SPECTRAL ENHANCED TRANSFORMER: MULTI-RESOLUTION CAUSAL CONVOLUTION FOR TRANSFORMER LANGUAGE MODELING

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce the **Spectral Enhanced Transformer (SET)**, which augments Transformer blocks with a learned multi-resolution causal convolutional path parallel to self-attention and causal token mixing after the MLP. FFT evaluation with zero padding preserves autoregressive causality and is numerically equivalent to direct evaluation of the same kernels. A complete 30-run WikiText-103 matrix crosses two positional encodings, five parameter-matched arms, and three paired seeds, with 61.44M training tokens per run. Full SET reduces mean validation NLL from 4.01723 to 3.88255 with learned positions and from 3.87901 to 3.84695 with RoPE, with every paired seed favoring full SET over the wider-FFN baseline. MLP-only mixing underperforms the baseline, and a fixed schedule beats depth-varying full SET in every RoPE seed, so the results do not establish a depth-varying-schedule advantage. Earlier synchronized timing measures 1.85× baseline step time, and longer baseline training wins under wall-clock matching. The supported result is a finite-budget equal-token quality–compute tradeoff, not an efficiency or scaling advantage; larger historical runs remain descriptive because of incomplete metadata and undertraining.

1 INTRODUCTION

Self-attention mixes token representations using content-dependent weights (Vaswani et al., 2017). Causal convolution applies a learned filter across relative offsets. Combining the two gives a language model both kinds of token mixing, but adds parameters and computation. We study whether this addition lowers loss when the training-token budget is fixed, and whether the improvement survives parameter and wall-clock controls.

Natural language contains dependencies at multiple ranges, and some sequence transformations are approximately translation invariant. Causal convolution provides a direct inductive bias for such transformations, while attention provides content-dependent interactions. Frequency-domain evaluation is one efficient way to compute long convolutions, but it does not change the function represented by a fixed time-domain kernel. Related work has explored Fourier mixing and learned filters: FNet (Lee-Thorp et al., 2022) replaces attention with a static Fourier transform in encoders, and GFNet (Rao et al., 2021) studies learned spectral filters for vision.

In an *autoregressive* language model, spectral mixing must be implemented causally. Although a Fourier coefficient depends on the complete input sequence, a correctly constrained convolution can remain causal. An unconstrained global spectral filter can instead expose future tokens during next-token evaluation. Our implementation diagnostic illustrates the resulting target leakage: its non-causal model reaches validation loss 0.09, which is not a valid autoregressive result.

We address this implementation requirement using classical causal convolution via FFT—a standard signal-processing technique also used in SSMs (Gu et al., 2022). The contribution is the integration of learned multi-resolution causal mixing with attention and the MLP, not the convolution theorem itself. We make three contributions:

1. SET combines a multi-resolution convolutional path parallel to attention with causal token mixing after the MLP (§5).

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.

2. A complete 30-run component matrix compares five parameter-matched arms under learned positions and RoPE. The full model lowers loss at equal training tokens, while MLP-only mixing does not (§6).
3. Matched FFT/direct runs, schedule controls, synchronized timing, and longer baseline training separate the architecture's quality effect from its implementation cost. SET's equal-token gain does not survive the tested wall-clock comparison (§7.3).

2 Related Work

Efficient Attention and Hybrid Architectures. The $O(n^2)$ cost of self-attention has motivated linear attention (Katharopoulos et al., 2020), kernel approximations (Choromanski et al., 2021), hardware-optimized exact attention (Dao et al., 2022), and structured state space models (S4 (Gu et al., 2022), Mamba (Gu & Dao, 2024)). H3 studies SSM layers and hybrid attention–SSM models (Fu et al., 2023). Hyena interleaves implicit long convolutions and data-controlled gates as an attention replacement (Poli et al., 2023). These are direct precedents for efficient gated convolutional language modeling. SET retains dense attention and integrates spectral processing *within* each block (as a parallel path, not alternating layers), uses multi-resolution kernels with a depth-dependent length schedule, and adds spectral gating to the MLP. The current data support this finite-budget integration rather than novelty of causal FFT convolution or a general advantage over optimized Hyena/H3 systems. Table 7 includes one Mamba-style baseline under the submitted finite-step recipe; it is not a general comparison with optimized SSM training.

Spectral Methods in Deep Learning. FNet (Lee-Thorp et al., 2022) replaces attention with an unparameterized 2D DFT, retaining most of BERT's quality with 7× speedup. However, FNet is limited to bidirectional (encoder) settings and uses a static, non-learned transform. GFNet (Rao et al., 2021) learns frequency-domain filters for image classification. The Gated Attention mechanism (Qiu et al., 2025) adds per-head sigmoid gates after scaled dot-product attention, improving stability and eliminating attention sinks. SET instead uses learned gates to combine attention with causal convolution evaluated in the frequency domain.

Representation Motivation. The superposition framework of Liu et al. (2025) relates feature interference to representation capacity. It motivates asking whether separating translation-invariant and content-dependent mixing can reduce interference, but it does not imply a scaling law for SET. We treat this connection as a hypothesis rather than a theoretical or empirical result.

3 Causality in Fourier-Based Token Mixing

3.1 Unmasked Global Fourier Mixing Leaks Future Tokens

An autoregressive language model predicting token x_{t+1} must condition only on $x_1, \ldots, x_t$. Standard self-attention enforces this via a causal mask. The DFT, however, computes frequency components $X_k = \sum_{t=0}^{n-1} x_t \, e^{-2\pi i k t/n}$, each depending on *all* positions. Applying a learned frequency filter H_k and inverting produces outputs where position t depends on positions $t+1, \ldots, n$, violating causality.

Empirical demonstration. To quantify this violation, we trained a 25M-parameter model where half the attention heads were replaced with naive (non-causal) FFT filters. The model achieved a validation loss of **0.09** and perplexity of **1.1** because future-token information reaches the prediction. This is target leakage, not evidence of memorizing validation examples. In contrast, the same architecture with standard causal attention achieves loss 4.33 / PPL 75.9 *in this controlled short-training ablation* (Table 6; the main-experiment 25M baseline in Table 13 reaches 4.032/56.4 with the full 8k-step training). This $48\times$ gap shows that spectral methods for autoregressive models must explicitly enforce causality.

3.2 Causal Convolution via FFT

We use **linear causal convolution**—a classical signal processing operation—because it is (i) causal, (ii) computable via FFT, and (iii) learnable. A causal kernel $\mathbf{h} = [h_0, h_1, \ldots, h_{K-1}]$ (with implicit

zero-padding beyond index $K-1$) produces:

$$y_t = \sum_{\tau=0}^{\min(t,K-1)} h_\tau \cdot x_{t-\tau} \tag{1}$$

where y_t depends only on $x_0, \ldots, x_t$. By the convolution theorem, this operation is equivalent to element-wise multiplication in the frequency domain:

$$\mathbf{y} = \text{IFFT}\Big(\text{FFT}(\mathbf{x};\ n{+}K{-}1)\ \odot\ \text{FFT}(\mathbf{h};\ n{+}K{-}1)\Big)_{0:n} \tag{2}$$

where both $\mathbf{x}$ and $\mathbf{h}$ are zero-padded to length $n + K - 1$ (to avoid circular convolution artifacts), and the output is truncated to n elements. The total cost is $O(n \log n)$ per feature dimension.

4 Architectural Motivation

We give architectural intuition rather than a general lower bound or scaling theorem. The two arguments below state the positional and modeling assumptions under which they are intended to apply.

4.1 Restricted Positional Intuition

Consider a single dot-product-attention layer with learned absolute positional embeddings, no relative-position bias, and no RoPE. Representing an arbitrary full-rank Toeplitz positional-logit matrix requires the positional representation to distinguish a growing number of relative offsets. A causal convolution represents the corresponding translation-invariant linear map directly through its impulse response. This comparison motivates a dedicated convolutional path under these restricted assumptions; it is not a width lower bound for general multi-layer Transformers, relative-position attention, or RoPE. Indeed, the controlled RoPE experiment in Table 3 shows that relative-position structure reduces, but does not eliminate, SET's empirical gain.

4.2 Complementary Inductive Biases

Attention supplies content-dependent interactions, whereas causal convolution applies the same learned transformation at each offset. A parallel architecture can therefore allocate different features to the two paths. The superposition framework of Liu et al. (2025) offers one possible interpretation: separating feature types may reduce interference. This explanation is heuristic, however. The exploratory model-size runs use unequal and incomplete training records, and the controlled fixed-schedule experiment does not show an independent benefit from depth-varying kernels. We therefore do not use the observed loss percentages to claim a superposition mechanism or a scaling law.

5 Method: Spectral Enhanced Transformer

SET adds two causal token-mixing components to each Transformer block: a *multi-resolution convolutional residual path* parallel to attention and a *causal convolutional gate* after the MLP. The submitted implementation varies residual-path kernel lengths with depth; Section 7.3 evaluates this schedule separately. Figure 1 illustrates one SET block.

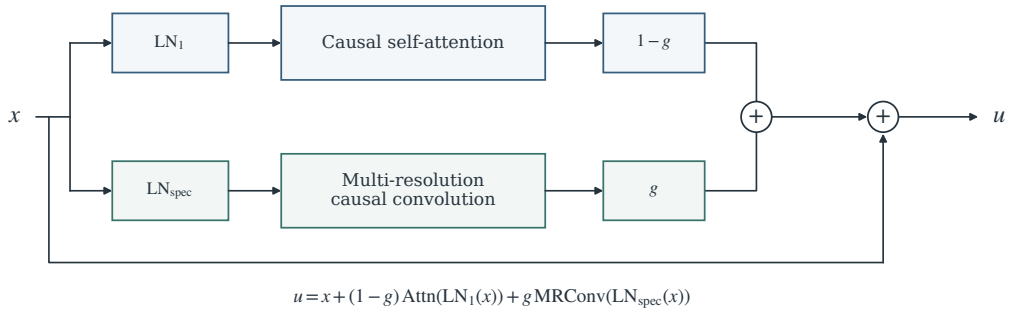
5.1 Multi-Resolution Causal Convolution

Motivation. Natural language contains dependencies at multiple temporal scales. SET provides a dedicated path with several kernel lengths, thereby parameterizing translation-invariant mixing over several receptive fields. The path costs $O(n \log n)$, while the complete block remains quadratic because attention is retained.

Design. We partition the d-dimensional hidden state into $G = 4$ groups, each processed by a causal convolution kernel of different size via FFT (Eq. 2). Each per-dimension kernel $h_\tau^{(g,d)}$ includes an exponential decay prior to encourage locality and stabilize training:

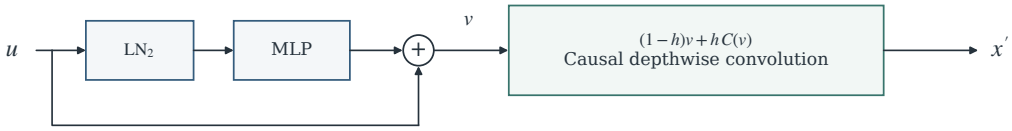
$$h_\tau^{(g,d)} = w_\tau^{(g,d)} \cdot \exp(-\tau/\sigma_g), \qquad \sigma_g = K_g/4 \tag{3}$$

(a) Parallel token mixing



$$u = x + (1-g)\,\mathrm{Attn}(\mathrm{LN}_1(x)) + g\,\mathrm{MRConv}(\mathrm{LN}_{\mathrm{spec}}(x))$$

(b) Feed-forward update and convolutional gating



$g, h \in (0, 1)$: learned sigmoid gates; C is applied after the MLP residual addition.

Figure 1: A SET block as implemented. (a) Separate LayerNorms feed attention and multi-resolution causal convolution; a sigmoid gate mixes their outputs before the first residual addition. Kernel lengths follow a predefined depth schedule. (b) The MLP residual addition precedes the convolutional gate, which blends the entire updated stream with its causal convolution. Convolutions use zero-padded FFTs or equivalent direct evaluation.

where $w_\tau^{(g,d)}$ is a learned parameter. The multi-resolution outputs are concatenated and linearly projected. The implementation then interpolates this projection with its input using a learned per-dimension sigmoid gate: $\mathrm{MRCS}(z) = \sigma(\gamma) \odot W_o \mathrm{concat}_g(h_g * z_g) + (1 - \sigma(\gamma)) \odot z$. This internal vector gate is distinct from the scalar gate between attention and the spectral path.

Integration. The spectral path runs in *parallel* with self-attention, with a learned per-layer scalar gate $\alpha^{(l)}$ controlling allocation:

$$\mathbf{h}^{(l)} = \mathbf{x}^{(l)} + \big(1 - \sigma(\alpha^{(l)})\big) \cdot \mathrm{Attn}\big(\mathrm{LN}(\mathbf{x}^{(l)})\big) + \sigma(\alpha^{(l)}) \cdot \mathrm{MRCS}\big(\mathrm{LN}_{\mathrm{spec}}(\mathbf{x}^{(l)})\big) \tag{4}$$

where σ is the sigmoid function and $\alpha^{(l)}$ is initialized to 0 (equal allocation).

5.2 Predetermined Depth-Dependent Kernel Lengths

Kernel lengths grow with layer depth, increasing the available receptive field as in convolutional networks (Lin et al., 2017):

$$K_g^{(l)} = \min\Big(\big\lfloor K_g^{\mathrm{base}} \cdot \big(1 + 3(l+1)/L\big)\big\rfloor,\ \ n\Big), \qquad K_g^{\mathrm{base}} \in \{8, 16, 32, 64\} \tag{5}$$

For the implemented $L = 12$ model, at layer $l = 0$ the four groups use kernels of size $\{10, 20, 40, 80\}$; at layer $l = L - 1$, they grow up to $\{32, 64, 128, 256\}$. The lengths follow a predefined depth-varying schedule; they are neither learned nor input-adaptive. The filter weights remain learned. The fixed-length control likewise learns its filter weights. This is a design choice in the submitted model, not an independently validated source of accuracy: a fixed $\{8, 16, 32, 64\}$ schedule performs similarly in the controlled experiment (Table 16).

5.3 Causal Mixing After the MLP

Motivation. The standard Transformer MLP operates independently at each position, with no cross-position information flow. While attention provides inter-position communication *before* the MLP, the MLP's output at position t depends only on the representation at t. We add a lightweight spectral layer *after* the MLP to provide position-local context.

Design. A single shared causal kernel (with per-dimension scaling) is applied to $z = h + \mathrm{MLP}(\mathrm{LN}(h))$, the entire updated residual stream, rather than just the MLP branch:

$$\mathrm{SpectralMLP}(\mathbf{x}) = \sigma(g) \cdot (\mathbf{k} * \mathbf{x}) + (1 - \sigma(g)) \cdot \mathbf{x} \tag{6}$$

where g is initialized to -2 ($\sigma(-2) \approx 0.12$, nearly closed) to give the convolution a smaller initial mixing weight. This adds $K + d + 1$ parameters per layer. Its implemented depth schedule is $K = 16 + \lfloor 48(l+1)/L \rfloor$; the fixed-schedule control uses $K = 40$. The identity mixture can also rescale the residual stream, so an effect cannot be attributed solely to adding token context.

5.4 Computational Analysis

Table 1: Asymptotic costs of the submitted SET block. SET retains quadratic attention and adds sub-quadratic convolutional paths.

Component	Complexity	Notes
Self-Attention	$O(d \cdot n^2)$	Dominates; unchanged
Spectral Residual Path	$O(d \cdot n \log n)$	Via FFT; sub-quadratic
Spectral MLP Gate	$O(d \cdot n \log n)$	Lightweight; shared kernel
Total SET	$O(d \cdot n^2 + d \cdot n \log n)$	Attention remains dominant asymptotically

The added paths are sub-quadratic, but SET does not remove attention and therefore remains $O(n^2)$ overall. In a synchronized H100 measurement under the archived "85M" configuration label, batch 24, and length 256, the baseline takes 41.73 ms/step (147,235 tokens/s) and SET takes 77.34 ms/step (79,437 tokens/s), or 1.85× as much elapsed time per step. Matched direct convolution takes 81.28 ms/step, so FFT is 5.1% faster for the same convolutional function in this implementation. Fused kernels such as FlashFFTConv (Fu et al., 2024) may change these constants, but we do not claim an end-to-end efficiency advantage from the present implementation.

6 Parameter-Matched Component Comparison

Our primary 30-run frozen matrix crosses two positional encodings, five arms, and three paired seeds. Every run uses 10,000 updates, batch 24, sequence length 256 (61.44M training tokens), and the same 247,040 validation tokens. FFN widths match parameter budgets within 0.1%. The deterministic batching implementation differs from the older runs, so these matrices are not pooled.

Table 2: Parameter-matched component comparison: mean validation NLL over three seeds.

Arm	Learned positions	RoPE
Full	3.88255	3.84695
Residual only	3.90730	3.86817
MLP only	4.03598	3.91221
None (wider FFN)	4.01723	3.87901
Full, fixed schedule	3.88372	3.84033

Full improves over none in every seed, with mean differences -0.13469 and -0.03207 NLL, and over residual-only by -0.02475 and -0.02122. MLP-only underperforms none in both settings. Depth-varying full versus fixed full has mixed signs with learned positions (mean -0.00117), while fixed full wins all three RoPE seeds (depth-varying-minus-fixed mean $+0.00662$). Consequently,

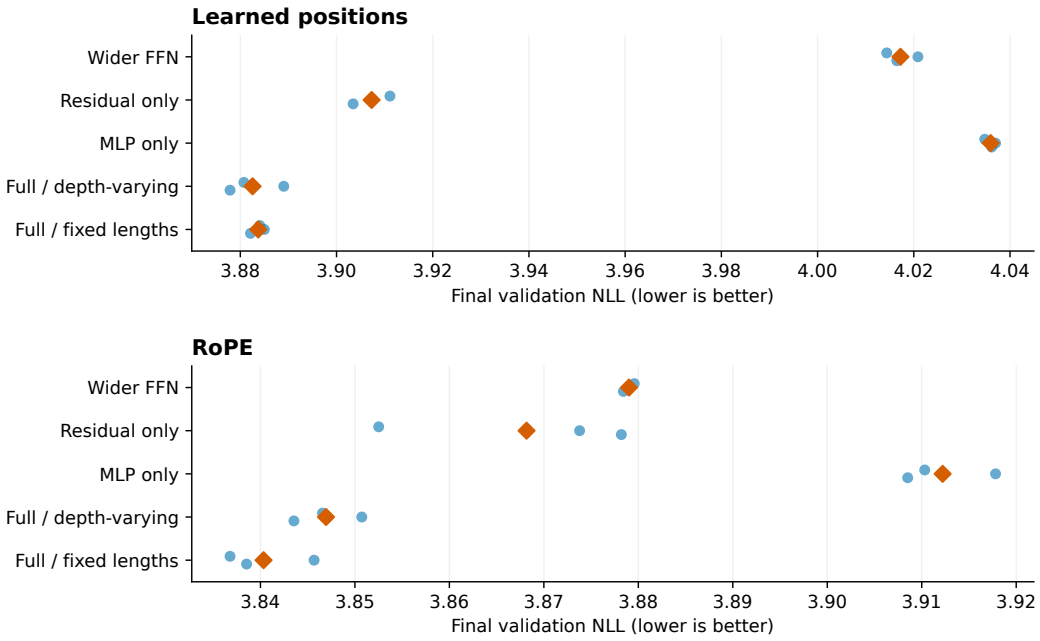


Figure 2: Complete 30-run matched-parameter matrix at 61.44M training tokens. Small points retain all three seed results; diamonds show the mean, with no confidence interval inferred from three seeds. Kernel schedules vary by depth or use fixed lengths; both learn filter weights. This equal-token comparison does not match total computation.

these data support the combined path under equal-token training but do not support a stable benefit from the depth-varying schedule. Three seeds and one corpus do not establish a general architecture advantage. This comparison controls parameters and training tokens, not FLOPs or wall-clock cost. All 30 runs contribute their final validation losses at the same training-token budget; no run is selected by its best intermediate checkpoint.

7 Implementation and Computational-Cost Controls

7.1 Experimental Setup

Datasets. Primary: WikiText-103 (Merity et al., 2017) (118M training tokens). Generalization: C4 (Raffel et al., 2020) (500M tokens). Tokenization: GPT-2 BPE vocabulary (50,257 tokens).

Models. GPT-style autoregressive LMs at 8 scales from 25M to 1B parameters. All models use learned position embeddings, pre-norm architecture, and weight-tied output heads. SET adds ∼5–6% parameter overhead via spectral kernels and gates.

Submitted-run training records. The recorded common settings are AdamW (β_1=0.9, β_2=0.95, weight decay 0.1), peak learning rate 3×10^{-4}, 500-step linear warmup followed by cosine decay to 3×10^{-5}, BF16 mixed precision, and gradient clipping at 1.0. Archived loss histories verify 8k steps at 25M and 10k steps at 50M–200M. They do not retain the original global batch, gradient accumulation, or measured FLOPs, and the 350M and context-length records lack complete training commands. A later release script provides a reproduction recipe but is not exact provenance for every submitted result. We report the submitted outcomes for completeness, but do not treat them as controlled model- or context-scaling sweeps.

Controlled protocol. New controls use the released 85M configuration, WikiText-103, sequence length 256, global batch 24, 10,000 steps, and 61,440,000 training tokens (0.521 passes over the

Table 3: Controlled 85M results on WikiText-103. Each entry is validation loss after 10,000 steps and 61.44M training tokens; rows use identical seeds and data order.

Position encoding / model	**Seed 604**	**Seed 605**	**Seed 606**	**Mean**
Learned-position Transformer	4.027726	4.028550	4.021857	4.026045
Learned-position SET	**3.885801**	**3.886889**	**3.881468**	**3.884719**
RoPE Transformer	3.894675	3.891497	3.887873	3.891348
RoPE SET	**3.844272**	**3.844629**	**3.846424**	**3.845108**

Table 4: New contemporaneous WikiText-103 matrix: final 10,000-step validation NLL. All 18 runs use the reconstructed data files; these are not pooled with archived runs.

Configuration	Seed 604	Seed 605	Seed 606	Mean $\pm$ SD
Learned baseline	4.027726	4.028550	4.021857	4.026045 ± 0.003650
Learned SET	3.885801	3.886889	3.881468	3.884719 ± 0.002868
Learned wider FFN	4.020317	4.016723	4.010125	4.015722 ± 0.005169
RoPE baseline	3.894675	3.891497	3.887873	3.891348 ± 0.003403
RoPE SET	3.844272	3.844629	3.846424	3.845108 ± 0.001154
RoPE wider FFN	3.884752	3.880440	3.876556	3.880582 ± 0.004100

117,920,140-token training file). All paired comparisons use seeds 604/605/606 and identical data order. Each run uses one H100 80GB GPU.

Here "85M" is the archived configuration name, not the actual parameter count in the controlled logs. Learned-position baseline/SET have 91,342,720 and 96,893,968 parameters; RoPE baseline/SET have 91,178,880 and 96,730,128. The controlled SET comparison therefore adds about 6.1% parameters. FFT/direct variants match parameters exactly, but the fixed schedule has 96,519,568 parameters versus 96,893,968 for the depth-varying schedule. Equal training tokens do not imply parameter- or compute-matched comparisons. The older submitted parameter table describes a different archived configuration and is not used to override these recorded controlled-run counts.

Exploratory comparisons. Larger-model and context-length results are retained in Appendix I. Their incomplete budget metadata and undertrained baselines prevent scaling claims; the controlled comparisons below are the primary evidence.

7.2 Controlled Equal-Token Results

SET reduces mean loss by 3.51% with learned absolute positions and by 1.19% with RoPE. The reduction occurs in every paired seed. The smaller RoPE difference is consistent with positional encoding affecting the comparison, but this experiment does not identify the cause of the remaining difference. These are equal-token results; Section 7.3 separately measures their computational cost.

The earlier component ablation is retained in Appendix K; the complete equal-parameter matrix above is the primary component comparison.

7.3 Controlled Implementation and Compute Ablations

Additional parameter-budget comparison. We rerun baseline, SET, and wider-FFN baseline for both positional encodings and all three seeds, using reconstructed WikiText-103 files. The FFN width 2,921 is chosen from parameter counts alone. Learned-position SET/wider models have 96,893,968/96,887,680 parameters; RoPE has 96,730,128/96,723,840, a difference of 6,288 parameters (about 0.0065%). All receive the same 61.44M training tokens. Appendix A specifies the frozen protocol; archived and new runs are not pooled as additional seeds.

Table 4 shows lower SET NLL in every paired comparison with the wider control. Mean SET-minus-wider differences are -0.131002 (learned; 3.26% reduction) and -0.035474 (RoPE; 0.91%). Thus simply expanding this FFN to the same budget does not close the observed gap. The intervention

Table 5: Matched FFT and direct causal convolution at the nominal 85M configuration. Observed losses are close across three paired seeds; this is not an equivalence test.

Seed	FFT	Direct	Direct − FFT
604	3.885801	3.885374	-0.000428
605	3.886889	3.886134	-0.000755
606	3.881468	3.882691	$+0.001224$
Mean	3.884719	3.884733	$+0.000014$

Table 6: Causality enforcement is essential. Without it, spectral processing produces deceptively low loss by accessing future tokens.

Method	Val Loss	PPL	Causal?
Standard Attention	4.332	75.9	✓
Naive FFT (non-causal)	0.091	1.1	✗ (target leakage)
Causal FFT Convolution (ours)	4.265	71.1	✓

does not isolate a unique convolutional mechanism, match FLOPs, or establish generalization across architectures. New baseline/SET values reproduce the archived values to the displayed precision, but the missing historical data files prevent certifying byte-identical preprocessing.

Submitted architectural controls. Appendix B retains the historical wider-Transformer, convolutional, and Mamba-style results. Their different recipes do not isolate the convolution algorithm or provide a contemporaneous control for new runs.

Matched FFT versus direct convolution. We implemented direct causal convolution using the identical learned time-domain kernels, exponential decay, kernel orientation, padding, groups, gates, projection, and spectral MLP path. Only the convolution algorithm changes. In FP64, maximum absolute output, input-gradient, and kernel-gradient differences are 8.44×10^{-15}, 8.88×10^{-15}, and 2.84×10^{-14}; FP32 relative errors are below 3.2×10^{-7}.

The large submitted 3.888-versus-4.059 difference is not reproduced when the implementations are matched. FFT evaluates the same mathematical convolution; finite-precision training trajectories need not be identical. The three-seed mean direct-minus-FFT difference is 0.000014, with an assumption-dependent paired-t 95% interval $[-0.002621, 0.002648]$. Such an interval neither certifies exact training equivalence nor establishes a general implementation advantage.

The three-seed depth-schedule comparison has no consistent winner (Appendix L); it does not support an independent accuracy benefit.

Wall-clock matching. The synchronized timing ratio is $1.85\times$, rather than the $1.53\times$ ratio used by the submitted "compute-matched" comparison. Rounding the measured ratio up, 18,600-step baselines reach losses 3.7404 and 3.7361 for seeds 605 and 606 (mean 3.7383), compared with SET losses 3.8869 and 3.8815 for the same seeds (mean 3.8842). Using the submitted throughput ratio, a 21,200-step baseline reaches 3.6951 for seed 604, compared with 3.8858 for SET. Thus SET improves loss at equal training tokens but not at equal wall-clock time in the present implementation.

Gate diagnostics. The submitted learned gates vary with depth and retain nonzero convolutional weights. Appendix C reports the values. Gate magnitude is not a controlled attribution of syntax, semantics, or reasoning capability.

7.4 The Causality Demonstration

The $\sim 47\times$ loss gap (25M scale) between non-causal naive FFT (0.091) and causal FFT convolution (4.265) demonstrates the severity of future-token leakage. Enforcing causality through zero-padded

linear convolution is standard signal processing; this table is an implementation diagnostic rather than a novelty claim.

Downstream evaluation. Token-level, completion, and throughput results are reported in Appendix H.

7.5 Generalization to Larger Dataset

On the submitted C4 runs (500M tokens), SET has 2.9–5.1% lower loss than the paired standard Transformer configurations from 125M to 1B (Appendix E). The direction of the difference is also observed on C4 under these recipes, but unequal optimization quality prevents a scaling-law or broad-generalization interpretation.

8 Discussion

Why does spectral integration help? The new wider-FFN control narrows the parameter-budget confound but does not identify a unique mechanism. Content-dependent attention and translation-invariant convolution may provide complementary biases. The smaller difference with RoPE is consistent with positional structure affecting the comparison, not proof of that explanation. Periodic linguistic patterns and improved gradient flow require direct tests.

Scaling behavior. The exploratory tables contain larger relative differences at some larger model sizes and contexts, but the original global batches, token budgets, and FLOPs are not fully recoverable, and the baseline losses show substantial undertraining. We therefore make no model- or context-scaling claim. A controlled scaling study would need fixed data and optimization budgets, complete run metadata, stronger training recipes, and contexts beyond 1024.

Relationship to SSMs. Unlike S4/Mamba hybrids that alternate SSM and attention layers, SET retains full attention in every block and adds multi-resolution spectral processing as a parallel path—a complementary augmentation, not a replacement.

Limitations. SET takes 1.85× the original baseline step time, and longer baseline training wins under wall-clock matching. The new wider-FFN comparison addresses parameter budget, not compute. Controlled evidence remains limited to the nominal 85M configuration and 61.44M training tokens. Larger-model and context runs have incomplete metadata and unequal optimization quality. Contexts extend only through 1024, without broad LM capability suites. The complete component matrix shows no depth-varying-schedule advantage; fixed full wins every RoPE seed. Mechanistic explanations remain hypotheses.

Choosing the relevant compute constraint. The equal-token and equal-time comparisons answer different resource questions. At a fixed token count, the convolution adds a trainable path that can improve validation loss under the tested parameter budgets. At a fixed wall-clock budget, its extra step cost reduces the number of updates that can be completed. The observed equal-token gain therefore cannot be translated directly into a training-time saving. Likewise, close FFT and direct-convolution losses concern the implementation of the same operation; they do not identify the fastest implementation on an untested accelerator or at a longer sequence length.

9 Conclusion

SET integrates learned multi-resolution causal convolution with attention and the MLP. Its complete 30-run matrix lowers validation NLL by 3.35% with learned positions and 0.83% with RoPE against parameter-matched wider-FFN controls. Matched FFT/direct runs have close losses without establishing statistical equivalence; the depth-varying schedule has no consistent advantage over the fixed schedule. SET incurs 1.85× the baseline step time and does not win under wall-clock matching. The supported contribution is an equal-token architectural improvement under tested budgets, not an efficiency or scaling claim.

Reproducibility Statement

The main text and supplementary parameter-budget protocol specify the architecture, tokenization, training schedule, paired seeds, and evaluation endpoint. The retained implementation and saved analysis records support the complete 30-run component matrix; the architecture figure follows the implementation's separate normalization branches and post-MLP residual gate. Historical runs are identified separately because their original commands, token files, or complete budget records are not all recoverable. Those records do not support an exact independent reconstruction of every historical result.

AI Use Statement

OpenAI Codex was used to aid and polish manuscript writing and to assist with literature retrieval and discovery, including finding related work. The authors are responsible for the final manuscript and citations.

References

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.

Tri Dao, Daniel Y Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. *Advances in Neural Information Processing Systems*, 2022.

Daniel Y. Fu, Tri Dao, Khaled K. Saab, Armin W. Thomas, Atri Rudra, and Christopher Ré. Hungry hungry hippos: Towards language modeling with state space models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=COZDy0WYGg.

Daniel Y Fu, Hermann Kumbong, Eric Nguyen, and Christopher Ré. FlashFFTConv: Efficient convolutions for long sequences with tensor cores. In *International Conference on Learning Representations*, 2024.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *Conference on Language Modeling*, 2024. URL https://colmweb.org/2024/AcceptedPapers.html.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *International Conference on Learning Representations*, 2022.

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

James Lee-Thorp, Joshua Ainslie, Ilya Eckstein, and Santiago Ontanon. FNet: Mixing tokens with Fourier transforms. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics*, 2022.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Yizhou Liu, Ziming Liu, and Jeff Gore. Superposition yields robust neural scaling. In *Advances in Neural Information Processing Systems*, 2025. Best Paper Runner-Up.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *International Conference on Learning Representations*, 2017.

Michael Poli, Stefano Massaroli, Eric Nguyen, Daniel Y. Fu, Tri Dao, Stephen Baccus, Yoshua Bengio, Stefano Ermon, and Christopher Ré. Hyena hierarchy: Towards larger convolutional language models. In *Proceedings of the International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 28043–28078, 2023. URL https://proceedings.mlr.press/v202/poli23a.html.

Zihan Qiu, Zekun Wang, Bo Zheng, Zeyu Huang, Kaiyue Wen, Songlin Yang, Rui Men, Le Yu, Fei Huang, Suozhi Huang, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free. In *Advances in Neural Information Processing Systems*, 2025. Best Paper Award.

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*, 21:1–67, 2020.

Yongming Rao, Wenliang Zhao, Zheng Zhu, Jiwen Lu, and Jie Zhou. Global filter networks for image classification. In *Advances in Neural Information Processing Systems*, 2021.

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*, 2017.

A Supplementary Parameter-Budget Protocol

The supplementary matrix fixes two positional encodings (learned absolute and RoPE), three architectures (original Transformer, SET, and wider-FFN Transformer), and seeds 604/605/606 before observing its results, for 18 runs. This is a local frozen protocol, not a third-party preregistration. We reconstruct the training and validation token files from the WikiText-103 raw parquet release using GPT-2 tokenization, concatenating nonempty text records without added EOS separators. The files contain 117,920,140 and 247,289 tokens respectively. The lost historical binary files cannot be checked for byte identity, so these new comparisons are analyzed together rather than pooled with archived runs.

All architectures use 12 layers, width 640, 10 heads, length 256, batch 24, 10,000 optimizer steps, and 61,440,000 training tokens. The baseline FFN width is 2,560; the wider control uses 2,921, chosen from parameter counts alone to differ from SET by less than 0.01%. This changes the architecture and is not an algorithm-only intervention. Model initialization is seeded before construction; a separately seeded data-loader generator gives the same training order within each seed. AdamW uses peak learning rate 3×10^{-4}, betas $(0.9, 0.95)$, weight decay 0.1, gradient clipping at 1.0, 500-step warmup, and cosine decay to one tenth of the peak rate. Training uses bfloat16 autocasting.

Validation runs every 500 steps on the fixed sequential validation loader, capped at 50 batches. With this file and drop-last batching there are 40 complete batches, or 245,760 scored next-token targets. The primary endpoint is the final 10,000-step NLL, not the best checkpoint. We retain all seed results and report sample standard deviations; three seeds do not establish broad statistical reliability. Concurrent-job wall times are not used as exclusive-hardware efficiency measurements. Preliminary smoke tests are excluded.

B Additional Architectural Controls

Table 7 reports the exploratory wider-Transformer, convolutional, and Mamba-style controls. The wider Transformer probes parameter count only in that training recipe, not the supplementary matrix. The original multi-resolution `nn.Conv1d` row used different initialization and padding/orientation and omitted Eq. 3's decay, the spectral inner gate, and the spectral MLP gate. It therefore cannot isolate FFT from direct evaluation. These finite-step results also do not establish a general comparison to optimized SSM training.

Table 7: Architectural controls under a separate training recipe. Convolutional alternatives are not matched FFT/direct comparisons.

Model	Params	85M Loss	125M Loss	Throughput
Standard Transformer	84M / 124M	4.021	4.242	214k / 194k
Wider Transformer (param-matched)	96M / 130M	4.017	4.223	115k / 143k
Conv-Augmented (single K=64)	97M / 131M	4.082	4.240	89k / 129k
Multi-resolution Conv control	97M / 131M	4.059	4.217	173k / 124k
Mamba-style SSM	92M / 124M	4.087	4.213	29k / 19k
SET	89M / 131M	3.888	4.015	101k / 91k

C Learned Gate Analysis

The gate values vary systematically with depth:

- **Shallow layers** assign approximately half of the parallel-path weight to convolution.
- **Deep layers** reduce that weight to approximately 35–40%, while retaining a nonzero convolutional contribution.
- **MLP gates** increase from approximately 0.12 to 0.16 across depth.

These coefficients stay away from zero, but the effective contribution also depends on branch-output scale and the internal vector gate. The coefficients alone do not measure how much information each path supplies or distinguish local syntax from semantic reasoning.

Table 8: Converged gate values in the 85M model. Values are descriptive and do not by themselves identify the function of either path.

Layer Group	Spectral Gate	MLP Gate	Scalar mixing weights
Layers 0–3	0.52–0.55	0.11–0.13	Convolution coefficient near one half
Layers 4–7	0.44–0.48	0.13–0.15	Attention coefficient slightly larger
Layers 8–11	0.35–0.42	0.14–0.16	Attention coefficient 0.58–0.65

D Model Configurations

Table 9: Detailed model configurations. SET adds spectral kernels (<6.5% parameter overhead).

Size	d_{model}	Heads	Layers	Std Params	SET Params	Overhead
25M	384	6	8	33.6M	35.4M	5.4%
50M	512	8	10	57.3M	60.5M	5.6%
85M	640	10	12	84.2M	89.1M	5.8%
125M	768	12	12	124M	131M	5.6%
200M	896	14	16	182M	193M	6.0%
350M	1024	16	24	327M	348M	6.4%

E Generalization to C4

Table 10: Submitted paired results on C4 (500M tokens). Training adequacy and compute metadata are not controlled across sizes.

Scale	Standard (C4)	SET (C4)	Improvement
125M	4.902	**4.713**	-3.9%
350M	5.398	**5.121**	-5.1%
760M	5.906	**5.672**	-4.0%
1B	6.465	**6.275**	-2.9%

SET has lower loss in each submitted C4 pair. Because training adequacy and complete compute metadata differ across sizes, the percentages should not be interpreted as a scaling trend.

F Fine-Grained Evaluation and Throughput

In these submitted evaluations, hard-token accuracy is higher for SET at both sizes and 4-way selection is 2.0–3.1 percentage points higher. These limited metrics do not identify the underlying mechanism or establish broad downstream capability.

G Implementation Details

Causal convolution. Each kernel $\mathbf{h} \in \mathbb{R}^{d_g \times K}$ (where $d_g = d/G$ is the group dimension) is initialized from $\mathcal{N}(0, 0.02^2)$ and multiplied element-wise by an exponential decay $\exp(-\tau/\sigma)$ with $\sigma = K/4$. FFT zero-pads both input and kernel to length $n + K - 1$ before transformation, and the output is truncated to n to ensure linear (non-circular) convolution. All FFT operations use PyTorch's `torch.fft.rfft/irfft` for real-valued efficiency.

Gate initialization. Spectral path gates $\alpha^{(l)}$ are initialized to 0 ($\sigma(0) = 0.5$, equal allocation between attention and spectral paths). MLP spectral gates are initialized to -2 ($\sigma(-2) \approx 0.12$,

Table 11: Submitted fine-grained evaluation and throughput (85M and 125M, WikiText-103 test set). Throughput is reported as the reduction in tokens/s, not an elapsed-time overhead.

	Overall Acc		**Hard (Rare) Acc**		**4-way Select**		**Tok/s**
Scale	Std	SET	Std	SET	Std	SET	**Reduction**
85M	35.1	**36.5**(+1.3)	6.4	**7.9**(+23%)	47.9	**51.0**(+3.1)	53%
125M	33.4	**35.7**(+2.3)	4.3	**6.6**(+53%)	45.9	**47.9**(+2.0)	53%

nearly closed). The parallel attention and convolutional paths therefore start with equal scalar weight. This initialization does not reproduce a standard Transformer block. The post-MLP gate initially assigns about 0.12 weight to its convolution; subsequent contributions depend on the learned gates and branch outputs.

Training stability. We use an exponential decay prior on kernels, conservative gate initialization, and gradient clipping at 1.0. We observed no training instabilities in the reported runs, but did not ablate these choices separately.

Reproducibility. The controlled 85M experiments use three paired seeds, identical data order, batch 24, sequence length 256, 10,000 steps, and 61.44M tokens. Learned-position baseline/SET means are 4.026045/3.884719; RoPE baseline/SET means are 3.891348/3.845108. Matched FFT/direct means are 3.884719/3.884733. The fixed-schedule mean is 3.884436. Runs use one NVIDIA H100 80GB GPU without model parallelism. The earlier 4.059 direct-convolution row is not used as statistical evidence because it did not match the SET architecture.

Table 12: Submitted token-level and completion evaluation. Models have unequal optimization quality across sizes, so cross-size trends should not be interpreted as scaling.

	Test PPL ↓		Next-tok Acc ↑		Completion Acc ↑	
Scale	Std	SET	Std	SET	Std	SET
25M	55.9	**52.7** (−5.7%)	36.1	**36.3**	77.6	76.9
50M	51.9	**47.0** (−9.5%)	36.9	**37.7**	78.8	77.9
85M	55.8	**48.7** (−12.8%)	36.8	**38.0**	75.5	**77.5**
125M	72.1	**57.1** (−20.8%)	33.4	**36.9**	72.7	**75.2**
350M	123.6	**91.0** (−26.4%)	30.2	**32.2**	68.8	**69.5**

Table 13: Validation loss from the exploratory WikiText-103 runs (sequence length 256). Token budgets and optimization quality are not controlled across sizes; percentages should not be interpreted as a scaling law.

Size	**Standard (Loss / PPL)**	**SET (Loss / PPL)**	**Δ Loss**	**Improvement**
25M	4.032 / 56.4	**3.970** / 53.0	−0.062	−1.5%
50M	3.950 / 52.5	**3.857** / 47.3	−0.093	−2.4%
85M	4.021 / 55.8	**3.888** / 48.8	−0.133	−3.3%
125M	4.242 / 69.6	**4.015** / 55.4	−0.227	−5.4%
200M	4.426 / 83.6	**4.155** / 63.8	−0.271	−6.1%
350M	4.777 / 118.7	**4.484** / 88.5	−0.294	−6.1%
760M†	5.781 / 323.9	**5.518** / 249.0	−0.263	−4.5%
1B†	6.379 / 589.1	**6.180** / 482.9	−0.199	−3.1%

H Downstream Evaluation

SET has lower test perplexity and higher next-token accuracy in each submitted pair. Completion accuracy decreases at 25M–50M and increases at 85M–350M. These token-level metrics do not establish broad language-model capabilities.

Fine-grained evaluation and throughput. The submitted evaluation shows higher rare-token accuracy at 85M and 125M and a 2.0–3.1 percentage-point increase in 4-way selection; Appendix F reports the values. The synchronized timing and corrected wall-clock comparison are reported in Section 7.3; the earlier 1.53× step comparison was not wall-clock matched.

I Exploratory Model and Context Sweeps

I.1 Results across Model Configurations

Key observations. (i) SET has lower loss than the corresponding standard Transformer in every submitted configuration. (ii) The standard-Transformer loss worsens with size, reaching 4.777 at 350M versus 4.032 at 25M. This behavior indicates unequal optimization quality and substantial undertraining. (iii) The relative differences therefore describe these individual finite-step runs; they do not establish that SET's advantage grows with model size.

I.2 Results at Different Sequence Lengths

The reported relative difference is larger at sequence length 1024 than at 256. Because the original batch, token budget, and FLOPs were not retained for this sweep, we do not attribute this pattern specifically to context length or claim long-context scaling. The maximum evaluated context is 1024 tokens.

Table 14: Submitted validation losses at three sequence lengths (85M model). Complete command metadata were not retained, so this table is descriptive rather than a controlled context-scaling experiment.

Seq Length	Standard (Loss / PPL)	SET (Loss / PPL)	Improvement
256	4.021 / 55.8	**3.888** / 48.8	-3.3%
512	4.510 / 91.0	**4.210** / 67.4	-6.7%
1024	4.824 / 124.5	**4.466** / 87.0	-7.4%

Table 15: Component ablation (25M, 8k training steps). The spectral residual path accounts for most of the observed difference.

Configuration	Val Loss	Δ vs Baseline
Standard Transformer (baseline)	4.032	—
+ Spectral MLP Gating only	4.042	$+0.3\%$
+ Spectral Residual Path only	3.992	-1.0%
+ Full SET (all three components)	**3.970**	$\mathbf{-1.5\%}$

J Fixed-checkpoint test sensitivity

We evaluate the 18 existing final checkpoints for none, full, and full-fixed, crossed with learned and rotary positions and the three original seeds. Evaluation uses the same 283,136 next-token targets from the public WikiText-103 raw test split, nonoverlapping length-256 chunks, and no retraining. We drop the remaining 150 possible targets. Because this dataset and historical test evaluations were available during the broader project, this is retrospective fixed-checkpoint evidence, not a previously unseen or selection-blind benchmark. Values are mean NLL and sample standard deviation across three seeds.

Position	Arm	Mean NLL	Seed SD
learned	full	3.8820	0.0031
learned	full-fixed	3.8828	0.0017
learned	none	4.0186	0.0033
rope	full	3.8510	0.0017
rope	full-fixed	3.8471	0.0036
rope	none	3.8835	0.0024

The fixed kernel-length schedule is almost tied with the depth-varying full model under learned positions and has lower mean NLL under rotary positions. Both arms still learn their filter parameters; this comparison does not isolate learning the frequency response. It does not establish an advantage for adaptive kernel-length scheduling. The full 30-run validation matrix remains the primary factorial experiment; this test subset does not replace its residual-only or MLP-only arms.

K Component Ablation under a Fixed Training Recipe

The spectral residual path provides the dominant difference (-1.0%). Spectral MLP gating alone is slightly worse than the baseline, whereas the full configuration is lower than either isolated component. Because this table does not hold the kernel schedule fixed, it does not isolate the contribution of depth adaptation; the controlled schedule comparison below addresses that question.

L Fixed and Depth-Varying Kernel Schedules

Depth schedule. The fixed control uses $\{8, 16, 32, 64\}$ in every layer and retains all other SET components. It has 96.520M parameters versus 96.894M for the depth-varying model because its kernels are shorter in some layers.

Table 16: Depth-varying and fixed kernel schedules at 85M. Neither schedule has a consistent advantage across seeds.

Seed	**Depth-varying**	**Fixed**	**Fixed − varying**
604	3.885801	3.887724	$+0.001923$
605	3.886889	3.881695	-0.005194
606	3.881468	3.883890	$+0.002422$
Mean	3.884719	3.884436	-0.000283

This result does not support an independent accuracy benefit from depth-varying scheduling in the controlled setting.