FUSION MATTERS: LENGTH-AWARE ANALYSIS OF POSITIONAL-ENCODING FUSION IN TRANSFORMERS

Mohamed Amine Hallam
School of Computer Science and Technology
Harbin Institute of Technology
25sf51027@stu.hit.edu.cn

Kuo-Kun Tseng
School of Computer Science and Technology
Harbin Institute of Technology
kktseng@hit.edu.cn

January 6, 2026

ABSTRACT

Transformers require positional encodings to represent sequence order, yet most prior work focuses on designing new positional encodings rather than examining how positional information is fused with token embeddings. In this paper, we study whether the fusion mechanism itself affects performance, particularly in long-sequence settings. We conduct a controlled empirical study comparing three canonical fusion strategies—element-wise addition, concatenation with projection, and scalar gated fusion—under identical Transformer architectures, data splits, and random seeds. Experiments on three text classification datasets spanning short (AG News), medium (IMDB), and long (ArXiv) sequences show that fusion choice has negligible impact on short texts but produces consistent gains on long documents. To verify that these gains are structural rather than stochastic, we perform paired-seed analysis and cross-dataset comparison across short, medium, and long sequence regimes. Additional experiments on the ArXiv dataset indicate that the benefit of learnable fusion generalizes across multiple positional encoding families. Finally, we explore a lightweight convolutional gating mechanism that introduces local inductive bias at the fusion level, evaluated on long documents only. Our results indicate that positional-encoding fusion is a non-trivial design choice for long-sequence Transformers and should be treated as an explicit modeling decision rather than a fixed default.

1 Introduction

Transformers have become the dominant architecture for sequence modeling across natural language processing tasks, including text classification, machine translation, and long-document understanding. Because self-attention is inherently permutation-invariant, positional information must be injected explicitly in order to represent token order. As a result, positional encodings have been a core component of Transformer architectures since their introduction.[1]

A substantial body of prior work has focused on *what* positional information should be encoded. This includes sinusoidal encodings, learned absolute embeddings[1], relative position representations[2], rotary embeddings[3], and bias-based formulations[4]. These approaches differ in how they represent order, distance, or relative position, and many have been proposed specifically to address limitations of Transformers on long sequences.

However, an orthogonal and largely overlooked design choice concerns *how* positional encodings are fused with token embeddings. In most standard implementations, positional encodings are injected via simple element-wise addition. This practice implicitly assumes that positional information should contribute uniformly across tokens and layers, and that its influence should be fixed rather than learned. While this assumption has become a de-facto default, it is rarely justified explicitly and is seldom questioned in empirical studies.

This omission is notable given recent trends in long-context modeling. Modern benchmarks increasingly involve long documents, such as scientific articles, legal texts, and multi-paragraph narratives, where positional relevance may vary substantially across the sequence. In such settings, treating positional information as a uniform additive signal may be unnecessarily restrictive. Yet, despite extensive work on alternative positional encoding *representations*, the fusion mechanism itself is typically held fixed.[5, 6]

In this work, we isolate the positional-encoding fusion operator as an independent modeling variable. Rather than proposing a new positional encoding, we ask whether the *method used to combine* token embeddings and positional encodings materially affects Transformer performance, particularly as sequence length increases. Our focus is intentionally narrow: we hold the Transformer architecture, optimization procedure, data splits, and random seeds constant, and vary only the fusion mechanism.

We investigate three core questions:

1. **Does the choice of positional-encoding fusion operator affect downstream performance when all other factors are controlled?**
2. **Is any observed effect dependent on sequence length, or does it persist across short, medium, and long documents?**
3. **Can inductive bias be introduced at the fusion level—without modifying attention or the Transformer encoder—to better handle long sequences?**

To answer these questions, we conduct a controlled empirical study across three text classification datasets spanning short (AG News), medium (IMDB), and long (ArXiv) sequence regimes. We compare element-wise addition, concatenation with projection, and learnable gating under identical experimental conditions. To rule out stochastic explanations, we perform paired-seed analyses in which fusion methods are compared using identical random seeds and data splits.

Beyond global gating, we further explore a lightweight convolutional gating mechanism applied solely to positional encodings. This design introduces local inductive bias at the fusion stage while leaving the Transformer core unchanged. The goal is not to outperform specialized long-sequence architectures, but to assess whether modest structural bias at the fusion level can yield consistent gains on long documents.

Our results show that positional-encoding fusion is not a neutral design choice. While fusion differences are negligible on short and medium-length datasets—where performance saturates—they produce consistent and statistically stable improvements on long documents. These findings suggest that positional-encoding fusion should be treated as an explicit modeling decision rather than a fixed default.

Contributions.

This paper makes the following contributions:

- We provide the first controlled, cross-dataset study isolating positional-encoding fusion as a modeling variable in Transformers.
- We show that learnable fusion mechanisms yield consistent improvements on long-sequence classification, while offering little benefit on short or medium-length texts.
- We introduce and evaluate a lightweight convolutional gating mechanism that injects local inductive bias at the fusion level without modifying attention.
- We demonstrate robustness of these findings through paired-seed analysis and evaluation across multiple positional encoding families.

2 Related Work

The original Transformer introduces sinusoidal positional encodings that are combined with token embeddings through element-wise addition [1]. Subsequent work explored alternative representations, including learned absolute embeddings, relative position representations [2, 7, 8], rotary embeddings [3], and bias-based approaches such as ALiBi [4]. Variants of relative positional bias have also been adopted in large-scale pretrained models, such as the T5 architecture [9]. Several studies analyze the limitations and extrapolation behavior of existing positional encodings [10, 11], highlighting that the choice of positional representation can significantly affect long-context modeling.

A separate line of research addresses the challenge of modeling long sequences by modifying attention patterns or improving computational efficiency. Sparse-attention architectures such as Longformer [6] and BigBird [12] reduce the quadratic complexity of self-attention, while approximate or low-rank methods such as Reformer [13], Performer [14], and Linformer [15] enable linear-time attention. Hierarchical encoders such as ETC further target structured long inputs [16]. While effective, these approaches typically introduce architectural changes and retain the standard additive fusion of token embeddings and positional encodings.

Gating and modulation mechanisms have been explored within Transformer architectures to control information flow in attention and feed-forward layers. Examples include gated linear units [17], stabilization techniques for training Transformers [18], and feature-wise modulation mechanisms such as FiLM [19]. Related ideas have also been applied to

positional signals within attention [8]. However, these mechanisms are generally applied to intermediate representations rather than to the fusion of token embeddings and positional encodings at the model input.
Across most Transformer variants, the fusion of token embeddings and positional encodings is inherited from the original architecture and implemented as a fixed additive operation [1]. Even when alternative positional representations or attention mechanisms are introduced, the fusion operator itself is rarely varied or analyzed in isolation. In contrast, our work focuses on this under-explored design dimension by systematically evaluating different fusion operators while holding architecture, optimization, and data splits fixed across sequence-length regimes.

3 Method

3.1 Base Model and Positional Encoding

All experiments use a fixed encoder-only Transformer architecture. To avoid confounding factors, the architecture, optimization procedure, and training schedule are identical across fusion variants.

We use **sinusoidal positional encodings** throughout the main experiments. This choice enables direct comparison across datasets of different lengths and avoids introducing additional variability from alternative encoding schemes.

3.2 Positional-Encoding Fusion Operators

Let $\boldsymbol{E} \in \mathbb{R}^{L\times d}$ denote token embeddings and $\boldsymbol{P} \in \mathbb{R}^{L\times d}$ positional encodings, where L is sequence length and d is model dimension.

3.2.1 Additive fusion (Add)

The standard approach combines token embeddings and positional encodings via element-wise addition:

$$\boldsymbol{H} = \boldsymbol{E} + \boldsymbol{P}. \tag{1}$$

This method does not introduce additional parameters and assumes a fixed, uniform contribution of positional information across tokens.

3.2.2 Concatenation with projection (Concat).

Token embeddings and positional encodings are concatenated and projected back to the model dimension:

$$\boldsymbol{H} = \boldsymbol{W}\,[\boldsymbol{E};\boldsymbol{P}], \qquad \boldsymbol{W} \in \mathbb{R}^{d\times 2d}, \tag{2}$$

Where [E;P] denotes concatenation along the feature dimension. This operator allows the model to learn a linear combination of content and positional features.

3.2.3 Gated Addition (Gate-Scalar).

To relax the assumption that positional information should contribute uniformly across all tokens, we introduce a *gated fusion* mechanism that learns a position-dependent trade-off between token embeddings and positional encodings.

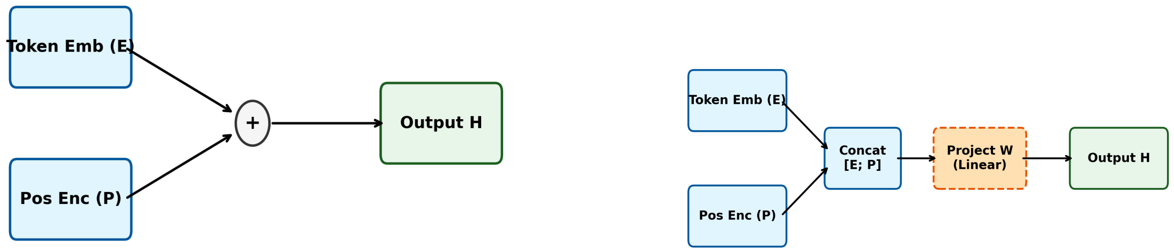
Let $\mathbf{E} \in \mathbb{R}^{L\times d}$ denote token embeddings and $\mathbf{P} \in \mathbb{R}^{L\times d}$ positional encodings. For each token position i, we compute a scalar gate based on the concatenation of token and positional representations:

$$g_i = \sigma\big(\mathbf{w}^\top [\mathbf{E}_i; \mathbf{P}_i] + b\big)\,, \tag{3}$$

We refer to this variant as *Gate-Scalar* to distinguish it from more expressive MLP-based gating mechanisms evaluated separately in Appendix A. where $[\mathbf{E}_i; \mathbf{P}_i]$ denotes feature-wise concatenation, $\mathbf{w} \in \mathbb{R}^{2d}$ and $b \in \mathbb{R}$ are learnable parameters, and $\sigma(\cdot)$ is the sigmoid function.
In all main experiments, the gate is a scalar computed from the concatenation of token and positional representations, This design introduces minimal interaction, ensuring that fusion remains a lightweight modulation rather than a full content–position transformation.

The gate is a scalar value computed independently at each position and shared across feature dimensions.

$$\mathbf{H}_i = g_i\,\mathbf{E}_i + (1 - g_i)\,\mathbf{P}_i. \tag{4}$$



(a) Additive (Add)

(b) Concat + projection (Concat)

Figure 1: Fusion operators in the evaluated experiments.

3.3 Local Convolutional Positional Gating

In addition to global gated fusion, we explore a local convolutional gating variant evaluated on long documents only. The motivation is to introduce a mild inductive bias that allows the contribution of positional information at a given token to depend on its local neighborhood, without modifying the Transformer encoder or attention mechanism. We instantiate this local gating using a depth-wise 1D convolution; other local sequence models (e.g., TCNs or shallow MLPs over local windows) are possible but not explored here.

For each token position i, we compute a scalar gate by applying a depth-wise one-dimensional convolution over the positional encodings:

$$g_i = \sigma\left(\sum_{k=-K}^{K} \mathbf{w}_k \odot \mathbf{P}_{i+k}\right), \tag{5}$$

where $\{\mathbf{w}_k\}$ are learnable depth-wise convolution kernels of size $2K+1$, $\sigma(\cdot)$ denotes the sigmoid function, and $\odot$ represents element-wise multiplication.

The fused representation is then obtained using the same convex combination as in global gated fusion:

$$\mathbf{H}_i = g_i\,\mathbf{E}_i + (1-g_i)\,\mathbf{P}_i. \tag{6}$$

Unlike global scalar gating (Gate-Scalar), which determines the gate independently at each position, this formulation allows positional relevance to be modulated based on nearby positions. Importantly, the convolution operates solely on the positional encodings and introduces no changes to the self-attention mechanism or Transformer architecture. We evaluate this variant as an exploratory design to assess whether locality-aware fusion provides additional benefits in long-sequence settings.

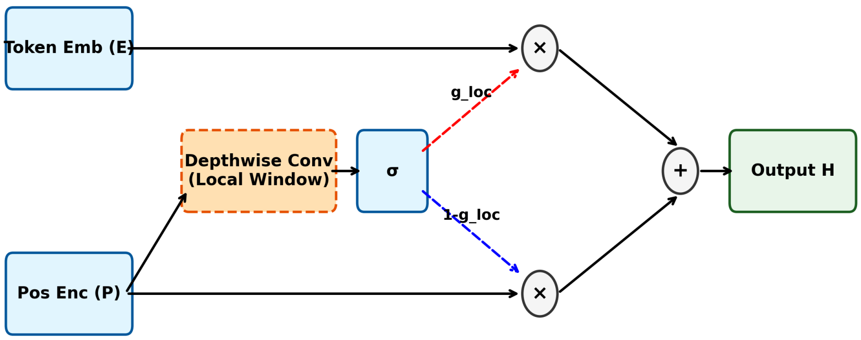


Figure 2: Local convolutional positional gating (Gate-CNN). The gate is computed from a depth-wise 1D convolution over positional encodings only, then used to mix token and positional features.

4 Experimental Setup

4.1 Dataset

We evaluate on three text classification datasets with increasing sequence length:

1. Ag News (Short sequences)
2. IMDB (medium-length sequences)
3. ArXiv (long documents, up to several thousand tokens)

We focus on classification tasks because they isolate sequence representation quality and avoid confounding factors introduced by decoding or generation.

4.2 Protocol

Across all experiments, data splits and hyperparameters are held fixed across fusion variants, and models are trained using multiple random seeds. For paired comparisons, the same random seed is used across fusion methods, ensuring identical initialization and data ordering. This experimental protocol isolates the effect of the fusion operator and ensures that observed performance differences cannot be attributed to stochastic variation. All fusion operators differ only in the embedding fusion step; the Transformer encoder and attention mechanism are unchanged. Code and reproduction materials are available online [20].

5 Fusion Benchmark Lengths

Table 1: Mean test accuracy (± standard deviation) across fusion strategies using sinusoidal positional encodings

Dataset	Add	Concat	Gate-Scalar
AG News	91.15 ± 0.08	90.93 ± 0.11	91.07 ± 0.09
IMDB	83.28 ± 0.15	83.78 ± 0.13	83.40 ± 0.14
ArXiv	59.22 ± 0.32	63.44 ± 0.28	**65.73** ± **0.30**

On AG News and IMDB, performance differences between fusion methods are small and inconsistent, indicating rapid saturation on short and medium-length sequences. In contrast, on ArXiv, Gate-Scalar consistently outperforms additive fusion, yielding a substantial absolute improvement in accuracy. A more expressive MLP-based fusion baseline is evaluated separately on the ArXiv dataset and reported in the **(Appendix A)**.

6 Robustness Across Sequence-Length Regimes

6.1 Paired-Seed Delta Analysis

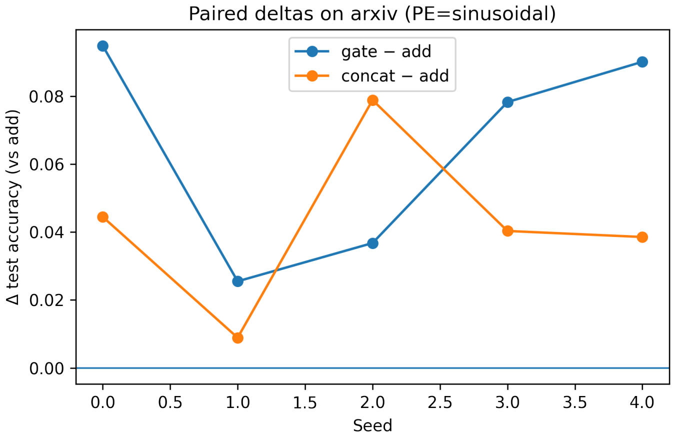


Figure 3: Paired per-seed accuracy deltas (Gate Scalar − Add) on the ArXiv dataset.

To verify that the observed improvements on ArXiv are not due to randomness, we compute paired per-seed accuracy deltas between gated and additive fusion. All seeds exhibit positive deltas, indicating a consistent structural effect.

6.2 Cross-Dataset Comparison

We analyze the effect of fusion strategies across datasets that naturally differ in sequence length, rather than post-hoc slicing within a single corpus. AG News contains short texts, IMDB represents medium-length documents, and ArXiv consists of substantially longer inputs.

As shown in **Table 1**, fusion choice has negligible and inconsistent impact on AG News and IMDB, where performance saturates and differences across fusion operators fall within statistical variation. Additional paired-seed delta analyses for AG News and IMDB are provided in the **(Appendix A)** . In contrast, on ArXiv—where documents span thousands of tokens—Gate-Scalar fusion consistently outperforms additive fusion, with improvements that are stable across random seeds (**Figure 3**).

This cross-dataset comparison indicates that the benefit of learnable positional fusion emerges primarily in long-sequence regimes. The absence of gains on shorter datasets suggests that fusion effects are masked when positional structure is simple or local, whereas long documents expose limitations of uniform additive injection.

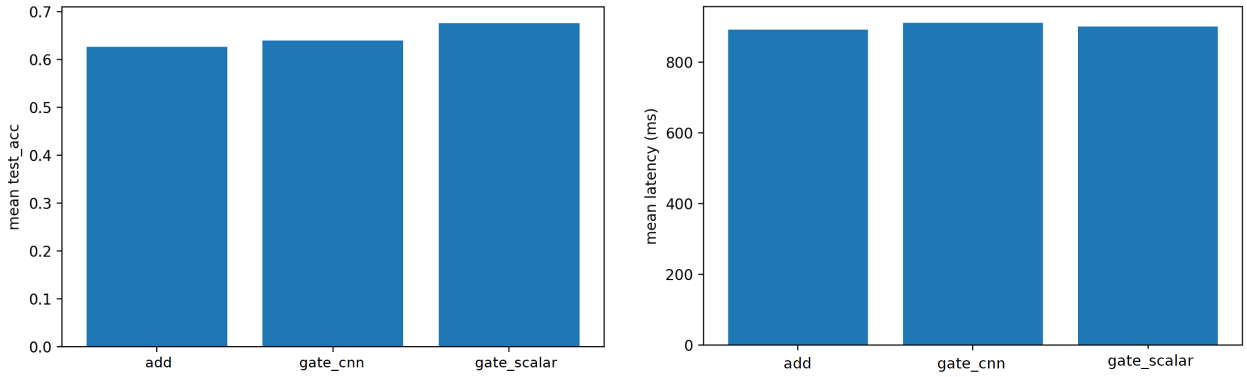
7 Local Inductive Bias in Positional Fusion

This section evaluates whether inductive bias can be introduced at the fusion level without modifying attention.

Table 2: Mean test accuracy and relative inference latency on ArXiv for different fusion mechanisms. Accuracy values are rounded to two decimals; latency is reported qualitatively due to implementation-dependent variance.

Fusion Mechanism	Mean Test Accuracy	Inference Latency
Add	~ 0.62	Lowest
Gate-CNN	~ 0.64	Slightly higher
Gate-Scalar	$\sim \mathbf{0.67}$	Highest

Gate-Scalar achieves the highest mean accuracy.. Convolutional gating improves over additive fusion, but paired-seed deltas show less consistent sign than Gate, indicating less stable improvements under the current setting.



(a) Mean test accuracy on ArXiv by fusion operator.

(b) Mean inference latency on ArXiv by fusion operator.

Figure 4: Mean accuracy and inference latency across fusion mechanisms.

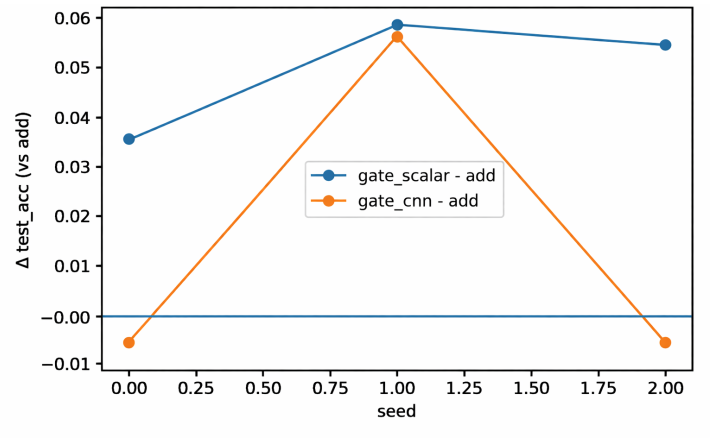


Figure 5: Paired per-seed accuracy deltas on ArXiv relative to Add. Positive values indicate improvement over Add under the same seed.

Paired-seed deltas show that gated fusion (Gate-Scalar) yields consistently positive improvements relative to additive fusion, whereas convolutional gating exhibits less consistent improvements than Gate-Scalar, indicating less stable gains. While these results focus on sinusoidal positional encodings, **(Appendix A)** reports complementary experiments on the ArXiv dataset showing that the advantage of learnable fusion generalizes across multiple positional encoding families.

8 Discussion

Our results show that positional-encoding fusion is not a neutral design choice for long-sequence Transformers. While short-sequence benchmarks saturate and mask fusion effects, long-document classification reveals consistent gains from Gate-Scalar fusion. Cross-dataset comparison across sequence-length regimes shows the effect emerges primarily on long documents. Convolutional gating demonstrates that local inductive bias can be introduced at the fusion level, although its benefits are less stable than those of Gate-Scalar in our experiments. Extended experiments on the ArXiv dataset (reported in the **(Appendix A)**) indicate that the advantage of learnable fusion persists across multiple positional encoding families, suggesting that the observed limitation arises from the fusion mechanism rather than from a specific encoding formulation.

9 Limitations

This study focuses on classification tasks and a single Transformer architecture. While we evaluated four major positional encoding families on ArXivClassification, future work could extend this analysis to generative tasks, hybrid encodings (e.g., ALiBi), or multimodal settings.

10 Conclusion

We have shown that the mechanism used to fuse positional encodings with token embeddings materially affects Transformer performance on long documents. Through controlled experiments, paired-seed analysis and cross-dataset comparison across sequence-length regimes, we demonstrate that gated fusion consistently outperforms the default additive scheme for long-sequence classification. These findings suggest that positional-encoding fusion should be treated as an explicit design choice when building Transformers for long-context tasks.

A Appendix: Additional Experiments

This appendix provides additional supporting results beyond the main ArXiv analysis. We first report extended ArXivClassification results across multiple positional encoding families (**Table 3**) to assess whether fusion effects generalize beyond sinusoidal encodings. We then present paired per-seed accuracy deltas for AG News and IMDB under sinusoidal positional encodings. These short- and medium-length datasets complement the ArXiv findings by showing that fusion effects are small and less consistent across seeds in saturated sequence-length regimes.

A.1 Cross-Positional Encoding Results on ArXivClassification

Table 3: Test accuracy (mean ± standard deviation) on the long-sequence ArXiv classification dataset across four positional encoding families and four fusion strategies. Results are averaged over five random seeds using identical architectures and hyperparameters, where MLP denotes a content–position fusion baseline evaluated **only on ArXiv**.

Positional Encoding	Add	Concat	Gate-Scalar	MLP-Gate
Sinusoidal	59.22 ± 1.07	63.44 ± 2.27	65.73 ± 4.01	65.50 ± 2.31
Learned Absolute	62.29 ± 3.24	60.83 ± 2.77	64.61 ± 2.83	64.28 ± 5.08
RoPE	58.47 ± 2.58	64.55 ± 1.82	65.61 ± 2.81	64.33 ± 3.67
Relative	62.48 ± 2.53	59.83 ± 2.11	65.55 ± 3.15	65.23 ± 2.34

Across all four positional encoding families, Here, Gate-Scalar refers to the scalar positional gate used in the main paper, while Gate-MLP denotes a feature-wise MLP-based gating variant evaluated only in Appendix A. Neither gating variant dominates universally: scalar gating performs slightly better for sinusoidal and RoPE encodings, while MLP-based gating achieves comparable or marginally higher accuracy for learned absolute and relative encodings. These results indicate that the primary source of improvement lies in introducing a learnable fusion mechanism, rather than in a specific gating parameterization or positional encoding formulation.

A.2 AG News

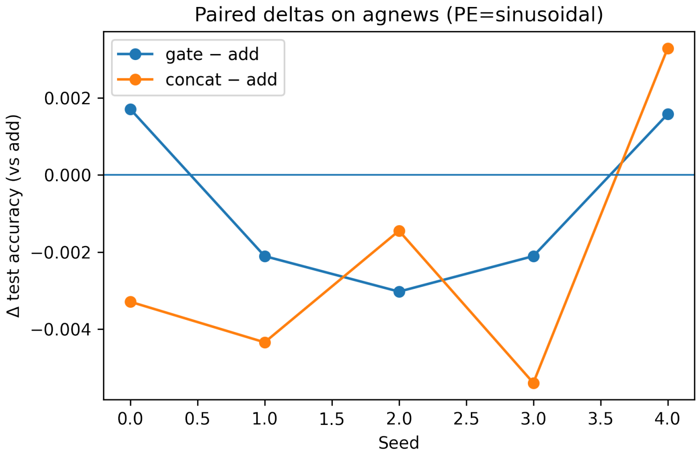


Figure 6: Paired per-seed accuracy deltas (Gate Scalar - Add) on AG News using sinusoidal positional encoding. Deltas are small may change sign across seeds, consistent with saturation effects on short sequences

Figure 6 shows paired per-seed accuracy deltas between gated fusion and additive fusion on the AG News dataset. The deltas are small in magnitude and exhibit sign changes across seeds. This indicates that, for short sequences, the choice of fusion operator does not produce a stable or systematic advantage, and performance differences fall within statistical variation.

A.3 IMDB

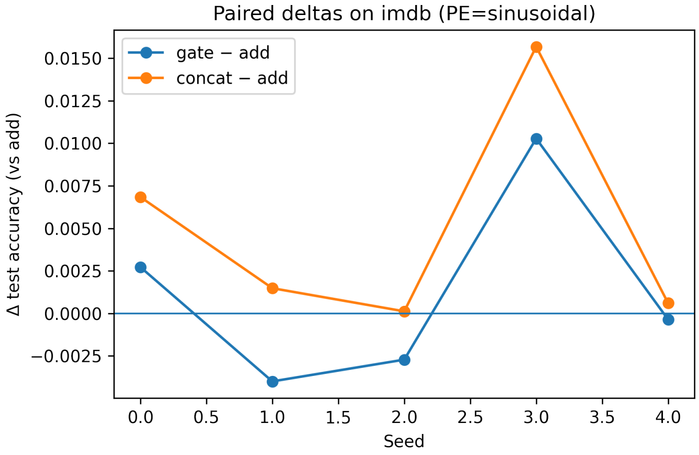


Figure 7: Paired-per-seed accuracy deltas (Gate Scalar - Add) on IMDB using sinusoidal positional encoding. Deltas are small and less sign-consistent than on ArXiv, consistent with limited differentiation between fusion strategies on medium-length sequences.

Figure 7 presents paired per-seed accuracy deltas for the IMDB dataset. Similar to AG News, the deltas are modest and less sign-consistent than those observed on ArXiv. This suggests that, for medium-length documents, fusion mechanisms offer limited separation and do not yield robust improvements across random seeds.

Summary of Appendix Findings

Taken together, the appendix results reinforce the main conclusions of the paper. Learnable fusion mechanisms provide clear benefits on long-document classification, and this effect generalizes across multiple positional encoding families. In contrast, on short and medium-length datasets, fusion effects are small and unstable, consistent with performance saturation in these regimes.

References

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017. arXiv:1706.03762.

[2] 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*, 2018. arXiv:1803.02155.

[3] Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. In *International Conference on Learning Representations*, 2021. arXiv:2104.09864.

[4] Ofir Press, Noah A. Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations*, 2022. arXiv:2108.12409.

[5] Yi Tay, Mostafa Dehghani, Dara Bahri, Philip Pham, Nal Kalchbrenner, Jie Qin, and Donald Metzler. Long range arena: A benchmark for efficient transformers. In *International Conference on Learning Representations*, 2021. arXiv:2011.04006.

[6] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020. arXiv:2004.05150.

[7] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019. arXiv:1901.02860.

[8] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*, 2021. arXiv:2006.03654.

[9] 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. In *Journal of Machine Learning Research*, 2020.

[10] Pei Ke, Boxin He, Xiaoyan Liu, Jianfeng Wang, and Wei Liu. Rethinking positional encoding in language pre-training. In *International Conference on Learning Representations*, 2021.

[11] Ethan Chi, Zi Lin, and Yuxuan Sun. Position information in transformers: An empirical study. *Transactions of the Association for Computational Linguistics*, 2022.

[12] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. BigBird: Transformers for longer sequences. In *Advances in Neural Information Processing Systems*, 2020. arXiv:2007.14062.

[13] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In *International Conference on Learning Representations*, 2020. arXiv:2001.04451.

[14] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. Rethinking attention with performers. In *International Conference on Learning Representations*, 2021. arXiv:2009.14794.

[15] Sinong Wang, Belinda Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. In *International Conference on Machine Learning*, 2020.

[16] Joshua Ainslie, Santiago Ontanon, Chris Alberti, Arman Cohan, Michael Collins, et al. Etc: Encoding long and structured inputs in transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 2020.

[17] Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

[18] Emilio Parisotto, Francis Song, Jack W. Rae, Razvan Pascanu, Caglar Gulcehre, Siddhant Jayakumar, Max Jaderberg, Raphael Lopez Kaufman, Aidan Clark, Sebastien Noury, and Matthew Botvinick. Stabilizing transformers for reinforcement learning. In *International Conference on Machine Learning*, 2020. arXiv:1910.06764.

[19] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[20] Mohamed Amine Hallam. Fusion matters: Code and reproduction materials. `https://github.com/MoAmineHallam/Fusion-matters`, 2026. GitHub repository.