$\mathcal{D}^2$-SPARSE: NAVIGATING THE LOW DATA LEARNING REGIME WITH COUPLED SPARSE NETWORKS

Diganta Misra[α, β, λ] **& Sparsha Mishra** [α]
[α]Mila Quebec AI Institute
[β]Carnegie Mellon University
[λ]Landskape AI
{diganta.misra,sparsha.mishra}@mila.quebec

Niklas Nolte
MIT

Lu Yin
University of Aberdeen

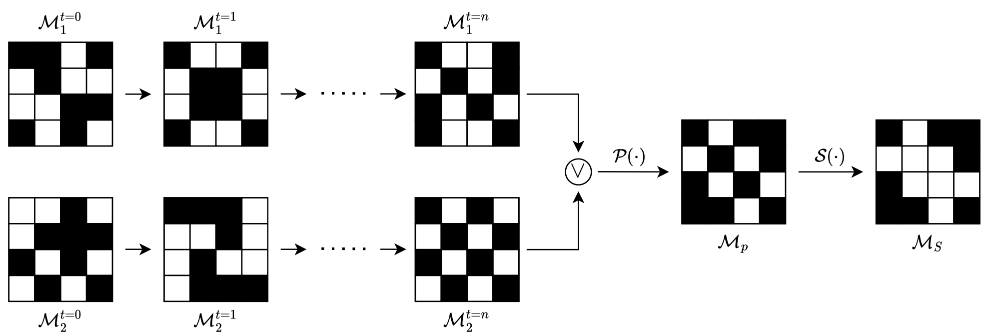

Figure 1: $\mathcal{D}^2$**-Sparse** system overview: For every state of pruning, two parallel sparse masks are co-learned before merging and subsequent final pruning to achieve state's target sparsity.

ABSTRACT

Research within the realm of deep learning has extensively delved into learning under diverse constraints, with the incorporation of sparsity as a pragmatic constraint playing a pivotal role in enhancing the efficiency of deep learning. This paper introduces a novel approach, termed $\mathcal{D}^2$**-Sparse**, presenting a dual dynamic sparse learning system tailored for scenarios involving limited data. In contrast to conventional studies that independently investigate sparsity and low-data learning, our research amalgamates these constraints, paving the way for new avenues in sparsity-related investigations. $\mathcal{D}^2$-Sparse outperforms typical iterative pruning methods when applied to standard deep networks, particularly excelling in tasks like image classification within the domain of computer vision. In particular, it achieves a notable 5% improvement in top-1 accuracy for ResNet-34 in the CIFAR-10 classification task, with only 5000 samples compared to iterative pruning methods.

1 INTRODUCTION

The investigation into sparsity as a research domain can be traced back to early contributions in signal processing and compressed sensing Gorodnitsky & Rao (1997); Rao (1998); Zibulevsky & Pearlmutter (2001). However, in the wake of the recent surge in deep learning, sparsity has emerged as a frontrunner in the discourse on efficient learning Liu et al. (2015); Srinivas et al. (2016); Zhang et al. (2016); Frankle & Carbin (2018). The increasing importance of sparsity in deep learning is tied to the continuous growth and overparameterization of deep neural networks, driven by improvements in optimization efficiency, scaling laws exploration, and computational cost reduction.

Practically, sparsity in neural networks serves different purposes: it enhances implicit parametric capacity, as seen in the Sparse Mixture of Experts (S-MoE) Shazeer et al. (2017); Riquelme et al. (2021); acts as regularization Louizos et al. (2017); Scardapane et al. (2016); and is utilized for efficiency gains Hoefler et al. (2021); Dao et al. (2022). In terms of efficiency, sparsity involves weight pruning, using structured and unstructured methods to reduce parametric complexity while maintaining performance with minimal impact compared to the full dense model.

From an intuitive perspective, sparsity can be perceived as a constraint that is applied either ad-hoc or post-hoc within the training protocol of a neural network. Within the spectrum of constraints, another fundamental one revolves around the scarcity of data volume. Research on low-data learning (Mustafa et al., 2020; Gutierrez et al., 2021; Camilleri et al., 2023; Pappu & Paige, 2020; Sanderson & Kalgonova, 2022) has occupied a significant space in the realm of learning under constraints. However, the majority of literature in this domain tends to examine constraints in isolation, thereby fundamentally constraining our comprehension in two key aspects: (i) the transferability of methods proposed to address a specific constraint when another constraint is introduced, and (ii) the impact of one constraint on another constraint within a learning problem.

Recognizing this limitation, our objective is to explore the interplay between low-data learning and sparse models. In pursuit of this goal, we introduce a pioneering dual-dynamic coupled sparse network learning framework, denoted as $\mathcal{D}^2$**-Sparse**. This framework is designed to make sparse networks feasible when subjected to the constraints of a low-data learning regime.

In brief, we succinctly summarize our contributions below.

1. Introduce a novel dual-dynamic coupled sparse network learning framework, denoted as $\mathcal{D}^2$**-Sparse**, specifically crafted for training sparse models in low-data learning scenarios.
2. Offer inaugural insights and evidence on complementary sparse learning systems.
3. Conduct comprehensive experiments to show the efficacy of $\mathcal{D}^2$-Sparse across diverse data budgets and varying levels of sparsity.
4. Provide a detailed analysis of the robustness and calibration of the trained models within the $\mathcal{D}^2$-Sparse framework.

2 Related Work

2.1 Sparse Training

Sparse neural network training aims to train initial sparse networks from scratch, achieving competitive performance with dense counterparts using fewer resources. It is commonly categorized into static sparse training (SST) and dynamic sparse training (DST), depending on whether the sparse connectivity remains static or changes dynamically during training.

Static sparse training involves methods that train initial sparse neural networks with a fixed sparse connectivity pattern throughout the training. Although the sparse connectivity remains static, the choices for layer-wise sparsity vary. Approaches range from uniform sparsity (Gale et al., 2019) to non-uniform methods like those of Mocanu et al. (2016), showing improved performance in Restricted Boltzmann Machines (RBMs). Other strategies, such as the use of expander graphs, demonstrate comparable performance to dense CNNs (Prabhu et al., 2018; Kepner & Robinett, 2019).Inspired by the graph theory, *Erdős-Rényi* (ER) (Mocanu et al., 2018) and its CNNs variant *Erdős-Rényi-Kernel* (ERK) (Evci et al., 2020) allocates lower sparsity to smaller layers, avoiding the layer collapse problem (Tanaka et al., 2020) and achieving stronger results than uniform sparsity in general.

Dynamic sparse training (DST) dynamically adjusts sparse neural network connectivity during training. Originating from Sparse Evolutionary Training (SET) (Mocanu et al., 2018), DST includes weight redistribution to optimize layer-wise sparsity ratios (Mostafa & Wang, 2019; Dettmers & Zettlemoyer, 2019). Commonly using magnitude pruning, DST varies in weight regrowth criteria. Gradient-based regrowth, such as momentum (Dettmers & Zettlemoyer, 2019), excels in image classification, while random regrowth outperforms in language modeling (Dietrich et al., 2021). Recent advances aim to improve accuracy by relaxing memory constraints (Jayakumar et al., 2020; Yuan et al., 2021; Liu et al., 2021; Huang et al., 2022). Yin et al. (2022b) introduced Suptickets, an ensemble framework that surpasses the generalization of dense models by averaging weights and sparse connections. Building on this, Yin et al. (2022a) extend the method to interpretation, further boosting performance.

Weight Averaging. Explored in convex optimization and neural networks (Polyak & Juditsky, 1992; Zhang et al., 2019), methods such as stochastic weight averaging (SWA)(Izmailov et al., 2018) and Exponential Moving Average (EMA)(Polyak & Juditsky, 1992) average model checkpoints within an optimization trajectory, achieving ensemble-level performance. (Yin et al., 2022b) introduced SWA in sparse training without pre-training, and Greedy soup (Wortsman et al., 2022) improved performance by averaging independent dense models. Weight interpolation, beyond simple averaging, gained attention. Empirical evidence from (Nagarajan & Kolter, 2019) revealed a linear path between solutions for the MNIST dataset starting from identical initializations. Subsequently, the linear interconnection of models fine-tuned from the same pre-trained model yielded equivalent performance (Neyshabur et al., 2020). Linear mode connectivity, introduced by (Frankle et al., 2020) and applied in (Yin et al., 2023), showed that interpolating between linearly connected subnetworks results in a more accurate network without extra costs.

3 $\mathcal{D}^2$-SPARSE

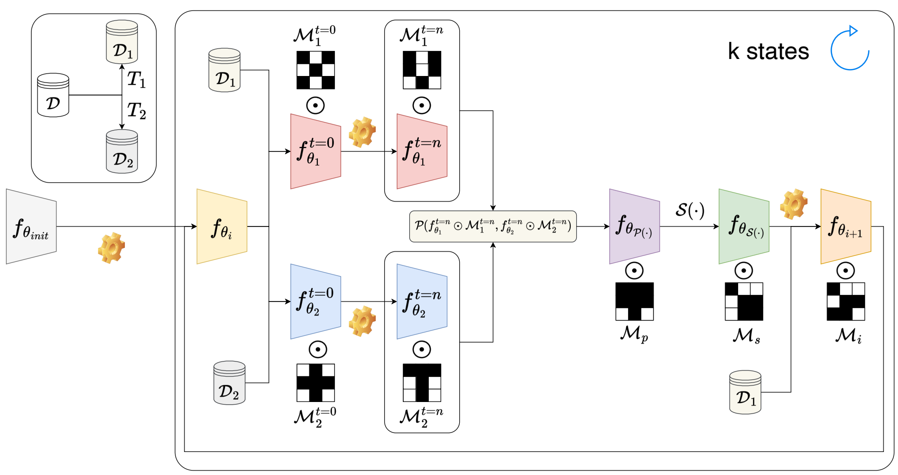

Figure 2: $\mathcal{D}^2$**-Sparse** learning framework

We illustrate the architecture of the $\mathcal{D}^2$-Sparse training framework in Fig. 2.

Preliminaries: Commencing with a randomly initialized dense model $f_{\theta_{init}}$ and a dataset $\mathcal{D}$ with limited data volume, our goal is to iteratively train $f_{\theta_{init}}$ using sparse training on $\mathcal{D}$ across $\mathtt{k}$ states, each corresponding to a specific target sparsity.

Warm-up: Initially, we create two transformed copies of $\mathcal{D}$, each transformed using different spatial augmentation strategies denoted as T_1 and T_2. The resulting low-data volumes are referred to as $\mathcal{D}_1$ and $\mathcal{D}_2$. Subsequently, we warm-up train $f_{\theta_{init}}$ on $\mathcal{D}$ for a limited number of iterations. This ensures that the model utilized for sparse training initiates from a non-random state. The resulting model from warm-up fine-tuning is denoted as f_{θ_i}, where i represents the state index ($i \in \mathtt{k}$).

Dual Branch: Next, f_{θ_i} is duplicated into two branches, each trained on $\mathcal{D}_1$ and $\mathcal{D}_2$. The initial copied models are denoted as $f_{\theta_1}^{t=0}$ and $f_{\theta_2}^{t=0}$. Two random sparse binary matrices, $\mathcal{M}_1^{t=0}$ and $\mathcal{M}_2^{t=0}$, are initialized for each model at the predefined state sparsity δ. Independently, $f_{\theta_1}^{t=0}$ and $f_{\theta_2}^{t=0}$ are trained on $\mathcal{D}_1$ and $\mathcal{D}_2$ for n iterations, evolving masks $\mathcal{M}_1^{t=0}$ and $\mathcal{M}_2^{t=0}$ using the ERK algorithm (Evci et al., 2020).The resulting trained models and updated masks are denoted as ($f_{\theta_1}^{t=n}$, $\mathcal{M}_1^{t=n}$) and ($f_{\theta_2}^{t=n}$, $\mathcal{M}_2^{t=n}$), respectively.

Merging: To merge the two independently trained sparse models and masks, ($f_{\theta_1}^{t=n}$, $\mathcal{M}1^{t=n}$) and ($f\theta_2{}^{t=n}$, $\mathcal{M}2^{t=n}$), we use a weight merging function denoted $\mathcal{P}(\cdot)$. The resulting model and mask after merging are denoted as $f_{\theta_{\mathcal{P}(\cdot)}}$ and $\mathcal{M}_{\mathcal{P}}$, respectively.

Dynamic Finetuning: Following the merging with $\mathcal{P}(\cdot)$, the resulting sparse model $f_{\theta_{\mathcal{P}(\cdot)}}$ may deviate from the target state sparsity δ. To address this, a one-shot sparsification step $\mathcal{S}(\cdot)$ is applied using the SNIP pruning algorithm (Lee et al., 2019). The resulting sparse model and mask, denoted as $f_{\theta_{\mathcal{S}(\cdot)}}$ and $\mathcal{M}_{\mathcal{S}}$, undergo further minimal fine-tuning on $\mathcal{D}_1$ using the ERK algorithm (Evci et al., 2020). This process yields the final sparse model and mask for the state i, denoted as $f_{\theta_{i+1}}$ and $\mathcal{M}_i$, with $f_{\theta_{i+1}}$ serving as the initialization for the next state $i+1$.

What is the reasoning for having two independent dynamic sparse training systems?

Our goal is to obtain two complementary sparse models through the process, ensuring that, upon merging, the resulting model is stronger than the two individual sparse models. We provide more insight into complementary sparse learning systems in Appendix A.

4 Evaluation

4.1 Experimental Setup

We evaluated the proposed $\mathcal{D}^2$-Sparse training framework on the ResNet (Krizhevsky & Hinton, 2009) family models (R-18, 34) using the CIFAR-10 (Krizhevsky, 2009) dataset. Both iterative and 1-shot sparse training variants are evaluated across different data fractions (0.5%, 1%, 2%, 5% and 10% of the total dataset size). The performance of the models obtained through the $\mathcal{D}^2$ -Sparse training is compared to Baseline and Baseline LB.

To ensure uniformity throughout the paper, our objective is to establish clear definitions of specific terminologies that are frequently used in reference to the proposed method and its corresponding evaluation.

- **Baseline LB.** This denotes the lower bound, where the model f_θ is trained using the ERK algorithm barring a dual branch, avoiding any merging or subsequent sparsification step.
- **Baseline:** This signifies training f_θ using the same process as Baseline LB. However, instead of training solely on $\mathcal{D}$, we train it on the joint dataset $\mathcal{D}_1 \cup \mathcal{D}_2$. This ensures that each backward pass is computed after two consecutive forward propagations on batches from each of the two datasets. This is done to maintain a consistent data volume with $\mathcal{D}^2$-Sparse for a fair comparison.
- **Dense:** This refers to the model f_θ being trained in a non-iterative manner in full dense capacity on $\mathcal{D}$ for total iterations of e which is the same for each sparsity state k' training iteration budget. Thus, the dense model is trained with $\approx \frac{1}{k}$ iterations budget compared to the iterative sparse variants.

4.2 Hyperparameters

Consistent hyperparameters are maintained across all discussed settings. In the warm-up phase, the total number of epochs is fixed at 25. At each sparsity state i, the sparse models undergo 50 epochs of training. For $\mathcal{D}^2$-Sparse, the subsequent sparsification and fine-tuning post-merging involve 5 epochs. Weight merging $\mathcal{P}(\cdot)$ uses a simple weight interpolation method, with the interpolation coefficients determined through a grid search over the values $[0.1, 0.25, 0.5, 0.75, 0.95]$. The final target sparsity for the k-th state is set to 5%, with a total of 10 states. Additionally, each configuration is run for 3 seeds for statistical significance. Different dataset fractions are constructed by uniformly subsetting across classes, ensuring the same partition for every training configuration with that specific data budget. For the 1-shot variants, we do only one state-based sparse training at 5% sparsity.

Ablation results on varying hyperparameters are provided in the Appendices B.1 and B.2.

4.3 Results

As illustrated in the detailed performance analysis presented in Figs. 3 and 4, both the 1-shot and iterative variants of the $\mathcal{D}^2$-Sparse training framework exhibit consistently superior performance compared to the Baseline and Baseline LB methods. The performance is particularly notable at a mere 0.5% data fraction on ResNet-34 (5% sparsity), where the iterative variant of $\mathcal{D}^2$-Sparse showcases a remarkable top-1 accuracy improvement of over 7%, outperforming both Baseline and Baseline LB.

This superior performance extends to various experimental settings, providing a robust and consistent profile. In the 1-shot variant, specifically at a 2% data capacity for ResNet-50, $\mathcal{D}^2$-Sparse demonstrates a significant performance enhancement of approximately 5% over Baseline and an impressive 8% improvement over Baseline LB. These compelling results offer strong evidence supporting the efficacy and effectiveness of the proposed $\mathcal{D}^2$-Sparse training approach.

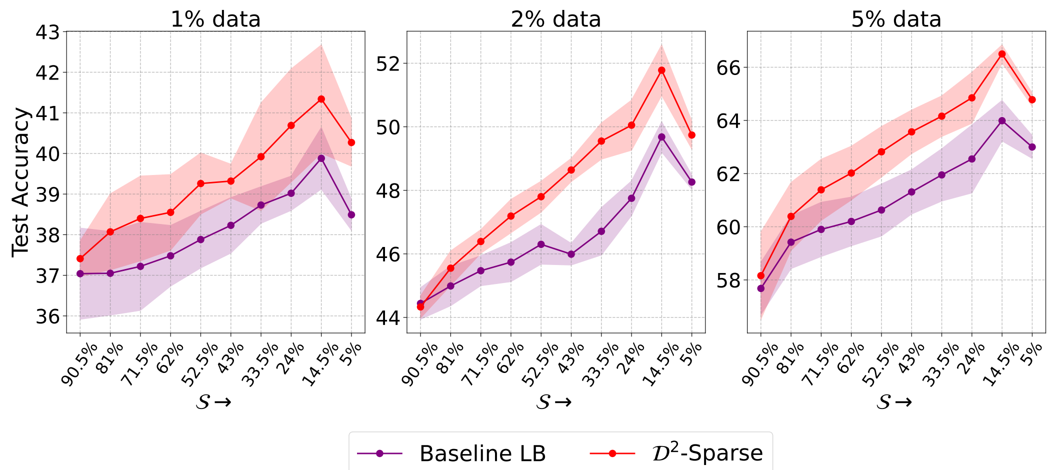

Figure 3: Results on ResNet-18 with CIFAR-10 dataset

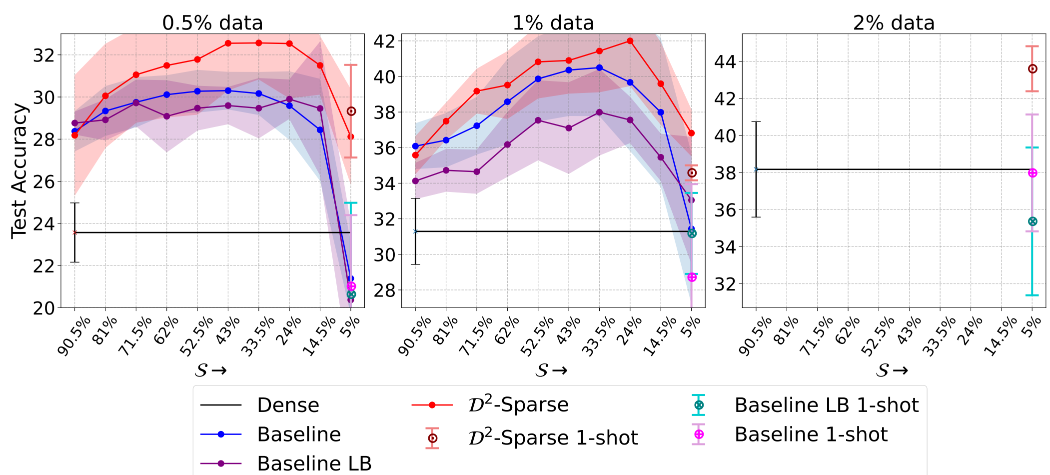

Figure 4: Results on ResNet-34 with CIFAR-10 dataset

Additonal ablation experiments on robustness and calibration are provided in the appendix B.2.

5 Conclusion

Exploring the interplay between different constraints in machine learning remains a crucial but often overlooked avenue of research. Recognizing this, our objective is to delve into the intricate dynamics between sparsity and the challenges posed by a low-data learning regime. To accomplish this, we introduce a novel approach, the $\mathcal{D}^2$-Sparse, a dual dynamic coupled sparse training framework.

This novel framework was conceived with the aim of unraveling the complex relationships between sparsity and the constraints imposed by limited data availability. Through a meticulous and extensive set of experiments, we aim to demonstrate not only the efficacy but also the superiority of our proposed $\mathcal{D}^2$-Sparse framework compared to baseline methods, especially when dealing with varying data budgets.

Although our current findings are rooted in small-scale settings, we view them as the foundation for a more comprehensive and rigorous exploration. Our future endeavors will involve expanding this work to conduct in-depth studies, benchmarking $\mathcal{D}^2$-Sparse across a diverse array of model families and datasets. This ambitious trajectory aims to elevate the depth and applicability of our research, providing valuable insights into the interplay of constraints in machine learning scenarios.

REFERENCES

Samuel K. Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries, 2023.

Elahe Arani, Fahad Sarfraz, and Bahram Zonooz. Learning fast, learning slow: A general continual learning method based on complementary learning system, 2022.

Sam Blakeman and Denis Mareschal. A complementary learning systems approach to temporal difference learning, 2019.

Romain Camilleri, Andrew Wagenmaker, Jamie Morgenstern, Lalit Jain, and Kevin Jamieson. Fair active learning in low-data regimes, 2023.

Tri Dao, Beidi Chen, Kaizhao Liang, Jiaming Yang, Zhao Song, Atri Rudra, and Christopher Ré. Pixelated butterfly: Simple and efficient sparse training for neural network models. *ICLR*, 2022.

Tim Dettmers and Luke Zettlemoyer. Sparse networks from scratch: Faster training without losing performance. *arXiv preprint arXiv:1907.04840*, 2019.

Anastasia Dietrich, Frithjof Gressmann, Douglas Orr, Ivan Chelombiev, Daniel Justus, and Carlo Luschi. Towards structured dynamic sparse pre-training of bert. *arXiv preprint arXiv:2108.06277*, 2021.

Utku Evci, Trevor Gale, Jacob Menick, Pablo Samuel Castro, and Erich Elsen. Rigging the lottery: Making all tickets winners. In *International Conference on Machine Learning*, pp. 2943–2952. PMLR, 2020.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv: Arxiv-1803.03635*, 2018.

Jonathan Frankle, Gintare Karolina Dziugaite, Daniel Roy, and Michael Carbin. Linear mode connectivity and the lottery ticket hypothesis. In *International Conference on Machine Learning*, pp. 3259–3269. PMLR, 2020.

Trevor Gale, Erich Elsen, and Sara Hooker. The state of sparsity in deep neural networks. *arXiv preprint arXiv:1902.09574*, 2019.

Irina F Gorodnitsky and Bhaskar D Rao. Sparse signal reconstruction from limited data using focuss: A re-weighted minimum norm algorithm. *IEEE Transactions on signal processing*, 45(3):600–616, 1997.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR, 2017.

Pierre Gutierrez, Antoine Cordier, Thaïs Caldeira, and Théophile Sautory. Data augmentation and pre-trained networks for extremely low data regimes unsupervised visual inspection, 2021.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations, 2019.

Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty, 2020.

T. Hoefler, Dan Alistarh, Tal Ben-Nun, Nikoli Dryden, and Alexandra Peste. Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks. *Journal Of Machine Learning Research*, 2021.

Tiansheng Huang, Shiwei Liu, L Shen, Fengxiang He, Weiwei Lin, and Dacheng Tao. On heterogeneously distributed data, sparsity matters. In *Submitted to The Tenth International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=AT0K-SZ3QGq`.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

Siddhant Jayakumar, Razvan Pascanu, Jack Rae, Simon Osindero, and Erich Elsen. Top-kast: Top-k always sparse training. *Advances in Neural Information Processing Systems*, 33:20744–20754, 2020.

Jeremy Kepner and Ryan Robinett. Radix-net: Structured sparse matrices for deep neural networks. In *2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pp. 268–274. IEEE, 2019.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited, 2019.

A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

Jung H. Lee. Dynmat, a network that can learn after learning, 2019.

Namhoon Lee, Thalaiyasingam Ajanthan, and Philip H. S. Torr. Snip: Single-shot network pruning based on connection sensitivity, 2019.

Baoyuan Liu, Min Wang, Hassan Foroosh, Marshall Tappen, and Marianna Penksy. Sparse convolutional neural networks. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 806–814, 2015. doi: 10.1109/CVPR.2015.7298681.

Shiwei Liu, Tianlong Chen, Xiaohan Chen, Zahra Atashgahi, Lu Yin, Huanyu Kou, Li Shen, Mykola Pechenizkiy, Zhangyang Wang, and Decebal Constantin Mocanu. Sparse training via boosting pruning plasticity with neuroregeneration. *Advances in Neural Information Processing Systems.*, 2021.

Christos Louizos, Max Welling, and Diederik P. Kingma. Learning sparse neural networks through l_0 regularization. *arXiv preprint arXiv: 1712.01312*, 2017.

Decebal Constantin Mocanu, Elena Mocanu, Phuong H. Nguyen, Madeleine Gibescu, and Antonio Liotta. A topological insight into restricted boltzmann machines. *Machine Learning*, 104(2): 243–270, Sep 2016.

Decebal Constantin Mocanu, Elena Mocanu, Peter Stone, Phuong H Nguyen, Madeleine Gibescu, and Antonio Liotta. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science. *arXiv:1707.04780. Nature communications.*, 9(1):2383, 2018.

Hesham Mostafa and Xin Wang. Parameter efficient training of deep convolutional neural networks by dynamic sparse reparameterization. *International Conference on Machine Learning*, 2019.

Basil Mustafa, Carlos Riquelme, Joan Puigcerver, André Susano Pinto, Daniel Keysers, and Neil Houlsby. Deep ensembles for low-data transfer learning, 2020.

Vaishnavh Nagarajan and J Zico Kolter. Uniform convergence may be unable to explain generalization in deep learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Behnam Neyshabur, Hanie Sedghi, and Chiyuan Zhang. What is being transferred in transfer learning? *Advances in neural information processing systems*, 33:512–523, 2020.

Hung Nguyen and Morris Chang. Complementary ensemble learning, 2021.

Aneesh Pappu and Brooks Paige. Making graph neural networks worth it for low-data molecular machine learning, 2020.

Quang Pham, Chenghao Liu, and Steven Hoi. Dualnet: Continual learning, fast and slow, 2021.

Quang Pham, Chenghao Liu, and Steven C. H. Hoi. Continual learning, fast and slow, 2023.

Boris T Polyak and Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM journal on control and optimization*, 30(4):838–855, 1992.

Ameya Prabhu, Girish Varma, and Anoop Namboodiri. Deep expander networks: Efficient deep networks from graph theory. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 20–35, 2018.

Bhaskar D Rao. Signal processing with the sparseness constraint. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, volume 3, pp. 1861–1864. IEEE, 1998.

Carlos Riquelme, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, and Neil Houlsby. Scaling vision with sparse mixture of experts. *Advances in Neural Information Processing Systems*, 34:8583–8595, 2021.

Dominic Sanderson and Tatiana Kalgonova. Maintaining performance with less data, 2022.

Simone Scardapane, D. Comminiello, A. Hussain, and A. Uncini. Group sparse regularization for deep neural networks. *NEUROCOMPUTING*, 2016. doi: 10.1016/j.neucom.2017.02.029.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL `https://openreview.net/forum?id=B1ckMDqlg`.

Suraj Srinivas, Akshayvarun Subramanya, and R. Venkatesh Babu. Training sparse neural networks. *Ieee Conference On Computer Vision And Pattern Recognition Workshops (cvprw)*, 2016. doi: 10.1109/CVPRW.2017.61.

Hidenori Tanaka, Daniel Kunin, Daniel LK Yamins, and Surya Ganguli. Pruning neural networks without any data by iteratively conserving synaptic flow. *Advances in Neural Information Processing Systems. arXiv:2006.05467*, 2020.

Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. *arXiv preprint arXiv:2203.05482*, 2022.

Jay Zhangjie Wu, David Junhao Zhang, Wynne Hsu, Mengmi Zhang, and Mike Zheng Shou. Label-efficient online continual object detection in streaming video, 2023.

Lu Yin, Shiwei Liu, Fang Meng, Tianjin Huang, Vlado Menkovski, and Mykola Pechenizkiy. Lottery pools: Winning more by interpolating tickets without increasing training or inference cost. *arXiv preprint arXiv:2208.10842*, 2022a.

Lu Yin, Vlado Menkovski, Meng Fang, Tianjin Huang, Yulong Pei, and Mykola Pechenizkiy. Superposing many tickets into one: A performance booster for sparse neural network training. In *Uncertainty in Artificial Intelligence*, pp. 2267–2277. PMLR, 2022b.

Lu Yin, Shiwei Liu, Meng Fang, Tianjin Huang, Vlado Menkovski, and Mykola Pechenizkiy. Lottery pools: Winning more by interpolating tickets without increasing training or inference cost. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 10945–10953, 2023.

Geng Yuan, Xiaolong Ma, Wei Niu, Zhengang Li, Zhenglun Kong, Ning Liu, Yifan Gong, Zheng Zhan, Chaoyang He, Qing Jin, et al. Mest: Accurate and fast memory-economic sparse training framework on the edge. *Advances in Neural Information Processing Systems*, 34, 2021.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features, 2019.

Michael Zhang, James Lucas, Jimmy Ba, and Geoffrey E Hinton. Lookahead optimizer: k steps forward, 1 step back. *Advances in Neural Information Processing Systems*, 32, 2019.

Shijin Zhang, Zidong Du, Lei Zhang, Huiying Lan, Shaoli Liu, Ling Li, Qi Guo, Tianshi Chen, and Yunji Chen. Cambricon-x: An accelerator for sparse neural networks. In *2016 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pp. 1–12, 2016. doi: 10.1109/MICRO.2016.7783723.

Xiang Zhang, Ziyuan Zhao, Theodoros Tsiligkaridis, and Marinka Zitnik. Self-supervised contrastive pre-training for time series via time-frequency consistency, 2022.

Michael Zibulevsky and Barak Pearlmutter. Blind source separation by sparse decomposition in a signal dictionary. *Neural computation*, 13:863–82, 05 2001. doi: 10.1162/089976601300014385.

A Complementary Sparse Learning

The exploration of Complementary Learning Systems stands as a pivotal paradigm, offering essential insights into how humans acquire knowledge from limited experiences and data. This area has garnered significant attention, with extensive contributions emerging from both the neuroscience and machine learning perspectives (Blakeman & Mareschal, 2019; Arani et al., 2022; Nguyen & Chang, 2021; Lee, 2019; Pham et al., 2023; Wu et al., 2023; Pham et al., 2021).

At its core, the Complementary Learning Systems framework can be intuitively understood through the lens of information maximization. The overarching objective within this paradigm is to extract the maximum information from a given context or sample. An illustrative example is found in the work of Zhang et al. (2022), where a contrastive time-series model was proposed. In this model, two complementary views of the same time-series sample, namely, the frequency domain obtained through a Fourier transform and the original temporal domain, were utilized. Although the total information content of the time-series sample remains constant, incorporating the extra frequency domain provides an additional perspective for the model to learn more features about the sample, thereby accelerating the learning process. This conceptualization underscores the essence of information maximization within the framework of Complementary Learning Systems.

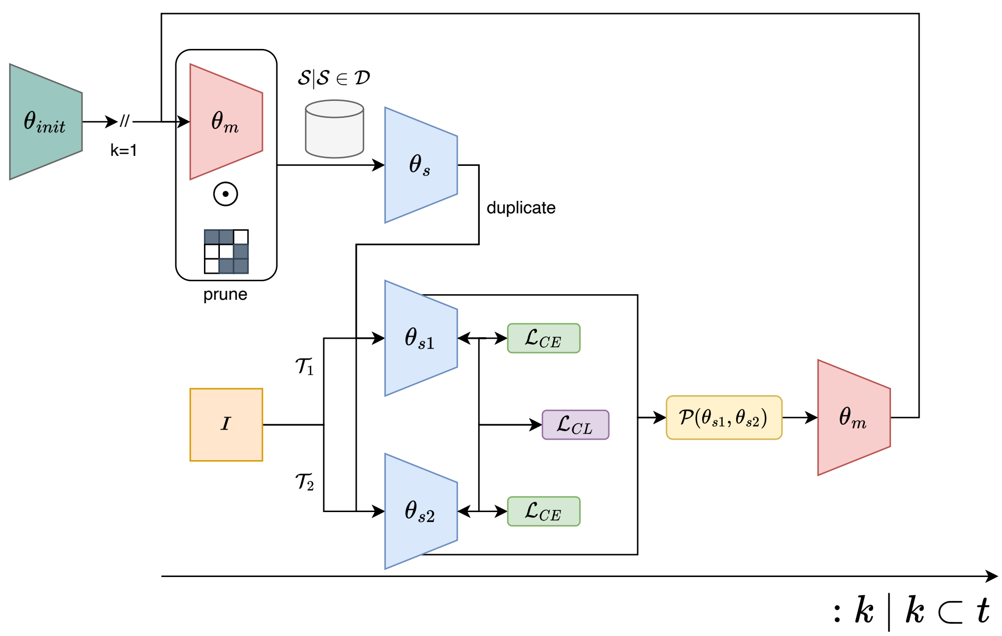

Figure 5: Schematic of the proposedl CSL framework.

Motivated by this concept, prior to crafting the $\mathcal{D}^2$-Sparse framework, we developed the Complementary Sparse Learning (CSL) framework, as illustrated in Fig. 5. Although most components remain consistent with $\mathcal{D}^2$-Sparse, the key modification lies in the incorporation of the complementary loss, denoted as $\mathcal{L}_{CL}$. Unlike $\mathcal{D}^2$-Sparse, where the two branches undergo independent training, our objective in CSL is to train two sparse models with individual cross-entropy losses ($\mathcal{L}_{CE}$) alongside a shared complementary loss.

The complementary loss serves the purpose of promoting diversity by encouraging the feature representations or weights of the two models to diverge. Various loss candidates can fulfill this role, with the simplest being an L2 penalty applied to the distance computed between the pre-final layer outputs (feature vectors) of each sparse model. Consequently, the overall training process can be conceptualized as a Minmax game, where each model aims to minimize its individual $\mathcal{L}_{CE}$ while simultaneously maximizing the joint $\mathcal{L}_{CL}$ - effectively maximizing the distance between their respective feature representations.

Alternative candidates for the complementary loss include minimizing the diagonal of neural representation similarity metrics such as Canonical Correlation Analysis (CCA) and Centered Kernel Alignment (CKA) (Kornblith et al., 2019). These approaches offer diverse avenues for promoting complementarity during the training process.

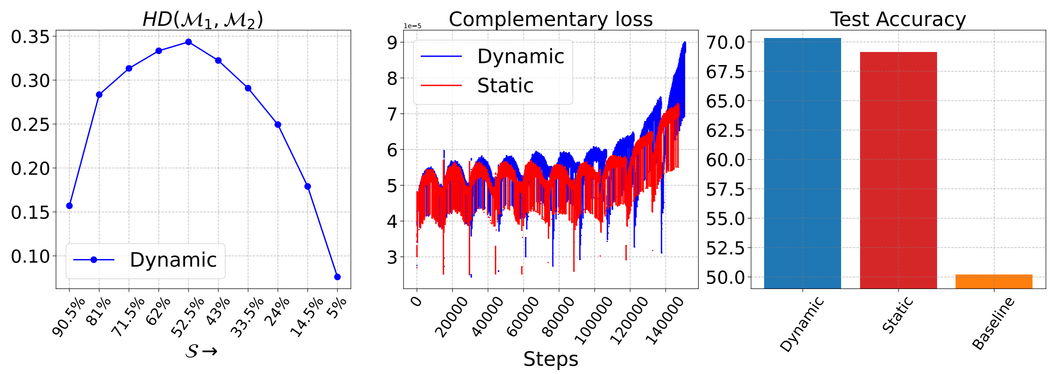

Figure 6: From left to right: (a) Hamming distance between the sparse masks of the two models. (b) L2-complementary loss profile. (c) Test accuracy on 50% fraction of CIFAR-10 using ResNet-34.

Using the L2 penalty as the complementary loss function, we conducted experiments using a ResNet-34 on a 50% data fraction of CIFAR-10. Diverging from the $\mathcal{D}^2$-Sparse approach, we initiated our exploration with *static* masks. In this context, static masks are set during initialization through pruning using SNIP (Lee et al., 2019) for each state. Only the unpruned parameters are allowed to be trained. The static mask is duplicated for both branches, eliminating the need for subsequent sparsification and fine-tuning after merging, as the merged model consistently maintains the same sparsity as the two individual models.

In contrast, the *dynamic* variant introduces ERK (Evci et al., 2020)-based mask learning, mirroring the approach in $\mathcal{D}^2$-Sparse. As depicted in Fig. 6, the dynamic variant achieves the highest test accuracy, and the complementary loss exhibits a desirable positive increasing profile, indicating optimal behavior where we aim for an increasing distance between their feature representations. However, two primary concerns arose during this exploration, prompting the decision to abandon the framework based on the complementary loss function.

First, as illustrated in Fig. 6 (a), the Hamming distance between the masks of the two models decreases with increasing sparsity. This suggests a higher overlap in the sparse patterns identified by each model, undermining the intended diversity. Second, the training process proved to be quite unstable, and tuning the loss coefficients posed significant challenges. These concerns collectively contributed to the decision to forego the use of the complementary loss function-based framework.

Hence, given the optimal and stable performance demonstrated by $\mathcal{D}^2$-Sparse, coupled with the observed positive trend in the Hamming distance profile between the learned masks of the two models, as shown in Fig. 8, we made the strategic decision to deviate from incorporating a complementary loss function into our framework.

B Additional Results

B.1 Ablation experiments

In addition to test accuracy, we assess the calibration metric for the sparse models generated by our proposed framework, $\mathcal{D}^2$-Sparse, in comparison to the baseline method. Expected calibration error (ECE) serves as the primary metric for computing the calibration scores of the sparse models on the held-out test set.

The formulation Guo et al. (2017) for ECE we use for our computations is given by,

$$\mathrm{ECE} = \sum_{b=1}^{B} \frac{|M_b|}{N} \; |\mathrm{acc}(b) - \mathrm{conf}(b)|$$

where B is the total number of bins, $|M_b|$ is the number of predictions in bin b, N is the total number of samples, and $\mathrm{acc}(b)$ and $\mathrm{conf}(b)$ are the accuracy and the confidence of bin b respectively.

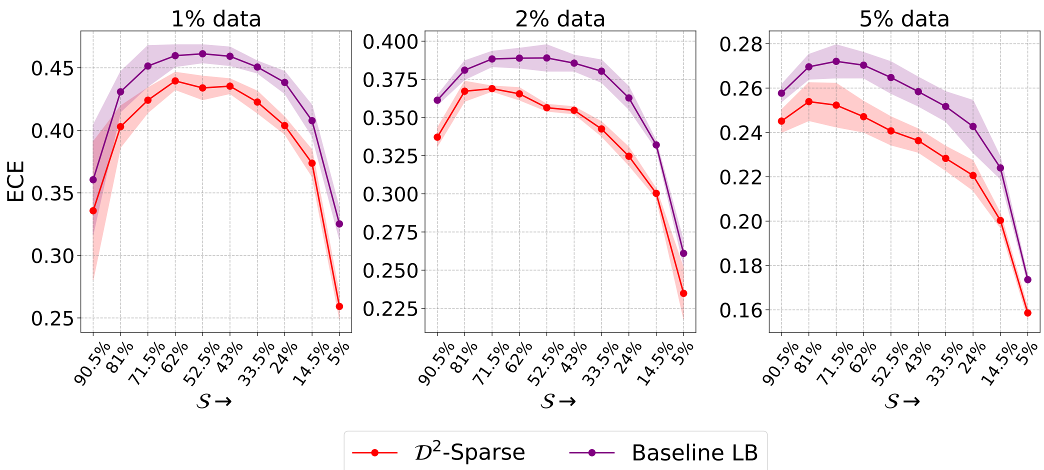

Figure 7: Calibration reported via ECE for ResNet-18 sparse models trained on CIFAR-10 dataset.

As shown in Fig. 7, $\mathcal{D}^2$-Sparse models obtain superior (lower) ECE at every sparse state across every data budget when compared to Baseline LB sparse models, suggesting their superior reliability.

B.2 Hyperparameter experiments

B.2.1 Augmentation

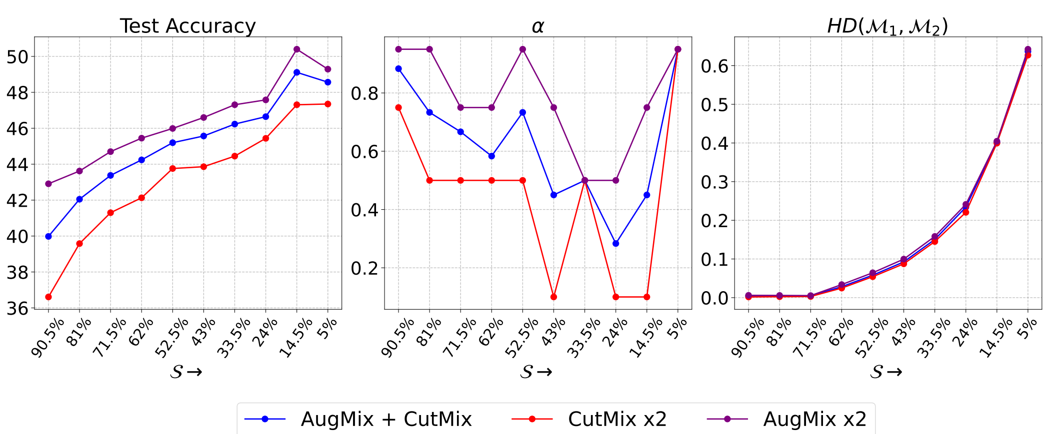

Figure 8: From left to right: (a) Test Accuracy on CIFAR-10 (2% data budget) with ResNet-34, (b) Grid-searched weight merging co-efficient (α), (c) Hamming distance between the sparse model masks.

As detailed in the main manuscript (paragraph 3), we create two transformed copies of the original data $\mathcal{D}$ by employing two unique spatial augmentation strategies, denoted as T_1 and T_2. In our experimentation, we explore two widely recognized augmentation techniques, namely CutMix (Yun et al., 2019) and AugMix (Hendrycks et al., 2020). We investigate three different configurations: utilizing CutMix for both T_1 and T_2 with varying strengths, using AugMix for both T_1 and T_2 with

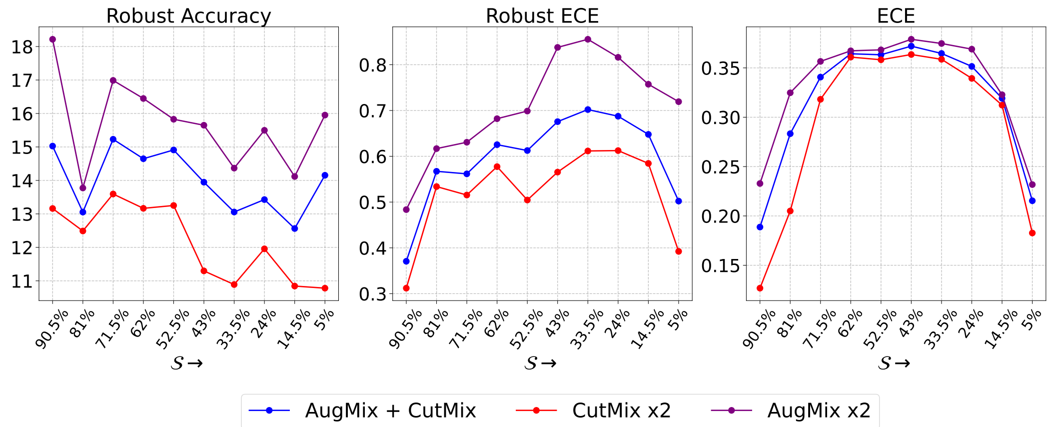

Figure 9: Robustness and Calibration results on varying T_1 and T_2 for ResNet-34 sparse training on 2% data budget of CIFAR-10.

varying strengths, and employing CutMix for T_1 and AugMix for T_2. As evident in both Figures 8 and 9, the second variant, involving AugMix with varying strengths for both T_1 and T_2, yields the highest test accuracy and the highest robust accuracy[1]. Thus, we fix the second variant as the default strategy for $\mathcal{D}^2$-Sparse.

B.2.2 Merging Strategy

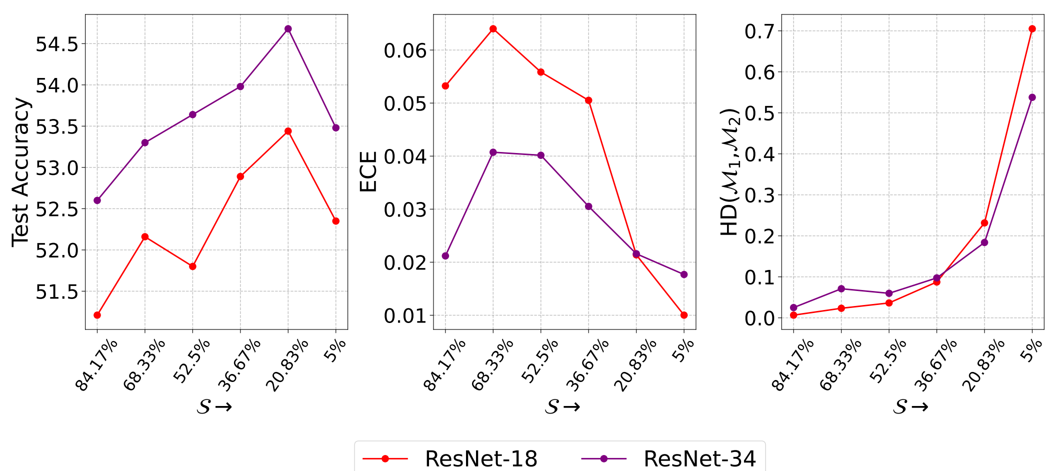

Figure 10: Git Rebasin results.

As described in the main manuscript (paragraph 3), the standard merging strategy involves a straightforward grid-search-based point-wise weight interpolation between the two sparse models. The merging function $\mathcal{P}(\cdot)$ can be mathematically expressed as:

$$\mathcal{P}(\cdot) = \alpha \times f_{\theta_1} + (1 - \alpha) \times f_{\theta_2} \tag{1}$$

where α is the optimal coefficient for the interpolation found using the grid search.

Nevertheless, we also explored GitRebasin (Ainsworth et al., 2023), a more recent and widely adopted model merging technique. Our results, presented in Figure 10, showcase our experiments with GitRebasin on ResNet-18 and ResNet-34 with a 50% data budget on CIFAR-100. Although

[1]Robust accuracy is computed on the CIFAR-10 C dataset (Hendrycks & Dietterich, 2019).

the initial findings seemed promising, we observed significant training instability, particularly when increasing the frequency of merging via GitRebasin, especially at high-sparsity states. Consequently, due to the heightened instability and the simplicity of grid search-based interpolation, we opt to retain the default approach of grid search interpolation.