Deep Learning through Generative and Developmental Systems

Phillip Verbancsics
SPAWAR Systems Center Pacific
53560 Hull Street
San Diego, CA, USA
phillip.verbancsics@navy.mil

Joshua Harguess
SPAWAR Systems Center Pacific
53560 Hull Street
San Diego, CA, USA
joshua.harguess@navy.mil

ABSTRACT

Deep learning through supervised and unsupervised learning has demonstrated human competitive performance on some visual tasks; however, evolution played an important role in the development of biological visual systems. Thus evolutionary approaches, specifically the Hypercube-based NeuroEvolution of Augmenting Topologies, are applied to deep learning tasks in this paper. Results indicate HyperNEAT alone struggles in image classification, but trains effective feature extractors for other machine learning approaches.

Categories and Subject Descriptors

I.5.4 [**Pattern Recognition**]: Computer Vision

General Terms

Algorithms; Experimentation

Keywords

Generative and Developmental Systems; Deep Learning

1. INTRODUCTION

Evolution is a significant contributor in creating biological visual systems [3], thus a path for creating computational approaches as effective as their biological counterparts is neuro-evolution [7]. Research in deep learning has demonstrated that mimicking the biological systems can provide pathways to human-level capability on visual tasks [4]. This paper investigates evolution for deep learning through the Hypercube-based NeuroEvolution of Augmenting Topologies (HyperNEAT). The capability of learning representations through HyperNEAT is examined by combining HyperNEAT with other learning approaches by transforming inputs thro–ugh the HyperNEAT substrate, thereby extracting features from the raw images. These features provide the training data for the other machine learning (M)L approaches and the performance of the solution trained by the ML approach informs the quality of the representation learned. Results show that HyperNEAT can effectively extract features for machine learning approaches.

GECCO'14, July 12–16, 2014, Vancouver, BC, Canada.
ACM 978-1-4503-2881-4/14/07.
http://dx.doi.org/10.1145/2598394.2598401.

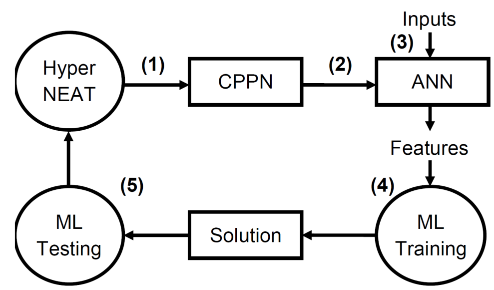

Figure 1: HyperNEAT Feature Learning. HyperNEAT trains CPPNs (1) that generate ANN connectivity (2). The ANN produces features from images (3). These features are given to another machine learning algorithm (4) that learns classifications of the features. The trained solution is evaluated on testing data (5). Solution performance on unseen data provides the fitness for HyperNEAT.

2. FEATURE LEARNING HYPERNEAT

Hypercube-based NEAT (HyperNEAT; [5]) is a generative and developmental system (GDS) extension of NEAT that enables effective evolution of high-dimensional ANNs. The effectiveness of the geometry-based learning in HyperNEAT has been demonstrated in multiple domains, such as multi-agent predator prey [1] and RoboCup Keepaway [6]. A full description of HyperNEAT is in Stanley et al. [5]. Although the HyperNEAT succeeds in a number of challenging tasks [5, 2], it has not yet been applied to tasks where deep learning is showing promise. To this end, HyperNEAT is modified in two ways. First, the alternative convolutional neural network architecture is substituted for the ANN substrate. Second, HyperNEAT is applied as a feature learner, rather than directly performing the task. In this way, HyperNEAT trains an ANN that extracts features that are given to another machine learning approach to solve the problem. Thus, the performance of this learned solution then defines the fitness score of the compositional pattern producing network (CPPN) for HyperNEAT (figure 1). In this way, HyperNEAT acts as a trainer for extracting the best features for another machine learning approach.

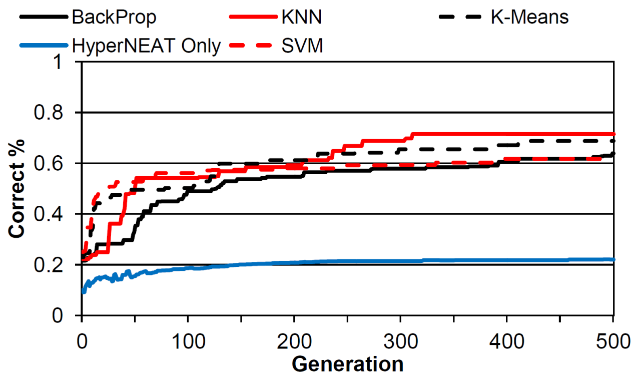

Figure 2: HyperNEAT-FL Classification Performance. HyperNEAT alone performs worst with a correct classification rate of 22.1%. BackProp, KNN, K-Means, and SVMs have classification rates of 63.9%, 71.4%, 68.8%, and 61.8% respectively.

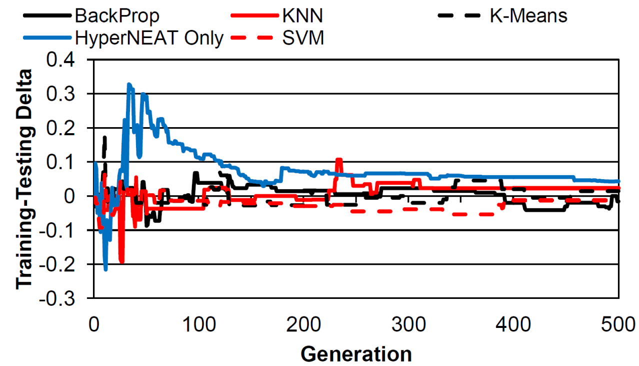

Figure 3: Generalization Performance. Each generation champion's generalization performance is measured by the percent difference between training performance during evolution and testing performance, where the ML approach is both trained and tested on images unseen during evolution.

3. EXPERIMENTS: MNIST

These investigations are conducted on the MNIST dataset. Two substrate architectures are investigated; a traditional ANN architecture (feed-forward, fully-connected, sigmoid activation functions) and the CNN architecture (constrained connectivity, alternating sigmoid and pooling layers). To operate as a feature extractor, each image is passed through the substrate to produce an associated feature vector and this set of features is evaluated by learning through another approach; either Backward Propagation, K-Nearest Neighbors, K-Means clustering, or Support Vector Machines.

For each of these experiments, results are averaged over 30 independent runs wherein each run randomly selects 300 images for the training set and 300 images for the evaluation set, evenly spread across the classes. For regular HyperNEAT (i.e. not feature learning), fitness is determined by applying the substrate to the training images. For the ML approaches, the fitness is determined by applying the training solution to the evaluation set.

In the first experiment, different substrate architectures are examined. A significant ($p < 0.01$, Student's t-test) difference is observed in the classification correctness between the ANN and CNN substrates as feature learners for BackProp. The CNN substrate reaches a 61.8% correct classification rate versus the ANN substrate with a 41.1% classification rate. The second experiment examines HyperNEAT's performance as a feature learner with the CNN substrate architecture (figure 2) contrasted with the performance of HyperNEAT directly classifying images. HyperNEAT as a feature learner significantly ($p < 0.01$) exceed the performance of HyperNEAT alone on the task. HyperNEAT with BackProp achieves a classification success rate of 63.9%. Learning for SVMs has a similar performance level at 61.8% correct classifications. HyperNEAT with KNN has the best classification rate of 71.4% over K-Means's 68.8%. Finally, generalization is examined (figure 3), by evaluating the champions of each generation on images not seen during evolution. The percentage change in training and testing performance is then measured. At the beginning of training, all the approaches significantly fluctuate in the difference between training and testing performance. However, as training continues, the features generalize better to the testing set, such that, by the end of training most approaches are less than two percent off the training performance.

4. CONCLUSION

This paper investigated deep learning through neuro-evolution, specifically HyperNEAT. HyperNEAT alone is shown to struggle in finding ANNs that perform well directly in image classification. However, HyperNEAT demonstrates an effective ability to learn feature extractors for other machine learning approaches. Thus HyperNEAT provides a potentially interesting path for combining reinforcement learning and supervised learning in image classification, as evolution and lifetime learning combine to create the capabilities in biological neural networks.

5. ACKNOWLEDGMENTS

This work was supported and funded by the SSC Pacific Naval Innovative Science and Engineering (NISE) Program.

6. REFERENCES

[1] D. D'Ambroiso and K. O. Stanley. Evolving policy geometry for scalable multiagent learning. In *Proc. of the 9th International Conf. on AAMAS*, page 8, New York, NY, USA, 2010. ACM Press.

[2] J. Gauci and K. O. Stanley. Autonomous evolution of topographic regularities in artificial neural networks. *Neural Computation*, page 38, 2010.

[3] T. Gliga and G. Dehaene-Lambertz. Structural encoding of body and face in human infants and adults. *Journal of Cognitive Neuroscience*, 17(8), August 2005.

[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to docment recognition. *IEEE*, 86(11):2278–2324, 1998.

[5] K. O. Stanley, D. B. D'Ambrosio, and J. Gauci. A hypercube-based indirect encoding for evolving large-scale neural networks. *Artificial Life*, 15, 2009.

[6] P. Verbancsics and K. O. Stanley. Evolving static representations for task transfer. *Journal of Machine Learning Research*, 11, 2010.

[7] X. Yao. Evolving artificial neural networks. *Proceedings of the IEEE*, 87(9):1423–1447, 1999.