

Differentiable Logic Gate Networks for Low-Latency EEG Classification on Edge Devices

Shyamal Y. Dharia[†], Stephen D. Smith^{†,‡}, Camilo E. Valderrama^{†, ◊, §*}

[†] The University of Winnipeg, Department of Applied Computer Science

[‡] The University of Winnipeg, Department of Psychology

[◊] University of Manitoba, Department of Electrical and Computer Engineering

[§] University of Calgary, Department of Community Health Sciences

Abstract

Real-time EEG classification on edge devices is bottlenecked by the floating-point arithmetic of conventional neural networks. We investigated Differentiable Logic Gate Networks (Diff-Logic) as a hardware-native alternative that compiles models into pure Boolean circuits executable via bitwise CPU operations. Through rigorous iso-parameter experiments across four EEG datasets spanning two classification tasks, binary dementia detection and 3-class emotion recognition, we compared Diff-Logic against matched-capacity Multi-Layer Perceptron (MLP) and Binarized Neural Network (BNN) baselines at four complexity tiers (50k–500k parameters). On dementia screening, Diff-Logic achieved 80.2% Macro F1, outperforming the MLP baseline by 6.8%. On emotion recognition, the MLP retained a moderate performance advantage but incurred a 2.3× higher latency and 14× larger model size when deployed on a power-constrained (7W) Nvidia Jetson Orin Nano CPU (Single-core). Critically, Diff-Logic inference time remained nearly constant across a 10× increase in model scale, achieving a peak speedup of 2.9× over MLPs at the largest complexity tier. Our results establish logic-based neural architectures as a practical paradigm for resource-constrained brain-computer interfaces, achieving competitive or superior performance while natively satisfying the latency and memory constraints of portable edge deployment. Code is available on GitHub: <https://github.com/Shyamal-Dharia/eeg-difflogic>.

Keywords: Differentiable Logic Gate Networks, Edge AI, Brain-Computer Interfaces

1. Introduction

Although Artificial Neural Networks (ANNs) have revolutionized pattern recognition in complex biosignals [1, 2], their evolution toward increasingly high-parameter architectures has created a significant deployment barrier [3, 4]. This barrier is present, for instance, in the brain-computer interface (BCIs) and real-time EEG monitoring, where the computational overhead of standard deep learning models often exceeds the power and thermal envelopes of portable edge devices. Consequently, practical deployments are often forced to offload processing to the cloud, thereby introducing latency that disrupts the real-time feedback loop essential for BCIs while exposing sensitive neural data to security risks [5].

To address the latency limitations of deployed deep learning models in real-time BCI systems, it is essential to move away from traditional multiply-accumulate (MAC) operations toward hardware-native formulations capable of delivering high-speed inference within microsecond-scale budgets [6]. One promising approach in this direction is Differentiable Logic Gate Networks (Diff-Logic) [7–10]. By relaxing discrete Boolean operations, such as approximating AND gates through the product of binary inputs, researchers have enabled the training of logic-based models via traditional gradient descent. Unlike conventional networks that rely on high-precision floating-point weights, these models converge into a hard, sparse network of Boolean gates. This transition from floating-point arithmetic to bitwise logic allows for extreme efficiency, as the resulting model can be executed directly

*c.valderrama.uwinnipeg.ca

on the Arithmetic Logic Unit (ALU) of edge processors, bypassing the energy-intensive floating-point bottlenecks of conventional AI accelerators.

Although Diff-Logic shows competitive accuracy on standard benchmarks, its application to real-world EEG processing under edge constraints remains unexplored. We investigate whether its theoretical efficiency translates to complex, non-stationary brain signals.

In this paper, we present the first rigorous application of Diff-Logic to the constraints of real-time EEG signal decoding. To this end, we investigate how bitwise learning dynamics respond to the high-dimensional, non-stationary nature of brain signals, comparing them with traditionally floating-point intensive models used for an EEG-based classification task. Our main contributions are summarized below:

- We show that Diff-Logic effectively encodes high-dimensional EEG features, matching or outperforming floating-point baselines in emotion recognition and dementia detection.
- We propose a pipeline from soft probabilistic training to compiled Boolean inference, achieving up to a $2.91\times$ speedup over Float32 MLPs on embedded hardware.
- We provide an iso-parameter evaluation across four EEG benchmarks (50k–500k parameters), comparing Diff-Logic against MLPs and BNNs to characterize performance-efficiency trade-offs.

2. Related Works

2.1. Deep Learning in EEG-based Brain-Computer Interfaces

The evolution of EEG decoding has transitioned from traditional feature engineering to deep representation learning. Early successes were marked by architectures such as Deep4Net [11], which used deep convolutional networks to capture spatiotemporal hierarchies in brain signals. However, the high parameter count of such models posed challenges for real-time deployment. This led to the development of compact architectures, most notably EEGNet [12], which introduced depthwise and separable convolutions to significantly reduce the computational burden while maintaining competitive accuracy across diverse BCI tasks. Despite these optimizations, recent trends have shifted toward even more complex structures, such as EEG-Conformers [13], which leverage self-attention mechanisms to model long-range temporal dependencies. Furthermore, the emergence of EEG foundation models, such as LaBraM [14], BIOT [15], and the recent DeeperBrain [16], has introduced a new paradigm of large-scale pre-training. While these models offer unprecedented generalization across subjects and tasks, they are highly parametric, often requiring millions to billions of floating-point operations (FLOPs) per inference. Even lightweight foundation models using Mamba-based linear scaling, such as LUNA [17] and FEMBA [18], remain fundamentally dependent on high-precision floating-point arithmetic. Although knowledge distillation [19] and structured pruning [20] offer partial remedies, they retain a fundamental dependence on MAC operations, thereby creating a persistent thermal and energy bottleneck for battery-powered wearable devices and leaving a critical efficiency gap for portable edge deployment.

2.2. Differentiable Logic and Hardware-Native AI

The pursuit of hardware-efficient AI has historically focused on quantization and pruning. Binarized Neural Networks (BNNs) [21] represented a significant leap, constraining weights and activations to $\{-1, +1\}$ and replacing multiplications with XNOR-popcount operations. However, BNNs retain the dense connectivity of standard deep learning architectures, requiring accumulation steps that limit throughput on embedded BCI platforms. A distinct

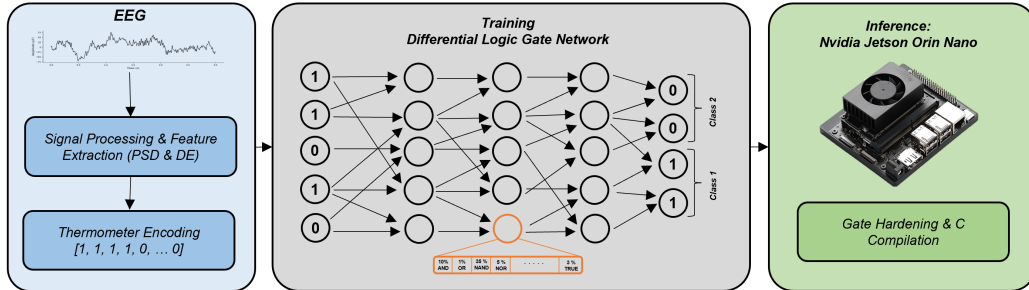


Figure 1. Proposed workflow for EEG classification using Diff-Logic.

paradigm has emerged through the work of Petersen et al. [7], who introduced *Differentiable Logic Gate Networks*. Unlike BNNs, which quantize a fixed topology, this framework learns the topology itself. By relaxing Boolean logic gates (AND, OR, XOR) into continuous differentiable operators, these networks can be trained via gradient descent to identify sparse, optimal logic circuits. Upon convergence, each gate hardens to a single Boolean function, yielding a network that requires no accumulation registers and executes entirely through bitwise logic. Recent advancements have extended this framework to convolutional structures [8] and recurrent units [9], demonstrating efficacy on tabular and image datasets (e.g., MNIST, CIFAR). However, the application of diff-logic to continuous, non-stationary physiological time-series remains unexplored. While BNNs have been applied to EEG [22], true logic gate networks, which eliminate accumulation in favour of pure bitwise operations, have not yet been rigorously evaluated for neurophysiological signal classification. This gap is particularly critical for edge computing platforms, such as the NVIDIA Jetson, where efficient CPU-bound execution is required to decouple the BCI control loop from AI accelerators, thereby reserving the GPU for concurrent, computationally intensive downstream workloads.

3. Methods

3.1. Dataset and Feature Extraction

To evaluate the robustness of Diff-Logic across clinical and affective domains, we utilized four EEG datasets spanning two distinct EEG-based applications. In total, the proposed framework was evaluated across a diverse cohort of 119 unique subjects (88 from the clinical dataset and 31 from SEEDs; see Table A1).

3.1.1. Clinical Diagnosis: Dementia

The dementia dataset [23] focuses on detecting neurodegenerative diseases using resting-state EEG from 88 subjects: 29 Cognitively Normal (CN), 36 Alzheimer’s Disease (AD), and 23 Frontotemporal Dementia (FTD). Signals were recorded at 500 Hz using a 19-channel montage (10–20 system). We extracted Power Spectral Density (PSD) features across five frequency bands ($\delta, \theta, \alpha, \beta, \gamma$), yielding a feature vector of $19 \times 5 = 95$ dimensions per 10-second non-overlapping window. The analysis was divided into two binary classification tasks: CN vs. AD (65 subjects) and CN vs. FTD (52 subjects). To ensure clinical validity, we employed subject-level 10-fold cross-validation, ensuring that no windows from the same subject appear in both training and testing partitions. Final per-subject predictions were obtained via majority voting across all constituent windows.

3.1.2. Emotion Recognition: SEED Family

We utilized the SEED dataset [24] and its cross-cultural variants (France and Germany) [25] for three-class emotion recognition (Negative, Neutral, Positive). All variants used a 62-channel setup downsampled at 200 Hz, segmented into 4-second non-overlapping windows. The primary feature was Differential Entropy (DE) across five bands, yielding a $62 \times 5 = 310$ dimensional vector per window. These datasets were evaluated using trial-level 10-fold cross-validation, where each trial corresponds to a specific film clip viewed during a session. This approach is subject-dependent, where the model learns from a subset of a subject’s trials to predict emotional state in held-out trials. Unlike clinical diagnosis, where generalization to unseen patients is critical, emotion recognition systems for BCIs often benefit from an initial on-device calibration phase [26]. Our protocol, therefore, mimics a practical deployment scenario where the model is personalized to a specific user’s EEG.

3.2. Per-Fold Feature Normalization

To ensure robust scaling of PSD and DE features without data leakage, we employed a fold-wise Min-Max normalization strategy. Scaling parameters were derived exclusively from the training partitions within each cross-validation iteration and subsequently applied to both training and test samples. In detail, we defined $\mathcal{T}_{\text{train}}$ as the training set indices for a given fold. For each feature dimension j , we computed the minimum ($\min^{(j)}$) and maximum ($\max^{(j)}$) values solely from $\mathcal{T}_{\text{train}}$. The normalized value x'_j for any sample (train or test) was calculated as:

$$x'_j = \frac{x_j - \min^{(j)}}{\max^{(j)} - \min^{(j)}}. \quad (3.1)$$

3.3. Deep Learning Architectures

Using the normalized features, we evaluated three deep learning architectures. The first architecture is the Diff-Logic. The other two architectures are baseline architectures: a multi-layer perceptron (MLP) representing traditional continuous neural networks, and a BNN representing quantized deep learning. The baseline models were used to rigorously evaluate the Diff-Logic.

3.3.1. Differentiable Logic Gate Network

Input Quantization: Thermometer Encoding. Given that the Diff-Logic operates entirely on bitwise logic, the first step was to quantize the normalized features, thus mapping continuous EEG features to binary inputs. To preserve the ordinal relationships of the continuous signal, we employed *Thermometer Encoding* (unary coding). Unlike one-hot encoding, which treats distinct quantization levels as orthogonal, thermometer encoding represents magnitude through cumulative bit activation (e.g., encoding 2 as 0011 and 3 as 0111). This allows the network to exploit the inherent ordering of signal amplitudes, a property shown to help neural networks learn highly non-linear decision boundaries [27].

The *Thermometer Encoding* discretized the continuous normalized feature x'_j into a binary vector of length T using uniformly spaced thresholds. We set $T = 15$ because, according to empirical studies [28], this T serves as a practical lower bound for deep learning, preserving task-relevant signals with negligible accuracy loss relative to Float32. For our EEG tasks, this granularity provides necessary signal resolution without excessively inflating the search space.

The procedure partitioned the feature space into $T + 1 = 16$ distinct levels. Although this encoding used 15 physical bits, it was mathematically equivalent to 4-bit quantization

($\log_2(16) = 4$), as illustrated in Table A2. The thresholds were defined as $\tau_k = \frac{k}{T+1}$ for $k \in \{0, \dots, T-1\}$. The binary representation $\mathbf{b}^{(j)} = [b_1^{(j)}, \dots, b_T^{(j)}]$ was generated via:

$$b_k^{(j)} = \mathbb{I}(x'_j \geq \tau_k) \quad (3.2)$$

where $\mathbb{I}(\cdot)$ was the indicator function. For example, if $x'_j = 0.6$, the resulting encoding was a sequence of k ones followed by $T - k$ zeros (e.g., $[1, 1, \dots, 1, 0, \dots, 0]$).

The *Thermometer Encoding* expanded input dimensionality by a factor of T : the 19-channel dementia dataset (95 features) yielded 1,425 binary inputs, while the 62-channel SEED dataset (310 features) yielded 4,650 binary inputs.

Differentiable Logic Architecture. Unlike standard neural networks that rely on linear transformations followed by non-linear activations, Diff-Logic networks are composed of layers of learnable Boolean logic gates. The core innovation lies in the *soft* logic gate, which allows gradients to flow through discrete Boolean operations during training.

In a Diff-Logic layer, each neuron n connects to two distinct inputs $(x_1, x_2) \in \{0, 1\}^2$. Therefore, there are total of $2^2 = 4$ combinations (i.e., 00, 01, 10, 11), and as each combination could be mapped to either 0 or 1, there are a total of $2^{2^2} = 16$ possible Boolean functions $f_i : \{0, 1\}^2 \rightarrow \{0, 1\}$ (e.g., AND, OR, XOR, NAND). During the training of the Diff-Logic layer, instead of selecting a fixed function initially, the neuron maintains a probability distribution $\alpha_n \in \mathbb{R}^{16}$ over all 16 possible functions (see Table A3 for relaxations and Figure 1 for illustration). The *soft* output y_{soft} is the probabilistic sum of all gate outputs:

$$y_{\text{soft}} = \sum_{i=1}^{16} \frac{\exp(\alpha_{n,i})}{\sum_j \exp(\alpha_{n,j})} \cdot f_i(x_1, x_2) \quad (3.3)$$

This formulation is fully differentiable with respect to both gate weights α and inputs \mathbf{x} , allowing the network to learn through minimizes classification loss via gradient descent algorithm. As training progresses, the distribution α_n sharpens.

Upon convergence, the network is *hardened* by permanently assigning the single function with the highest probability ($k = \arg \max_i \alpha_{n,i}$) to the neuron. The result is a static, sparse Boolean circuit that executes $y = f_k(x_1, x_2)$ using efficient bitwise operators (AND, OR, NOT, XOR), requiring no floating-point arithmetic.

Our architecture stacked multiple Logic Gate layers (Table A4) with random, sparse connectivity. To map the final Boolean outputs to class predictions, we employed a Group Sum operator, where the final layer’s bits are partitioned into C groups. The integer logit for class c is the sum of its active bits: $z_c = \sum_{i \in G_c} b_i$. This score was then passed through a temperature-scaled Softmax, defined as $p_c = \exp(z_c/\tau) / \sum_{j=1}^C \exp(z_j/\tau)$, to produce class probabilities. Following [7], we set $\tau = 30$ to sufficiently smooth the gradients during training, preventing overconfidence from the discrete integer summations.

3.3.2. MLP and BNN models

The MLP baseline consisted of 2–3 fully connected hidden layers with ReLU activations and dropout ($p = 0.3$) after each hidden layer. Crucially, MLP models received the continuous min-max normalized features directly (95 or 310 dimensions), preserving their standard operating conditions.

The BNN baseline followed the architecture of Courbariaux et al. [21]. Each hidden block consisted of a binarized linear layer, batch normalization, and a sign activation. During the forward pass, weights were binarized to $\{-1, +1\}$ via the sign function, with gradients propagated through a straight-through estimator (STE) that clips the gradient to zero when $|w| > 1$. The latent full-precision weights were retained for gradient updates. Like the MLP, BNN models received continuous normalized inputs.

4. Experimental Setup

4.1. Iso-Parameter Comparison

A core tenet of our experimental design was an *iso-parameter comparison*: at each complexity tier, all three architectures were constrained to approximately equal trainable parameter counts, ensuring that performance differences reflect representational efficiency rather than model capacity. MLP and BNN operated on min-max normalized continuous features ($n=95$ for dementia, $n=310$ for SEED), while Diff-Logic operated on thermometer-encoded binary inputs ($n=1,425$ and $n=4,650$, respectively). Hidden layer widths were chosen so that total parameter counts match across architectures at each tier, accounting for BNN’s additional BatchNorm parameters per layer. Overall, we evaluated at four tiers: 50k, 100k, 200k, and 500k parameters (Table A4 provides detailed configurations). An exception to the 50k tier was made for the SEED dataset. LogicLayers enforce a structural constraint requiring a minimum of $\lceil n_{\text{in}}/2 \rceil$ outputs per layer. Given the large binary input space for SEED, this constraint establishes a hard lower bound of $\sim 74\text{k}$ parameters for the Diff-Logic architecture, dictating the starting tier for this dataset.

4.2. Training Protocol

All models were implemented in PyTorch and trained on an Nvidia RTX A6000 using the Adam optimizer for 100 epochs with a batch size of 128. To address class imbalance, we computed inverse-frequency class weights via scikit-learn’s *balanced* mode and passed them to the cross-entropy loss function. Furthermore, the Diff-Logic requires a higher learning rate to effectively update the gate probability distributions, so we used $\eta = 0.01$ following Petersen et al. [7]. MLP and BNN baselines were trained with $\eta = 0.001$. To account for stochasticity in initialization and training dynamics, every experiment was repeated across five independent random seeds. We report results as mean \pm standard deviation. The primary evaluation metric is the *Macro F1-score*, chosen for its robustness to class imbalance compared to standard accuracy.

4.3. Edge Hardware Benchmarking

To validate real-time feasibility for portable BCI applications, we benchmarked inference performance on an NVIDIA Jetson Orin Nano (8 GB RAM, ARM Cortex-A78AE CPU, Ampere-architecture GPU, 7W TDP) [29]. This platform represents the modern embedded systems targeted for wearable neurotechnology [30]. Furthermore, all latency measurements were conducted under single-threaded CPU execution to simulate the worst-case deployment scenario typical of low-power microcontroller environments. Thread counts for PyTorch, OpenMP, and Math Kernel Library (MKL) were explicitly set to one. Each model was evaluated over 10,000 timed iterations following a 5,000-iteration warm-up phase, using a fixed batch size of 8.

For edge deployment benchmarking, all model architectures were instantiated directly on the Jetson Orin Nano. Diff-Logic models were compiled on-device based on prior work [7]. Because inference latency depends solely on gate count and connectivity, not on which Boolean functions were assigned, untrained instances yield latency measurements identical to those of trained networks, avoiding the need to compile all $5 \times 10 = 50$ trained checkpoints per tier. Boolean input arrays were passed to the compiled shared library via foreign function interface calls, and execution proceeds entirely through ALU bitwise operations without floating-point arithmetic. Finally, MLP and BNN baselines were benchmarked using ONNX Runtime (CPUExecutionProvider, single-threaded) with a batch size of 8 for fair comparison. All measurements use 10,000 timed iterations after a 5,000-iteration warmup. This

Table 1. Impact of Model Scale on Classification Performance (Macro F1-Score).

Model	Type	Params	Dementia		Emotion (SEED)		
			CN vs AD	CN vs FTD	Chinese	French	German
<i>Tier 1 (Small Scale)</i>							
MLP _{50k}	Float32	≈ 50k / 75k	74.8% ± 2.6	67.4% ± 2.8	62.3% ± 0.1	60.1% ± 1.0	54.4% ± 1.5
BNN _{50k}	Binary		55.1% ± 7.3	59.6% ± 12.0	49.3% ± 2.2	55.8% ± 2.7	55.4% ± 3.5
Diff-Logic_{50k}	Logic		79.5% ± 2.1	71.1% ± 2.8	54.8% ± 0.8	49.8% ± 1.6	43.8% ± 2.0
<i>Tier 2 (Medium Scale)</i>							
MLP _{100k}	Float32	≈ 100k	73.4% ± 4.7	75.8% ± 2.4	62.5% ± 0.5	60.2% ± 1.6	55.2% ± 0.6
BNN _{100k}	Binary		61.2% ± 7.3	57.7% ± 5.8	50.1% ± 1.1	56.3% ± 1.3	55.3% ± 2.0
Diff-Logic_{100k}	Logic		80.2% ± 1.8	79.1% ± 2.3	55.5% ± 0.3	50.6% ± 1.5	45.7% ± 2.1
<i>Tier 3 (Large Scale)</i>							
MLP _{200k}	Float32	≈ 200k	70.6% ± 2.2	70.5% ± 5.8	61.5% ± 0.3	56.0% ± 3.1	50.2% ± 2.1
BNN _{200k}	Binary		58.1% ± 7.2	58.2% ± 3.6	48.7% ± 3.2	57.9% ± 1.9	56.2% ± 2.9
Diff-Logic_{200k}	Logic		80.0% ± 3.7	76.4% ± 2.8	56.4% ± 0.7	52.9% ± 0.8	50.2% ± 1.3
<i>Tier 4 (X-Large Scale)</i>							
MLP _{500k}	Float32	≈ 500k	72.0% ± 5.1	69.8% ± 5.7	61.3% ± 0.4	56.6% ± 2.3	46.6% ± 2.8
BNN _{500k}	Binary		59.3% ± 11.0	56.0% ± 3.3	52.3% ± 0.6	57.9% ± 0.7	54.6% ± 2.2
Diff-Logic_{500k}	Logic		78.9% ± 0.7	74.1% ± 2.2	61.0% ± 0.5	54.7% ± 1.3	50.7% ± 2.0

Note: Chance-level performance for the binary dementia dataset is 50%, while the 3-class SEED datasets have a chance level of 33.3%. Parameter count 50k for dementia and 74k for SEED (Diff-Logic minimum)

Table 2. Inference Speed on Nvidia Jetson Orin Nano with Single-core CPU at 960 MHz.

Model	Runtime	Size (KB)	Dementia			Emotion (SEED)		
			F1-Score	Latency	Speedup	F1-Score	Latency	Speedup
<i>Tier 1 (Small Scale)</i>								
MLP	ONNX (Float32)	≈195	74.8%	0.107 ms	1.00×	62.3%	0.136 ms	1.00×
BNN	ONNX (Binary)	≈198	55.1%	0.178 ms	0.61×	49.3%	0.192 ms	0.70×
Diff-Logic	C-Compiled	≈52	79.5%	0.193 ms	0.56×	54.8%	0.223 ms	0.61×
<i>Tier 2 (Medium Scale)</i>								
MLP	ONNX (Float32)	≈389	73.4%	0.167 ms	1.00×	62.5%	0.165 ms	1.00×
BNN	ONNX (Binary)	≈394	61.2%	0.268 ms	0.62×	50.1%	0.231 ms	0.71×
Diff-Logic	C-Compiled	≈44	80.2%	0.191 ms	0.87×	55.5%	0.224 ms	0.74×
<i>Tier 3 (Large Scale) – The Efficiency Crossover</i>								
MLP	ONNX (Float32)	≈779	70.6%	0.277 ms	1.00×	61.5%	0.279 ms	1.00×
BNN	ONNX (Binary)	≈781	58.1%	0.426 ms	0.65×	48.7%	0.382 ms	0.73×
Diff-Logic	C-Compiled	≈64	80.0%	0.191 ms	1.45×	56.4%	0.265 ms	1.05×
<i>Tier 4 (X-Large Scale)</i>								
MLP	ONNX (Float32)	≈1959	72.0%	0.631 ms	1.00×	61.3%	0.625 ms	1.00×
BNN	ONNX (Binary)	≈1963	59.3%	0.939 ms	0.67×	52.3%	0.819 ms	0.76×
Diff-Logic	C-Compiled	≈140	78.9%	0.216 ms	2.91×	61.0%	0.271 ms	2.30×

protocol isolates the raw computational cost of each architecture, independent of parallelism or hardware-specific acceleration.

5. Experimental Results

5.1. Classification Performance and Robustness

As shown in Table 1, Diff-Logic demonstrated superior performance on the dementia classification tasks, consistently outperforming both the MLP (Float32) and BNN (Binary) baselines. Specifically, on the CN vs. AD task, Diff-Logic achieved an F1-score of 78.9–80.2% across all model scales, representing a significant improvement of nearly 10 percentage points over the MLP baseline at the 200k tier. In contrast, the BNN baseline exhibited high instability, particularly on the dementia tasks, where standard deviations reached ± 12 . This suggests that the gradient approximation methods required for training standard binary

neural networks struggled to converge on these high-dimensional EEG data. Diff-Logic, by learning connectivity and logic directly, appeared to avoid these optimization pitfalls, offering a more robust alternative for quantized inference.

On the SEED emotion recognition benchmarks, the Float32 MLP maintained a moderate performance lead (≈ 62 vs 55–61). However, it is crucial to note that this accuracy gap **vanished** at the 500k parameter tier (Diff-Logic 61.0 vs. MLP 61.3). This indicates that while Diff-Logic requires slightly higher capacity to model the complex emotional features in SEED, it can match full-precision performance when scaled, without the associated latency penalties discussed below.

5.2. Inference Efficiency and Scaling Laws

The most critical advantage of the proposed method was revealed in the scaling behaviour (Figure 2). While the inference latency of the MLP and BNN baselines scaled linearly with parameter count, Diff-Logic latency remained effectively constant (~ 0.19 – 0.22 ms) even as the training model size grew by an order of magnitude. This efficiency, and the resulting *flat* scaling curve, stems from two key properties of the Diff-Logic architecture. First, through *structural condensation*, the training formulation explored a dense search space (e.g., 16 operators per neuron), but the inference model collapsed to a single hard logic gate per node. This effectively reduced the active parameter count by a factor of $16\times$ relative to the training capacity. Second, *bitwise compilation* translated these sparse logic circuits into flat bitwise operations. This allowed the model to leverage the full width of CPU registers (SWAR) and Single Instruction, Multiple Data (SIMD) instructions, minimizing memory access bottlenecks. Consequently, at the 500k tier, Diff-Logic achieved a $2.91\times$ speedup over the MLP and a $4.33\times$ speedup over the BNN. Notably, Diff-Logic outperformed the BNN despite both using binary operations, highlighting that logic gate sparsity is fundamentally more hardware-efficient than simple weight quantization.

Finally, Figure 3 summarizes the performance–efficiency trade-off. On the dementia task (Figure 3a), Diff-Logic dominated the pareto frontier, simultaneously achieving the highest F1-score and the lowest latency at scale, a combination unmatched by either baseline. On the SEED task (Figure 3b), while the Float32 MLP retained a slight peak performance advantage, it did so at a disproportionate computational and spatial cost. Specifically, at the 500k parameter tier, the MLP required nearly 2 MB of storage and 0.63 ms per inference. In stark contrast, the equivalent Diff-Logic model completed the forward pass in just 0.22 ms and consumed only 140 KB of memory, representing a substantial $14\times$ reduction in storage footprint. For constrained embedded environments like the Jetson Orin Nano, where L1/L2 cache sizes are critical bottlenecks, this dramatic reduction in footprint and latency suggests that Diff-Logic is a highly viable candidate for always-on deployment, easily justifying the minor trade-off in peak performance observed in more complex datasets (SEEDs).

6. Discussion

6.1. Discrete Logic as an Effective Inductive Bias for EEG

Our results indicate that Diff-logic matches or exceeds deep learning performance on EEG tasks while using significantly fewer resources. Notably, Diff-Logic outperformed the MLP on dementia tasks by 9.4 percentage points in F1-score with identical parameter budgets (Tier 3), suggesting a superior inductive bias for clinical signals. While MLPs must approximate sharp decision boundaries with continuous activations, Diff-Logic naturally aligns with the threshold-based rules common in clinical EEG biomarkers (e.g., specific band-power cutoffs).

Scaling analysis revealed no benefit from increasing parameters beyond 100k (Table 1). In fact, MLP accuracy on the CN vs. AD task dropped from 74.8% to 72.0% when scaled from

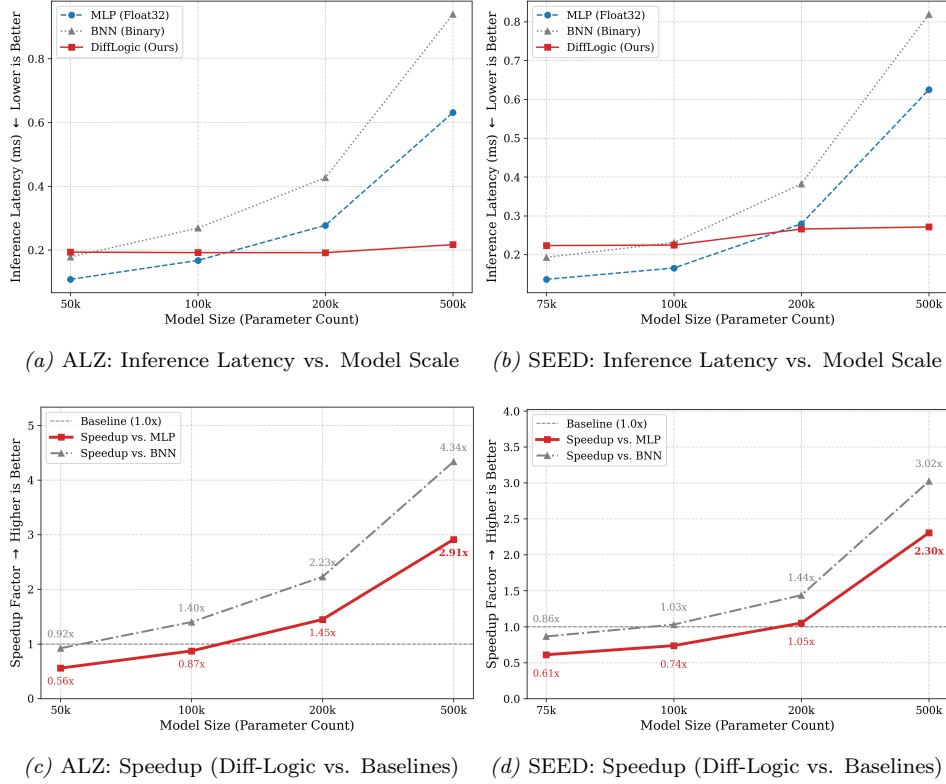


Figure 2. Inference scaling on the Jetson Orin Nano embedded CPU (single-threaded).

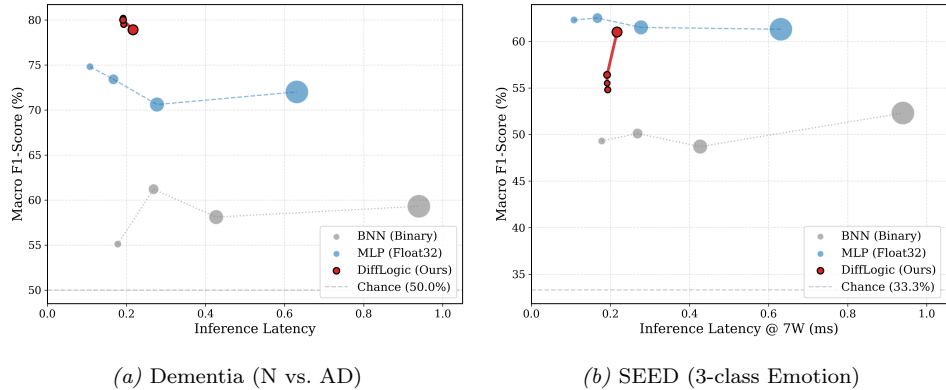


Figure 3. Macro F1-Score vs. Inference Latency on the Jetson Orin Nano embedded CPU (single-threaded). Each point represents one model tier (50k–500k parameters) and bubble area is proportional to compiled model storage size in KB.

50k to 500k, indicating overfitting. This highlights again the practical value of logic-based models in clinical settings with smaller cohorts, where they offer superior generalization alongside deployment efficiency.

The BNN baseline consistently lagged on dementia tasks, likely due to the restricted capacity caused by binarizing both weights and activations. However, this extreme quantization acted as an implicit regularizer at higher parameter counts.

While the Float32 MLP degraded by nearly 8 percentage points on SEED German tasks when scaled to 500k, the BNN maintained stable performance across all scales (Table 1, SEED German) and even overtook the MLP on larger-scale SEED French tasks, effectively avoiding catastrophic overfitting. Finally, regarding inference efficiency, Diff-Logic retains a fundamental architectural advantage over BNNs even when discounting implementation overheads. While BNNs require an accumulation step (*popcount*) after the *XNOR* operation to sum bitwise products, compiled Diff-Logic models are accumulation-free, operating as discrete combinational circuits.

6.2. The scaling advantage of Boolean circuits

The most striking efficiency result was not Diff-Logic’s absolute speed, but its *scaling behaviour*. The inference latency of the MLP and BNN baselines exhibits an exponential behaviour as parameter count increases. In contrast, compiled Diff-Logic latency remained nearly constant across a $10\times$ increase in model size (Figure 2). This occurred because the compiled Boolean circuit is evaluated via bitwise operations over packed registers. Consequently, the computational cost was bounded by the circuit’s depth (the number of sequential layers) rather than its width (the number of parallel gates). At the 500k parameter tier, this architectural advantage yielded a $2.91\times$ speedup over the MLP on a single ARM core.

6.3. Implications for edge deployment

Real-time clinical tasks like seizure detection require ultra-low latency to track neural dynamics on portable, battery-powered devices. At the 500k tier, Diff-Logic achieved a latency of 0.22 ms with a total storage footprint of just 140 KB. This was nearly two orders of magnitude below the real-time threshold and compact enough to fit entirely within the L1/L2 cache hierarchy of most modern microcontrollers. This effectively eliminated costly off-chip memory accesses, positioning logic-based networks as a highly viable architecture class for always-on neural monitoring, where the dominant engineering constraints are power and memory rather than raw continuous accuracy.

6.4. Limitations and Future Work

A primary limitation of our current evaluation is its reliance on pre-extracted tabular EEG features. The standard Diff-Logic architecture does not inherently support temporal or spatial convolutions, restricting its immediate applicability to raw time-series or image-based modalities. Furthermore, the thermometer encoding expanded the input dimensionality by $15\times$, which may become computationally expensive for very high-dimensional raw signals. Nevertheless, for real-time applications, our single-core results remain highly representative of true edge deployment constraints.

Several directions merit further investigation to bridge these gaps. First, recent extensions of Diff-Logic to convolutional [8] and recurrent [9] gate architectures offer a path toward temporal and spatial EEG feature extraction using logic primitives. Integrating these variants could enable end-to-end learning directly from raw EEG while preserving the accumulation-free inference efficiency demonstrated in this work. Second, for logic-based networks to reach true clinical viability, robust interpretability methods must be developed. Although we can currently observe gate-type distributions and node-selection frequencies per layer, designing systematic approaches to extract and validate clinically meaningful Boolean decision rules from these compiled circuits remains an open challenge. Finally, true Field-Programmable Gate Array (FPGA) or Application-Specific Integrated Circuit (ASIC) implementations of the compiled Boolean circuits, where logic gates map directly to silicon primitives without

software emulation, could bring Diff-Logic substantially closer to ultra-low-power, real-time brain-computer interface applications.

7. Conclusion

We presented an iso-parameter comparison of Diff-Logic against standard continuous (MLP) and binary (BNN) baselines for EEG classification. Our experiments show that Diff-Logic achieved superior performance on clinical dementia detection (up to 80.2% Macro F1, outperforming the MLP by 9.4 percentage points) and comparable accuracy on the SEED emotion recognition task at scale. Crucially, these logic networks compiled to accumulation-free Boolean circuits that were $2.91\times$ faster and $14\times$ smaller than equivalent floating-point models at inference time. Furthermore, we showed that Diff-Logic’s latency remained effectively constant as model complexity grew, a hardware scaling property unmatched by conventional multiply-accumulate architectures. Ultimately, these results demonstrate that logic-based neural networks offer a compelling performance–efficiency trade-off for resource-constrained biomedical applications. Our findings emphasize that architectural inductive bias is a decisive factor for embedded neural inference, positioning Diff-logic as a deployment-ready alternative to standard deep learning for structured biomedical signals.

Acknowledgements

This work was supported by NSERC Discovery Grants (RGPIN-2023-03443 and RGPIN-2024-05575) and the Manitoba Medical Service Foundation (grant number 2026-04).

References

- [1] A. Raza and M. Z. Yusoff. “Deep Learning Approaches for EEG-Motor Imagery-Based BCIs: Current Models, Generalization Challenges, and Emerging Trends”. In: *IEEE Access* 13 (2025), pp. 151866–151893.
- [2] J. Lai, J. Wei, L. Yao, and Y. Wang. “A Simple Review of EEG Foundation Models: Datasets, Advancements and Future Perspectives”. In: *arXiv preprint arXiv:2504.20069* (2025).
- [3] A. Joel. “Challenges, Limitations, and Future Directions in EEG-Driven Image Classification using Machine Learning”. In: (May 2024).
- [4] G. Kuruppu, N. Wagh, V. Kremen, S. Pati, G. Worrell, and Y. Varatharajah. “Eeg foundation models: A critical review of current progress and future directions”. In: *arXiv preprint arXiv:2507.11783* (2025).
- [5] S. Muter, F. Alqurashi, and A. Badr. “A Comprehensive Survey on Wearable Computing for Mental and Physical Health Monitoring”. In: *Electronics* 14.17 (2025), p. 3443.
- [6] O. Sen, R. Soni, D. Virmani, A. Parekh, P. Lehman, S. Jena, A. Katikhaneni, A. Khalifa, and B. Chatterjee. “A low-latency neural inference framework for real-time handwriting recognition from EEG signals on an edge device: O. Sen et al.” In: *Scientific Reports* 15.1 (2025), p. 41040.
- [7] F. Petersen, C. Borgelt, H. Kuehne, and O. Deussen. “Deep differentiable logic gate networks”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 2006–2018.
- [8] F. Petersen, H. Kuehne, C. Borgelt, J. Welzel, and S. Ermon. “Convolutional differentiable logic gate networks”. In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 121185–121203.
- [9] S. Bühner, A. Plesner, T. Aczel, and R. Wattenhofer. “Recurrent deep differentiable logic gate networks”. In: *Proceedings of the 2nd International Workshop on Edge and Mobile Foundation Models*. 2025, pp. 31–36.
- [10] L. Rüttgers, T. Aczel, A. Plesner, and R. Wattenhofer. “Light Differentiable Logic Gate Networks”. In: *arXiv preprint arXiv:2510.03250* (2025).

- [11] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggenberger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball. “Deep learning with convolutional neural networks for EEG decoding and visualization”. In: *Human brain mapping* 38.11 (2017), pp. 5391–5420.
- [12] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance. “EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces”. In: *Journal of neural engineering* 15.5 (2018), p. 056013.
- [13] Y. Song, Q. Zheng, B. Liu, and X. Gao. “EEG conformer: Convolutional transformer for EEG decoding and visualization”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 31 (2022), pp. 710–719.
- [14] W.-B. Jiang, L.-M. Zhao, and B.-L. Lu. “Large brain model for learning generic representations with tremendous EEG data in BCI”. In: *arXiv preprint arXiv:2405.18765* (2024).
- [15] C. Yang, M Westover, and J. Sun. “Biot: Biosignal transformer for cross-data learning in the wild”. In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 78240–78260.
- [16] J. Wang, S. Zhao, Y. Zhou, Y. Kang, S. Li, and G. Pan. “DeeperBrain: A Neuro-Grounded EEG Foundation Model Towards Universal BCI”. In: *arXiv preprint arXiv:2601.06134* (2026).
- [17] B. Döner, T. M. Ingolfsson, L. Benini, and Y. Li. “LUNA: Efficient and topology-agnostic foundation model for EEG signal analysis”. In: *arXiv preprint arXiv:2510.22257* (2025).
- [18] A. Tegon, T. M. Ingolfsson, X. Wang, L. Benini, and Y. Li. “FEMBA: Efficient and Scalable EEG Analysis with a Bidirectional Mamba Foundation Model”. In: *arXiv preprint: 2502.06438* (2025).
- [19] G. Hinton, O. Vinyals, and J. Dean. “Distilling the knowledge in a neural network”. In: *arXiv preprint arXiv:1503.02531* (2015).
- [20] S. Han, H. Mao, and W. J. Dally. “Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding”. In: *arXiv preprint arXiv:1510.00149* (2015).
- [21] M. Courbariaux, I. Hubara, D. Soudry, R. El-Yaniv, and Y. Bengio. “Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1”. In: *arXiv preprint arXiv:1602.02830* (2016).
- [22] S. A. Mirsalari, S. Sinaei, M. E. Salehi, and M. Daneshmand. “MuBiNN: Multi-level binarized recurrent neural network for EEG signal classification”. In: *2020 IEEE international symposium on circuits and systems (ISCAS)*. IEEE. 2020, pp. 1–5.
- [23] A. Miltiadous, K. D. Tzimourta, T. Afrantou, P. Ioannidis, N. Grigoriadis, D. G. Tsalikakis, P. Angelidis, M. G. Tsipouras, E. Glavas, N. Giannakeas, et al. “A dataset of scalp EEG recordings of Alzheimer’s disease, frontotemporal dementia and healthy subjects from routine EEG”. In: *Data* 8.6 (2023), p. 95.
- [24] W.-L. Zheng and B.-L. Lu. “Investigating Critical Frequency Bands and Channels for EEG-based Emotion Recognition with Deep Neural Networks”. In: *IEEE Transactions on Autonomous Mental Development* ().
- [25] W. Liu, W.-L. Zheng, Z. Li, S.-Y. Wu, L. Gan, and B.-L. Lu. “Identifying similarities and differences in emotion recognition with EEG and eye movements among Chinese, German, and French People”. In: *Journal of Neural Engineering* 19.2 (2022), p. 026012.
- [26] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, and F. Yger. “A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update”. In: *Journal of Neural Engineering* 15.3 (2018), p. 031005.
- [27] J. Buckman, A. Roy, C. Raffel, and I. Goodfellow. “Thermometer encoding: One hot way to resist adversarial examples”. In: *International conference on learning representations*. 2018.
- [28] R. Banner, Y. Nahshan, and D. Soudry. “Post training 4-bit quantization of convolutional networks for rapid-deployment”. In: *Advances in neural information processing systems* 32 (2019).
- [29] NVIDIA Corporation. *NVIDIA Jetson Orin Nano Super Developer Kit Datasheet*. 3575392-r2. Accessed: 2025-02-16. NVIDIA Corporation. Dec. 2024. URL: <https://nvdam.widen.net/s/zkfqjmtds2/jetson-orin-datasheet-nano-developer-kit-3575392-r2>.
- [30] I. Rakhmatulin. “Device JNEEG to convert Jetson Nano to brain-Computer interfaces. Short report”. In: *arXiv preprint arXiv:2301.11110* (2023).

Appendix A. Dataset Summary

Table A1. Summary of EEG datasets. Features denote Power Spectral Density (PSD) or Differential Entropy (DE). Windows denotes total non-overlapping segments across all subjects.

Dataset	Type	Subj.	Chan.	f_s (Hz)	Feat. Dim	Windows	CV Protocol
Dementia (N vs AD)	Clinical	65	19	500	95 (PSD)	5,292	Subject-level
Dementia (N vs FTD)	Clinical	52	19	500	95 (PSD)	4,043	Subject-level
SEED-China	Emotion	15	62	200	310 (DE)	37,890	Trial-level
SEED-France	Emotion	8	62	200	310 (DE)	20,016	Trial-level
SEED-Germany	Emotion	8	62	200	310 (DE)	17,420	Trial-level

Appendix B. Thermometer Encoding

Table A2. Mapping between continuous input ranges, Thermometer Encoding ($T = 15$), and equivalent 4-bit unsigned integer precision.

Normalized Input x'	Thermometer Code (15-bit)	Sum of Bits	UINT4 Equiv.
$[0, \frac{1}{16})$	$[0, 0, 0, \dots, 0]$	0	0000 (0)
$[\frac{1}{16}, \frac{2}{16})$	$[1, 0, 0, \dots, 0]$	1	0001 (1)
$[\frac{2}{16}, \frac{3}{16})$	$[1, 1, 0, \dots, 0]$	2	0010 (2)
\vdots	\vdots	\vdots	\vdots
$[\frac{15}{16}, 1]$	$[1, 1, 1, \dots, 1]$	15	1111 (15)

Appendix C. Boolean operators for Differential Logic Gate Network

Table A3. The 16 Boolean logic gates and their differentiable real-valued relaxations used in Diff-Logic. The inputs $a, b \in [0, 1]$ represent the probabilities of the input bits being true.

ID	Boolean Operator	Description	Real-Valued Relaxation $f(a, b)$
0	False	Constant False	0
1	$a \wedge b$	AND	$a \cdot b$
2	$a \wedge \neg b$	A and not B	$a - a \cdot b$
3	a	Identity A	a
4	$\neg a \wedge b$	Not A and B	$b - a \cdot b$
5	b	Identity B	b
6	$a \oplus b$	XOR	$a + b - 2(a \cdot b)$
7	$a \vee b$	OR	$a + b - (a \cdot b)$
8	$\neg(a \vee b)$	NOR	$1 - (a + b - a \cdot b)$
9	$\neg(a \oplus b)$	XNOR	$1 - (a + b - 2a \cdot b)$
10	$\neg b$	NOT B	$1 - b$
11	$a \vee \neg b$	Implication ($B \rightarrow A$)	$1 - b + a \cdot b$
12	$\neg a$	NOT A	$1 - a$
13	$\neg a \vee b$	Implication ($A \rightarrow B$)	$1 - a + a \cdot b$
14	$\neg(a \wedge b)$	NAND	$1 - a \cdot b$
15	True	Constant True	1

Appendix D. Model Architectures Structure for Comparison

Table A4. Model architectures at each iso-parameter tier. Diff-Logic uses H gates/layer; MLP and BNN use H neurons/layer. Diff-Logic receives thermometer-encoded binary inputs; MLP and BNN receive continuous inputs.

Tier	Diff-Logic (Binary In)	MLP (Continuous In)	BNN (Continuous In)
<i>Dementia (input: 1,425 binary / 95 continuous, 2 classes)</i>			
50k	2L: [1532, 1532]	2L: [310, 64]	2L: [306, 64]
100k	4L: [1940, 1940, 970, 970]	2L: [620, 64]	2L: [616, 64]
200k	6L: [2988×2, 1992×2, 996×2]	3L: [870, 128, 32]	3L: [860, 128, 32]
500k	6L: [7782×2, 5188×2, 2594×2]	3L: [2200, 128, 64]	3L: [2180, 128, 64]
<i>SEED (input: 4,650 binary / 310 continuous, 3 classes)</i>			
74k	2L: [2325, 2325]	2L: [216, 32]	2L: [214, 32]
100k	2L: [3219, 3219]	2L: [292, 32]	2L: [290, 32]
200k	2L: [6156, 6156]	3L: [530, 64, 32]	3L: [520, 64, 32]
500k	4L: [10314×2, 5157×2]	3L: [1120, 128, 64]	3L: [1120, 128, 64]

Appendix E. Exact Model Configurations

Table A5. Exact parameter and gate counts. For MLP and BNN, inference cost scales with the full parameter count. For Diff-Logic, inference cost scales only with the compiled gate count.

Model	Dementia		SEED	
	Params	Gates	Params	Gates
MLP _{50k}	49,794	—	74,219	—
BNN _{50k}	49,894	—	74,025	—
Diff-Logic _{50k}	49,024	3,064	74,400	4,650
MLP _{100k}	99,394	—	100,287	—
BNN _{100k}	100,114	—	100,245	—
Diff-Logic _{100k}	93,120	5,820	103,008	6,438
MLP _{200k}	199,202	—	200,993	—
BNN _{200k}	199,002	—	198,475	—
Diff-Logic _{200k}	191,232	11,952	196,992	12,312
MLP _{500k}	501,314	—	500,259	—
BNN _{500k}	501,578	—	502,883	—
Diff-Logic _{500k}	498,048	31,128	495,072	30,942