Investigating Sparse Artificial Neural Network Topology through Similarity and Balance scores

Amirhossein Douzandeh Zenoozi[1,†], Christos Aronis[2,†], Antonio Liotta[1] and Lucia Cavallaro[2,*]

[1]Faculty of Engineering, Free University of Bozen, Piazza Università 1, 39100 Bolzano, Italy
[2]Institute for Computing and Information Sciences, Radboud University, Toernooiveld 200-222, 6525 EC Nijmegen, The Netherlands

Abstract

While sparse artificial neural networks offer computational efficiency, their internal structural evolution during training is often treated as a black box, masking critical phenomena like vanishing gradient flow. This discussion paper explores the use of graph theory to monitor the training dynamics of pruned Multi-Layer Perceptrons (MLPs) as they update their connections. Through an exploratory case, the training dynamics of MLPs are studied and their evolving network topology is evaluated using the DeltaCon similarity metric. The preliminary analysis herein presented suggests distinct behavioral patterns: stable learning is captured by a smooth, continuous increase in structural similarity, whereas poor generalization manifests as stunted growth and stagnating DeltaCon scores. Next, this work also introduces *local* and *global* balance scores of the network, grounded in a signed graph representation of the MLP's model parameters. Through these lens, important phases of the model training can be observed, such as convergence and overfitting (creation of redundant paths). It is worth to notice that the balance scores consistently increases as the model learns. Ultimately, graph similarity metrics and the notion of balance can provide fruitful insights into the structural signature of the network that can be used to enhance the understanding of the learning procedure.

Keywords

Graph Similarity, Sparse Artificial Neural Networks, Signed Networks, Structural Balance

1. Introduction

Deep learning has revolutionized data analysis by offering high-accuracy predictions. However, these models typically require expensive high-performance computing units. To address this issue, the use of Sparse Artificial Neural Networks (ANNs) has become increasingly popular [1, 2, 3, 4, 5]. In these networks, connections are removed using various pruning techniques. This process makes the model more efficient without significantly reducing its accuracy.

However, despite their success, deep learning models are often still treated as "black boxes" [6, 7, 8, 9] where a big part of the learning procedure remains relatively unclear. Gaining a better understanding of these internal structures can lead to significant improvements. It may allow to design more efficient networks of smaller sizes and provide faster inference time. In addition, deep insight into the model's structure might help to prevent common training issues, such as overfitting problems.

Training Sparse ANNs presents unique challenges that are not typically found in dense models (*i.e.*, where the neurons of one layer are fully connected with the ones of the adjacent layers). One such issue is the alternation of data flow and signal propagation caused by the reduced connectivity. Wang *et al.* demonstrated that sparse networks often suffer from poor gradient flow, particularly at initialization [10]. In these scenarios, the error signals necessary for learning may vanish or fail to reach earlier layers because the required pathways are absent. This impaired flow significantly hinders the model's ability

1st Workshop on Graphs Across AI: From Structural Reasoning to Foundation Models, held in conjunction with the IEEE World Congress on Computational Intelligence (WCCI 2026), June 21–26, 2026, Maastricht, The Netherlands
*Corresponding author.
†These authors contributed equally.
adouzandehzenoozi@unibz.it (A. Douzandeh Zenoozi); christos.aronis@ru.nl (C. Aronis); antonio.liotta@unibz.it (A. Liotta); lucia.cavallaro@ru.nl (L. Cavallaro)
0009-0002-2839-1202 (A. Douzandeh Zenoozi); 0009-0000-1611-1192 (C. Aronis); 0000-0002-2773-4421 (A. Liotta); 0000-0002-2367-6084 (L. Cavallaro)
© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-11-08

CEUR-WS.org/Vol-4319/paper4.pdf

to converge. To address this, dynamic approaches such as the RigL method proposed by Evci *et al.* suggest that the network topology must evolve during training [11]. By removing weak connections and growing new ones, the model can re-establish active paths for information flow. Furthermore, Fruengel *et al.* highlighted that without these adjustments, naive sparsity heavily impairs the network's information processing capabilities compared to biological systems [12]. These studies collectively emphasize that the structural topology of a sparse network is a critical factor in determining its training stability. However, monitoring this structural evolution remains a challenge, as standard metrics do not explicitly reveal the state of the underlying graph.

Beyond training dynamics, recent research suggests that the structural topology of an ANN can strongly influence its performance. You *et al.* [13] showed that model accuracy of a model is not arbitrary with respect to graph structure; rather, it varies systematically with graph properties such as the clustering coefficient and average path length. They identified an optimal range of relational graphs where the network structure naturally facilitates better learning. This supports the work of Xie *et al.* [14], in which the authors demonstrated that it is not always needed to design the network structure by hand. Instead, they used graph algorithms to wire the neurons together randomly. Specifically, they used models like *small-world* graphs, which naturally contain efficient shortcuts for data. Their findings suggest that these random structures often perform better than standard human designs because they allow information to flow more easily [14]. Furthermore, the "*Lottery Ticket Hypothesis*" by Frankle *et al.* frames the pruning process as a search for a specific, optimal subgraph within the dense model [15]. This implies that the success of a sparse model depends on finding this "winning" structure. Consequently, graph theory can be used as a tool to quantify and optimize those structures.

To this end, in this paper ANNs are treated as a graph, adopting a graph-theoretic perspective that enables the investigation of neuron behavior and their connections, with the aim of obtaining insights potentially correlated to learning mechanisms.

Specifically, the aim is to understand how the structural changes in one epoch relate to the next, or how pruning techniques can affect the model as a whole. To achieve this, the ANN model is converted into a graph representation at every training epoch and measure the similarity between them.

Then, the first set of experiments discussed in Sect. 2 builds upon our previous research [4, 16, 17] and selects Sparse Evolutionary Training (SET) [2] as the representative in-training pruning technique. The choice of SET relies in its explicit use of the *Erdős-Rényi* [18] random graph model for sparse network initialization, making it ideal for graph-based analysis. To quantify the changes between epochs, this work employs the DeltaCon [19] similarity metric, chosen for its ability to capture how the flow of data within the network structure evolves over time at a topological level.

Next, the experiments discussed in Sect. 3 extend the analysis by making use of signed network theory, treating the model's edge signs as in a signed graph. In particular, the notion of structural balance [20] finds integration into ANNs through the introduction of two novel metrics: the *local balance score* (LBS), measuring the balance of every input-output feed-forward path, and the *global balance score* (GBS), applied at the overall network level. The evolution of both the network's balance and that of individual input-output paths during training is then explored, following a similar approach to the previous set of experiments.

These preliminary analyses seem to indicate that observing the network's evolution through similarity and balance measures can give indications on its structural properties, deepening the understanding of the learning process, as detailed in Sect. 4.

In this paper, Multilayer Perceptron (MLP) architectures are used to investigate the structural evolution of the network. This choice is motivated by their natural compatibility with graph representations. In an MLP, mapping the network to a graph is a more straightforward process compared with other ANNs. Indeed, neurons act as nodes, and weighted links between layers act as edges. In contrast, other architectures, such as Convolutional Neural Networks (CNNs), add significant complexity to the experiment. Nevertheless, more complex architectures undoubtedly play a critical role in solving real-world problems, and their investigation represents a necessary and important direction for future work. For the sake of completeness, the datasets used in the experiments herein described is MNIST [21] and FashionMNIST [22].

2. Similarity Metrics Analysis

In this first set of experiments, two MLP architectures are considered, namely 9k-MLP and 96-MLP, where the number represent the total neurons. The first one, follow the one structure mentioned in [2], whereas the second one is smaller having only two hidden layers with 64 and 32 neurons, respectively.

By comparing these two sizes, where one is nearly 100 times larger than the other, the behavior of the model can be studied across very different scales.

The experimental setup (*e.g.*, learning rate, batch size, number of epochs, *etc.*) follows the same settings used in our previous works for the standard training [4, 16, 17].

To provide a clear baseline for comparison, we also tested the models under poor generalization conditions. By increasing the training rounds from 60 to 150 epochs and reducing the dataset to only 100 samples, we created a reference state of poor accuracy to compare against generalized learning.

In this study, we focus specifically on in-training pruning strategies. In contrast, pre-training and post-training methods are excluded because their network structure remains constant throughout training, meaning the architecture does not change between the first and last epochs. In particular, we make use of the sparse-to-sparse paradigm, where the network maintains a sparse structure throughout the training process and in particular, Sparse Evolutionary Training (SET) [2]. All experiments operate at a sparsity level of 80%, retaining only 20% of the possible connections as active. To enable effective learning, SET updates the network structure at the end of every training epoch by removing the weakest connections and replacing them with new random ones, also called rewiring. In the setup adopted herein, 20% of the remaining connections undergo this update at each round. To examine alternative sparsification strategies, we also consider L1-pruning and Random selection in place of the standard *Erdős-Rényi* pruning used in standard SET.

Erdős-Rényi This is the original approach used in the SET algorithm. It uses the *Erdős-Rényi* model from graph theory. In this method, the connections between neurons are created based on a specific probability. This ensures that the initial graph is randomly sampled according to an Erdős-Rényi type probability law, rather than being manually designed or completely arbitrary. Specifically, in [2] the authors define a probability $\mathrm{P}(W_{ij}) = \frac{\epsilon \times (n_{l-1}+n_l)}{(n_{l-1} \times n_l)}$ of having a connection between neuron i in layer $l-1$ and neuron j in layer l, where $\epsilon \in \mathbb{R}^+$ is a parameter that controls the sparsity level.

L1-Norm Selection The second technique used the L1-Norm criterion. This is one of the most common methods used in pre-training and post-training pruning. Instead of randomly selecting connections, this method looks at the weight values. It keeps the connections with the largest magnitude weights (strongest connections) and removes the small ones. This initialization serves as a comparison point between a "strength-based" approach and the "graph-based" ones. Mathematically, for a weight matrix W, the absolute value $|w_{ij}|$ of every connection is computed. All connections are, then, ranked based on these values, keeping only the top connections satisfying the condition $|W_{keep}| \geq |W_{removed}|$.

Random Selection The third technique uses random masking. Unlike *Erdős-Rényi* initialization, this method does not use a layer size dependent connectivity rule. Instead, connections are kept uniformly at random until the target sparsity level is reached. Thus, if S denotes sparsity, each connection has a marginal probability simply defined as $P(w_{ij}) = 1 - S$.

For example, in this study the sparsity S is 0.8 (*i.e*, 80%), so the probability will be $P(w_{ij}) = 0.2$. This means every connection has a marginal probability 20% chance of surviving in the network. This differs from the *Erdős-Rényi* used in SET, because in the layer sizes will not be equal. As a result, the *Erdős-Rényi* model assigns layer dependent connection probabilities, while random selection treats all connections uniformly. Therefore, in this setting, the same overall sparsity does not imply the same expected topology, since the two methods induce different random connectivity patterns.

2.1. DeltaCon Similarity of Artificial Neural Networks

To calculate the DeltaCon similarity metric, each epoch of the training process was first converted into a distinct graph representation. This transformation was achieved by using the adjacency matrix of the neural network. Subsequently, each individual epoch was compared to the one immediately following it to calculate the DeltaCon score.

This metric was chosen because it quantifies structural similarity by comparing node affinities, thereby capturing how the network's connectivity patterns evolve overtime. These calculations were then used to generate the similarity plots shown in Figure 1. By monitoring these epoch-to-epoch transitions, it becomes possible to track the dynamic optimization of the network and identify exactly how the internal architecture adapts or fails during different training conditions.

2.2. Results Discussion

As shown in Figure 1, under stable training conditions (*i.e.*, 60 epochs) the DeltaCon similarity metrics exhibited a smooth, increasing trend across both the 96-MLP and 9k-MLP architectures for MNIST (Fig. 1a-1d) and FashionMNIST (Fig. 1e-1h). For instance, in MNIST dataset, in the 96-MLP model, the DeltaCon metric demonstrated a steady rise, beginning at 0.1 at initialization and reaching 0.3 by the end of the 60-epoch training phase, see Figure 1. Similarly, the 9k-MLP model mirrored this smooth increasing trend, with the metric starting at 0.05 and concluding at 0.12. This continuous upward trajectory indicates a stable evolution of the network's internal structure as the model successfully learns and adapts. As the network continually updates its connections to re-establish active paths, the smooth growth in graph similarity seems to reflect the convergence of the model. During these experiments, three types of pruning initialization (*Erdős-Rényi*, L1-Norm, and Random) were compared to observe their specific effects on these trends. The starting point for *Erdő*s-Rényi initialization was found to be consistently higher than the other two methods.

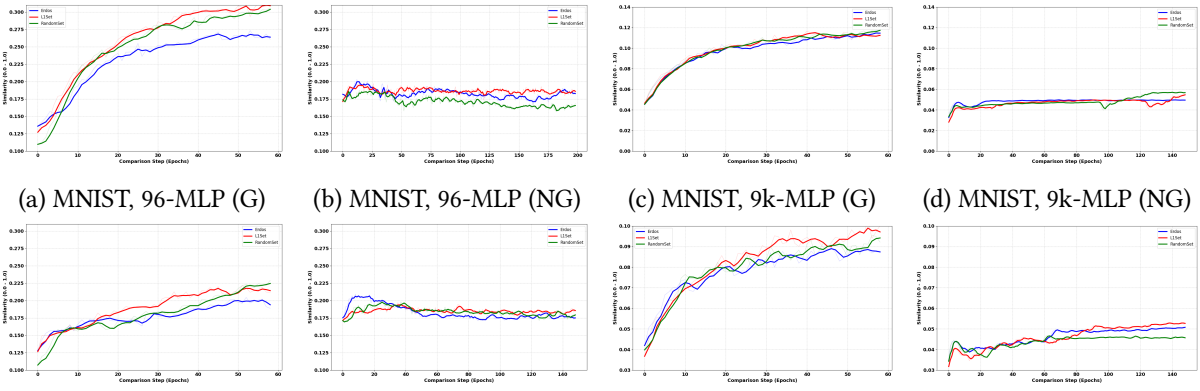
Conversely, focusing still on the MNIST dataset, when the training parameters are altered and the number of epochs is increased to 150 (to force the models into a state of poor generalization.), the DeltaCon metrics captured a disruption in structural evolution. In the 9k-MLP model, the similarity metric exhibited highly stunted growth, only increasing from 0.03 to 0.05 over the entire 150 epochs. The 96-MLP model provided an even clearer indication of the structural failed state: the DeltaCon metric initially increased from 0.17 to 0.20 by epoch 15, but then regressed, decreasing to 0.18 where it stagnated for the remainder of the training.

These preliminary results seems to suggest that graph similarity metrics, particularly DeltaCon, can indicate training stability. While a smooth, increasing trend is associated with stable training and good generalization, stagnant, or regressing scores may suggest poor generalization. These patterns are not unique to MNIST; as shown in the bottom row of Fig. 1, we observe identical behavior in the FashionMNIST experiments.

3. Structural Balance Analysis

A traditional way to better understand ANN models is by inspecting the role of weighted edges, and this can be achieved by considering pruning edge connections of the ANNs as early mentioned. However, the role of edge signs, although being important in various contexts, has not been examined systematically yet. Given that ANNs are composed of signed weighted edges, it seems natural to continue our analysis by considering the signs thus considering the model's signs as a signed graph.

Signed networks are essentially modeling the coexistence of two competing types of relationships which is essential to shape the system performance, *e.g.*, activatory or inhibitory functions in biological systems, and cooperative or antagonistic relationships in social networks [23, 24]. Therefore, *signed networks*, where connections can be positive or negative, have become important ingredients of models in many research fields over recent years. A central notion in the study of signed networks is *structural balance* [20]. In particular, a signed network is structurally balanced if and only if all its cycles are



(a) MNIST, 96-MLP (G) (b) MNIST, 96-MLP (NG) (c) MNIST, 9k-MLP (G) (d) MNIST, 9k-MLP (NG)

(e) FMNIST, 96-MLP (G) (f) FMNIST, 96-MLP (NG) (g) FMNIST, 9k-MLP (G) (h) FMNIST, 9k-MLP (NG)

Figure 1: DeltaCon similarity for pruned MLPs on MNIST (a-d) and FMNIST (*i.e.* FashionMNIST) (e-h). Models are categorized by being able (G) or not (NG) to generalize during the training. Growth indicates stability, while stagnation denotes poor performance.

so-called positive, *i.e.*, have an even number of negative edges, see Figure 2. Balanced signed networks exhibit a range of desirable properties that are supported both theoretically and empirically, *e.g.*, [25]. However, the application of signed network theory, specifically the notion of balance, in ANNs is relatively unknown.

This paper investigates the role of signs by considering ANNs as signed graphs. The aim is to explore the role of edge signs over the training process, and whether it is possible to interpret and utilize the insights to lead the training or further applications. To address those questions, it is worth to examine the role of signs by using the fundamental concept of structural balance from signed network theory. Thus ANNs can be represented as signed graphs, enabling the examination of feed-forward paths from inputs to outputs. In order to achieve this, a translation layer is needed so that the structural balance can be used in the ANN setting, where the graphs consist of multipartite ones.

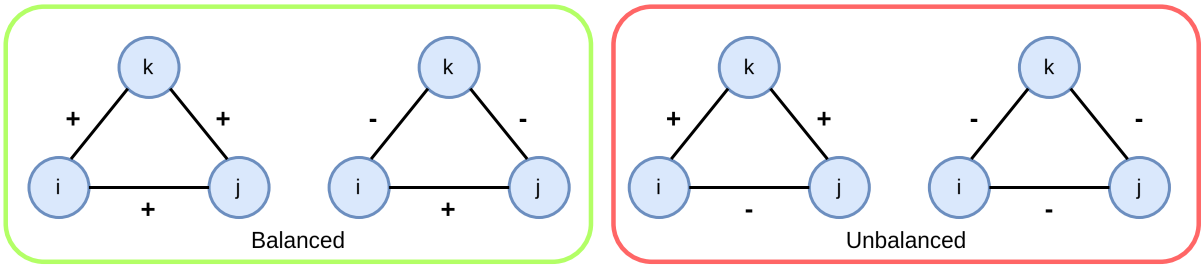


Figure 2: Example of balance *vs.* unbalance triangle (or circle). It is unbalanced if it contains an odd number of negative signs.

3.1. Structural Balance of Artificial Neural Networks

Before describing the structural balance on ANNs, the following concepts of *local* and *global* balance must be introduced. In particular, the sign of a path P is defined as $\operatorname{sign}(P) = \prod_{e \in P} \operatorname{sign}(w_e) \in \{-1, 1\}$,

where $\operatorname{sign}(P) = 1$ if the parity of negative edges on the paths is even, and -1 if odd.

Based on the signs of the paths the notion of *Local Balance Score* (LBS) is proposed as follows. For a pair of nodes (i, j), we have that $\operatorname{LBS}(i, j) = \frac{|N_{ij}^{+} - N_{ij}^{-}|}{N_{ij}^{+} + N_{ij}^{-}}$, where N_{ij}^{+} and N_{ij}^{-} represent, respectively, the number of positive and negative paths from nodes i to j. Thus, $\operatorname{LBS}(i, j) = 1$ if and only if $N_{ij}^{+} = 0$ or $N_{ij}^{-} = 0$ (*i.e.*, all cycles involving i, j are positive) or $\operatorname{LBS}(i, j) = 0$ if and only if $N_{ij}^{+} = N_{ij}^{-}$, which maximizes the number of negative cycles with i, j.

Having defined LBS score, it is now feasible to describe the score of a network as a whole. Therefore for all forward paths between all input nodes, $\mathcal{I}$, and all output nodes, $\mathcal{O}$, the *Global Balance Score* (GBS) is defined as $\text{GBS} = \frac{\sum_{i\in\mathcal{I},j\in\mathcal{O}} |N_{ij}^{+} - N_{ij}^{-}|}{\sum_{i\in\mathcal{I},j\in\mathcal{O}} N_{ij}^{+} + N_{ij}^{-}}$. Thus, $\text{GBS} = 1$ if and only if all cycles are positive, (*i.e.*, the signed network is balanced) and 0 otherwise. Alternatively, the GBS score can be considered as a natural extension of LBS but this time on the network level.

It is also worth noting that the computation of LBS and GBS incurs additional computational costs however this aspect is not addressed in this work for two reasons. First, their computation is polynomial and second, it is performed post training without interfering with the learning process. Therefore, a detailed analysis of these costs is beyond the scope of this study.

3.2. Results Discussion

The aforementioned scores can be used to obtain insights about the model training of and possibly about the learning as well. In more detail, the LBS score can be seen as a quantification of the meaningful information flow between every input-output pair. Extending this analogy to a global scale, the GBS score may be used to characterize the balance of the network as a whole. In [26], we observed that the local and, subsequently, the global balance scores increase during training. That findings were more apparent after the introduction of unstructured sparsity (10% $\sim$ 70%) thus suggesting the emergence of strongly negative or positive paths, as shown in Figure 3. These results were obtained under the SET paradigm where during the model initialization *Erdős-Rényi* sparsification was employed along with rewiring during training. These results can be seen as an indication of meaningful structural patterns encoded in edge signs as it was also noted by Gadhikar et. al. [27], that are captured by our notion of positive (N^{+}) or negative (N^{-}) paths.

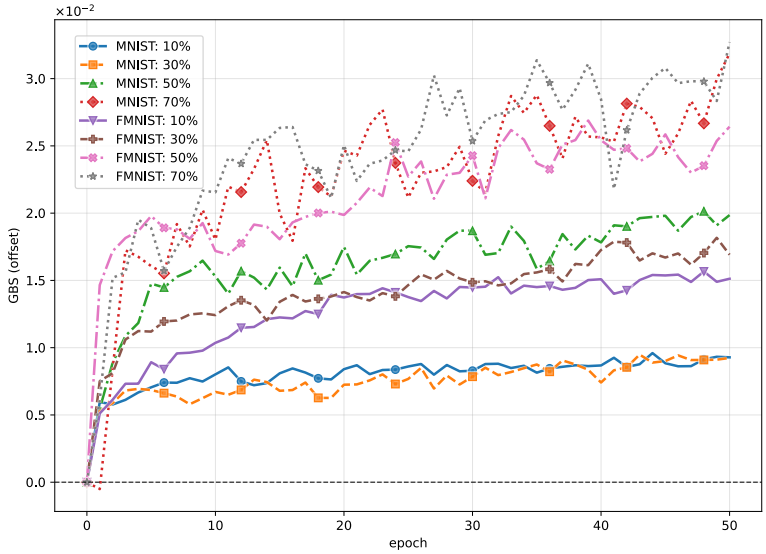


Figure 3: Evolution of GBS over the epochs over different levels of model sparsity (10% up to 70%) using two distinct benchmarking datasets, namely MNIST and FashionMNIST. For those experiments an MLP has been used with 2 hidden layers and 64 neurons each.

4. Conclusion and Future Work

This work presents a preliminary study of the structural evolution of neuron connections during training, monitored through the DeltaCon similarity metric and further investigated by treating the model's edge signs as in a signed graph. The findings suggest that similarity metrics capture important stages of the learning process, such as when the model learns meaningful representations (generalized)

or fails to do so (non-generalized). Furthermore, sign flipping occurring during training appears to drive an increase in both LBS and GBS, suggesting the creation of more positive (N^+) or negative (N^-) paths, leading the network toward a more balanced state.

Several promising directions emerge for future work. From the perspective of structural balance, the notion of balance on ANNs opens the door to feature pruning based on balance scores, where the least impactful features could be identified and removed in a more informed manner. Desired sparsity levels could similarly be determined by observing how network balance evolves during training, with balance serving as a tool to identify redundant paths and guide pruning decisions. An important open question concerns the existence of a clear correlation between balance and accuracy. Beyond this, investigating how different sparsification paradigms (*e.g.*, dense-to-sparse *vs.* sparse-to-sparse, or random *vs.* ℓ_1-pruning) affect network balance could reveal further insights into the learning dynamics of sparse models.

From the perspective of similarity metrics, future investigation could explore whether measures such as DeltaCon serve as predictors of key transition phases during training. Of particular interest is whether changes in structural similarity correlate with training stability and convergence speed, and whether these metrics can identify critical epochs, non-convergent runs, or the impact of different pruning and sparsification strategies.

Finally, examining the relationship between similarity metrics and structural balance could determine whether large topological changes coincide with shifts in sign organisation and network balance, potentially bridging the two analytical perspectives introduced in this work.

Declaration on Generative AI

The authors declare that GenAI tools (i.e., ChatGPT and Copilot) were minimally and limitedly used only for language editing and refinement. It had no role in shaping the research content, results, nor their interpretations. All AI-generated text polishing suggestions were carefully reviewed and edited by the authors to ensure scientific accuracy and compliance with academic standards.

References

[1] S. Han, J. Pool, J. Tran, W. Dally, Learning both weights and connections for efficient neural network, in: C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, R. Garnett (Eds.), Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1, volume 28 of *NIPS'15*, MIT Press, Cambridge, MA, USA, 2015, p. 1135–1143. doi:10.48550/arXiv.1506.02626.

[2] D. C. Mocanu, E. Mocanu, P. Stone, P. H. Nguyen, M. Gibescu, A. Liotta, Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science, Nature Communications 9 (2018) 2383. doi:10.1038/s41467-018-04316-3.

[3] O. Galchonkov, A. Nevrev, M. Glava, M. Babych, Exploring the efficiency of the combined application of connection pruning and source data preprocessing when training a multilayer perceptron, Eastern-European Journal of Enterprise Technologies 2 (2020) 104. doi:10.15587/1729-4061.2020.200819.

[4] A. Douzandeh Zenoozi, L. Erhan, A. Liotta, L. Cavallaro, A comparative study of neural network pruning strategies for industrial applications, Frontiers in Computer Science 7 (2025) 1563942. doi:10.3389/fcomp.2025.1563942.

[5] T. Hoefler, D. Alistarh, T. Ben-Nun, N. Dryden, A. Peste, Sparsity in deep learning: pruning and growth for efficient inference and training in neural networks, J. Mach. Learn. Res. 22 (2021). URL: http://jmlr.org/papers/v22/21-0366.html. doi:10.48550/arXiv.2102.00554.

[6] B. Chan, Black-box assisted medical decisions: AI power vs. ethical physician care, Med. Health Care Philos. 26 (2023) 285–292. doi:10.1007/s11019-023-10153-z.

[7] C. Rudin, Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead, Nat. Mach. Intell. 1 (2019) 206–215. doi:10.1038/s42256-019-0048-x.
[8] J. Liu, Y. Lin, L. Jiang, J. Liu, Z. Wen, X. Peng, Improve interpretability of neural networks via sparse contrastive coding, in: Findings of the Association for Computational Linguistics: EMNLP 2022, Association for Computational Linguistics, Stroudsburg, PA, USA, 2022, pp. 460–470. doi:10.18653/v1/2022.findings-emnlp.32.
[9] D. Klindt, C. O'Neill, P. Reizinger, H. Maurer, N. Miolane, From superposition to sparse codes: interpretable representations in neural networks, 2025. doi:10.48550/arXiv.2002.07376.
[10] C. Wang, G. Zhang, R. Grosse, Picking winning tickets before training by preserving gradient flow, 2020. doi:10.48550/arXiv.2002.07376.
[11] U. Evci, T. Gale, J. Menick, P. S. Castro, E. Elsen, Rigging the lottery: making all tickets winners, in: Proceedings of the 37th International Conference on Machine Learning, ICML'20, JMLR.org, 2020. doi:10.48550/arXiv.1911.11134.
[12] R. Fruengel, M. Oberlaender, Sparse connectivity enables efficient information processing in cortex-like artificial neural networks, Frontiers in Neural Circuits Volume 19 - 2025 (2025). doi:10.3389/fncir.2025.1528309.
[13] J. You, J. Leskovec, K. He, S. Xie, Graph structure of neural networks, in: Proceedings of the 37th International Conference on Machine Learning, ICML'20, JMLR.org, 2020. doi:10.48550/arXiv.2007.06559.
[14] S. Xie, A. Kirillov, R. B. Girshick, K. He, Exploring randomly wired neural networks for image recognition, 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (2019) 1284–1293. doi:10.1109/ICCV.2019.00137.
[15] J. Frankle, M. Carbin, The lottery ticket hypothesis: Finding sparse, trainable neural networks, arXiv: Learning (2018). doi:10.48550/arXiv.1803.03635.
[16] A. Douzandeh Zenoozi, L. Erhan, A. Liotta, L. Cavallaro, A comparative study of neural network pruning strategies for AI deployment on edge devices, in: Studies in Computational Intelligence, Studies in Computational Intelligence, Springer Nature Switzerland, Cham, 2025, pp. 3–14. doi:10.3389/fcomp.2025.1563942.
[17] A. Douzandeh Zenoozi, L. Erhan, A. Liotta, L. Cavallaro, Inference and training efficiency in pruned multilayer perceptron networks, PLOS Complex Systems 3 (2026) 1–28. doi:10.1371/journal.pcsy.0000095.
[18] P. Erdős, A. Rényi, On random graphs. I, Publ. Math. Debrecen 6 (2022) 290–297.
[19] D. Koutra, J. T. Vogelstein, C. Faloutsos, Deltacon: A principled massive-graph similarity function, in: Proceedings of the 2013 SIAM international conference on data mining, SIAM, 2013, pp. 162–170. doi:10.1137/1.9781611972832.18.
[20] F. Harary, On the notion of balance of a signed graph, Michigan Math. J. 2 (1953) 143–146. doi:10.1307/mmj/1028989917.
[21] L. Deng, The MNIST database of handwritten digit images for machine learning research, IEEE Signal Processing Magazine 29 (2012) 141–142. doi:10.1109/MSP.2012.2211477.
[22] H. Xiao, K. Rasul, R. Vollgraf, Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, arXiv preprint arXiv:1708.07747 (2017).
[23] G. Facchetti, G. Iacono, C. Altafini, Computing global structural balance in large-scale signed social networks, Proc. Natl. Acad. Sci. 108 (2011) 20953–20958. doi:10.1073/pnas.1109521108.
[24] Y. Tian, S. Lautz, A. Wallis, R. Lambiotte, Extracting complements and substitutes from sales data: a network perspective, EPJ Data Sci. 10 (2021) 45. doi:10.1140/epjds/s13688-021-00297-4.
[25] C. Altafini, Dynamics of opinion forming in structurally balanced social networks, PLoS ONE 7 (2012) 1–9. doi:10.1371/journal.pone.0038135.
[26] C. Aronis, Y. Tian, L. Cavallaro, Bridging signed and artificial neural networks, (in press). Accepted at the 17th International Conference on Complex Networks (CompleNet2026), Zaragoza, Spain.
[27] A. Gadhikar, T. Jacobs, C. Zhou, R. Burkholz, Sign-in to the lottery: Reparameterizing sparse training from scratch (2025). doi:10.48550/arXiv.2504.12801.