Dynamic Deep Learning*

Claes Strannegård[1,2], Herman Carlström[1], Niklas Engsner[2], Fredrik Mäkeläinen[2], Filip Slottner Seholm[1], and Morteza Haghir Chehreghani[1]

[1] Department of Computer Science and Engineering
Chalmers University of Technology, Gothenburg, Sweden
[2] Dynamic Topologies Sweden AB, Gothenburg, Sweden

Introduction

The standard approach to deep learning has been first to design the network architecture manually, then train the network with data. Architecture design remains an art that requires manual input by machine-learning experts.

A number of methods have been suggested for automating architecture design. Among the earliest dynamic models, one finds the *cascade-correlation architecture* [3], which adds one hidden neuron at a time while freezing the network to avoid catastrophic forgetting. *Progressive neural networks* [6] add new layers of neurons successively while blocking changes to those parts of the network that were trained on earlier data, so that previously acquired knowledge is retained. Other incremental methods exist: e.g., based on incremental training of an autoencoder whereby neurons are added either in response to high failure rate with new data [7] or based on reconstruction error [2]. AdaNet [1] gradually extends its network based on evaluation and selection among candidate sub-networks. A *dynamically expandable network* [4] expands its capacity via network split and duplication operations, retraining the old network only when necessary.

Despite the advent of such methods, architecture design remains on the whole a manual exercise based on some predefined heuristics. Moreover, these methods tend to be computationally inefficient, involving extensive search among alternative architectures or repeated training of dense architectures with randomly initialized parameters. We propose a principled dynamic approach for deep learning that starts as a blank slate and develops a deep neural network gradually, in a fully automatic and energy-efficient manner.

Dynamic model LL0

Our dynamic model LL0 for supervised learning uses a neural network that starts as a blank slate and develops continuously over time. LL0 adds and removes nodes and connections dynamically through four network-modification mechanisms, each inspired by neuroplasticity [5]: (i) back-propagation that adjusts parameters, inspired by synaptic plasticity; (ii) extension that adds new

* Supported by the Torsten Söderberg Foundation Ö110/17. Corresponding author: Claes Strannegård. Email: claes.strannegard@chalmers.se

nodes, inspired by neurogenesis; (iii) forgetting that removes nodes, inspired by programmed cell death; and (iv) generalization that abstracts from existing nodes, inspired by synaptic pruning. In this way, LL0 models four forms of neuroplasticity, rather than one (i) as in standard deep learning, or two (i+ii) as in the dynamic approaches mentioned above.

LL0 receives a continuous stream of data points of the form (x, y), where x and y are real-number vectors of fixed dimensions and y is the target vector associated with x. Algorithm 1 shows the main loop.

Algorithm 1: Main loop of LL0.

```
receive the first data point $(x, y)$
form $|x|$ input nodes and $|y|$ output nodes
while true do
    compute network output $\hat{y}$ produced by input $x$
    if $prediction(\hat{y}) \neq y$ then
        generalization
        extension
    else
        backpropagation
    end
    forgetting
    receive a new data point $(x, y)$
end
```

Results

LL0 has been analyzed with respect to four data sets from different domains, adapted from `playground.tensorflow.org` and `scikit-learn.org`:

spirals : 2,000 two-dimensional data points in the form of two intertwined spirals as shown in Figure 1 (right).
digits : 1,797 labeled 8x8 pixel grayscale images of hand-written digits.
radiology : 569 data points, each a 30-dimensional vector describing features of a radiology image labeled benign or malignant.
wine : 178 data points, each a 13-dimensional vector describing taste features of a wine identified by one of three regions of origin.

LL0 was compared to four fully connected, layered networks:

FC0 : No hidden layer.
FC10 : One hidden layer with 10 nodes.
FC2*10 : Two hidden layers with 10+10 nodes.
FC3*10 : Three hidden layers with 10+10+10 nodes.

All four baseline models were trained using stochastic gradient descent with individually optimized learning rates for each data set. Despite their simplicity, these baseline models are highly useful. Dynamic models that construct dense architectures and randomize initial weights generally learn slower and consume more energy than hand-crafted networks, because they are required to search for architectures in addition to undergoing the standard training procedure.

Figure 1 shows the network generated (left) when LL0 ran on the spirals data set, along with the network's decision boundaries (right).

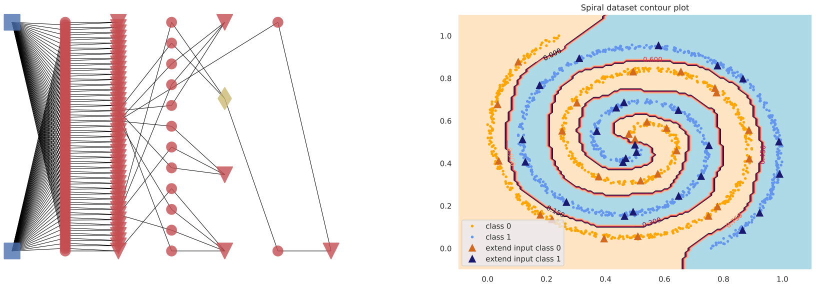

Fig. 1. Left: The network produced by LL0 on the spirals data set, with the two output nodes and their connections omitted for sake of readability. The architecture converged after less than one epoch with about 160 nodes, depth six, and max fan-in five. **Right**: The spirals data set with the generated decision boundary. Nodes created through extension are marked by triangles.

Figure 2 shows the result of the five models in terms of accuracy (left) and energy consumption (right) with respect to the spirals data set.

The outcome on the other three data sets can be summarized as follows, all because LL0 uses a form of one-shot learning that speeds up that learning while reducing energy consumption:

- LL0 reached at least the same accuracy as the baselines on all datasets;
- LL0 learned several times faster than all baselines on all datasets;
- LL0 used much less energy than the baselines.

It is important to remember that the four data sets stem from very different sources: i.e., mathematical functions, handwriting, clinical judgment, and chemical measurements. LL0 used the same hyper-parameter settings on all data sets and still, for each data set, it performed at the level of the best baseline model or better.

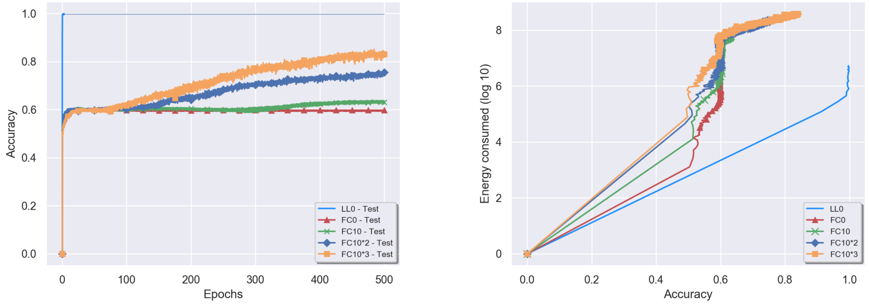

Fig. 2. Performance on the spirals data set. **Left**: LL0 reaches 1.0 accuracy on the test set after six epochs. The best baseline model FC3*10 has not reached 0.9 accuracy after 500 epochs. **Right**: Estimated energy consumption plotted on a logarithmic scale. Energy consumption was estimated in terms of the number of floating-point operations. FC3*10 consumes over 1,000 times more energy than LL0 to reach 0.8 accuracy on the test set.

Conclusion

The LLO model can be used for constructing networks automatically instead of manually. It starts from a blank slate and develops its deep neural network continuously. It uses no randomization, builds no fully connected layers, and engages in no search among candidate architectures: properties that set it apart from other dynamic models. LL0 compares favorably to all static baseline models in terms of learning speed, energy efficiency, and versatility.

References

1. Cortes, C., et al.: Adanet: Adaptive structural learning of artificial neural networks. In: Proceedings of the 34th International Conference on Machine Learning. pp. 874–883. JMLR.org (2017)
2. Draelos, T.J., et al.: Neurogenesis deep learning: Extending deep networks to accommodate new classes. 2017 International Joint Conference on Neural Networks (IJCNN) pp. 526–533 (2017)
3. Fahlman, S.E., Lebiere, C.: The cascade-correlation learning architecture. In: Advances in neural information processing systems. pp. 524–532 (1990)
4. Lee, J., Yoon, J., Yang, E., Hwang, S.J.: Lifelong learning with dynamically expandable networks. CoRR **abs/1708.01547** (2018)
5. Power, J.D., Schlaggar, B.L.: Neural plasticity across the lifespan. Wiley Interdisciplinary Reviews: Developmental Biology **6**(1), e216 (2017)
6. Rusu, A.A., Rabinowitz, N.C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., Hadsell, R.: Progressive neural networks. arXiv preprint arXiv:1606.04671 (2016)
7. Zhou, G., Sohn, K., Lee, H.: Online incremental feature learning with denoising autoencoders. In: Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics. pp. 1453–1461 (2012)