The Trichromatic Strong Lottery Ticket Hypothesis: A Unifying View of Supermask-Based Learning

Ángel López García-Arias *‡, Yasuyuki Okoshi†, Hikari Otsuka†,
Daiki Chijiwa*, Yasuhiro Fujiwara*, Susumu Takeuchi*, and Motomura Masato†
*NTT, Inc., Japan. †Institute of Science Tokyo, Japan. ‡Corresponding Author: lopez@ieee.org

***Abstract*—The Strong Lottery Ticket Hypothesis (SLTH) posits that high-performing subnetworks can be extracted from randomly initialized neural networks using binary pruning masks, or supermasks. This paper introduces the Trichromatic Strong Lottery Ticket Hypothesis (T-SLTH), a unified generalization of the SLTH that decomposes supermasks into three conceptually orthogonal components: connectivity, sign, and magnitude. This trichromatic decomposition unifies existing SLTH variants, enables new configurations, and clarifies the connection between supermask-based training and quantization-aware training. The proposed framework is evaluated on image classification benchmarks with ResNet architectures, achieving state-of-the-art accuracy-compression tradeoffs among supermask-based models. ResNet-50 is compressed $38\times$ on CIFAR-100 and $25\times$ on ImageNet without training any weights. Additional experiments across datasets, architectures, and modalities demonstrate that trichromatic supermasks act as robust structural priors. These results position partially random networks trained via supermasks as a compelling alternative to conventional weight optimization for efficient inference systems.**

***Index Terms*—supermasks, strong lottery ticket hypothesis, sparsity, pruning, neural network compression**

I. Introduction

Traditional neural network training involves optimizing real-valued weights via gradient descent, simultaneously determining which connections are important and how they interact. This process implicitly entangles multiple forms of inductive bias—such as sparsity, polarity, and scaling—within a single optimization procedure. In contrast, recent work has shown that strong performance can be achieved without training weights at all, by identifying *which structures* enable effective computation in randomly initialized networks. The Strong Lottery Ticket Hypothesis (SLTH) [1] formalizes this observation, positing that sparse subnetworks defined by binary *supermasks* over fixed random weights can match the performance of fully trained models. However, most existing supermask-based approaches apply binary masking at individual weights, implicitly conflating three structural roles: *connectivity* (whether a connection exists), *sign* (whether its influence is excitatory or inhibitory), and *magnitude* (its relative strength). This lack of disentanglement limits expressiveness, interpretability, and systematic design.

In this work, we propose the *Trichromatic Strong Lottery Ticket Hypothesis (T-SLTH)*, which provides a unifying view of supermask-based learning by decomposing a supermask into three conceptually orthogonal components: *connectivity*, *sign*, and *magnitude*. These components are applied over fixed random weights and may be learned or fixed independently, enabling a spectrum of training regimes that ranges from pure pruning to quantization-aware training (QAT). This decomposition unifies a variety of existing supermask methods within a single structural abstraction, and reframes sparse subnetwork discovery as a problem of architectural design rather than solely one of optimization. Under this formulation, supermasks become interpretable structural artifacts rather than monolithic binary objects. By controlling which components are learned and which are fixed, T-SLTH provides a flexible interface for studying and exploiting sparsity, quantization, randomness, and architectural constraints within a unified framework.

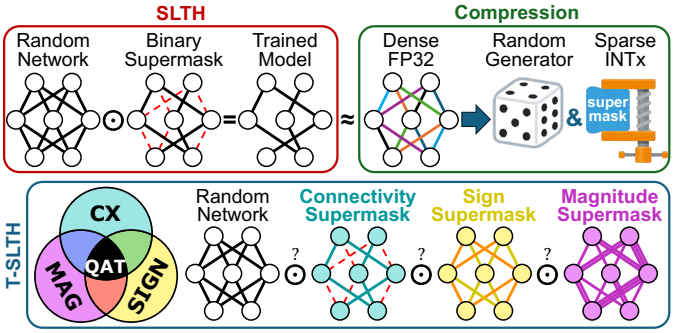


Fig. 1: The T-SLTH decomposes the SLTH into three orthogonal components: connectivity, sign, and magnitude.

This view is validated across vision and natural language processing tasks, spanning multiple datasets, architectures, and model scales. Across these settings, T-SLTH-based masks consistently match or outperform dense baselines while achieving strong accuracy–compression tradeoffs relative to existing supermask approaches, including on large-scale benchmarks. The main contributions of this work are:

1) A unifying view of supermask learning that generalizes the SLTH and connects it to QAT.
2) A trichromatic supermask construction framework for extending supermask methods to arbitrary connectivity.
3) Demonstration of flexibility across architectures, datasets, and modalities.
4) Achievement of state-of-the-art accuracy–compression tradeoffs among supermask methods.

This paper extends a preliminary non-archival workshop version [2] with a reframed conceptual perspective and a substantially broader experimental evaluation.

II. Background and Motivation

Deep neural networks are known to be heavily overparameterized, enabling post-training compression techniques such as pruning and quantization to reduce memory and computation costs. Pruning removes redundant connections [3], [4], while quantization reduces numerical precision. The traditional FP32 full numerical precision format is known to be unnecessarily large for deep learning, encouraging efforts to transition to an FP8 format [5] and implementations with even more aggressive quantization [6]–[9] down to binarization [10]–[12]. These techniques have become staples in efficient model deployment and hardware-aware optimization [7], [13], [14].

A paradigm shift was introduced by the *Strong Lottery Ticket Hypothesis* (SLTH) [1], illustrated in Fig. 1: instead of compressing trained models, one can directly discover high-performing sparse subnetworks at initialization—called *strong lottery tickets* (SLTs)—using unstructured binary pruning masks, or *supermasks*, over fixed random weights, with only the mask parameters optimized during training. These SLTs achieve high test accuracy *without updating any weights*, revealing that some subnetworks are already "lucky" at initialization. This view aligns with prior work showing that certain architectures can exhibit strong performance even with random weights [15], [16], suggesting the presence of implicit structural priors: patterns in connectivity or activation that align with the task distribution even in the absence of learned weights. Identifying and leveraging such priors at initialization may offer a path toward sparse models that are not only efficient, but also reusable and interpretable. Moreover, supermask optimization can be viewed as imposing structure on the optimization landscape itself, guiding learning toward these favorable configurations.

Since weights can be regenerated from the random seed, storing only the sparse supermask is sufficient for model reconstruction. Storage overhead is minimal: binary masks require only 1 bit per weight, and scalar or ternary masks can be compressed using nested run-length compression [17] or entropy coding [18]. Supermask sparsity can also be exploited for computation reduction via data gating [19], sparse execution, or replacing multipliers with shift-add operations [18]. Even in analog devices, structured pruning can guide physical design [20], though reconfigurability remains a challenge. Further compression can be obtained by optionally initializing connectivity with random pruning, for example by overlaying a learned supermask with a fixed random mask that can be reconstructed from the seed in the same manner as weights, reducing the number of edges from the beginning [21], [22].

A. Supermask-Based Training

Supermask-based training seeks to identify which structural elements of a network are essential for task performance, even without learning weights. Supermasks are commonly learned using the `Edge-Popup` algorithm [1]. A score tensor $\boldsymbol{Z} \in \mathbb{R}^n$ is initialized (one score per weight) and updated during training using backpropagation gradients computed from the masked weights. The supermask is derived from $\boldsymbol{Z}$ via thresholding or top-k selection and is applied to the fixed random weights $\boldsymbol{W}_{\text{rand}}$ only during the forward pass. During backpropagation, gradients are applied to $\boldsymbol{Z}$ using straight-through estimation [23], allowing the mask to evolve over time. Sparsity can be enforced using per-layer [1] or global constraints [24], fixed thresholds [25], [26], preset distributions [27], or alternative search strategies [28].

Several extensions of the SLTH paradigm have expanded what the supermask encodes:

- The **Connectivity Supermask (*C*)** [1] learns a binary mask by retaining the positions with the top-$k\%$ score magnitudes, effectively quantizing scores into $\{0, 1\}$.
- The **Signed Supermask (*SSup*)** [26] increases expressivity by encoding sign, quantizing scores into balanced ternary values $\{-1, 0, +1\}$.
- The **Multicoated Supermask (*MSup*)** [17] method represents magnitude using scalar-valued supermasks by aggregating multiple thresholded binary masks (*coats*), effectively quantizing scores into unsigned integer scalars $\{0, \ldots, N\}$, for N coats.

Collectively, these approaches progressively quantize different structural aspects of the score tensor. In parallel, it was shown that transforming the underlying ResNet into a recurrent architecture by sharing weights and masks across iterations—i.e., **folding (*F*)** the supermask [29]—can substantially improve both accuracy and compression. Folded supermasks are not a distinct supermask type, but an orthogonal composition technique that can be combined with any of the above variants.

B. Motivation for a Unified Framework

All existing supermask variants rely on learnable scores to control which parts of the network are active. However, they differ in what the scores encode, conflating the effects of randomness, connectivity, sign, and magnitude. In practice, this makes it unclear which components are responsible for performance gains and how design choices should transfer across architectures or hardware constraints. This entanglement limits interpretability, reusability, and systematic design, and prior work lacks a unified view explaining how these variants relate or can be composed. This motivates a unified framework for supermask design that explicitly separates connectivity, sign, and magnitude. As the next section shows, this trichromatic perspective generalizes prior approaches while enabling new combinations tailored to specific constraints such as limited reconfigurability, memory budgets, or transfer scenarios. By disentangling structural components, the proposed formulation reframes sparsity methods as general-purpose structural priors rather than one-shot compression techniques.

III. Trichromatic Supermask Framework

This section introduces a unified framework for supermask-based training that explicitly disentangles three fundamental aspects of weight behavior: connectivity, sign, and magnitude. Rather than treating pruning as a privileged operation, the proposed perspective exposes a broader design space in which different subsets of these properties may be learned,

Supermask	W. Init	Randomness			R
		CX	SIGN	MAG	
C	KN	×	✓	✓	2
	SK	×	✓	×	1
S	KN	✓	×	✓	2
	SK	✓	×	×	1
M	KN	✓	✓	×	2
	SK	✓	✓	×	2
CS *(SSup)*	KN	×	×	✓	1
	SK	×	×	×	0
CM *(MSup)*	KN	×	✓	×	1
	SK	×	✓	×	1
SM	KN	✓	×	×	1
	SK	✓	×	×	1
CSM *(SMSup)*	KN	×	×	×	0
	SK	×	×	×	0

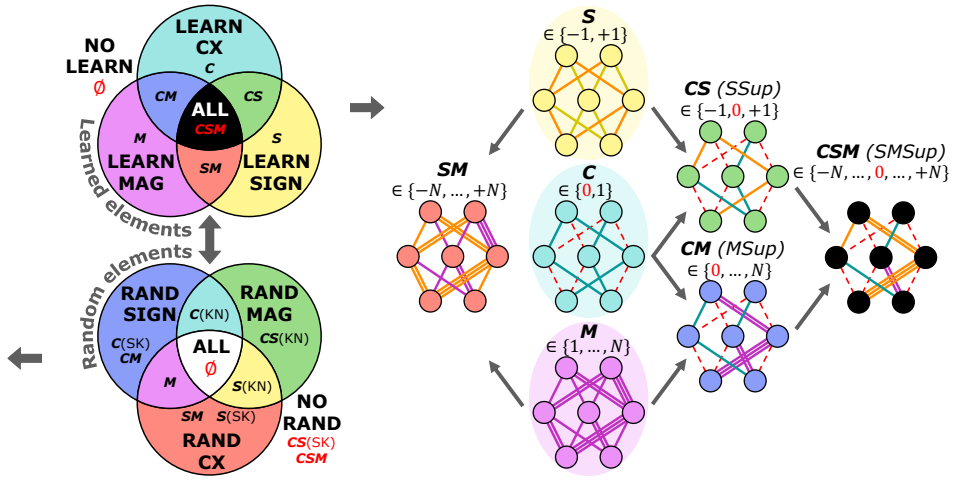


Fig. 2: The T-SLTH decomposes supermask learning into connectivity, sign, and magnitude, yielding seven possible configurations that learn different structural subsets while leaving the remaining components fixed random.

fixed, or randomized. This trichromatic view both unifies prior supermask methods and enables novel configurations with practical relevance for efficient learning and hardware-constrained scenarios.

A. Trichromatic Reinterpretation of the SLTH

Existing extensions of the Strong Lottery Ticket Hypothesis (SLTH) progressively expand what is encoded in the supermask. The SSup method augments connectivity learning with sign information, while MSup further introduces learnable magnitudes. Combining both yields the **Signed Multicoated Supermask (*SMSup*)**, originally proposed in the workshop version of this work [2] and later validated in hardware implementations [18]. SMSup quantizes scores into signed integers $\{-N, \ldots, 0, \ldots, +N\}$, simultaneously encoding connectivity, sign, and magnitude.

While empirically effective, this formulation marks a conceptual departure from the original SLTH: it eliminates fixed randomness entirely and encodes all structural degrees of freedom in a single object. From this viewpoint, SMSup is functionally equivalent to quantization-aware training (QAT) weights, where the score tensor acts as a set of *latent weights* and the uncovered subnetwork corresponds to the effective quantized model. Under this interpretation, other supermask methods can be viewed as instances of *partially random weight quantization*, in which some dimensions of each weight are learned while others are held fixed and random.

From this perspective, the pruning enforced by existing supermasks is not a privileged operation. Rather, connectivity is only one of several structural dimensions that may be learned or fixed. It should be possible to train supermasks where the part left fixed is the connectivity—i.e., to obtain SLTs with *random*, *dense*, or otherwise *arbitrarily predetermined connectivity*—by training signs or magnitudes instead. This observation leads to a generalization of the SLTH:

The *Trichromatic* Strong Lottery Ticket Hypothesis (T-SLTH). *A randomly initialized neural network of arbitrary predefined sparsity contains sufficient representational capacity such that, after training only its weight connectivity, signs, magnitudes, or any subset thereof, it can achieve competitive test accuracy.*

This generalization of the SLTH expands the central role that previous work has given solely to network topology to a framework that considers three orthogonal structural biases based on the fundamental properties of a weight:

- **Connectivity (CX)**: whether a connection exists.
- **Sign (SIGN)**: the polarity of a connection.
- **Magnitude (MAG)**: the strength of a connection.

B. Trichromatic Supermask Construction

While SSup expanded the supermask's learned *connectivity* with learned *signs*, MSup expanded it with learned *magnitudes*, and the proposed SMSup learns all three. This work proposes to detach these three elements into a framework formed by *three primary supermasks*, analogous to the three primary colors, illustrated in Fig. 2 (center):

- **Connectivity supermask ($\boldsymbol{C}$)**: quantizes score magnitude into $\{0, 1\}$ to encode which connections are active.
- **Sign supermask ($\boldsymbol{S}$)**: quantizes scores to $\{-1, +1\}$, encoding polarity while leaving magnitude unchanged.
- **Magnitude supermask ($\boldsymbol{M}$)**: encodes unsigned integer magnitudes $\{1, \ldots, N\}$ via multiple thresholded masks.

Each of these structures may be learned using an algorithm such as `Edge-Popup`, be set random, or to a predetermined pattern, which may be useful for example to represent a specific underlying analog substrate.

The full trichromatic supermask is constructed as $\boldsymbol{T} = \boldsymbol{C} \odot \boldsymbol{M} \odot \boldsymbol{S}$, where $\odot$ denotes elementwise multiplication. This formulation yields seven possible supermask configurations depending on which components are learned, shown in Fig. 2 (right). These include previously proposed methods—SSup (*CS*), MSup (*CM*), and SMSup (*CSM*)—as well as new configurations: *S*, *M*, and *SM*, which are introduced here.

Algorithm 1 Trichromatic Supermask Training

Input: Fixed random weights $\boldsymbol{W}_{\text{rand}}$; trainable scores $\boldsymbol{Z}$
Config: Flags `c`, `m`, `s`; list of densities $\{k_0, \ldots, k_{N-1}\}$; optional $\boldsymbol{C}_{\text{init}}, \boldsymbol{M}_{\text{init}}, \boldsymbol{S}_{\text{init}}$
Initialize masks
$\boldsymbol{C} \leftarrow \boldsymbol{C}_{\text{init}}$ if provided, else $\mathbf{1}$
$\boldsymbol{M} \leftarrow \boldsymbol{M}_{\text{init}}$ if provided, else $\mathbf{1}$
$\boldsymbol{S} \leftarrow \boldsymbol{S}_{\text{init}}$ if provided, else $\mathbf{1}$
Training loop: Optimize trichromatic supermask $\boldsymbol{T}$
for each training step **do**
  $\boldsymbol{C} \leftarrow \texttt{threshold}(\boldsymbol{Z}, k_0)$ **if** `c` *# Top-k*
  $\boldsymbol{M} \leftarrow \mathbf{1} + \sum_{i=1}^{N-1} \texttt{threshold}(\boldsymbol{Z}, k_i)$ **if** `m`
  $\boldsymbol{S} \leftarrow \texttt{sign}(\boldsymbol{Z})$ **if** `s` *#Outputs in* $\{-1, +1\}$
  $\boldsymbol{T} \leftarrow \boldsymbol{C} \odot \boldsymbol{M} \odot \boldsymbol{S}$ *# Trichromatic Supermask*
  $\boldsymbol{W} \leftarrow \boldsymbol{W}_{\text{rand}} \odot \boldsymbol{T}$ *# Effective weights*
  Forward pass; update $\boldsymbol{Z}$ via e.g. `Edge-Popup`
end for

Algorithm 1 formalizes the proposed trichromatic training procedure. Given fixed random weights $\boldsymbol{W}_{\text{rand}}$ and a trainable score tensor $\boldsymbol{Z}$, three primary supermasks are constructed and optionally optimized: a connectivity mask $\boldsymbol{C}$, a sign mask $\boldsymbol{S}$, and a magnitude mask $\boldsymbol{M}$. Each mask may be learned, fixed random, or set to a predetermined pattern. During training, enabled masks are updated from $\boldsymbol{Z}$ using thresholding or sign operations, while gradients are propagated to $\boldsymbol{Z}$ via straight-through estimation as in `Edge-Popup`. The effective weights used in the forward pass are obtained by elementwise multiplication of the random weights with the combined trichromatic supermask.

Fig. 2 (left) collects the randomness left by each supermask type in weight connectivity, signs, and magnitudes, tallying the total as a degree of randomness R from 0 (completely learned) to 3 (completely random). Notably, this degree also depends on the initialization of the fixed weights. For example, under Signed Kaiming initialization (SK) [1], weights take values in $\{-\sigma, +\sigma\}$ (where σ is the standard deviation of the continuous-valued Kaiming Normal initialization (KN) [30]), making sign the only source of randomness. In this setting, SSup (*CS*) fully removes all randomness, becoming functionally equivalent to balanced-ternary QAT weights. The novel *S*, *M*, and *SM* settings enable training supermasks while leaving connectivity dense, random, or arbitrarily preset, depending on the desired inductive bias or hardware constraints.

IV. Experiments and Results

This section evaluates the proposed trichromatic supermask framework across a broad range of tasks, architectures, and modalities. Experiments proceed from controlled sparsity sweeps to generalization tests across datasets, architectures, and modalities, followed by an evaluation of compression performance and comparison against state-of-the-art quantization-aware training (QAT) methods.

A. Experimental Setup

a) Mask Construction: Supermasks are trained using the original `Edge-Popup` [1] algorithm, which employs a top-k% threshold to match a target density. Unless otherwise stated, a constant (non-annealed) top-k% is applied independently at each layer. For magnitude supermasks ($\boldsymbol{M}$), the density schedule across coats is determined using the Linear method from [17].

b) Initialization: To evaluate configurations with non-learned connectivity, we employ random pruning at initialization (RPaI) by applying a top-k% threshold to the initial random score tensor $\boldsymbol{Z}$, yielding an initial connectivity mask $\boldsymbol{C}_{\text{init}}$. This allows connectivity to be fixed, random, or learned depending on the selected supermask configuration.

c) Training Protocol: Due to the diversity of architectures and datasets considered, hyperparameters are necessarily heterogeneous. For each model–dataset pair, we adopt commonly used training settings and lightly tune hyperparameters on a validation split, after which they are fixed across all supermask configurations in that setting. An extended arXiv version will report full training configurations. Results show average top-1 test accuracy over five runs, with shaded standard deviation. For ImageNet [31], due to computational cost, we report single-run top-1 validation accuracy.

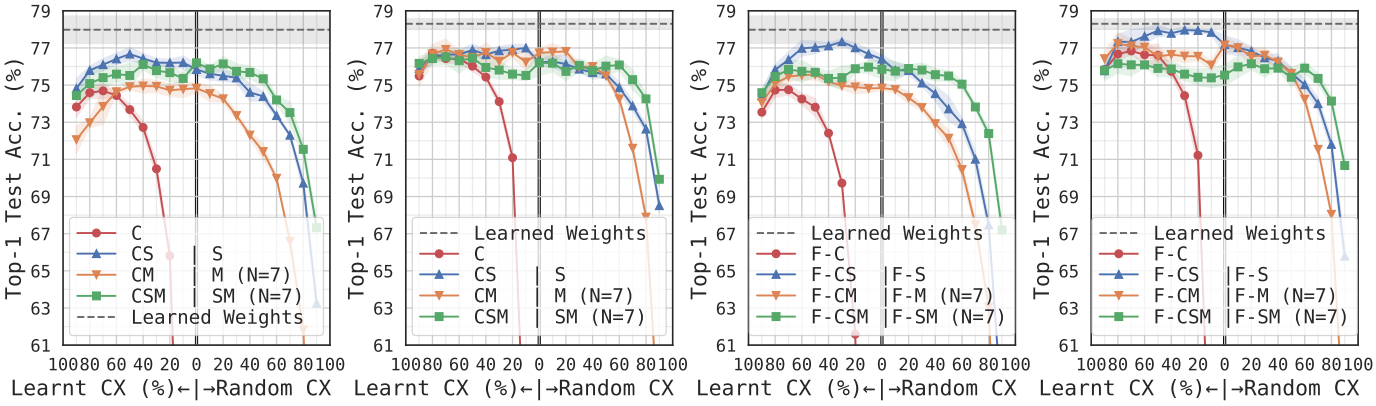
d) Memory Size Accounting: Model size is computed assuming an architecture equipped with an on-chip random number generator that regenerates fixed random weights and random connectivity patterns from a seed, following prior work [19], [32], [33]. Supermasks are stored in a nested binary representation [17], without further compression. Batch normalization parameters are stored in full precision.

The memory size in bits of a trichromatic supermask $\boldsymbol{T}$ is given by $|\boldsymbol{T}| = (|\boldsymbol{C}| + |\boldsymbol{M}| + |\boldsymbol{S}|) \cdot p_{\text{init}}$, where p_{init} is the density used for RPaI when applicable, or 1 otherwise. The connectivity supermask $\boldsymbol{C}$ requires one bit per underlying weight (i.e., $|\boldsymbol{C}| = |\boldsymbol{W}_{\text{rand}}|$). The sign supermask $\boldsymbol{S}$ also requires one bit per active weight and can be nested under $\boldsymbol{C}$ when sparse by only storing the $\boldsymbol{S}$ elements non-pruned by $\boldsymbol{C}$ (i.e., $|\boldsymbol{S}| = |\boldsymbol{W}_{\text{rand}}| \cdot k_0$). The magnitude supermask $\boldsymbol{M}$ is represented using nested binary coats, yielding $|\boldsymbol{M}| = |\boldsymbol{W}_{\text{rand}}| \sum_{n=0}^{N-2} k_n$, where k_n denotes the density of the n-th coat. Although these sparse binary representations can be further compressed via lossless entropy coding, we do not apply it here to isolate algorithmic compression from implementation-dependent gains.

B. Supermasks Across the Sparsity Spectrum

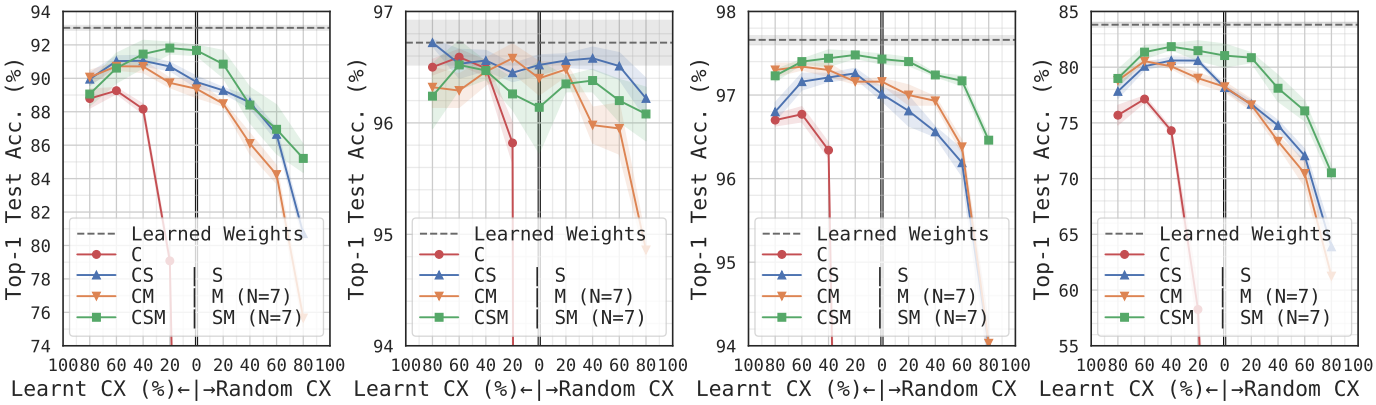
We first examine the behavior of supermasks across the full sparsity spectrum, using either learned or random connectivity. Dense connectivity (0% sparsity) is plotted as the midpoint of a unified sparsity axis. Following prior work, experiments use a ResNet-50 [34] trained on CIFAR-100 [35].

Fig.s 3a and 3b compare all seven supermask configurations under Kaiming Normal (KN) and Signed Kaiming (SK) initializations, respectively. Consistent with prior observations, SK initialization yields higher accuracy across all configurations.



(a) KN initialization (b) SK initialization (c) Folded, KN initialization (d) Folded, SK initialization

Fig. 3: Supermasks compared on ResNet-50 and CIFAR-100 with connectivity (CX) that is learned (left semiaxis), dense (0%), and random (right semiaxis), expressed in sparsity. Note that at random CX, the *C* is lost (e.g., *CSM* becomes *SM*).



(a) CIFAR-10 (b) SVHN (c) K-49 (d) CINIC-10

Fig. 4: Comparison across vision datasets using ResNet-56 and all supermasks types on the same sparsity axis as above.

When connectivity is learned, augmenting the connectivity mask $\boldsymbol{C}$ with sign or magnitude learning (*CS* or *CM*) consistently improves accuracy and expands the usable sparsity range. At the point of dense connectivity, masks retain most of their performance, indicating that topology plays a limited role when signs or magnitudes are trained.

On the random pruning side, results strongly support the T-SLTH: high accuracy can be achieved by learning only a subset of the weight structure, even when connectivity is fixed and random. While random connectivity performs slightly worse than learned connectivity, this gap may be attributed to the naive RPaI strategy used here, which lacks the structured variance induced by Kaiming initialization. More sophisticated random pruning strategies [36]–[39] may reduce this gap.

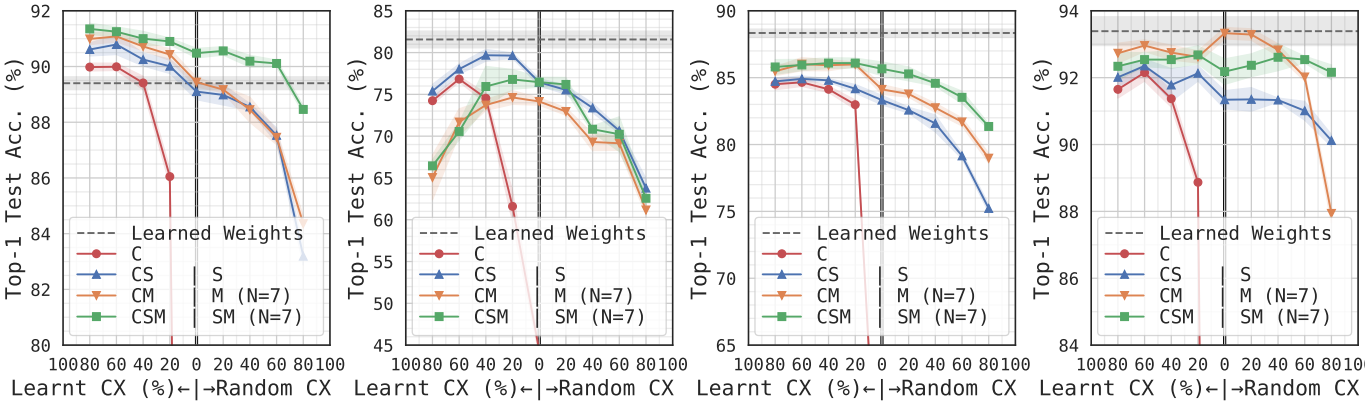
Fig.s 3c and 3d report the same comparison using folded (*F*) supermasks. Folding improves performance across all configurations, with *F-CS* at 10–70% sparsity nearly matching the learned-weight baseline. Remarkably, a randomly initialized *F-SM* mask performs on par with the best learned-connectivity supermasks (*F-C*) up to 60% sparsity.

C. Generalization Across Dataset, Architecture, and Modality

We next evaluate whether the proposed framework generalizes beyond a single dataset or architecture. Fig. 4 compares all supermask configurations across four vision benchmarks—CIFAR-10, SVHN [40], K-49 [41], and CINIC-10 [42]—using a fixed ResNet-56. Despite substantial variation in resolution, dataset size, and label granularity, supermasks maintain strong performance across all datasets.

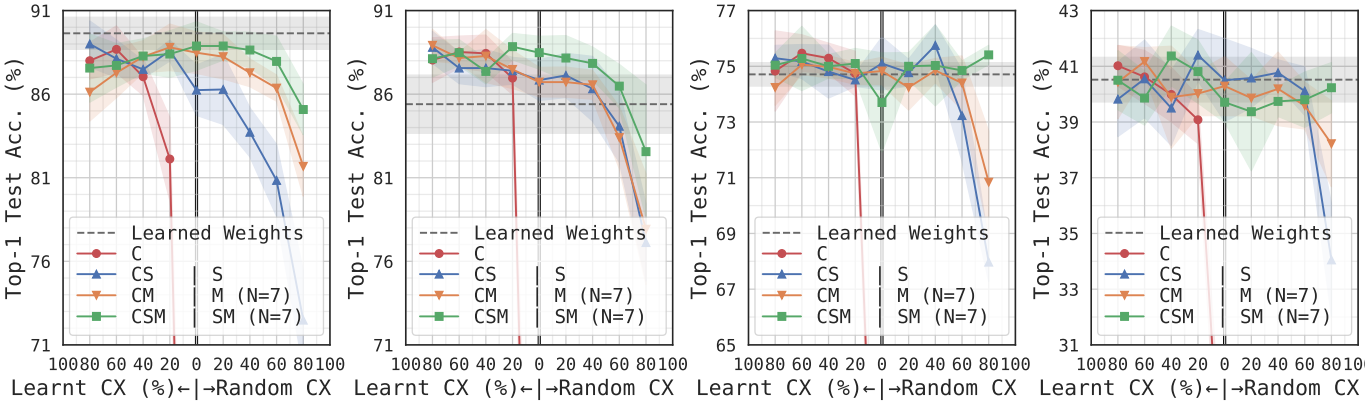
These datasets vary in size, resolution, label granularity, and domain statistics. Supermasks consistently perform well when trained independently on different datasets, demonstrating that the proposed framework applies robustly across varied data distributions. On K-49, an *SM* supermask with up to 60% random connectivity outperforms most learned-connectivity baselines while offering additional compression advantages.

Fig. 5 evaluates architectural generalization on CIFAR-10 using using four alternative vision models: Conv6 [43], MLP-Mixer [44], ViT [45], and MobileNetV2 [46]. Normalization layers were added after masked layers when required to support magnitude scaling. Across all architectures, supermasks retain competitive accuracy. Notably, on MobileNetV2, the random connectivity models outperform the learned connec-



(a) Conv6 (b) MLP-Mixer (c) ViT (d) MobileNetV2

Fig. 5: Comparison across vision architectures using CIFAR-10 and all supermasks types on the same sparsity axis as above.



(a) TextCNN on TREC10 (b) Transformer on TREC10 (c) Transformer on SST-2 (d) Transformer on SST-5

Fig. 6: Comparison across modality using NLP datasets and architectures on the same sparsity axis as above.

tivity counterparts, even matching the dense baseline. This suggests that supermasks operate independently of specific architectural priors such as CNN locality or ViT global attention.

Finally, Fig. 6 explores applicability across modalities by applying supermasks to NLP tasks: topic classification (TREC10) [47], binary sentiment (SST-2) [48], and fine-grained sentiment (SST-5), using TextCNN [49] and Transformer [50] models. Despite limited data and minimal regularization, supermasks achieve robust performance and often outperform dense baselines, particularly in Transformer settings. These results suggest that supermask-based training acts as a form of implicit regularization, mitigating overfitting without the need for additional regularization/augmentation. This is further supported by the strong results of random connectivity versions, especially in Fig. 6a and Fig. 6c. These findings are consistent with recent findings in masked Transformer and GNN models [51]–[54], suggesting that these priors extend beyond vision and text.

D. Compression Performance

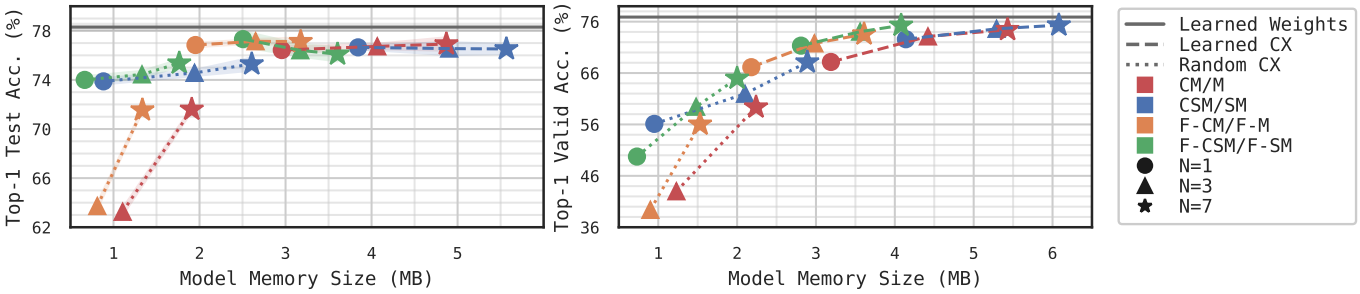
Fig.s 7a and 7b compare accuracy–size tradeoff on CIFAR-100 and ImageNet, respectively, using a 70% sparse ResNet-50. Results are grouped by the number of magnitude coats N, following the convention that *C* is a special case of *CM* where N=1 [17].

On CIFAR-100, an *F-CS* supermask achieves 77.32% accuracy in 2.51 MB, within 1% of the dense baseline (78.30%) but 38× smaller. Under random connectivity, *F-S* achieves 74.00% accuracy in 0.66 MB, yielding a 144× reduction suitable for on-chip deployment.

On ImageNet, accuracy and model size scale predictably with the number of learned primary supermasks and coats. The 7-coat *CSM* model achieves 75.32% accuracy in 6.1 MB, while folding yields a parallel tradeoff that improves both accuracy and compression:*F-CSM* matches its feedforward counterpart while raising compression to 25× (4.1 MB vs. 102.2 MB). Random-connectivity supermasks further extend this tradeoff toward ultra-compact regimes.

E. Comparison With Other QAT Methods

TABLE I compares T-SLTH against prior compression methods on ImageNet. Compared to earlier supermask-based approaches [1], [19], [55], T-SLTH achieves higher accuracy under tighter memory budgets. For example, *F-CSM* attains 75.28% accuracy in only 4.1 MB, outperforming all prior SLTH-based methods at this size.



(a) CIFAR-100 (Baseline: 95 MB)

(b) ImageNet (Baseline: 102 MB)

Fig. 7: Accuracy-size tradeoff compared on a 70% sparse ResNet-50. CX: Connectivity; F: folded.

TABLE I: QAT comparison on ImageNet and ResNet-50.

Method	Acc. (%)	W/A (bits)	Sparsity (%)	Size (MB)
Dense Baseline	**76.89**	32/32	0	**102.2**
`Edge-Popup` (*C*) ∗	68.6	1/32	70	3.1
Hiddenite (*C*) ∗	70.09	1/BFP8	70	3.1
WD (*F-CS*) ∗	71.54	2/FP8	70	2.8
PaB (≈*S*) ∗	63.58	2/32	30	5.4
LSQ	73.7	2/2	0	6.4
LCQ	**75.1**	2/2	0	**6.4**
LSQ	75.8	3/3	0	> 6.4
LCQ	76.3	3/3	0	> 6.4
LSQ	76.7	4/4	0	> 6.4
F-SM (*N*=7)∗	64.97	4/32	70	2.0
F-C∗	67.14	1/32	70	2.2
M (*N*=7)∗	59.36	3/32	70	2.2
F-CS∗	**71.33**	2/32	70	**2.8**
SM (*N*=7)∗	68.12	4/32	70	2.9
F-CSM (*N*=7)∗	**75.28**	4/32	70	**4.1**
CSM (*N*=7)∗	75.32	4/32	70	6.1

W/A: Weight/Activation precision; ∗: Supermask-based

Against non-sparse QAT methods, supermasks remains competitive. LCQ [8] and LSQ [6] require over 6.4 MB to reach comparable accuracy, despite also quantizing activations. In contrast, supermasks maintain their performance with full-precision activations, though prior work has demonstrated that supermasks remain effective even when activations are also quantized [18], [19], [56]. Random connectivity further enables deployment on substrates that do not support sparsity, such as photonic accelerators, while achieving substantially smaller memory footprints.

V. Discussion and Conclusion

This work presents a unified framework for training partially random neural networks by decomposing supermasks into three orthogonal components: connectivity, sign, and magnitude. This trichromatic view generalizes existing SLTH variants and expands the design space of supermask-based learning beyond pruning and quantization.

Across vision and language tasks, the resulting configurations—including those with fixed random connectivity—achieve accuracy comparable or superior to prior SLTH methods while approaching dense baselines at a fraction of the model size. Their consistent behavior across datasets, architectures, and modalities suggests that trichromatic supermasks capture robust structural priors, with potential implications for interpretable or transferable sparse training when retraining is constrained.

By connecting SLTH to QAT and supporting arbitrary connectivity, the proposed framework exposes the modular structure of the supermask landscape and enables systematic exploration of new training regimes. This flexibility also facilitates deployment in settings such as analog accelerators, where reconfigurable sparsity is limited, positioning T-SLTH as a practical interface for algorithm–hardware co-design.

References

[1] V. Ramanujan, M. Wortsman, A. Kembhavi, A. Farhadi, and M. Rastegari, "What's hidden in a randomly weighted neural network?" in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, 2020, pp. 11 893–11 902.

[2] Á. García-Arias, Y. Okoshi, H. Otsuka, D. Chijiwa, Y. Fujiwara, S. Takeuchi, and M. Motomura, "The trichromatic strong lottery ticket hypothesis: Neural compression with three primary supermasks," in *Workshop on Machine Learning and Compression, NeurIPS*, 2024.

[3] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," in *Proc. Adv. Neural Inform. Process. Syst.*, 2015, pp. 1135–1143.

[4] D. Blalock, J. J. G. Ortiz, J. Frankle, and J. Guttag, "What is the state of neural network pruning?" *arXiv preprint arXiv:2003.03033*, 2020.

[5] P. Micikevicius, D. Stosic, N. Burgess, M. Cornea, P. Dubey, R. Grisenthwaite, S. Ha, A. Heinecke, P. Judd, J. Kamalu *et al.*, "FP8 formats for deep learning," *arXiv preprint arXiv:2209.05433*, 2022.

[6] S. K. Esser, J. L. McKinstry, D. Bablani, R. Appuswamy, and D. S. Modha, "Learned step size quantization," in *Proc. Int. Conf. Learn. Repr.*, 2020.

[7] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," *Proc. Int. Conf. Learn. Repr.*, 2016.

[8] K. Yamamoto, "Learnable companding quantization for accurate low-bit neural networks," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, 2021, pp. 5029–5038.

[9] Z. Zhang, W. Shao, J. Gu, X. Wang, and P. Luo, "Differentiable dynamic quantization with mixed precision and adaptive resolution," in *Proc. Int. Conf. Mach. Learn.* PMLR, 2021, pp. 12 546–12 556.

[10] Z. Liu, Z. Shen, M. Savvides, and K.-T. Cheng, "ReActNet: Towards precise binary neural network with generalized activation functions," in *Proc. European Conf. Comput. Vis.* Springer, 2020, pp. 143–159.

[11] M. Neseem, C. McCullough, R. Hsin, C. Leichner, S. Li, I. S. Chong, A. Howard, L. Lew, S. Reda, V.-M. Rautio *et al.*, "PikeLPN: Mitigating overlooked inefficiencies of low-precision neural networks," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, 2024.

[12] Y. Zhang, Z. Zhang, and L. Lew, "PokeBNN: A binary pursuit of lightweight accuracy," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, 2022, pp. 12 475–12 485.

[13] S. Han, X. Liu, H. Mao, J. Pu, A. Pedram, M. A. Horowitz, and W. J. Dally, "EIE: Efficient inference engine on compressed deep neural network," *Proc. Int. Symp. Comp. Archit.*, 2016.

[14] Y. Chen, T.-J. Yang, J. S. Emer, and V. Sze, "Eyeriss v2: A flexible accelerator for emerging deep neural networks on mobile devices," *IEEE J. Emerg. Sel. Top. Circuits Syst.*, vol. 9, no. 2, pp. 292–308, 2019.

[15] A. Gaier and D. Ha, "Weight agnostic neural networks," *Proc. Adv. Neural Inform. Process. Syst.*, vol. 32, 2019.

[16] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior." *International Journal of Computer Vision*, vol. 128, no. 7, pp. 1867–1889, 2020.

[17] Y. Okoshi, Á. López García-Arias, K. Hirose, K. Ando, K. Kawamura, T. Van Chu, M. Motomura, and J. Yu, "Multicoated supermasks enhance hidden networks," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 17 045–17 055.

[18] Á. López García-Arias, Y. Okoshi, J. Yu, J. Suzuki, H. Otsuka, K. Kawamura, T. Van Chu, D. Fujiki, and M. Motomura, "WhiteDwarf: A holistic co-design approach to ultra-compact neural inference acceleration," *IEEE Access*, vol. 13, pp. 86 509–86 527, 2025.

[19] K. Hirose, J. Yu, K. Ando, Y. Okoshi, Á. López García-Arias, J. Suzuki, T. V. Chu, K. Kawamura, and M. Motomura, "Hiddenite: 4K-PE hidden network inference 4D-tensor engine exploiting on-chip model construction achieving 34.8-to-16.0TOPS/W for CIFAR-100 and ImageNet," in *Proc. IEEE Int. Solid-State Circuits Conf.*, vol. 65, 2022, pp. 1–3.

[20] S. Banerjee, M. Nikdast, S. Pasricha, and K. Chakrabarty, "Pruning coherent integrated photonic neural networks," *IEEE J. Sel. Topics Quantum Electron.*, vol. 29, no. 2: Optical Computing, pp. 1–13, 2023.

[21] A. H. Gadhikar, S. Mukherjee, and R. Burkholz, "Why random pruning is all we need to start sparse," in *Proc. Int. Conf. Mach. Learn.* PMLR, 2023, pp. 10 542–10 570.

[22] H. Otsuka, D. Chijiwa, Á. López García-Arias, Y. Okoshi, K. Kawamura, T. V. Chu, D. Fujiki, S. Takeuchi, and M. Motomura, "Partially frozen random networks contain compact strong lottery tickets," *Transactions on Machine Learning Research*, 2025.

[23] Y. Bengio, N. Léonard, and A. Courville, "Estimating or propagating gradients through stochastic neurons for conditional computation," *arXiv preprint arXiv:1308.3432*, 2013.

[24] X. Zhou, W. Zhang, H. Xu, and T. Zhang, "Effective sparsification of neural networks with global sparsity constraint," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, 2021.

[25] H. Cheng, P. Zhao, Y. Li, X. Lin, J. Diffenderfer, R. Goldhahn, and B. Kailkhura, "Efficient multi-prize lottery tickets: Enhanced accuracy, training, and inference speed," *arXiv preprint arXiv:2209.12839*, 2022.

[26] N. Koster, O. Grothe, and A. Rettinger, "Signing the supermask: Keep, hide, invert," in *Proc. Int. Conf. Learn. Repr.*, 2022.

[27] U. Evci, T. Gale, J. Menick, P. S. Castro, and E. Elsen, "Rigging the lottery: Making all tickets winners," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 2943–2952.

[28] P. Altmann, J. Schönberger, M. Zorn, and T. Gabor, "Finding strong lottery ticket networks with genetic algorithms," *arXiv preprint arXiv:2411.04658*, 2024.

[29] Á. López García-Arias, Y. Okoshi, M. Hashimoto, M. Motomura, and J. Yu, "Recurrent residual networks contain stronger lottery tickets," *IEEE Access*, vol. 11, pp. 16 588–16 604, 2023.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1026–1034.

[31] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[32] Y.-C. Chen, S. Ando, D. Fujiki, S. Takamaeda-Yamazaki, and K. Yoshioka, "HALO-CAT: A hidden network processor with activation-localized CIM architecture and layer-penetrative tiling," *arXiv preprint arXiv:2312.06086*, 2023.

[33] J. Yan, H. Ito, Y. Nagahara, K. Kawamura, M. Motomura, T. Van Chu, and D. Fujiki, "Bingogcn: Towards scalable and efficient gnn acceleration with fine-grained partitioning and slt," in *Proc. Int. Symp. Comp. Archit.*, 2025, pp. 1910–1924.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, 2016, pp. 770–778.

[35] A. Krizhevsky, "Learning multiple layers of features from tiny images," Master's thesis, Department of Computer Science, University of Toronto, Toronto, 2009.

[36] N. Lee, T. Ajanthan, and P. Torr, "SNIP: Single-shot network pruning based on connection sensitivity," in *Proc. Int. Conf. Learn. Repr.*, 2019.

[37] C. Wang, G. Zhang, and R. Grosse, "Picking winning tickets before training by preserving gradient flow," in *Proc. Int. Conf. Learn. Repr.*, 2020.

[38] H. Tanaka, D. Kunin, D. L. Yamins, and S. Ganguli, "Pruning neural networks without any data by iteratively conserving synaptic flow," in *Proc. Adv. Neural Inform. Process. Syst.*, vol. 33, 2020, pp. 6377–6389.

[39] H. Otsuka, Y. Okoshi, Á. López García-Arias, K. Kawamura, T. V. Chu, D. Fujiki, and M. Motomura, "Restricted random pruning at initialization for high compression range," *Transactions on Machine Learning Research*, 2024.

[40] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A. Y. Ng *et al.*, "Reading digits in natural images with unsupervised feature learning," in *NeurIPS workshop on deep learning and unsupervised feature learning*, no. 5. Granada, 2011, p. 7.

[41] T. Clanuwat, M. Bober-Irizar, A. Kitamoto, A. Lamb, K. Yamamoto, and D. Ha, "Deep learning for classical japanese literature," in *NeurIPS 2018 Workshop on Machine Learning for Creativity and Design*, 2018.

[42] L. N. Darlow, E. J. Crowley, A. Antoniou, and A. J. Storkey, "CINIC-10 is not ImageNet or CIFAR-10," *arXiv preprint arXiv:1810.03505*, 2018.

[43] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[44] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit *et al.*, "MLP-Mixer: An all-MLP architecture for vision," in *Proc. Adv. Neural Inform. Process. Syst.*, vol. 34, 2021, pp. 24 261–24 272.

[45] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Repr.*, 2021.

[46] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobilenetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, 2018, pp. 4510–4520.

[47] X. Li and D. Roth, "Learning question classifiers," in *COLING*, 2002.

[48] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *EMNLP*, 2013, pp. 1631–1642.

[49] Y. Kim, "Convolutional neural networks for sentence classification," in *EMNLP*, 2014.

[50] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inform. Process. Syst.*, vol. 30, 2017.

[51] S. Shen, Z. Yao, D. Kiela, K. Keutzer, and M. Mahoney, "What's hidden in a one-layer randomly weighted transformer?" in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 2914–2921.

[52] Y. Okoshi, H. Otsuka, J. Suzuki, D. Fujiki, and M. Motomura, "Unlocking the potential of extremely low-bit sparse transformers through adaptive multi-bit supermasks and random weights," in *TTODLer-FM Workshop at ICML*, 2025.

[53] J. Yan, H. Ito, Á. López García-Arias, Y. Okoshi, H. Otsuka, K. Kawamura, T. Van Chu, and M. Motomura, "Multicoated and folded graph neural networks with strong lottery tickets," in *Learning on Graphs Conference*. PMLR, 2024, pp. 11–1.

[54] H. Ito, J. Yan, H. Otsuka, K. Kawamura, M. Motomura, T. Van Chu, and D. Fujiki, "Uncovering strong lottery tickets in graph transformers: A path to memory efficient and robust graph learning," *Transactions on Machine Learning Research*, 2025.

[55] X. Chen, J. Zhang, and Z. Wang, "Peek-a-Boo: What (more) is disguised in a randomly weighted neural network, and how to find it efficiently," in *Proc. Int. Conf. Learn. Repr.*, 2022.

[56] J. Diffenderfer and B. Kailkhura, "Multi-prize lottery ticket hypothesis: Finding accurate binary neural networks by pruning a randomly weighted network," in *Proc. Int. Conf. Learn. Repr.*, 2021.