# Designing Better Transformers: Using Amorphous Network Proxies to Engineer the Unified Highway Transformer

Anonymous authors
Paper under double-blind review

## Abstract

Standard transformer architectures define information flow through rigid, serialized sequences of discrete operators. In deep or very deep implementations, this leads to PreNorm Dilution, or Latent Amnesia, in which deep layers become blind to early layer signals in the residual stream, stalling convergence. Existing solutions to PreNorm Dilution are computationally expensive or add engineering complications. This work utilizes unconstrained Amorphous Neural Network (AmNN) clouds as diagnostic proxies allowing analysis of information flow that emerges free from the constraints of discrete operators. These paths empirically confirmed the demand for historical latent context in deeper layers. Modeling this flow back into a discrete operator transformer resulted in the creation of the Unified Highway Transformer (UHT). The UHT introduces an $O(1)$ parallel memory bus ($Z_{mem}$) that builds a history of latent context from the residual stream and makes that history available to deep layers. To counter initialization instability during training, a Dynamic Variance Annealer initially scales residual projections entering the $O(1)$ bus, phasing out scaling after initial epochs. The UHT was evaluated against a constrained 120-layer, 9.7-million total trainable parameter architectural stress test ($d_{model} = 64$) on WikiText-103. A gradient tracker revealed dynamic, high-variance, gradient norm activation at deep levels in the UHT, illustrating that the UHT restored the representational capacity of the deep layers. In contrast, a 120-layer Pre-LN control showed gradient stagnation in deep layers. The UHT reduced terminal perplexity in the tested configuration (118 for the Pre-LN to 66 for the UHT). An ablation UHT removing all layer normalization operations achieved a similar perplexity of 68. A 180-layer test of the UHT with layer normalization operations showed similar performance. The UHT provides a scalable, parameter-efficient architecture for extreme-depth modeling that solves gradient stagnation in deep networks.

## 1 Introduction

Transformers, the backbone of deep learning for language models (Vaswani et al., 2017), rely on sequences of discrete operators that inherently create a rigid information flow. This flow is strictly serialized, passing through specialized modules—such as Multi-Head Self-Attention (MHSA) and Feed-Forward Networks (FFNs)—which exchange data with a primary residual stream.

Literature discussing Latent Amnesia (Manson, 2025) describes it as a phenomenon where deep networks progressively lose access to their own early-layer representations. Recent work by Kimi Team et al. (2026) references the mathematical root of this behavior as 'PreNorm Dilution', proving that standard unscaled residual connections dilute early state contributions. To solve this, those authors introduced Attention Residuals, to interconnect layers with attention outputs from prior layers through a combination of cross-layer communication, distributed memory transfers, and cached attention states. Approaches addressing deep network trainability include ResiDual (Xie et al., 2023), which adds a concurrent stream



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.

to preserve gradients, and DeepNorm (Wang et al., 2022), which aggressively scales residual connections to limit sub-layer updates.

This work utilized Amorphous Neural Networks (AmNN) as a diagnostic tool. In a Pre-LN Transformer (Xiong et al., 2020), the series of discrete FFN operators and the series of self-attention operators were each replaced or supplemented with an amorphous cloud that sampled data from and injected data to the residual signal stream at various points. This created an environment where information flow could be governed by emergent dynamics within the clouds.

When tested on the WikiText-103 dataset (Merity et al., 2016), this structure successfully converged. These unstructured amorphous clouds effectively replaced multiple discrete neural network operators. Analyzing the emergent flow within the amorphous clouds yielded unexpected diagnostic insights. The organic routing patterns revealed new pathways that also indicated inefficiencies inherent to the traditional Transformer structure. The hybrid self-attention cloud preferred upstream cloud data to mix with newly injected data from the residual stream. The FFN cloud showed a similar, but weaker, desire for upstream data. This confirms that forcing the self-attention block to operate only from the normalized residual stream is a structural inefficiency.

The cloud flow analysis was used as a guide to make minimally invasive modifications to the baseline Transformer and improve its performance. The result is the Unified Highway Transformer (UHT). It includes a "memory stream," or "unified memory highway" ($\mathrm{Z_{mem}}$), that is periodically updated via memory projections and direct addition from the residual stream. The $\mathrm{Z_{mem}}$ bus is implemented as an O(1) parallel memory bus. The memory stream serves as an "upstream" flow for both the self-attention and the FFN operators. It acts as a collective latent space that provides context otherwise "forgotten" in a conventional Transformer. The $\mathrm{Z_{mem}}$ bus models the cloud behavior in the diagnostic workbench, allows every intermediate layer to actively read from the unnormalized O(1) bus, and mitigates amnesia caused by sole reliance on the residual stream.

In a 120-layer, 9.7M parameter intentional stress test on WikiText-103, the UHT mitigated deep-layer gradient stagnation and reduced terminal perplexity from 117 in the Pre-LN baseline to 66. A 180-Layer UHT confirmed the architecture works well at further depth. In a larger parameter scale test (910 million parameters) on WikiText-103, the UHT also mitigated deep-layer gradient stagnation and reduced terminal perplexity from 102 to 80. While WikiText-103 does not benchmark state-of-the-art language learning capability, it provides a structural stress test to confirm that the $Z_{mem}$ bus successfully bypasses Layer Norm dilution and restores gradient flow at extreme depths.

## 2 Architectural Methodology – the Amorphous Diagnostic

The foundational mechanics of Amorphous Neural Networks—referred to hereafter as AmNNs—were recently formalized in literature (Simon, 2026). As established by Simon (2026), these networks operate via unconstrained connections where a neuron's inputs are stochastically distributed to upstream neurons at multiple upstream depths. This creates a non-isomorphic, fluidic relationship between processing states, bypassing the rigid constraints of traditional discrete layers.

While this prior literature establishes the fundamental structural theory of amorphous topologies, this paper introduces their novel application as a diagnostic workbench to evaluate discrete-operator flow. Figure 1 illustrates the approach taken to develop a dual-cloud model used to diagnose transformer flow, implemented in PyTorch.

The A-Cloud created free forming fluid flow to all the self-attention operators, and the C-Cloud replaced all the FFN operators. To simulate a Directed Acyclic Graph (DAG) and limit parameters, both clouds utilized continuous recurrent loops made fluidic with stochastic binary and temporal masks. The A-Cloud processed the combined residual signal injections and upstream cloud signals through a series of fluidic states, applied linear projection and the standard SA mechanics (QKV matrices, causal masking, SoftMax), and

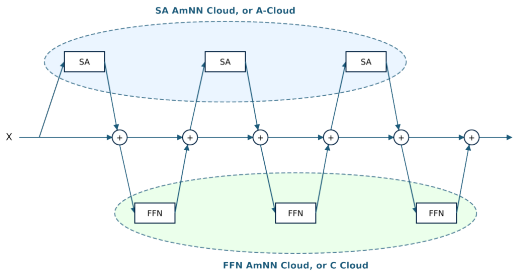


Figure 1: AmNN Cloud Overlay on Discrete Transformer Flow

extracted the result into the residual stream $X$. The C-Cloud processed flow through a series of fluidic states between injection and extraction points.

Both clouds were injected from the residual stream through linear projections with a non-linear activation (LeakyReLU). To observe unconstrained behavior, Layer Normalization (Ba et al., 2016) was not used in the injections. The C-Cloud injection points past the first layer were to the same state space as the extraction point, after the extraction. The A-Cloud injection point was the state space following the extraction point. The extractions from both clouds utilized dense linear projections, without masking, to bridge the wide amorphous latent spaces back to the dimensionality of the residual stream. These projected signals were added directly to the residual signal $X$ to complete the residual connection, without the application of activation functions or layer normalization at the exit boundary.

This structure created a workbench to test whether amorphous clouds could freely form to replace the functionality of the discrete SA and FFN operators. The workbench was configured with four injection and extraction points per cloud, representing an equivalent test structure of four transformer layers. The token dimension was set to 128, sequence length to 128, C-Cloud width to 256, and A-Cloud width to 128. This yielded approximately 1.4 million trainable parameters in the backbone cloud structure, or 7.8 million total trainable parameters when including the token embeddings. The baseline standard transformer was given 7 layers and an FFN width of 512 to equalize the nominal trainable parameter count.

## 3 Operation and Results of the Dual-Cloud Architecture

The dual-cloud and the 7-layer standard transformers were trained on the WikiText-103 dataset for 50 epochs across 10 runs each, using identical seed sets for both architectures. The lowest validation loss points for each run were recorded, and the 10-run average minimum was computed. The standard 7-layer transformer achieved an average minimum validation loss of 4.5189 (perplexity 91.73). The 4-layer dual-cloud transformer achieved an average minimum validation loss of 4.4512 (perplexity 85.73). This empirical test validates the use of AmNN clouds as substitutes for discrete neural network operators within an interconnected system.

During data capture for the dual-cloud transformer, mean activation-weighted signal strength values, calculated as:

$$Ratio = \frac{\text{mean}(|X_{upstream}| \cdot |W_{upstream}|)}{\text{mean}(|X_{injection}| \cdot |W_{injection}|)}$$

were recorded at two key boundaries per layer: the entry intersection (comparing new cloud injection to continuous upstream flow) and the exit boundary (comparing internal cloud flow to the final extraction). The most significant diagnostic data emerged from the entry intersection. Figure 2 displays the ten-run mean of Upstream-to-Injection ratios at Epoch 50, along with individual ratios from the first three runs:

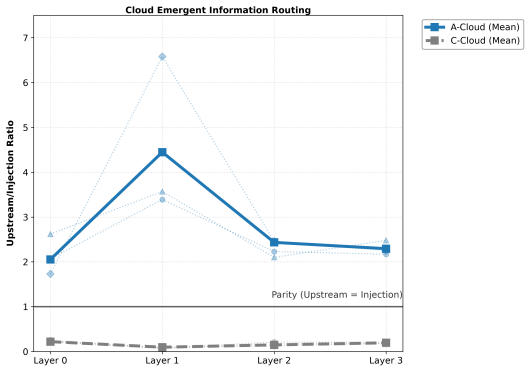


Figure 2: Upstream-to-Injection weight ratios for the Amorphous A-Cloud and C-Cloud across ten runs at Epoch 50. Bold lines represent the mean, while faint dotted lines represent data from three individual runs. The A-Cloud demonstrates a consistent, learned preference for upstream latent context over residual stream injection.

In each run, the self-attention processing in the A-Cloud at each layer weighted the continuous upstream fluidic data heavier than the state injection from the residual stream, $X$. Notably, over ten runs, the second layer weighted upstream fluidic data on average over four times heavier than the state injection from the residual stream.

This demonstrates the unconstrained amorphous cloud dynamically generating its own routing structure. Across all ten runs, the consistent natural internal structuring during training showed a demand for latent upstream information (early-layer "memories") during self-attention. The C-Cloud also showed a desire for upstream data, but not as a preference over injected data. In these tests, the AmNN clouds were a powerful diagnostic tool confirming structural inefficiency in the standard Transformer.

## 4 Building a Better Transformer

The primary requirement for translating the insights of the dual-cloud analysis into a practical transformer is that the modifications must be minimally invasive and compatible with highly optimized, modern hardware. Inserting a continuous, unconstrained amorphous structure directly into standard production pipelines is not a viable deep learning language model option for today's industry.

### 4.1 Memory Highways

Arriving at the final, data-guided transformer design was a two-step process. The initial approach (Model I) sought to directly replicate the dual continuous information "highways" observed in the A-Cloud and C-Cloud. This first discrete model added two parallel, unnormalized memory streams as a translation of flow observed in the A- and C-Clouds. These streams were constructed from latent data projected directly from the residual signal, $X$, with one stream providing unmitigated context to the self-attention operators, and the other to the FFN operators. This dual-highway architecture is illustrated in Figure 3.

Compared to a standard transformer, this architecture introduces a minor per-layer parameter overhead from the linear projections into the memory highways, alongside an increased VRAM requirement to maintain the dual highway states in memory. To enforce strict parameter equalization with the baseline, this dual-highway transformer was constrained to 6 layers. This configuration contained approximately 1.4 million core backbone parameters and 7.8 million total trainable parameters.

The average minimum validation loss reached across 10 runs of 50 epochs each was 4.4462 (perplexity 85.30). With one less layer, the dual highway transformer outperformed the 7-layer standard transformer baseline (validation loss 4.5189, perplexity 91.73).

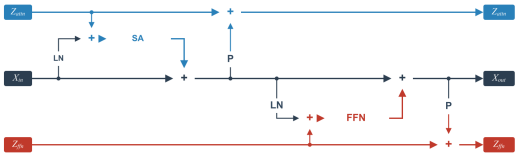


Figure 3: Transformer with dual memory highways

### 4.2 The Unified Memory Highway

The mathematically similar dual highways were combined into a single memory highway $\mathrm{Z_{mem}}$ to feed both the self-attention and FFN operators. At origin, the memory bus $\mathrm{Z_{mem}}$ is initialized as a clone of the input stream $X$. A first configuration included memory projections from the residual signal path after each of the discrete operators were added to $\mathrm{Z_{mem}}$.

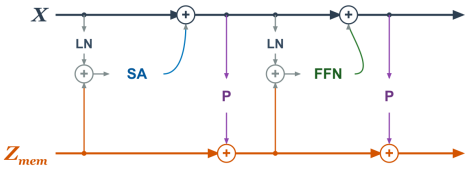


Figure 4: Unified Highway Transformer Layer

As the two memory streams did in the prior example, the unified memory preserves unmitigated context in latent states.

Before executing the Self-Attention (SA) operation, the normalized primary stream is summed with the current state of the memory stream. This provides the discrete operator with direct access to the preserved $Z_{mem}$ upstream flow:

$$H_{SA}^{(l)} = \mathrm{LN}\left(X^{(l)}\right) + Z_{mem}^{(l)}$$

$$X_{mid}^{(l)} = X^{(l)} + \mathrm{SA}\left(H_{SA}^{(l)}\right)$$

Following the SA block, the network executes the first memory projection P (linear projection followed by LeakyRELU activation (Maas et al., 2013)) to the unified highway. Projection P extracts and transforms latent states from the updated residual stream and strictly adds the result into the memory stream:

$$Z_{mem,mid}^{(l)} = Z_{mem}^{(l)} + \mathrm{LeakyReLU}\left(P_{SA}\left(X_{mid}^{(l)}\right)\right)$$

Accumulating LeakyReLU outputs may introduce a positive mean drift and variance expansion into the unnormalized $Z_{mem}$ stream. The stability shown during smaller scale test indicates that the network compensated for this drift through learned weight scaling and stabilization during early epochs from a Dynamic Variance Annealer. In a deep layer scaled test ($d = 768$), the activation function for P was changed to a zero-mean ShiftedLeakyReLU (Hendrycks & Gimpel, 2016) to prevent uncontrolled variance growth.

This aggregated context is then carried forward into the Feed-Forward Network (FFN) phase:

$$H_{FFN}^{(l)} = \mathrm{LN}\left(X_{mid}^{(l)}\right) + Z_{mem,mid}^{(l)}$$

$$X^{(l+1)} = X_{mid}^{(l)} + \mathrm{FFN}\left(H_{FFN}^{(l)}\right)$$

Finally, a second memory projection P operation captures the updated $X$ signal to inject the transformed state back into the memory stream before passing to the next layer:

$$Z_{mem}^{(l+1)} = Z_{mem,mid}^{(l)} + \mathrm{LeakyReLU}\left(P_{FFN}\left(X^{(l+1)}\right)\right)$$

This design fundamentally differs from standard residual skip connections. Stateless Skips: Standard residual connections are strictly additive pipelines that bypass discrete operators but are ultimately subjected to ablation via layer normalization at the next depth step. Stateful Folding: The memory stream acts as an independent, continuous accumulator. Because $Z_{mem}$ bypasses layer normalization entirely, the memory projection operator P iteratively builds a representation of the sequence's history. Measurements during deep layer tests show the resulting $Z_{mem}$ signal is highly orthogonal to the residual stream. By replacing the rigid, serialized flow with this interconnected highway, the architecture provides discrete operators with the continuous context routing observed in the AmNN diagnostic models.

### 4.3 Testing and Results of the Unified Highway Transformer

The Unified Highway Transformer (UHT) was evaluated across progressively deep configurations and one scale test, all on WikiText-103. Test 1 evaluated a 6-layer UHT against a 7-layer parameter-equivalent standard Pre-LN transformer for 50 epochs; Test 2 evaluated a 17-layer UHT against a 20-layer parameter-equivalent baseline for 100 epochs; Test 3 evaluated an 80-layer $d = 64$ UHT against an 80-layer standard transformer for 50 epochs; Test 4 evaluated a $d = 64$ 120-layer UHT against a 120-layer standard baseline; and scale Test 5 evaluates a $d = 768$ 120 layer UHT against a 120-layer standard baseline. In all tests, identical weight initialization seeds were used across architectures to isolate topological impacts. Across all test scales, the UHT achieved a lower validation loss and perplexity than the standard baseline.

Shallow and Moderate Depths (Tests 1 & 2). In Test 1 (6 layer UHT, 10 runs each), the UHT reached an average validation loss of 4.4256 (perplexity 85.30), compared to 4.5189 (perplexity 91.73) for the standard transformer. At this depth, the $Z_{mem}$ highway was updated with projections from the residual signal without any scaling, normalization, or clamping. Uncontrolled variance growth was not observed at this shallow scale.

In Test 2 (17 layer UHT, 3 runs each), an ablation run utilized a static scaler on the memory projections to address early signs of variance expansion. The UHT reached an average validation loss of 4.003 (perplexity 54.78), compared to 4.0477 (perplexity 57.26) for the baseline. In the final configuration, the memory projections were strided (every other layer) and a Dynamic Variance Annealer was introduced to scale the memory projections for the first five epochs, progressing to a 1.0 scaling factor. The resulting average low validation loss was 3.8779 (perplexity 48.32).

An ablation run of the UHT in Test 2 evaluates whether the signals on the $Z_{mem}$ bus contain both latent states and sufficient residual signal states together to feed the self-attention operators directly. If true, the memory projections are redundant. To test this hypothesis, an evaluation model (Model M) was configured to eliminate the separate memory projections and rout the FFN operator output directly to the $Z_{mem}$ bus. By eliminating the memory projection matrices, this modification has zero overhead relative to the standard transformer. In addition, the layer normalization was removed from the FFN operator and replaced with

the Dynamic Variance Annealer, configured with a final scaling divisor of $\sqrt{1.3}$. To maintain parameter equivalence with the baseline, this modified UHT was configured with 20 layers. It was tested over 100 epochs and given a higher max norm of 2.5. In a single run test, this modified UHT reached a low validation loss of 3.8320 (perplexity 46.15).

The 80-Layer Scaling Test and Dynamic Variance Annealing (Test 3). Given the variance observed in Test 2, Test 3 (3 runs each) was designed as a strict stress test of scalability. To allow rapid iteration, the token dimension was bottlenecked to $d = 64$. This intentionally forced the learning into a highly compressed vector space, may naturally exacerbate variance expansion. Gradient clipping was relaxed ($max_norm = 2.0$).

Initial attempts to train the 80-layer UHT revealed that static scaling was insufficient to control early-epoch variance. The Dynamic Variance Annealer was added to progressively anneal the scale factor for the memory projections over the first 5 epochs. This created a highly stable $Z_{mem}$ signal across all three runs. The stabilized UHT reached an average low validation loss of 4.2137 (perplexity 67.61), compared to an average low of 4.7422 (perplexity 114.69) for the 80-layer standard transformer. Because this test was conducted at Iso-Layer rather than Iso-Parameter depth, the UHT carried a 5.6% total trainable parameter (7.5% backbone) overhead.

More significant than the validation loss differential is the time-to-capability acceleration. In Test 3, the lowest validation loss achieved by the standard baseline was 4.6804 at Epoch 46. The UHT surpassed this capability limit at Epoch 18 in all three runs. Even penalizing for the 7.5% backbone overhead, the UHT demonstrated a meaningful Time-to-Capability acceleration.

Extreme Depth and Latent Amnesia (Test 4). Test 4 was a direct scale up of the 80 layer ($d = 64$) test. At 120 layers, the UHT had a 5.4% overhead. To absorb initialization shocks, the Dynamic Variance Annealer began with an aggressive residual scaling divisor of $\sqrt{8L}$. Over the first 5 epochs, both the scalar multiplier and the layer-count dependency ($L$) were decayed until the divisor reached exactly 1.0. This allowed full memory projection signals to accumulate on the memory highway, and forced the network to natively self-regulate its variance for the remainder of training. Due to latent amnesia, the standard Pre-LN transformer would be expected to perform only marginally better or worse than the 80 layer standard Pre-LN transformer in Test 3. To show that the memory highway efficiently addresses latent amnesia at extreme depth, the 120 layer UHT would need to outperform the 80 layer UHT.

Initialization and the Illusion of Convergence. The standard 120-layer Pre-LN baseline successfully survived initialization without experiencing numerical detonation (NaNs). During the first two epochs, the baseline demonstrated steady convergence, tracking closely with the UHT. This initial descent is likely driven entirely by the network's shallowest layers, effectively masking gradient stagnation in deeper layers.

Observation of Latent Amnesia and Structural Uncoupling. Monitoring the gradient decile profiles across all 120 layers revealed patterns associated with the rapid onset of Latent Amnesia in the baseline architecture. The standard Pre-LN model exhibited severe structural uncoupling. The gradient magnitudes in the deepest 84 layers flatlined near ~0.05. This indicated that deep Pre-LN layers become mathematically muted, forcing the network to operate with the effective capacity of a much shallower model.

Dynamic Routing and Scale-Free Stability. In contrast, the strided UHT bypassed the sequential dilution of LayerNorm efficiently. The UHT stabilized its unnormalized memory highway variance after the annealer was complete. More importantly, the $O(1)$ routing of the $Z_{mem}$ highway produced a healthy, non-monotonic "wave-like" gradient flow. Instead of pooling at the bottom of the stack, the UHT dynamically distributed compute, maintaining a strong foundational anchor at Layer 0 while spontaneously forming highly active, deep representational hubs.

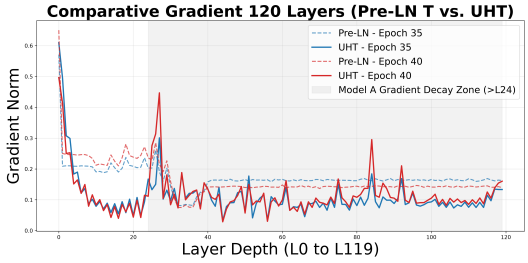


Figure 5: Gradient activity through 120 layers; the standard Pre-LN Transformer shows gradient stagnation in deep layers that is overcome in the UHT

Pushing the Extreme. To verify the capacity limits of the 64-dimension configuration, and to exercise the UHT at further depth, the UHT was configured with 180-Layers for three test runs at different seeds. Figure 6 illustrates that this UHT also showed dynamic gradients at deep layers and verified the stability provided by the Dynamic Variance Annealer.

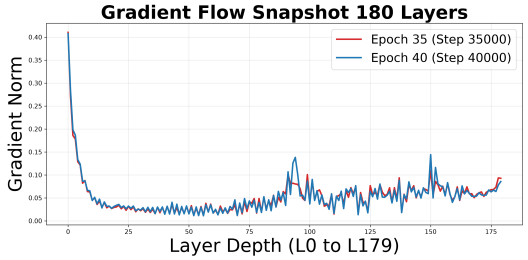


Figure 6: UHT Gradient activity through 180 layers; deep layer gradient stagnation is not observed

Results. In the 120 layer test, the baseline standard transformer prematurely slowed its ability to learn complex semantic dependencies, and achieved an average low validation loss of 4.7693 (Perplexity: 118). In contrast, the UHT easily bypassing the Pre-LN's performance and outperforming the 80 layer UHT, reaching an average low validation loss of 4.1950 (Perplexity: 66). Intersection flow measured on a 120 layer UHT test showed a deep layers shift to $> 99.9\%$ read reliance on $Z_{mem}$. A UHT ablation test, which completely removed the residual stream input to the self-attention operation and replaced the layer normalization of the input to the FFN with the scale function provided by the Dynamic Variance Annealer, reached an average low validation loss of 4.2183 (Perplexity 68). This implies that the $Z_{mem}$ signal contains a rich mixture of spine and latent data sufficient for self-attention operation, and opens the door to save resources through elimination of a Layer Normalization operation.

The 180-layer UHT showed similar results as the 120-layer test, reaching a low validation loss of 4.2834 (perplexity 72.49) over three runs.

The 120-Layer $d = 768$ Scaling Test (Test 5). While the WikiText-103 dataset is too small to exercise the full language learning capability of a 910 million parameter transformer, it serves as a diagnostic stress test to demonstrate information flow during training. The Pre-LN transformer's FFN dimension was increased ($ffn_dim = 3168$) to equalize parameters with the UHT. The UHT's combination of the larger model dimension ($d = 768$) and the configurations used in Tests 3 and 4 left the $Z_{mem}$ highway vulnerable to uncontrolled variance growth. Replacing LeakyReLU with ShiftedLeakyRELU in the memory projections and replacing GELU in the FFN operator also with ShiftedLeakyRELU helped stabilize the $Z_{mem}$ stream. Using Figure 2 as a guide, the $Z_{mem}$ contribution to the self-attention input was limited to 4x the size of the normalized residual stream input and the $Z_{mem}$ contribution to the FFN input was limited to 1x the size of the normalized residual stream input. In addition, the $Z_{mem}$ highway was decayed by 0.90 prior to each memory projection injection,

the injection stride was lengthened from 2 to 4, and the QK vectors were normalized. The UHT and the Pre-LN Transformer were each evaluated in a single run using the same seed.

Like in the prior tests, the UHT outperformed the Pre-LN Transformer reaching a low validation loss of 4.3773 (perplexity 79.62) compared to the baseline's low of 4.6279 (perplexity 102.30). More interesting is model behavior. The Pre-LN transformer layer update symmetry showed quick layer die-off (as measured by active logits at the self-attention input), with almost 85% of layer self-attention operators non-functioning by epoch 10 and over 87% of the layer self-attention operators non-functioning by epoch 35. In contrast, the UHT showed no layer die-off due to self-attention until close to epoch 30 and 40% layers functioning at the final epoch. Given the small size of the test dataset in this much larger Transformer, it is not surprising that eventually at least some layers would become pass-throughs. Also of note in the UHT, the $Z_{mem}$ alignment to the residual stream signal reached near orthogonality in deep layers. And more surprisingly, during the run, layers that had died off (by reaching a pass through state) periodically awoke and began contributing to the residual signal.

## 5 Conclusion and Future Work

This work demonstrates that the rigid, serialized information flow of standard discrete-operator deep learning systems may have inherent inefficiencies that can be identified using Amorphous Neural Network (AmNN) diagnostic clouds. When substituted into a Pre-LN Transformer, the AmNN clouds reveal demand for upstream latent states that would otherwise get trivialized through Layer Normalization.

Learnings from the diagnostic clouds led to the engineering of the UHT, which introduces the $O(1)$ parallel memory bus that accumulates and routes latent context directly each layer. The UHT outperformed its baseline Pre-LN counterpart in every test. In a scaled test of a small dataset at $d = 768$ the UHT preserved functionality through network depth long into training, while the baseline transformer quickly experienced layer die-off.

Limitations and Future Work. Having empirically validated the UHT's ability to improve Transformer performance and cure gradient stagnation in a bottlenecked ($d_{model} = 64$) model and demonstrating operation with a small dataset on an industrial scaled model ($d = 768$), validating the UHT on a full dataset at industrial dimensions remains an open empirical question. Future work will investigate the integration of the UHT design across varying transformer architectures and scaling of the Test 2 ablation that reroutes the FFN output and eliminates the separate memory projection. Further work will explore adapting the $O(1)$ memory bus beyond language modeling and continuing to optimize the architecture for even greater efficiency. Future work on AmNN diagnostics will address scaling limits of interfacing AmNN clouds into deeper layer diagnostic workbenches. Ultimately, this research proves that Amorphous Networks are not merely theoretical novelties. They are invaluable analytical tools capable of diagnosing hidden architectural bottlenecks and guiding the design of highly effective, hardware-sympathetic improvements for industrial AI systems.

AI use statement

An LLM (Gemini Pro) was utilized as a research assistant to generate initial Python code for the experiment baseline under the explicit structural and algorithmic specifications provided by the authors. All generated code was manually reviewed, verified, and debugged by the author to ensure scientific and mathematical accuracy. The LLM was not allowed access to the code development environment or the document drafting environment. The LLM was also used as a writing assistant to suggest alternative phrasing for selected sections and to assist with general text cleanup and readability improvements. Every AI-generated suggestion was thoroughly reviewed, critically evaluated, and completely rewritten by the human author. The final content, arguments, and claims presented in this work remain entirely the original contribution of the author, who accepts full responsibility for its integrity and accuracy.

Reproducibility statement

Python implementations of the cloud diagnostic workbench and of the unified Highway Transformer (UHT) are provided as an anonymous .zip archive to replicate primary claims.

## References

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. arXiv preprint arXiv:1607.06450, 2016.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415, 2016.

Kimi Team, Guangyu Chen, Yu Zhang, Jianlin Su, Weixin Xu, Siyuan Pan, Yaoyu Wang, et al. Attention residuals. arXiv preprint arXiv:2603.15031, 2026. URL https://doi.org/10.48550/arXiv.2603.15031.

Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In Proc. icml, volume 30, pp. 3. Atlanta, Georgia, USA, 2013.

Rob Manson. What makes LLMs so fragile (and brilliant)? Flux by Rob Manson (blog), December 1, 2025, 2025. URL https://flux.robman.fyi/p/what-makes-llms-so-fragile-and-brilliant.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. arXiv preprint arXiv:1609.07843, 2016.

Anthony Luke Simon. Amorphous neural networks, 2026. U.S. Patent Application Publication No. US 2026/0252856 A1. Published August 27, 2026.

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, 2017.

Hongyu Wang, Shuming Ma, Li Dong, Shaohan Huang, Dongdong Zhang, and Furu Wei. Deepnet: Scaling transformers to 1,000 layers. arXiv preprint arXiv:2203.00555, 2022. URL https://doi.org/10.48550/arXiv.2203.00555.

Shufang Xie, Huishuai Zhang, Junliang Guo, Xu Tan, Jiang Bian, Hany Hassan Awadalla, Arul Menezes, Tao Qin, and Rui Yan. Residual: Transformer with dual residual connections. arXiv preprint arXiv:2304.14802, 2023. URL https://doi.org/10.48550/arXiv.2304.14802.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. On layer normalization in the transformer architecture. In International Conference on Machine Learning, pp. 10524–10533. PMLR, 2020.