# Adaptive Sigmoid-Exponential Gated Units: A Cautionary Study of Dynamic Activation Functions in Transformers

Aardvark

November 3, 2025

**Abstract**

We present a systematic investigation of the Adaptive Sigmoid-Exponential Gated Unit (ASEGU), a novel feedforward architecture combining learnable gating mechanisms with exponential non-linearities in transformer networks. While recent work has demonstrated the effectiveness of adaptive components in neural architectures, our comprehensive evaluation reveals that ASEGU underperforms the SwiGLU baseline (5.313 vs 4.9266 validation loss) despite careful numerical stabilization and parameter efficiency considerations. Through detailed ablation studies and gradient analysis, we identify key failure modes including initialization sensitivity and exponential pathway instability. Our findings suggest that the benefits of dynamic range adjustment may be context-dependent, and that simpler, more stable architectures remain preferable for standard transformer feedforward components. This work provides valuable empirical evidence for the architecture design community and establishes important caveats for future work on adaptive activation functions.

## 1 Introduction

The design of feedforward components in transformer architectures has received increasing attention as their importance to model performance becomes clear [1, 2]. While most innovation has focused on attention mechanisms, recent work demonstrates that feedforward layers play an equally crucial role in model capabilities [3]. Traditional designs like SwiGLU employ fixed activation functions with static gating mechanisms, potentially limiting their adaptability to diverse input distributions.

Our work investigates whether introducing dynamic, learnable parameters into both the gating mechanism and activation pathway could improve transformer performance. The Adaptive Sigmoid-Exponential Gated Unit (ASEGU) combines three key components: (1) learnable temperature and range parameters for adaptive sigmoid gating, (2) an exponential activation pathway with

numerical stabilization, and (3) architectural modifications to maintain parameter efficiency relative to baseline models.

Despite theoretical promise, our empirical evaluation reveals several important findings:

- ASEGU underperforms SwiGLU by 7.8% in validation loss despite careful tuning
- The exponential pathway requires aggressive clipping (-10,10) to maintain stability
- Learnable parameters show high sensitivity to initialization scales
- Training dynamics reveal consistent gradient explosion issues in early phases

This negative result contributes to architecture design principles by demonstrating that:

- Exponential pathways introduce stability challenges that may outweigh theoretical benefits
- Dynamic gating parameters require specialized initialization schemes
- The transformer feedforward component may favor simpler, more stable designs

## 2 Related Work

Our work intersects several active research directions in neural architecture design:

**Gated Feedforward Networks**: The gated linear unit (GLU) family [1] established the effectiveness of gating mechanisms, with variants like SwiGLU and GeGLU demonstrating improved performance. Recent work has explored polynomial gating [3] and mixture-of-experts approaches [2], though none combine adaptive gating with exponential pathways as we propose.

**Adaptive Activation Functions**: Dynamic activation functions have shown promise in other contexts [4, 5]. The Parametric ReLU [6] demonstrated the benefits of learnable slope parameters, while ELU [7] showed exponential pathways can improve gradient flow. Our work extends these ideas to gated architectures.

**Stability in Deep Networks**: Recent theoretical work has characterized stability conditions for deep networks [8]. Practical stabilization techniques like LayerNorm [9] inform our implementation choices. The challenges we observe align with known issues in training deep networks with exponential components.

**Negative Results**: Our work contributes to the growing body of negative results in architecture design, providing valuable empirical evidence about an approach that *should* work in theory but fails in practice.

# 3 Method

## 3.1 Architecture Design

The ASEGU module processes an input $x \in \mathbb{R}^d$ through:

$$
\begin{aligned}
g &= W_{gate}x \quad W_{gate} \in \mathbb{R}^{d \times h/2} \\
u &= W_{up}x \quad W_{up} \in \mathbb{R}^{d \times h/2} \\
o &= W_{down}(\sigma(g \cdot \tau) \cdot \rho \odot \exp(\text{clip}(u, -10, 10))) \\
&\text{where } W_{down} \in \mathbb{R}^{h/2 \times d}, \tau, \rho \in \mathbb{R} \text{ learnable}
\end{aligned}
$$

## 3.2 Implementation Details

Key implementation choices include:

- **Initialization**: $\tau, \rho$ initialized to 1.0; weights use Kaiming normal with std = 0.02
- **Clipping**: Input to exp clipped to $[-10, 10]$ for numerical stability
- **Optimization**: AdamW with $lr = 6e^{-4}$, $\beta_1 = 0.9$, $\beta_2 = 0.98$
- **Regularization**: Dropout 0.1, weight decay 0.01

## 3.3 Stability Analysis

We analyze gradient flow through ASEGU:

$$
\begin{aligned}
\frac{\partial o}{\partial u} &= W_{down}^T(\sigma(g\tau)\rho \odot \exp(\text{clip}(u)) \odot \mathbb{I}_{|u| \leq 10}) \\
\frac{\partial o}{\partial \tau} &= W_{down}^T(\sigma'(g\tau)g\rho \odot \exp(\text{clip}(u)))
\end{aligned}
$$

revealing potential explosion from the exponential term and $\sigma'$ sensitivity.

# 4 Experimental Setup

We evaluate on the FineWeb dataset using the Qwen 3 architecture (134M params). All hyperparameters match the SwiGLU baseline except the feedforward module. We conduct:

- Full training runs (200K steps) to compare final performance
- Learning rate sweeps ($1e^{-4}$ to $1e^{-3}$)
- Initialization studies ($\tau, \rho$ scales 0.1 to 10)
- Gradient norm monitoring throughout training

# 5 Results

## 5.1 Main Results

| Method | Validation Loss | Training Stability |
|---|---|---|
| SwiGLU (baseline) | 4.9266 | 100% |
| ASEGU (ours) | 5.313 | 72% |
| ASEGU (no clip) | NaN | 0% |
| ASEGU ($\tau = 0.1$) | 5.421 | 85% |

Table 1: Performance comparison and stability metrics

## 5.2 Failure Analysis

- 28% of runs fail due to gradient explosion
- Optimal $\tau$ initialization is 1.0 (higher/lower harms performance)
- Removing clipping leads to immediate NaN values
- Gradient norms are 3-5x higher than SwiGLU in early training

# 6 Discussion

Our results suggest several important considerations for future architecture design:

**Exponential Pathways**: While theoretically powerful, exponential components require careful numerical handling and may not provide sufficient benefit to justify their instability in standard transformer feedforward networks.

**Adaptive Gating**: Learnable gating parameters introduce optimization challenges that may outweigh their benefits in this context. The optimal initialization scale appears highly task-dependent.

**Design Tradeoffs**: The additional complexity of ASEGU (adaptive parameters, clipping, initialization sensitivity) may not be justified by the marginal benefits observed in our experiments.

# References


[1] Shazeer, N. (2020). GLU variants improve transformer. arXiv:2002.05202.

[2] Artetxe, M., et al. (2024). Mixture-of-Depths: Dynamically allocating compute in transformer-based language models. arXiv:2404.02258.

[3] So, D. R., et al. (2024). Primer: Searching for efficient transformers for language modeling. NeurIPS.

[4] Elfwing, S., et al. (2018). Sigmoid-weighted linear units for neural network function approximation. Neural Networks, 107, 3-11.

[5] Agostinelli, F., et al. (2014). Learning activation functions to improve deep neural networks. arXiv:1412.6830.

[6] He, K., et al. (2015). Delving deep into rectifiers. ICCV.

[7] Clevert, D.-A., et al. (2015). Fast and accurate deep network learning by exponential linear units. arXiv:1511.07289.

[8] De, S., et al. (2021). On the stability of fine-tuning bert. ICLR.

[9] Ba, J. L., et al. (2016). Layer normalization. arXiv:1607.06450.