

Evolutionary Approaches for ANNs Design

Antonia Azzini

University of Milan, Italy

Andrea G. B. Tettamanzi

University of Milan, Italy

INTRODUCTION

Artificial neural networks (ANNs) are computational models, loosely inspired by biological neural networks, consisting of interconnected groups of artificial neurons which process information using a connectionist approach.

ANNs are widely applied to problems like pattern recognition, classification, and time series analysis. The success of an ANN application usually requires a high number of experiments. Moreover, several parameters of an ANN can affect the accuracy of solutions. A particular type of evolving system, namely neuro-genetic systems, have become a very important research topic in ANN design. They make up the so-called Evolutionary Artificial Neural Networks (EANNs), i.e., biologically-inspired computational models that use evolutionary algorithms (EAs) in conjunction with ANNs.

Evolutionary algorithms and state-of-the-art design of EANN were introduced first in the milestone survey by Xin Yao (1999), and, more recently, by Abraham (2004), by Cantu-Paz and Kamath (2005), and then by Castellani (2006).

The aim of this article is to present the main evolutionary techniques used to optimize the ANN design, providing a description of the topics related to neural network design and corresponding issues, and then, some of the most recent developments of EANNs found in the literature. Finally a brief summary is given, with a few concluding remarks.

ARTIFICIAL NEURAL NETWORK DESIGN

In ANN design, the successful application of an ANN usually demands much experimentation. There are many parameters to set. Some of them involve ANN type, others the number of layers and nodes defining

the architecture and the connection weights. Also the training data are an important factor, and a great deal of attention must be paid to the test data to make sure that the network will generalize correctly on data which has not been trained on.

Feature selection, structure design, and weight training can be regarded as three search problems in the discrete space of subsets of data attributes, the discrete space of the possible ANN configurations, and the continuous space of the ANN parameters, respectively.

Architecture design is crucial in the successful application of ANNs because it has a significant impact on their information-processing capabilities. Indeed, given a learning task, an ANN with only a few connections and linear nodes may not be able to perform the task at all, while an ANN with a large number of connections and nonlinear nodes may overfit noise in the training data and lack generalization. The main problem is that there is no systematic way to design an optimal architecture for a given task automatically.

Several methods have been proposed to overcome these shortcomings. This chapter focuses on one of them, namely EANNs. One distinct feature of EANNs is their adaptability to a dynamic environment. EANNs can be regarded as a general framework for adaptive systems, i.e., systems that can change their architectures and learning rules appropriately without human intervention.

In order to improve the performance of EAs, different selection schemes and genetic operators have been proposed in the literature. This kind of evolutionary learning for ANNs has also been introduced to reduce and, if possible, to avoid the problems of traditional gradient descent techniques, such as Backpropagation (BP), that lie in the trapping in local minima. EAs are known to be little sensitive to initial training conditions, due to their being global optimization methods, while a gradient descent algorithm can only find a local

optimum in a neighbourhood of the initial solution. EANNs provide a solution to these problems and an alternative for controlling network complexity.

ANN design can be regarded as an optimization problem. Tettamanzi and Tomassini (2001) presented a discussion about evolutionary systems and their interaction with neural and fuzzy systems, and Cantu-Paz and Kamath (2005) also described an empirical comparison of EAs and ANNs for classification problems.

EVOLUTIONARY ARTIFICIAL NEURAL NETWORKS

There are several approaches to evolve ANNs, that usually fall into two broad categories: *problem-independent* and *problem-dependent* representation of EAs. The former are based on a general representation, independent of the type and structure of the ANN sought for, and require the definition of an encoding scheme suitable for Genetic Algorithms (GAs). They can include mapping between ANNs and binary representation, taking care of decoders or repair algorithms, but this task is not usually easy.

The latter are EAs where chromosome representation is a specific data structure that naturally maps to an ANN, to which appropriate genetic operators apply.

EAs are used to perform various tasks, such as connection weight training, architecture design, learning rule adaptation, input feature selection, connection weight initialization, rule extraction from ANNs, etc. Three of them are considered as the most popular at the following levels:

- *Connection weights* concentrates just on weights optimization, assuming that the architecture of the network is given. The evolution of weights introduces an adaptive and global approach to training, especially in the reinforcement learning and recurrent network learning paradigm, where gradient-based training algorithms often experience great difficulties.
- *Learning rules* can be regarded as a process of “learning how to learn” in ANNs where the adaptation of learning rules is achieved through evolution. It can also be regarded as an adaptive process of automatic discovery of novel learning rules.

- *Architecture* enables ANNs to adapt their topologies to different tasks without human intervention. It also provides an approach to automatic ANN design as both weights and structures can be evolved. In this case a further subdivision can be made by defining a “pure” architecture evolution and a simultaneous evolution of both architecture and weights.

Other approaches consider the evolution of transfer functions of an ANN and input feature selection, but they are usually applied in conjunction with one of the three methods above in order to obtain better results.

The use of evolutionary learning for ANNs design is no more than two decades old. However, substantial work has been made in these years, whose main outcomes are presented below.

Weight Optimization

Evolution of weights may be regarded as an alternative training algorithm. The primary motivation for using evolutionary techniques instead of traditional gradient-descent techniques such as BP, as reported by Rumelhart et al. (1986), lies in avoiding trapping in local minima and the requirement that the activation function be differentiable. For this reason, rather than adapting weights based on local improvement only, EAs evolve weights based on the fitness of the whole network.

Some approaches use GAs with real encodings for biases and weights, like in the work presented by Montana and Davis (1989); others used binary weights encoding at first, and then implemented a modified version with real encodings as Whitley *et al.* (1990). Mordaunt and Zalzala (2002) implemented a real number representation to evolve weights, analyzing evolution with mutation and a multi-point crossover, while Seiffert (2001) described an approach to completely substitute a traditional gradient descent algorithm by a GA in the training phase.

Often, during the application of GAs, some problems, e.g., premature convergence and stagnation of solution can occur as reported by Goldberg (1992). In order to solve this problem, an improved algorithm was proposed by Yang et al. (2002), where a genetic algorithm, based on evolutionary stable strategy, was implemented to keep the balance between population diversity and convergence speed during evolution.

4 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/evolutionary-approaches-anns-design/10305

Related Content

Embracing Technological Disruption: Exploring the Adoption of AI-Driven Solutions in Tourism and Hospitality

Milan Sharma, Amrik Singhand - Rohit (2024). *Impact of AI and Tech-Driven Solutions in Hospitality and Tourism* (pp. 123-138).

www.irma-international.org/chapter/embracing-technological-disruption/354131

A Dynamic Spoken Dialogue Interface for Ambient Intelligence Interaction

Germán Montoro, Pablo A. Hayaand Xavier Alamán (2010). *International Journal of Ambient Computing and Intelligence* (pp. 24-51).

www.irma-international.org/article/dynamic-spoken-dialogue-interface-ambient/40348

Revolutionizing Agriculture Through Blockchain: A Bibliometric Analysis of Emerging Trends and Applications

Durga Prasad Singh Samanta, B.C.M. Patnaik, Ipseeta Satpathyand Alex Khang (2023). *Handbook of Research on AI-Equipped IoT Applications in High-Tech Agriculture* (pp. 295-312).

www.irma-international.org/chapter/revolutionizing-agriculture-through-blockchain/327841

An Agent-Based Approach to Process Management in E-Learning Environments

Hokyin Lai, Minhong Wang, Jingwen Heand Huaiqing Wang (2008). *International Journal of Intelligent Information Technologies* (pp. 18-30).

www.irma-international.org/article/agent-based-approach-process-management/2441

Integrating Effect of AR and VR in Sustained Learning Outcomes in Self-Directed Learning Environments

Debasis Pani, Narayana Maharana, Sunil Kumar Pradhanand Saumendra Das (2025). *Impacts of AI on Students and Teachers in Education 5.0* (pp. 317-348).

www.irma-international.org/chapter/integrating-effect-of-ar-and-vr-in-sustained-learning-outcomes-in-self-directed-learning-environments/368638