Expert Systems with Applications 41 (2014) 2660–2677

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Intelligent forecasting system based on integration of electromagnetism-like mechanism and fuzzy neural network

Peitsang Wu *, Yung-Yao Hung, Zi-po Lin

Department of Industrial Management, I-Shou University, Kaohsiung 840, Taiwan, ROC

ARTICLE INFO

Keywords:
Fuzzy neural networks
Electromagnetism-like mechanism
Back propagation algorithm
Weights elimination

ABSTRACT

Fuzzy neural network (FNN) architectures, in which fuzzy logic and artificial neural networks are integrated, have been proposed by many researchers. In addition to developing the architecture for the FNN models, evolution of the learning algorithms for the connection weights is also a very important. Researchers have proposed gradient descent methods such as the back propagation algorithm and evolution methods such as genetic algorithms (GA) for training FNN connection weights. In this paper, we integrate a new meta-heuristic algorithm, the electromagnetism-like mechanism (EM), into the FNN training process. The EM algorithm utilizes an attraction–repulsion mechanism to move the sample points towards the optimum. However, due to the characteristics of the repulsion mechanism, the EM algorithm does not settle easily into the local optimum. We use EM to develop an EM-based FNN (the EM-initialized FNN) model with fuzzy connection weights. Further, the EM-initialized FNN model is used to train fuzzy if–then rules for learning expert knowledge. The results of comparisons done of the performance of our EM-initialized FNN model to conventional FNN models and GA-initialized FNN models proposed by other researchers indicate that the performance of our EM-initialized FNN model is better than that of the other FNN models. In addition, our use of a fuzzy ranking method to eliminate redundant fuzzy connection weights in our FNN architecture results in improved performance over other FNN models.

1. Introduction

Application of artificial neural network (ANN) learning algorithms to fuzzy systems enhances their performance and is a demonstrably innovative approach. Takagi and Hayashi (1991) introduced an ANN model integrated with fuzzy inference. Lin and Lee (1991) proposed a neural network-based fuzzy logic control system that demonstrated the mighty learning power of neural networks. Ishibuchi, Fujioka, and Tanaka (1992), Ishibuchi, Fujioka, and Tanaka (1993) and Ishibuchi, Kwon, and Tanaka (1995a) also did a series of research in which neural networks learned from fuzzy if–then rules. These series of studies introduced neural network architectures with fuzzy inputs, outputs, and connection weights. ANNs that have these kinds of combinations of fuzzy functions are called fuzzy neural networks (FNN).

One of the most important factors that determine the efficacy of a neural network is its learning algorithm. In addition to the back propagation learning algorithm proposed by Ishibuchi et al. (1993) and Ishibuchi et al. (1995a), there are other methods for training the connection weights of an FNN. Buckley, Reilly, and Penmetcha (1996), Kuo (2001) and Kuo and Chen (2004) combined genetic algorithms (GA) with FNN and reported better training efficiency than that of the back propagation learning algorithm for FNN. Kuo, Horng, and Hwang (2010) further applied particle swarm optimization techniques to the FNN. Kuo, Tseng, Tien, and Warren Liao (2012) proposed an artificial immune system (AIS)-based FNN, to learn the relationship between RFID signals and a picking cart's position. Use of the electromagnetism-like mechanism (EM) for FNN training was also investigated by Wu and Hung (2010). They discovered that using EM for FNN training could result in better performance than using back-propagation training based on the results of experiments conducted in their study.

The EM algorithm is an innovative meta-heuristic algorithm for global optimization (Birbil & Fang, 2003). Some researchers have investigated and applied this method. Wu, Yang, and Wei (2004), applied this EM algorithm primarily to neural network-related training algorithms. Otherwise, an FNN model with the EM training algorithm but with the FNN architecture using crisp connection weights was also presented. However, application of using EM algorithm to generate initial fuzzy connection weights to fuzzy neural networks training has not been investigated yet. We are interested in the efficacy of FNN connection weights training in which the connection weights training is substituted with EM.

* Corresponding author.
E-mail addresses: pwu@isu.edu.tw (P. Wu), d9320001@cloud.isu.edu.tw (Y.-Y. Hung), isu69839@cloud.isu.edu.tw (Z.-p. Lin).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.11.007

In this paper, we establish an EM-initialized FNN system with the initial fuzzy connection weights generated by using EM. Further, we attempt to achieve the following objectives:

1. Establishment of an FNN system with fuzzy connection weights using EM connection weights training (the EM-initialized FNN with fuzzy connection weights).
2. Development of an error back propagation (EBP) type FNN system with fuzzy connection weights.
3. Establishment of these two FNN systems (FNN with fuzzy connection weights and EM-initialized FNN with fuzzy connection weights).
4. Illustration of the results of these two systems by means of computer simulations.
5. Investigation of the application of EM in FNN connection weights training by examining several scenarios.

The remainder of this paper is organized as follows. Section 2 presents background on FNNs and several heuristic methods for global optimization. Section 3 discusses the methodologies of FNNs, EM, and the FNN with fuzzy connection weights using EM connection weights training. Performance measures obtained and comparisons with the results from other models are discussed in Section 4. Finally, in Section 5, we conclude and outline future research.

2. Background

2.1. Fuzzy neural networks (FNN)

Artificial neural networks (ANN), fuzzy logic, and genetic systems are the three independent research fields constituting sixth generation systems (SGS) (see Fig. 2.1) (Kuo, Wu, & Wang, 2000). ANN and fuzzy logic have been used in many application areas (Lee, 1990a, 1990b; Lippmann, 1987); however, each brings its own advantages and disadvantages. Thus, methods that successfully combine these two approaches are a relevant concern and deserve further study.

Nakayama, Horikawa, Furuhashi, and Uchikawa (1992) proposed an FNN architecture with a special structure for realizing a fuzzy inference system. The concept underlying their architecture was that each membership function consisted of one or two sigmoid functions for each inference rule. However, because of the lack of a membership function setup procedure, the rule determination and membership function setup were decided by named experts, which made the decision very subjective. Ishibuchi et al. (1993) presented a learning method for neural networks that utilized expert knowledge represented by fuzzy if–then rules. It included two kinds of systems: classification systems and fuzzy control systems. For classification problems, it derived learning algorithms from cost functions. For fuzzy control problems, it derived learning algorithms from the actual fuzzy outputs and the fuzzy target outputs. However, there are no fuzzy connection weights in their FNN system. Ishibuchi et al. (1995a) subsequently proposed an architecture for FNNs with triangular fuzzy networks and triangular fuzzy connection weights. Their FNN can handle fuzzy input vectors and output fuzzy vectors with the fuzzy connection weights. The cost function is the same as that proposed by Ishibuchi et al. (1993). The most important aspect of their study is that they derived a learning algorithm from the cost function for adjusting three parameters of each fuzzy weight. Dunyak and Wunsch (1999, 1997) proposed a practical algorithm for training neural networks using fuzzy number connection weights, inputs, and outputs. They believed that fuzzy number neural networks are difficult to train because of the many α-cut constraints implied by the fuzzy connection weights. Therefore, they went a step further to create a transformation that eliminated these constraints and created standard unconstrained optimization methods. This innovative algorithm was demonstrated on a three-layer network. Kuo and Cohen (1998) introduced a feedforward ANN into the fuzzy inference represented by Takagi's fuzzy model and applied it to multi-sensor integration.

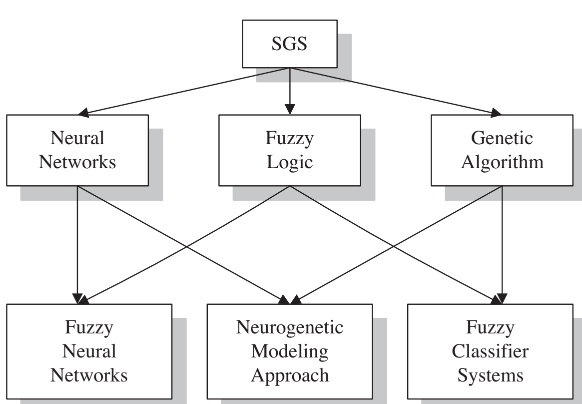

Fig. 2.1. Sixth generation systems (SGS).

Table 2.1 shows the architecture of the neural network fuzzifications. In Type 1, the FNN is used to classify the problem of the fuzzy input vectors into a crisp class. In order to implement fuzzy if–then rules using neural networks, FNNs with fuzzy inputs, fuzzy targets, and crisp connection weights are used (Type 2). Ishibuchi et al. (1995a) developed Type 3 and Type 4 FNNs, whose connection weights are fuzzy. Their models can handle classification problems and were derived for FNNs with fuzzy connection weights. One thing that needs to be noted in Table 2.1 is that the last three cases, Type 5, Type 6, and Type 7, are not practicable. If both the inputs and the connection weights of the FNNs are real numbers, the outputs must also be real numbers. In the same way, if the outputs are fuzzy numbers, there should be fuzzy connection weights for deriving these fuzzy outputs. In Type 7, the fuzzy connection weights are not necessary because the targets are real numbers, so using the fuzzy connection weights here is a waste of computation time in the FNN system.

In the case of Type 4, Kuo and Xue (1999) proposed an innovative FNN that has fuzzy inputs, fuzzy outputs, and fuzzy connection weights. They used asymmetric Gaussian functions as fuzzy numbers in this FNN to make it applicable to realistic situations. They also utilized an EBP-type learning procedure as their learning algorithm. Further, in development of their FNN architecture, Kuo, Wu, and Wang (2002) proposed a weight elimination method for FNN that actually can improve an FNN's performance.

In this paper, the FNN architecture with fuzzy connection weights, which was derived by Kuo et al. (2002), is used for FNN training. In the next step, we integrate EM into this FNN architecture and substitute the EBP-type learning procedure.

Table 2.1
Architecture of neural network fuzzifications.

	Weights	Inputs	Outputs
Conventional NN	Real number	Real number	Real number
Fuzzy NN: Type 1	Real number	Fuzzy number	Real number
Fuzzy NN: Type 2	Real number	Fuzzy number	Fuzzy number
Fuzzy NN: Type 3	Fuzzy number	Real number	Fuzzy number
Fuzzy NN: Type 4	Fuzzy number	Fuzzy number	Fuzzy number
Fuzzy NN: Type 5	Real number	Real number	Fuzzy number
Fuzzy NN: Type 6	Fuzzy number	Real number	Real number
Fuzzy NN: Type 7	Fuzzy number	Fuzzy number	Real number