Thesis Design

Motif-based Structural Optimization of Sparse MLPs

Tonny Le
tonny.le@student.uva.nl
University of Amsterdam
Amsterdam, The Netherlands

Hongyun Liu
h.liu@uva.nl
University of Amsterdam
Amsterdam, The Netherlands

1 INTRODUCTION

In an era with a growing demand for solutions based on artificial intelligence (AI), Deep Neural Networks (DNNs) have seen great success. However, this success is not without its challenges, as it entails a ravenous appetite for greater computational, energy, and memory resources, especially as the complexity and dimensionality of models continue to grow [9]. If not for its requirement to be trained on data with high dimensonality and the high computational, energy, and memory requirements, DNN's would be a widespread option to various machine learning (ML) tasks.

In [5], the author introduces a method called Sparse Evolutionary Training, applied to Multi-Layer Perceptrons (SET-MLP) for supervised feature selection, aiming to address the increasing computational demands of deep neural networks. It demonstrates robustness to changes in network topology, producing consistent feature selections across different topologies and recovering from poor initialization on selected datasets. The topology was also explored in [2], where a simple neural network was trained. The study revealed that as the neural network learned, the weights evolved in a way that generated recurrently branching trajectories when initialized to zero, forming trees that described the growth of effective capacity in each layer. When initialized to tiny random values, the weights exhibited smooth evolution along two-dimensional surfaces, and the natural coordinates on these surfaces corresponded to important factors of variation in the learning process.

In recent work [7], introduced the Sparse Multilayer Perceptrons (MLPs), which refer to neural networks where most of the connections between neurons are zero or close to zero. The authors replaced the fully-connected layers with sparse ones before training, reducing quadratically the number of trainable parameters, with no decrease in accuracy.

In work close theirs [1], RigL was introduced. This method introduces a dynamic sparsity, a novel training method for sparse neural networks with a fixed parameter count and computational cost throughout the training process. This approach updates the topology of the sparse network during training using parameter magnitudes and infrequent gradient calculations. This leads to more efficient and effective neural network training, outperforming traditional dense-to-sparse training methods.

These methods are all applied during the feature selection of training, optimizing the structure of each model. Since optimizing the structure based on motifs remains an unexplored territory, this thesis seeks to address this, assuming that certain motifs or patterns within a network's topology can significantly influence its performance. In [10], universal patterns in sparse Recurrent Neural Networks (RNNs) were found, highlighting the potential of motif-based optimizations of MLPs.

1.1 Research question

How can the integration of motif-based structural optimizations and the exploration of convergent structures in sparse multilayer perceptrons (MLPs) contribute to enhancing the efficiency, computational performance of deep neural networks?

1.2 Sub questions

(1) Motif-based Structural Optimizations:
   (a) What are motif-based structural optimizations in the context of deep neural networks?
   (b) How do motif-based structural optimizations enhance the efficiency of sparse multilayer perceptrons (MLPs)
(2) Improvement in Deep Neural Networks
   (a) Are there trade-offs between efficiency and other performance metrics?

1.3 Background/Theoretical Framework

1.3.1 Neural Networks. A neural network (NN) is an ML framework consisting of nodes, resembling the human brain [3]. This is an old concepts which was explored back in 1958 [8]. A NN is build out of multiple layers. The input layer serves as the initial layer for the network, receiving the input data. Between the input and output layers, are hidden layers that perform transformations on the input. The output layer is the last layer. The first NN consisted an input layer, a hidden layer with randomized weights that did not learn, and an output layer with learning connections. In modern NNs the node values and bias are updated when the input reaches the output layer, using backpropagation.

1.3.2 Multi Layer Perceptrons. MLPs are a specific type of neural network architecture characterized by having at least three layers of nodes: an input layer, one or more hidden layers, and an output layer. Each node in a layer is connected to every node in the subsequent layer, forming a fully connected network. The term MLPs are often used interchangeably as MLPs are some of the oldest forms of neural networks. But a key difference is the fact that MLPs are feedforward: all connections between the nodes go towards the output, and there are variants of NNs where connections can be cyclic, like in Recurent Neural networks (RNNs).

1.3.3 Topology. Topology is a branch of mathematics that studies the properties of space preserved under continuous deformations, such as stretching and bending. It focuses on the intrinsic properties of geometric shapes and their relationships. Topology is relevant in computer science too, particularly in the areas of network theory, distributed systems, and algorithms. In computer networks, topology refers to the arrangement of nodes and the links between them.

Graphs, a branch of discrete mathematics, are closely related to topology. Graph theory is used in computer science to model and analyze relationships between entities, for example Dijkstra shortest path algorithm.

2 RELATED WORK

There has a lot of work in the field of improving the computational efficiency of NNs. [4] proposed a Dense-Sparse-Dense training method, which addresses the challenge of training large neural networks by introducing a three-step process. In the Dense step, a dense network is trained to learn connection weights and importance. The Sparse step involves trimming unimportant connections with small weights, imposing sparsity constraints, and retraining the network. In the re-Dense step, the sparsity constraint is removed, trimmed parameters are re-initialized, and the entire dense network is retrained. [7] replaced fully connected layers to sparse layers, reducing the computational requirement, without loss in accuracy. [5] explored the potential of sparse training from high dimensional data. Both of these works form a core foundation to training NNs with sparse layers.

Furthermore, in [10], the evolution of RNNs subjected to different sparsification strategies was analyzed. The findings reveal a universal pattern of signed motifs in optimized sparse topologies. RNNs evolve towards structurally balanced configurations during sparsification, and structural balance can improve the performance of sparse RNNs in a variety of tasks. Such structural balance patterns also emerge in other state-of-the-art models, including neural ordinary differential equation networks and continuous-time RNNs. Overall, the study provides insights into universal structural features in optimized sparse neural networks.

Unlike existing works that either reduce computational requirements by introducing sparse layers [1, 7], introduce novel training architectures meeting lower computational power [4, 5], or identify motifs in current neural networks [2, 6, 10], this thesis aims to explore the gap where motifs and patterns are utilized to optimize MLPs. This will be done by introducing new methods to reduce the computational requirements, leveraging the current motif analyses and topologies done by existing work. This will not only further enhance our understanding of NNs, but also will lower the threshold of DNNs, making it more practical in the real world.

3 METHODOLOGY

The main focus of this thesis will be on optimizing structures of NNs/MLPs. These can either be assessed in accuracy and computational time required. Below are some options to assess the models trained:

3.1 Data

Free public datasets can be used like CIFAR and MNIST, which are all available through tensorflow. Models can be written using python's module torch.

3.2 Design

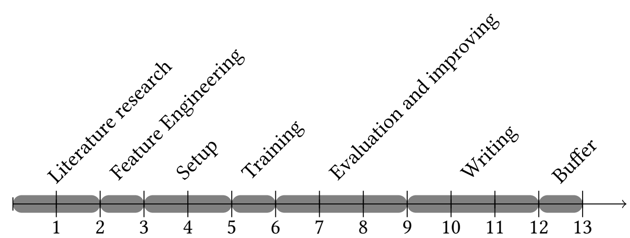
(1) use network science techniques to test around different motifs and recognise potential motifs
(2) design a new architecture with the motifs identified, and train the model with different types of sparse layers.
(3) in order to evaluate the architectures, for example the computational power required and accuracy can be measured. Recent state of the art work can be used as baseline. (potential trade-offs can be explored if any)
(4) assess the different motifs and check/retrain the models

In order to check our research question, statistical analysis will be done, by evaluating the results of the model variants. The significance of the improvements can be checked by training it on different datasets.

4 RISK ASSESSMENT

As Neural networks can become computationally very demanding, this could become an issue in work efficiency if one has to train multiple at once, while testing them. Secondly, time management could become a problem short-term, due to personal reasons.

5 PROJECT PLAN



REFERENCES

[1] Utku Evci, Trevor Gale, Jacob Menick, Pablo Samuel Castro, and Erich Elsen. 2020. Rigging the Lottery: Making All Tickets Winners. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 119)*, Hal Daumé III and Aarti Singh (Eds.). PMLR, 2943–2952. https://proceedings.mlr.press/v119/evci20a.html

[2] Maxime Gabella. 2019. Topology of learning in artificial neural networks. *arXiv preprint arXiv:1902.08160* (2019).

[3] Kevin Gurney. 1997. *An introduction to neural networks.* CRC press. https://doi.org/10.1201/9781315273570

[4] Song Han, Jeff Pool, Sharan Narang, Huizi Mao, Enhao Gong, Shijian Tang, Erich Elsen, Peter Vajda, Manohar Paluri, John Tran, et al. 2016. Dsd: Dense-sparse-dense training for deep neural networks. *arXiv preprint arXiv:1607.04381* (2016). https://doi.org/10.48550/arXiv.1607.04381

[5] Neil Kichler. 2021. Robustness of sparse MLPs for supervised feature selection. http://essay.utwente.nl/86886/

[6] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. 2002. Network motifs: simple building blocks of complex networks. *Science* 298, 5594 (2002), 824–827.

[7] Decebal Constantin Mocanu, Elena Mocanu, Peter Stone, Phuong H Nguyen, Madeleine Gibescu, and Antonio Liotta. 2018. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science. *Nature communications* 9, 1 (2018), 2383. https://doi.org/10.1038/s41467-018-04316-3

[8] Frank Rosenblatt. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review* 65, 6 (1958), 386.

[9] Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel S. Emer. 2017. Efficient Processing of Deep Neural Networks: A Tutorial and Survey. *Proc. IEEE* 105, 12 (2017), 2295–2329. https://doi.org/10.1109/JPROC.2017.2761740

[10] Xin-Jie Zhang, Jack Murdoch Moore, Gang Yan, and Xiang Li. 2023. Universal structural patterns in sparse recurrent neural networks. *Communications Physics* 6, 1 (2023), 243. https://doi.org/10.1038/s42005-023-01364-0