



Chapter VI

Using Genetic Programming to Extract Knowledge from Artificial Neural Networks

Daniel Rivero, University of A Coruña, Spain

Miguel Varela, University of A Coruña, Spain

Javier Pereira, University of A Coruña, Spain

Abstract

A technique is described in this chapter that makes it possible to extract the knowledge held by previously trained artificial neural networks. This makes it possible for them to be used in a number of areas (such as medicine) where it is necessary to know how they work, as well as having a network that functions. This chapter explains how to carry out this process to extract knowledge, defined as rules. Special emphasis is placed on extracting knowledge from recurrent neural networks, in particular when applied in predicting time series.

Introduction

Through the use of artificial neural networks (ANN), satisfactory results have been obtained in the most diverse fields of application, including the classification of examples, the identification of images, and processing of natural language.

However, in some fields there is still some reticence toward their use, mainly as a result of one single issue: the fact that they do not justify their answer. An ANN extracts information from a training series (formed by input-output pairs, or simply input if unsupervised training is used). Based on the information obtained, it will be able to offer output for input not previously seen during the learning process. Even if the answer provided is correct, the ANN does not provide any information about why one particular solution has been chosen: it behaves like a “black box”. This is unacceptable in certain fields. For example, any system used in medical diagnosis must not only reach the correct conclusions, but also be able to justify on what it has based its decision. For this reason, expert systems are commonly used in medicine.

Expert systems (ES) are able to explain the solution or response achieved, which is their main core and also their guarantee of success. Therefore, this chapter attempts to develop a system that carries out an automatic extraction of rules from previously trained ANN, thereby obtaining the knowledge that an ANN obtains from the problem it solves.

Different rule-extraction techniques using ANN have been used to date, always applied to multi-layer ANN, as they are easier to handle. These networks also have a limited capacity with regard to the knowledge that can be distributed among their connections.

As may be inferred, the extraction of rules and expressions from recurrent ANN is more complicated, due to the fact that past states intervene in neural activation, and that their distributed knowledge capacity is considerably higher than that of multi-layer ANN, as there are no restrictions to neural connectivity. Also, if recurrent ANNs are used in dynamic problems where certain time characteristics such as the prediction of time series intervene, the task of extracting using the methods developed so far becomes harder, if not impossible for most of them.

However, if ANN provides good results, why reject them? It would be enough to find a method that justifies the output offered by ANN based on the input values. This method would have to be able to be applied to networks of any type, meaning that they would have to comply with the following characteristics (Tickle, 1998):

- **Independence of the architecture.** The method for extracting rules should be able to be applied independently from the architecture of the ANN, including recurrent architectures.
- **Independence of the training algorithm.** The extraction of rules cannot depend on the algorithm used for the ANN’s learning process.
- **Correction.** Many methods for extracting rules only create approximations of the functioning of the ANN, instead of creating rules as precisely as possible.

23 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/using-genetic-programming-extract-knowledge/5366

Related Content

Trigonometric Polynomial Higher Order Neural Network Group Models and Weighted Kernel Models for Financial Data Simulation and Prediction

Lei Zhang, Simeon J. Simoff and Jing Chun Zhang (2009). *Artificial Higher Order Neural Networks for Economics and Business* (pp. 484-503).

www.irma-international.org/chapter/trigonometric-polynomial-higher-order-neural/5296

Reliability Study of Polymers

Amit Sachdeva and Pramod K. Singh (2020). *AI Techniques for Reliability Prediction for Electronic Components* (pp. 45-54).

www.irma-international.org/chapter/reliability-study-of-polymers/240489

Overview of Artificial Neural Networks and their Applications in Healthcare

Joarder Kamruzzaman, Rezaul Begg and Ruhul Sarker (2006). *Neural Networks in Healthcare: Potential and Challenges* (pp. 1-19).

www.irma-international.org/chapter/overview-artificial-neural-networks-their/27271

Electric Machines Excitation Control via Higher Order Neural Networks

Yannis L. Karnavas (2010). *Artificial Higher Order Neural Networks for Computer Science and Engineering: Trends for Emerging Applications* (pp. 366-396).

www.irma-international.org/chapter/electric-machines-excitation-control-via/41675

Analysis and Improvement of Function Approximation Capabilities of Pi-Sigma Higher Order Neural Networks

Junichi Murata (2010). *Artificial Higher Order Neural Networks for Computer Science and Engineering: Trends for Emerging Applications* (pp. 239-254).

www.irma-international.org/chapter/analysis-improvement-function-approximation-capabilities/41669