DynamicGate-MLP: A Gated Sparse Neural Network Architecture

Yong il Choi
Sorynorydotcom Co., Ltd.
hurstchoi@sorynory.com

October 29, 2025

Abstract

We propose **DynamicGate-MLP**, a novel multilayer perceptron (MLP) architecture designed to improve computational efficiency by dynamically gating individual connections. Traditional MLPs keep all input–hidden–output connections active, resulting in redundant computation and memory usage. DynamicGate-MLP introduces **gate parameters (gate logits)** associated with each weight, enabling the network to learn the importance of each connection during training. In the forward pass, connections are selectively activated through **hard gates** generated from sigmoid probabilities with a threshold, while in the backward pass, the **Straight-Through Estimator (STE)** ensures gradients flow through continuous probabilities. On the MNIST dataset, DynamicGate-MLP maintains comparable accuracy to a baseline MLP while reducing active connections and multiply-accumulate operations (MACs) by up to 70%. This approach contributes to efficient inference and improved interpretability of MLP-based models.

1. Introduction

Deep learning models have demonstrated remarkable performance across domains such as image recognition, speech processing, and natural language understanding. However, their high computational and memory requirements remain a major limitation. Multilayer perceptrons (MLPs), while structurally simple, serve as a fundamental benchmark but are inherently **dense** architectures where all connections are always active.

Numerous approaches have been explored to reduce computational cost. **Dropout** randomly deactivates connections during training for regularization but does not reduce inference cost. **Pruning** removes unimportant connections after training, improving inference efficiency but failing to reflect sparsity during training.

This paper introduces **DynamicGate-MLP**, which applies trainable gates at the level of individual connections. The model learns to deactivate unnecessary connections dynamically during training and converges to an efficient sparse structure for inference.

2. Related Work

- **Dropout**: Improves generalization by stochastic connection removal during training, but all weights are active during inference.
- **Pruning**: Post-training connection removal improves inference efficiency but lacks training-time sparsity.
- **L0 Regularization / Lottery Ticket Hypothesis**: Explore sparse sub-networks, but often involve complex or unstable training.
- **Dynamic Neural Networks**: Conditional execution and adaptive computation have been studied, but many approaches focus on layer-level gating rather than fine-grained connection-level control.

DynamicGate-MLP unifies these directions by offering **connection-level trainable gating** with efficient inference capability.

1 Methodology

1.1 Architecture

Given input $x \in \mathbb{R}^{B\times d}$, weight matrix $W \in \mathbb{R}^{out\times in}$, we introduce gate parameters $gate\_logit \in \mathbb{R}^{out\times in}$.

1.2 Gate Computation

$$G_{prob} = \sigma(gate\_logit), \quad G_{hard} = \mathbf{1}[G_{prob} > \tau]$$

Forward propagation uses effective weights:

$$W_{eff} = W \odot G_{hard}$$

1.3 Straight-Through Estimator (STE)

$$G = G_{hard} + (G_{prob} - G_{prob}.detach())$$

This ensures that the forward pass uses binary masks while the backward pass propagates gradients through continuous values.

1.4 Loss Function

$$\mathcal{L} = \mathcal{L}_{CE} + \beta \cdot \big(mean(G_{prob})\big)$$

The cross-entropy loss is combined with an L1-style penalty on gate probabilities to encourage sparsity.

2 Experiments

2.1 Dataset

MNIST handwritten digit dataset (28×28 grayscale images).

2.2 Setup

Baseline MLP: 784 $\rightarrow$ 256 $\rightarrow$ 10. DynamicGate-MLP: same architecture + gate parameters. Optimizer: Adam (lr=1e-3). Training epochs: 50.

2.3 Results

- Accuracy: Comparable to baseline (∼98%)
- Active connection ratio r: ∼0.3–0.5
- Inference MACs: 30–50% of baseline
- Parameters: Reduced to ∼60k–100k after pruning (vs. 203k baseline)

3 Discussion

During training, gate logits increase parameter count and computation relative to the baseline MLP. During inference, only hard gate masks remain, allowing pruning and conversion into a smaller dense MLP. Unlike dropout, DynamicGate-MLP achieves actual inference efficiency. Unlike pruning, sparsity is considered during training. The trade-off between accuracy and computational savings depends on β (regularization strength) and τ (threshold).

4 Conclusion

We presented DynamicGate-MLP, a novel gated sparse MLP architecture that dynamically learns to deactivate unnecessary connections. Experiments on MNIST demonstrate that the model achieves comparable accuracy to a baseline MLP while reducing active connections and inference cost substantially. Future work will extend this approach to larger datasets (CIFAR-10/100, ImageNet), explore gate scheduling strategies, and evaluate real-world inference speedups on optimized hardware.

References

[1] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[2] S. Han, J. Pool, J. Tran, and W. J. Dally. Learning both weights and connections for efficient neural network. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

[3] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991.

[4] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*, 2017.

[5] J. Frankle and M. Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[7] W. Fedus, B. Zoph, and N. Shazeer. Switch Transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2021.

[8] X. Wang, F. Yu, Z. Dou, T. Darrell, and J. E. Gonzalez. SkipNet: Learning dynamic routing in convolutional networks. In *European Conference on Computer Vision (ECCV)*, 2018.

[9] Y. Chen, Y. Dai, M. Liu, D. Chen, L. Yuan, and N. Vasconcelos. Dynamic convolution: Attention over convolution kernels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.