

Differing Artificial Neural Network Applications And Efficiencies

Abstract

Progression in the field of artificial neural networks had dwindled in 1969, but has recently seen more interest and varying applications. The full extent of the uses of these networks is not yet known, however they show varying effectiveness depending on the application, and their configuration. This proposal details an experiment in assessing factors and differences in efficiency and practicality of how a trained artificial neural network (ANN) can perform in natural language processing, dataset classification, and pattern analysis, or more specifically, solving a Rubik's Cube. These factors include training time, processing time, and degree of accuracy. Each subject area differs in the implementation of neural networks, as each ANN is designed specifically for the task in a separate environment, however one is expected to be timelier in terms of throughput or more cost effective in power consumption.

1. Background and Significance

Significance: Artificial Neural Networks have been utilized across many differing subjects such as complex pattern recognition, approximating and optimizing functions, prediction, and several other fields. Some specific applications include wind speed prediction/forecasting [1][2], location prediction [3], phonetic recognition [4], vast image pattern classification [5], student retention rates [6] and many more. Each application of a neural network may vary in effectiveness, depending on the task/problem it is attempting to solve, due to the differing algorithms used, methods of training, the topology/configuration of the layers within the

network, or even the type of ANN itself. Our objective is to conclude which application, out of three separate experiments involving dataset classification, pattern analysis, and natural language processing, an ANN is most efficient and suited for in terms of processing requirements, training time, and degree of accuracy.

Background: As a basic description, an Artificial Neural Network is a computational processing model or paradigm that is ultimately inspired by the way our human brains process information [7]. Humans learn largely from experience and example, and an Artificial Neural Network (ANN), alike the biological neural network in each of our heads, learns in a similar way from a given set of data. In the human brain, learning involves adjusting synaptic connections between neurons over exposure and experience with certain content. Likewise, an ANN is comprised of fully interconnected layers of artificial neurons. Depending on the task at hand, the ANN may be more or less complex in its configuration. Generally, a more complex neural network is composed of more layers and more neurons in each layer. Between these layers, connections to each neuron exist as weights, much alike the synapses in a real brain. There are many different types of artificial neural networks, including Radial Basis Function (RBF), Recurrent, Modular, Physical, and Dynamic neural networks, all with differing applications and learning methods. They all stem from the most typical, or simplest configuration known as a *feedforward* network, consisting of an input layer, one or more hidden layers, and an output layer. The input layer sends values to each of the interconnected neurons within the next hidden layer, and depending on the value of the resulting calculation made by multiplying the values by the weights of the neurons within that hidden layer (typically a value *greater* than a specified threshold), sending of the value to the next layer in line is triggered [7]. The weighted

connections between the neurons are generally changed many times over the course of training the network. In the feedforward style, this adjusting is done through feedback process called *backpropagation*. Through backpropagation, the network compares the output value it calculated with the actual output value it was attempting to reach, and adjusts the weights between its neurons in lieu of obtaining a more accurate result. These changes are normally organized by generations, which can be further broken down into species. The species are smaller changes to the previous generation that are given a score based on how accurate their results are. The best scoring species in the generation are used to create the next generation [8]. The intended end goal of training the ANN is to output the exact desired value through this method of backpropagation. At this point, one should be able to present the ANN with data it has never seen, with which it can make an educated prediction based on what it has learned [7].

2. Methodology

Once each separate experiment is carried out, we will compare the CPU/processing consumption, training times, and accuracy of the artificial neural networks within each project to deduce which of the three projects was benefited the most by the use of an ANN.

October	November	December
Gather resources and Begin Experimentation	Share findings from each member and compare networks	Manuscript preparation and submission
	Presentation preparation and	

	submission	
--	------------	--

Table 1. Our projected timeline for the remainder of the project

2a. Samantha

For my pattern analysis experiment, I will using the NeuroEvolution of Augmenting Topologies (NEAT); this method was developed by Kenneth O. Stanley at the University of Texas in Austin. I will be implementing this algorithm using Python. NEAT is a feedforward neural network, but it has a few differences than the typical feedforward neural network [9]. NEAT doesn't have clear defined layers that the inputs travel through, instead it is more of a web of connections. First part of training the network to solve a 2x2x2 Rubik's cube will include defining what all the possible moves that can be made are. Then looking at what the new configuration is to see if it has gotten closer to solving the cube, this will have to be given a score so that it will know if it has come closer or further away from achieving a solved cube. Starting out, I will have the cube only a few moves away from being solved and then increase this as the network learns. Each generation of the network will be documented with a fitness score that can later be graphed to see the evolution of the network and will be shared with the rest of the group.

2b. Josh

For my data classification experiment, I will be utilizing the standalone program WEKA (Waikato Environment for Knowledge Analysis) developed by the University of Waikato, New Zealand and written in Java (<https://www.cs.waikato.ac.nz/ml/weka/>). WEKA houses a multilayer perceptron neural network, which is a classification of the aforementioned

feedforward type. I will train this neural network with a database similar to the renowned IRIS dataset, and section the data into differing ratios of training and testing sets. Separate configurations of weights, number of hidden layers, and number of input nodes will be implemented in an effort to increase the accuracy of the classifications from the set. Each configuration change will be documented, along with records of the subsequent results from each configuration, and the computational statistics will be shared with the other group members for us to compare the effectiveness.

2c. Kelly

For my natural language processing research, I will be creating my own ANN in C++ and train it to learn what legitimate sentences are made of. First, I will train it using a dictionary that assigns an attribute to each word such as a noun or a verb, etc. Then, I will feed the network with sentences that are all correct. The machine learning algorithm should be able to look at all the successful sentences and recognize all the common traits between them. It will use the attributes to create rules for what a correct sentence is. To do this, I will be using an open source machine learning implementation called MLPack (<http://www.mlpack.org/about.html>) to do neighborhood analysis for the words in a sentence [10]. Neighborhood analysis will reveal in successful sentences, what is the likelihood of an adjective is right before a noun, or other rules based on closeness. In order to test the one that I create, I will feed it data containing both correct and incorrect grammar and assess its accuracy in determining if they are right or wrong. I will be finding the rate of efficiency for this neural network as more data is fed into it in a natural language processing scenario. My findings will be compiled along with the group.

3. Statement of Qualification

Through pursuing computer science degrees at Eastern Connecticut State University (ECSU), we have taken a sufficient amount of courses that have furthered our skills in the computational field, and obligated us to working with other students in group projects throughout our time enrolled. While we have not had any prior experience in the field of neural networks, we believe that our experiences at ECSU and knowledge gained from our mutual senior research course have given us sufficient insight with which to explore the complexities of artificial neural networks and their applications. For example, CSC 305 Data Mining and Applications is a course in which we learned about machine learning and its applications with statistical and computational models. We are each interested in artificial intelligence and machine learning in our own ways and we wish to gain insight into how neural networks operate under our unique applied functions.

4. Expected Outcomes

At the completion of this project, we will gain a greater understanding of the implementation of neural networks. Each team member will have implemented an ANN for a different application and measured its effectiveness. We will each know the accuracies of the Rubik's cube solver, data classifier, and sentence checker. This will allow us to determine which of the applications had the most success using an ANN. It will also provide us with a deeper knowledge of the behavior of neural networks such as their strengths and limitations. Additionally, we will gain an understanding of the sizes and scopes of current open source ANN implementations and how to

incorporate them into our own research projects. A presentation of the findings will be given at the end of November, and a manuscript of these findings will be submitted mid December.

Bibliography

- [1] M. Mohandes, S. Rehman, and T. O. Halawani, *A neural networks approach for wind speed prediction*, vol. 13. 1998.
- [2] D. Corne, A. Reynolds, S. Galloway, E. Owens, and A. Peacock, *Short term wind speed forecasting with evolved neural networks*. 2013.
- [3] Mantoro and A. K. Olowolayemo, "Mobile user location prediction enabled services in ubiquitous computing," *Proceedings of the 7th International Conference on Advances in Mobile Computing and Multimedia - MoMM 09*, Dec. 2009.
- [4] Waibel, T. *et al.*, "Phoneme Recognition Using Time-Delay Neural Networks," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, Mar. 1989.
- [5] Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.
- [6] M. Plagge, "Using Artificial Neural Networks to Predict First-Year Traditional Students Second Year Retention Rates," *Proceedings of the 51st ACM Southeast Conference on - ACMSE 13*, Apr. 2013.
- [7] C. Woodford. (2017) "How neural networks work - A simple introduction." [Online]. Available: <http://www.explainthatstuff.com/introduction-to-neural-networks.html>. [Accessed: 28-Sep-2017].
- [8] SethBling, *MarI/O - Machine Learning for Video Games*.
- [9] J. Heaton, "A NEAT Approach to Neural Network Structure." *Forecasting & Futurism*, Dec-2013.
- [10] "mlpack: a scalable c++ machine learning library", 2017. [Online]. Available: <http://www.mlpack.org/about.html>. [Accessed: 8-Oct-2017].