**DynamicGate-MLP: A Gated Sparse Neural Network Architecture**

**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.

**3. Methodology: DynamicGate-MLP**

**3.1 Architecture**

* Input:
* Weights:
* Gate parameters:

**3.2 Gate Computation**

Forward propagation uses effective weights:

**3.3 Straight-Through Estimator (STE)**

To enable gradient flow through discrete gates:

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

**3.4 Loss Function**

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

**4. Experiments**

**4.1 Dataset**

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

**4.2 Setup**

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

**4.3 Results**

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

**5. 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.
* Trade-off: Accuracy vs. computational savings depends on (regularization strength) and (threshold).

**6. 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.