131,072 effective tokens per update, five warmup steps, and ten measured complete steps.

G FlashRelation Systems Results

G.1 Scale-Matched FlashRelation and FlashAttention

FlashAttention is explicitly forced in these scale-matched production records. FlashRelation uses the final exact production dispatch for each head dimension.

Table 15: Scale-matched production FlashAttention and FlashRelation throughput; memory is allocated/reserved GiB.

Scale	T	FlashAttention tok/s	FlashRelation tok/s	FR/FA	FR micro/GA	FA micro/GA	FR alloc./res.	FA alloc./res.
10M	1024	841,833.930	755,088.370	0.897000×	32/4	32/4	5.24/5.70	5.38/5.88
30M	2048	361,793.115	334,028.010	0.923300×	16/4	16/4	9.68/10.04	10.03/10.45
100M	4096	107,783.242	100,166.770	0.929300×	4/8	4/8	17.19/18.21	17.71/18.63

H Linear Relation Systems and Long-Context Scaling

H.1 Steady-State Throughput

Table 16: Frozen Linear Relation steady-state throughput.

Scale	T	Micro/GA	Steady-state tok/s	Raw mean tok/s
10M	1024	64/2	517,602	282,578
30M	1024	32/4	256,034	199,875
100M	1024	16/8	90,592	87,865

H.2 Long-Context Scaling

Long-context runs use the production K_{L60} schedule with chunk 128, block 64, four warps, one stage, and row tile 1. Micro is fixed to 1 and the effective update size is 131,072 tokens. The $T = 65536$ point is a real OOM.

Table 17: Linear Relation long-context scaling.

T	Micro	GA	Status	tok/s	Allocated GiB
1024	1	128	PASS	10,129.47	1.140
2048	1	64	PASS	20,301.39	1.640
4096	1	32	PASS	39,990.24	2.648
8192	1	16	PASS	59,367.95	4.819
16384	1	8	PASS	52,894.96	8.827
32768	1	4	PASS	30,003.56	16.815
65536	1	2	OOM	–	–