IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 29 November 2022, accepted 28 December 2022, date of publication 2 January 2023, date of current version 5 January 2023.

Digital Object Identifier 10.1109/ACCESS.2022.3233596

SURVEY

A Survey on the Optimization of Artificial Neural Networks Using Swarm Intelligence Algorithms

BIBI AAMIRAH SHAFAA EMAMBOCUS, MUHAMMED BASHEER JASSER, (Member, IEEE), AND ANGELA AMPHAWAN, (Member, IEEE)
Department of Computing and Information Systems, School of Engineering and Technology, Sunway University, Petaling Jaya, Selangor 47500, Malaysia
Corresponding author: Muhammed Basheer Jasser (basheerj@sunway.edu.my)

This work was supported in part by the Sunway University Internal Grant Scheme 2022 under Grant GRTIN-IGS-DCIS[S]-11-2022, and in part by the Publication Support Scheme.

ABSTRACT Artificial Neural Networks (ANNs) are becoming increasingly useful in numerous areas as they have a myriad of applications. Prior to using ANNs, the network structure needs to be determined and the ANN needs to be trained. The network structure is usually chosen based on trial and error. The training, which consists of finding the optimal connection weights and biases of the ANN, is usually done using gradient-descent algorithms. It has been found that swarm intelligence algorithms are favorable for both determining the network structure and for the training of ANNs. This is because they are able to determine the network structure in an intelligent way, and they are better at finding the most optimal connection weights and biases during the training as opposed to conventional algorithms. Recently, a number of swarm intelligence algorithms have been employed for optimizing different types of neural networks. However, there is no comprehensive survey on the swarm intelligence algorithms used for optimizing ANNs. In this paper, we present a review of the different types of ANNs optimized using swarm intelligence algorithms, the way the ANNs are optimized, the different swarm intelligence algorithms used, and the applications of the ANNs optimized by swarm intelligence algorithms.

INDEX TERMS Artificial neural network, swarm intelligence, optimization.

I. INTRODUCTION

Artificial Neural Networks (ANNs) are computational models that simulate the biological neural network that constitutes the human brain to generate inferences based on certain given information. They are suitable for both supervised and unsupervised learning for solving a myriad of classification, regression, clustering, and association problems in a multitude of areas. Notably, ANN has been a prominent algorithm in the domain of machine learning, and has paved the way for the advancement in multiple areas such as natural language processing, fraud detection, computational biology, computer vision, unassisted control of vehicles, speech recognition, medical diagnosis and recommendation systems [1]. Recently, ANNs have been applied for making decisions in healthcare organizations [2], for forecasting the energy use in buildings [3], for the development of greenhouse technology [4], for detecting faults in photovoltaic technology [5], and for the forecast of solar power energy [6].

Similar to the biological neural network, ANNs consist of neurons, also called nodes, which are connected to nodes of other layers of the network by weighted links. They usually consist of one input layer, one or several hidden layers and one output layer. The network structure, that is the number of hidden layers and the number of neurons in the layers, is usually determined using a trial-and-error method. However, optimization algorithms can be employed to choose the optimal structure of a neural network in an intelligent way. As for the connection weights, they are specified during the training of the artificial neural network which is a requisite step prior to using the ANN. During the training process, the connection weights and biases are adjusted to enable the neural network to produce the right output based on the inputs given. For supervised learning, the training of an ANN is done by reducing the difference in error between the predicted output from the ANN and the target output from the training

The associate editor coordinating the review of this manuscript and approving it for publication was Yeliz Karaca.

data. Hence, artificial neural networks can be optimized by optimizing the connection weights and biases during the training process or by optimizing the network structure.

Artificial neural networks are conventionally trained using gradient-descent-based algorithms including back-propagation, Quasi-Newton, conjugate gradient, Levenberg-Marquardt, and Gauss-Newton [7]. These are local search algorithms that have high exploitation capabilities. However, since they do not have exploration capabilities, they are often incapable of locating the optimal connection weights for the ANN in the search space and they often get stuck in local optima [7]. This causes the trained neural networks to have a low accuracy. Recently, the use of swarm intelligence algorithms for the training of neural networks has been noticed to be more beneficial owing to their exploration and exploitation capabilities. Moreover, they are also useful for determining the network structure in an intelligent way.

Swarm intelligence algorithms are metaheuristic optimization algorithms that are inspired by the behaviour of groups of animals or swarms of insects as they interact among themselves and their environment. Specifically, they utilize the simple collective behaviour of a population of some biological organism which gives rise to a global intelligence. This allows swarm intelligence algorithms to solve complex optimization problems using a population of artificial search agents interacting among themselves and their environment.

Multiple swarm intelligence algorithms and their hybrids have been employed for optimizing artificial neural networks. However, to the best of our knowledge, there is no comprehensive survey on the different swarm intelligence algorithms that have been employed to optimize artificial neural networks. In this paper, we present a review of the swarm intelligence algorithms employed for optimizing ANNs in terms of the different types of ANNs that have been optimized using swarm intelligence algorithms, the different ways by which the ANNs have been optimized using swarm intelligence algorithms, the different original and hybrid algorithms used, and the different supervised learning tasks that the trained neural networks have been employed for. The remainder of the paper is structured as follows: in Section II, a background on artificial neural networks and swarm intelligence algorithms is given. In Section III, the different types of ANNs which have been optimized using swarm intelligence algorithms are presented, in Section IV, the way by which the ANNs have been optimized is presented, in Section V, the different swarm intelligence algorithms used for optimizing neural networks are presented, and in Section VI, the applications of the optimized neural networks are presented. Finally, in Section VII, the conclusions are given.

II. BACKGROUND

A. ARTIFICIAL NEURAL NETWORKS

The simplest ANN is called a perceptron which is a single-layer neural network consisting of input values,

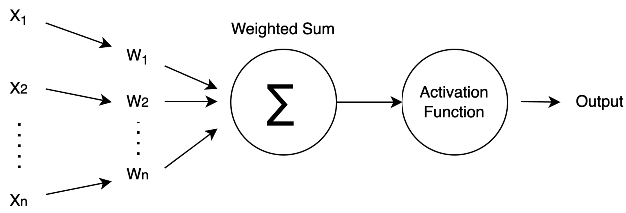

FIGURE 1. Structure of a perceptron.

weights, biases, net sum and an activation function. It is usually used as a binary classifier in supervised learning to categorize the inputs into one of two classes. The weights represent the importance of each input and the weighted sum of the inputs is used by the activation function to classify the data. Fig. 1 shows the structure of a perceptron.

The perceptron can only be used for simple binary problems. Hence, multiple other neural networks have been developed by taking the perceptron as the basic building block. This is usually executed using multiple layers of perceptrons with one or more perceptrons in each layer. The perceptrons, also called nodes or neurons, in each layer make conclusions based on the results obtained by those in the previous layer and they transmit the results to the perceptrons in the next layer.

There are three main types of ANNs; Feedforward Neural Networks (FNNs), Recurrent Neural Networks (RNNs), and Convolutional Neural Networks (CNNs). Feedforward neural networks are ANNs in which information is only transferred in one direction from the input nodes through hidden layer nodes and finally to the output nodes. They do not consist of any connection loops, and are usually used for identifying patterns between independent variables.

Recurrent neural networks are ANNs in which there are connection loops, that is the output of a layer is transmitted back to the input to determine the output. This acts like a memory storing all the information in order to generate the output. RNNs are usually used for variables which involve a sequence, that is they are dependent on one another or for time series data.

Convolutional neural networks are specialized neural networks designed mainly for working with two-dimensional images. They usually consist of several blocks such as convolution layers, pooling layers, and fully connected layers which identify features in the data.

B. SWARM INTELLIGENCE ALGORITHMS

Swarm intelligence algorithms are inspired by various populations of biological organisms in nature. They usually replicate certain characteristics of specific organisms as they interact among themselves and their environment for achieving certain tasks intelligently by making use of simple organized steps. Swarm intelligence algorithms consist of a population of artificial search agents that interact similarly to a specific group of biological organisms in a search space. These simple interactions allow the algorithm to look for the optimal solution for a problem in a heuristic manner. Hence, they are able to solve a myriad of optimization problems by providing either optimal or near-optimal solutions in a reasonable time period.

Swarm intelligence algorithms can be employed to solve different types of optimization problems including continuous, discrete or multi-objective optimization problems. Hence, they have numerous applications in a variety of domains. For example, they can be used in water resources engineering [8], in wireless networks [9], in cloud-based Internet of Things [10], in optical systems [11], in recommender systems [12], in anomaly detection systems [13], and in supply chain management [14]. They can also be used for clustering [15], for feature selection [16] and for solving the traveling salesman problem [17], [18]. Moreover, they have several applications in optimal designs, electrical engineering, networking, mechanical engineering, machine learning, resource allocation, and digital image processing [19].

III. OPTIMIZING DIFFERENT TYPES OF ARTIFICIAL NEURAL NETWORKS

There exist many different types of artificial neural networks having different architectures which are used for various applications. Most of the neural networks can be classified as one of the three main types of ANNs, namely the feedforward neural network, the recurrent neural network, or the convolutional neural network.

We have found that there are four types of feedforward neural networks which have been optimized using swarm intelligence algorithms; the Multi-Layer Perceptron (MLP), the Deep Feedforward network, the Extreme Learning Machine (ELM), and the Radial Basis Function (RBF) neural network. Five types of ANNs which have been optimized by swarm intelligence algorithms can be classified as recurrent neural networks, namely, the Nonlinear Auto-Regressive with Exogenous input (NARX), the Elman RNN, the deep RNN, the Long Short-Term Memory (LSTM), and the Jordan RNN. Moreover, there are four types of ANNs which can be classified as convolutional neural networks and which have been optimized using swarm intelligence algorithms, including the Visual Geometry Group Neural Network (VGGNet), the Residual Neural Network (ResNet), the GoogLeNet, and the U-Net.

The taxonomy in Fig. 2 depicts the classification of the different works in which swarm intelligence algorithms have been used for optimizing artificial neural networks based on the type, that is, the architecture of the neural network.

In [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45], and [46] swarm intelligence algorithms are employed for optimizing feedforward multilayer perceptrons. A multilayer perceptron is a neural network consisting of more than one perceptron. Usually, it is comprised of an input layer, a variable number of hidden layers, and an output layer, each with a varying number of perceptrons. The perceptrons of each layer are connected to the perceptrons of the precedent and subsequent layers via connection weights. In [20], six MLPs consisting of 3, 4, 4, 9, 9, and 22 neurons in the input layer, 7, 9, 9, 19, 19, and 45 neurons in the hidden layer, and 1, 3, 1, 1, 1, 1 neuron in the output layer are used. In [21], three MLPs consisting of seven, four and one node in the input, hidden, and output layers respectively are used. In [22], five MLPS with 20, 50, and one neuron in the input, hidden, and output layers respectively are used. In [23], one MLP with five, 10, and one neuron in the input, hidden, and output layers respectively is used. In [24], one MLP with six, ten and one node in the input, hidden, and output layers respectively is used. In [25], the MLP used consists of two, three, and one neuron in the input, hidden, and output layers respectively. In [26], two MLPs with seven, six, and one node in the input, hidden, and output layers respectively are used. In [27], three MLPs with structures 3-7-1, 4-9-1, and 22-45-1 respectively are used. In [28], one MLP with structure 8-9-1 is used. In [29], one MLP with structure 12-8-1 is used. In [31], one MLP with structure 14-5-5 is used. In [32], one MLP with structure 4-7-1 is used. In [33], one MLP with structure 2-12-1 is used. In [35], five MLPs with structures 8-1-1, 8-2-1, 8-3-1, 8-4-1 and 8-5-1 are used. In [36], an MLP with structure 20-10-1 is used. In [37], an MLP with structure 32-64-6 is used. In [40], an MLP with structure 7-5-1 is used. In [41], an MLP with structure 14-11-18 is used. In [43], 13 MLPs with structure 3-7-1, 4-9-1, 4-9-3, 13-27-3, 9-19-1, 22-45-1, 4-9-1, 6-13-1, 10-21-1, 14-29-1, 1-15-1, 1-15-1, and 1-15-1 are used. In [44], three MLPs with structures 9-19-1, 9-18-1, and 9-17-1 are used. In [46], 8 MLPs with structures 4-3-3, 4-3-1, 9-3-1, 9-3-1, 4-9-3, 4-9-1, 9-19-1, and 9-19-1 are used.

In [47], [48], [49], and [50], swarm intelligence algorithms have been used for optimizing deep feedforward neural networks. Deep feedforward networks are feedforward neural networks with more than one hidden layer. In [48], the architecture of the deep neural network employed consists of one input layer with 13 neurons, 4 hidden layers, each with six neurons, and one output layer with one neuron. In [51], an extreme learning machine is used to be trained by a swarm intelligence algorithm. An ELM is a feedforward neural network consisting of exactly one hidden layer, and the weights between the input and the hidden nodes are assigned at random and they do not change. During the training of the network, the weights between the hidden nodes and the output nodes are optimized. In [52], a swarm intelligence algorithm is used for training a radial-basis function neural network which is a feedforward neural network consisting of one input, one hidden, and one output layer and which uses Gaussian functions as the activation functions.

In [53], [54], [55], [56], [57], [58], [59], [60], [61], [62], [63], [64], and [65], swarm intelligence algorithms have been employed for optimizing different types of recurrent neural networks. In [53], the Nonlinear Auto-Regressive with Exogenous Input (NARX) recurrent neural network is used.

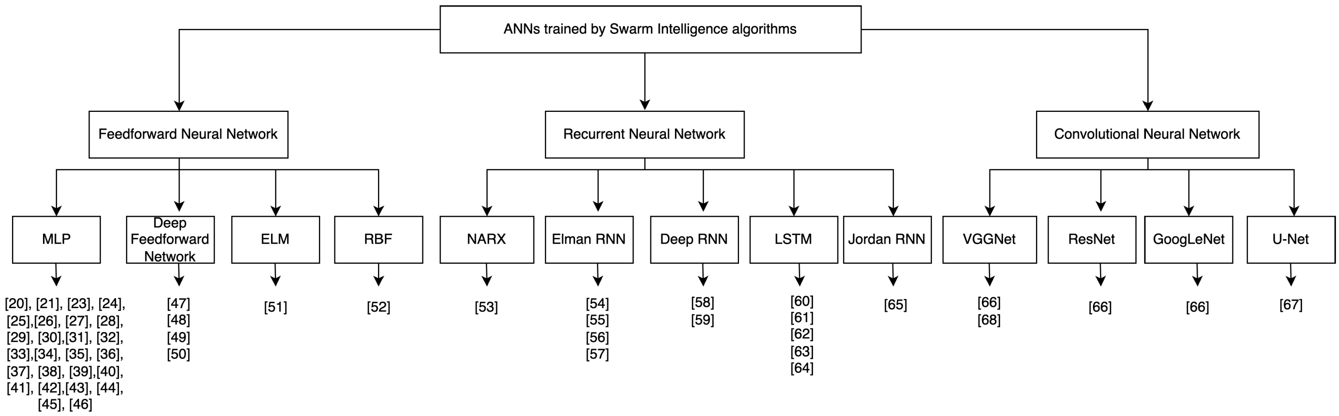

FIGURE 2. Taxonomy classifying the ANNs trained by swarm intelligence algorithms based on the type of the ANN.

Contrary to other recurrent neural networks, in NARX the feedback comes from only the output neuron instead of from all other neurons such as the hidden neurons. The NARX neural network used has one hidden layer with seven neurons. In [54], [55], [56], and [57], Elman recurrent neural networks are optimized using swarm intelligence algorithms. Elman RNNs are recurrent neural networks with an additional unit called the context unit which is connected to the hidden layer. In [54], the Elman RNN used has one input, one hidden, and one output layer with three, six, and one neurons respectively. In [55], the Elman neural network used consists of one input layer with five neurons, two hidden layers with 15 and 20 neurons respectively, and one output layer with one neuron. In [56], the RNN used consists of three hidden layers with a variable number of neurons in each layer. In [57], the Elman RNN used consists of one hidden layer with five nodes. In [58] and [59], deep recurrent neural networks with more than one hidden layer are used. In [60], [61], [62], [63], and [64], the Long Short Term Memory (LSTM) network is used. LSTM is a recurrent neural network with an additional unit called the memory cell which stores information for a long amount of time. This allows the network to learn longer-term dependencies by regulating the input to the layers and also the information that the network should forget. In [65], a swarm intelligence algorithm is used for training a Jordan RNN. A Jordan RNN is one which has a connection loop from the output of the network to the hidden layers.

In [66], [67], and [68], different types of convolutional neural networks have been optimized using swarm intelligence algorithms. In [66], three types of convolutional neural networks have been used, namely, the Visual Geometry Group Neural Network (VGGNet), the Residual Neural Network (ResNet), and the GoogLeNet. The VGGNet is a convolutional neural network with either 16 or 19 layers. It is comprised of convolutional layers followed by a pooling layer and it uses around 64 to 512 filters. The convolutional kernels and the maxpool kernels have a fixed size of 3×3 and 2×2 respectively. The ResNet is a CNN which consists of either 50, 101, or 152 layers and which incorporates gated recurrent units called skip connections. The GoogLeNet consists of 22 layers and 9 inception modules. The inception modules allow the network to choose the size of the convolutional filter in each block instead of using a fixed size. In [68], the VGGNet with 16 and 19 layers are used. In [67], a U-Net CNN is used. The U-Net is a fully connected CNN with a U-shaped encoder-decoder network architecture. There are four encoder blocks and four decoder blocks connected via a bridge.

Based on the above explanations and taxonomy, it can be seen mostly feedforward neural networks, specially MLPs have been optimized using swarm intelligence algorithms. Among the RNNS, mostly Elman RNNs and LSTMs have been optimized using swarm intelligence algorithms. CNN is the type of neural network which has been optimized by swarm intelligence algorithms the least. Only five CNNs have been optimized using swarm intelligence algorithms.

IV. OPTIMIZING ARTIFICIAL NEURAL NETWORK WEIGHTS, BIASES, AND STRUCTURE

Swarm intelligence algorithms can optimize the performance of artificial neural networks by either optimizing only the connection weights, optimizing the weights and biases, or optimizing the network structure.

The taxonomy in Fig. 3 shows the works in which swarm intelligence algorithms have been used to optimize the three main types of artificial neural networks, that is, feedforward neural networks, recurrent neural networks, and convolutional neural networks by means of optimizing the connection weights, the weights and biases, or the structure of the network.

In [22] and [24], swarm intelligence algorithms have been used for optimizing only the connection weights of feedforward neural networks. In feedforward neural networks, the neurons in one layer are linked to neurons of the previous and subsequent layers using weights which indicate the strength

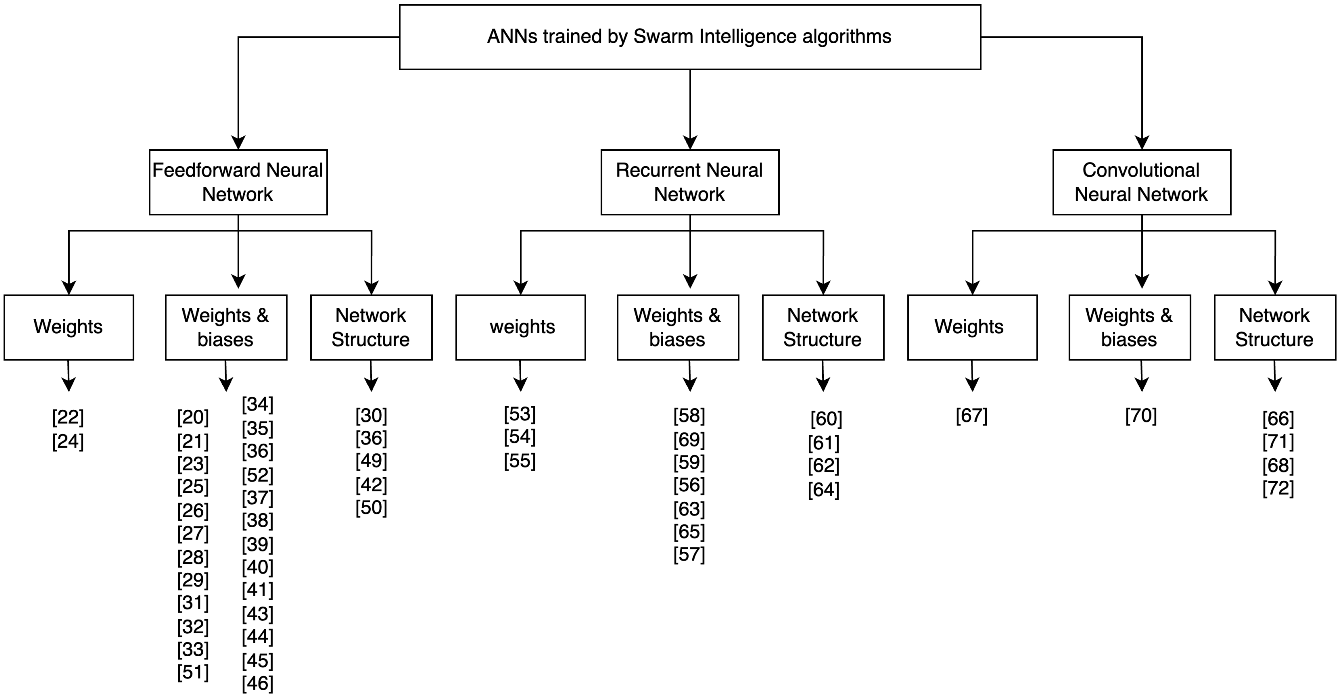

FIGURE 3. Taxonomy classifying the ANNs trained by swarm intelligence algorithms based on the parameter being optimized.

of the connection. During the training of the network, these weights are adapted such that the error between the predicted output of the network and the target output is reduced. Hence, by taking the error as the objective function, swarm intelligence algorithms are applied to find the optimal connection weights of the neural network during the training process.

During the training of feedforward neural networks, apart from the connection weights, constants called biases are also determined so as to allow the neural network to best predict the output by either adjusting the output or shifting the activation function. These biases are also optimized by taking the error of the neural network as the objective function. In [20], [21], [23], [25], [26], [27], [28], [29], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [43], [44], [45], [46], [51], and [52], swarm intelligence algorithms are used for optimizing the connection weights and biases of feedforward neural networks.

The structure of neural networks, that is the number of layers, the activation function, the number of neurons in each layer, and the learning rate, is usually determined on a trial-and-error basis prior to the training of the network. However, they can also be determined in an intelligent way using optimization algorithms. In [30], [36], [42], [49], and [50], swarm intelligence algorithms have been employed for optimizing the structure of feedforward neural networks. In [30], a swarm intelligence algorithm has been employed for optimizing the number of hidden layers, the number of neurons in the hidden layer, the activation function, and the learning rate of a feedforward neural network. In [36], in addition to the weights and biases, the number of neurons in the hidden layer is also optimized by a swarm intelligence algorithm. In [49], the number of epochs, learning rate, batch size, and the number of neurons in the first hidden layer of a feedforward neural network are optimized by a swarm intelligence algorithm. In [42], the number of hidden layers, the number of neurons in the hidden layer, and the learning rate are optimized by a swarm intelligence algorithm. In [50], the number of hidden layers of a feedforward neural network is optimized using a swarm intelligence algorithm.

In [53], [54], and [58], the connection weights of recurrent neural networks are optimized using swarm intelligence algorithms. In [56], [57], [58], [59], [63], [65], and [69], both the connection weights and the biases of recurrent neural networks are optimized during the training. In [60], [61], [62], and [64], the optimal structure of recurrent neural networks is determined using swarm intelligence algorithms prior to their training. In [61], the number of hidden layers, time lags, batch size, number of neurons in the hidden layers, and number of epochs of a recurrent neural network are optimized by a swarm intelligence algorithm. In [62], the number of neurons in the recurrent layer, the output dropout, the recurrent dropout value, and the embedding dropout are optimized using a swarm intelligence algorithm. In [64], the batch size, time step, and number of hidden layer units of a recurrent neural network are optimized using a swarm intelligence algorithm.

In [67], the connection weights of a convolutional neural network are optimized by a swarm intelligence algorithm.

In [70], the optimal connection weights and biases of a convolutional neural network are determined by a swarm intelligence algorithm. In [66], [68], [71], and [72], the structure of convolutional neural networks is optimized using swarm intelligence algorithms.

Based on the above explanations and taxonomy, it can be inferred that swarm intelligence algorithms have been primarily employed for optimizing the weights and biases, that is for training feedforward neural networks. They have also been substantially employed for the training of recurrent neural networks. However, in terms of optimizing the network structure, swarm intelligence algorithms have been scarcely employed for both FNNs and RNNs. As for CNNs, a few swarm intelligence algorithms have been employed to optimize the network structure. However, in only two works the weights and biases of CNNs have been optimized using swarm intelligence algorithms.

V. SWARM INTELLIGENCE ALGORITHMS USED FOR OPTIMIZING NEURAL NETWORKS

In recent years, multiple swarm intelligence algorithms that are inspired by different biological organisms have been proposed. A number of them have been applied for optimizing different kinds of neural networks. In this section, a description of the different types of swarm intelligence algorithms used for optimizing feedforward, recurrent, and convolutional neural networks is given.

A. FEEDFORWARD NEURAL NETWORK

The taxonomy in Fig. 4 shows the works in which swarm intelligence algorithms have been used for optimizing feedforward neural networks.

In [22], [23], [29], [32], [33], [35], [39], [40], [42], [44], [45], [48], and [50], the original Particle Swarm Optimization (PSO) algorithm is employed for training ANNs. PSO is a swarm intelligence algorithm which simulates the movement of bird flocks or fish schools. The position of each particle in the solution space represents one potential solution for the optimization problem. Each particle updates its position based on its own experience and that of the whole population. In [24], a hybrid PSO with a new variational inertia weight that balances the exploration and exploitation phases of PSO is used. In [30], a modified PSO with a center of gravity concept, called the PSOCoG is used. The PSOCoG makes use of the mean gravity of the particles in the population calculated by their distances from a reference point. In [51], a Modified PSO (MPSO) with time-varying acceleration coefficients is used for training an ANN. In [34], a Parallel PSO (PPSO) incorporating the Fork/Join framework is used for training an ANN. In [36], a hybrid of the discrete PSO, PSO, and Levenberg–Marquardt algorithms, called DPSO-PSO-FFNN is used. In [38], a hybrid of PSO, the HPSO-CA algorithm, which incorporates the cellular automata algorithm in the human-based PSO to balance its exploration and exploitation phases, is employed. In [45], a dynamic PSO called the Cooperative Quantum PSO (CQPSO) is employed for training a neural network.

In [21], [25], and [26], the original Dragonfly Algorithm (DA) is utilized for training feedforward neural networks. DA is inspired by swarms of dragonflies, particularly their hunting and migrating behaviours. In [20], a hybrid of DA and Artificial Bee Colony (ABC), called HAD is employed. In [27], a hybrid of DA and an improved Nelder-Mead algorithm, called INMDA is used for training a neural network. In the INMDA algorithm, the improved Nelder-Mead algorithm improves the local search ability of the original DA. In [46], a hybrid of DA and the hill climbing algorithm is used for training MLPs. The hill climbing algorithm is used to enhance the exploitation phase of the original DA.

In [22], in addition to PSO, the Ant Colony Optimization (ACO), the Artificial Bee Colony (ABC) and the firefly algorithms are employed for the training of ANNs. ACO is inspired by the foraging behaviour of ant colonies, that is the way by which they locate the shortest path from their colony to the food sources by leaving pheromone trails on their path. ABC is inspired by the food-searching behaviour of bees. There are three types of bees in a colony, namely the employed bees, the onlooker bees, and the scout bees. The employed bees keep track of the number of food sources discovered and their quality, the onlooker bees are responsible for choosing the best food sources, and the scout bees are responsible for searching for new food sources. The firefly algorithm is based on the flashing behaviour of firefly swarms which is used for attracting prey or other fireflies. The fireflies get attracted to other fireflies based on the intensity of the light emitted. In [47], a hybrid of ABC called ABC-ISB is employed for training a feedforward neural network. The ABC-ISB algorithm improves the local search ability of ABC by modifying the iterative formula for the employed and onlooker bees. It also improves its global search ability by introducing a new selection strategy for the scout bees.

In [26], in addition to the original DA, the original Harris Hawks Optimization (HHO) algorithm has been used for the training of a feedforward neural network. It has also been employed in [31] for the training of a feedforward neural network. The HHO algorithm is inspired by the cooperative hunting strategy of hawks as they trace, encircle, approach, and finally attack prey. The exploration is based on the strategies used by the hawks to detect prey and the exploitation is based on the strategies used to attack the prey. In [28], the original Grasshopper optimization Algorithm (GOA) and the original Grey Wolf Optimization (GWO) algorithm have been used for optimizing feedforward neural networks. GOA is based on the herding behaviour of grasshoppers during the two phases of their life cycles; the nymph phase and the adulthood phase. The exploitation phase of the algorithm is modelled based on the nymph phase which involves slow movement while the exploration phase is modelled based on the adulthood phase which comprises long and abrupt movements. GWO is based on the leadership hierarchy and hunting behaviour of grey wolfs. It simulates the

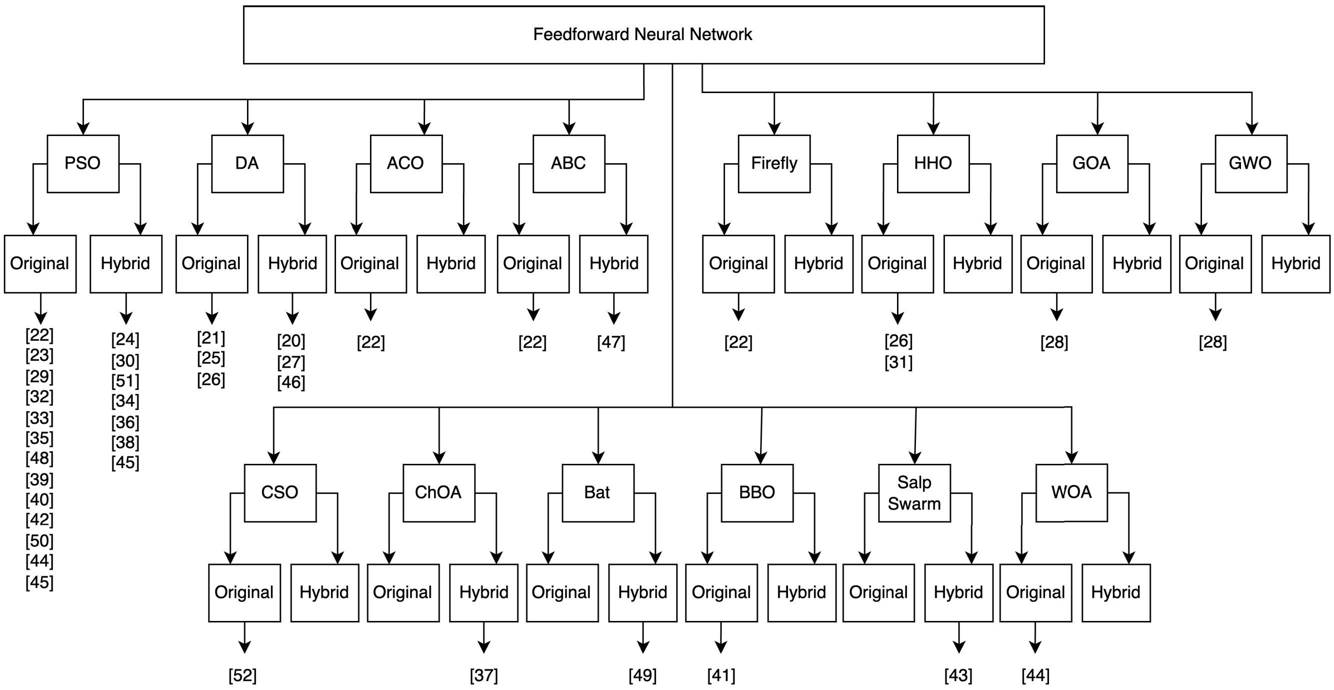

FIGURE 4. **Taxonomy classifying the FNNs trained by swarm intelligence algorithms based on the algorithm used.**

behaviour of the four types of grey wolfs, namely alpha, beta, delta, and omega as they hunt, search, encircle and attack prey.

In [52], the original Competitive Swarm Optimization (CSO) algorithm is employed for training a feedforward neural network. CSO is based on particle swarm optimization but it uses a competitive mechanism between the particles in the swarm. Unlike PSO, in CSO the historical best positions of particles are not stored and particles learn from competition winners. In [37], a modified Chimp Optimization Algorithm (ChOA) is used. ChOA is based on the attacker, driver, barrier, and chaser chimps as they hunt for prey. The modified ChOA proposes a technique for updating the chimps' positions based on proportional coefficients derived from the positions of the four types of chimps. In [49], a hybrid of the bat algorithm is employed. The bat algorithm is based on the echolocation of microbats in nature. The hybrid algorithm incorporates the mutation strategies of differential evolution in the bat algorithm. In [41], the original Biogeography-Based Optimization (BBO) algorithm is used for training ANN. BBO is inspired by the main processes of biogeography, that is the evolution, extinction, and dispersal of biological species. In [43] a hybrid of the salp swarm algorithm is used. The salp swarm algorithm is inspired by the behaviour of salps as they form salpchains and move towards food sources. The hybrid algorithm incorporates the use of multiple leaders instead of just one leader in the salp swarm algorithm so as to improve its exploration capabilities. In [44], the original Whale Optimization Algorithm (WOA) is used for training a feedforward neural network. WOA is inspired by the foraging behaviour of humpback whales. The main phases consist of the shrinking encircling and the spiral updating phases.

Table 1 shows a comparison of the different swarm intelligence algorithms used for training feedforward neural networks in terms of the population size and maximum number of iterations used and the final Root Mean Squared Error (RMSE) of the trained neural network. In the case where the RMSE is not used for evaluating the performance of the neural network, the performance metric used is mentioned within parentheses.

From Table 1, it can be deduced that the original PSO, and its hybrids have been widely used for optimizing feedforward neural networks. In 13 works, the original PSO has been employed and in seven works, its hybrids have been employed. Moreover, it can be seen that mostly the original version of swarm intelligence algorithms are used as opposed to their hybrids which are usually better performing.

Different population sizes and number of iterations have been employed for the different works. It can be noted that larger population sizes and higher number of iterations can result in low efficiency as more time and resources are needed for the optimization. Hence, it is better to achieve the lowest RMSE or highest accuracy of the neural network by using the least possible population size and number of iterations. To evaluate the performance of the optimized neural networks, the RMSE of the network is most widely used, followed by its accuracy.

TABLE 1. Comparison of Swarm Intelligence Algorithms used to train Feedforward Neural Networks.

Paper	Algorithm	Population size	Num of iterations	Range of RMSE of ANN
[22]	Original PSO, ACO, ABC, Firefly	20, 100, 40, 20	100, 200, 100, 100	64.5 - 96.2, 34.7 - 96.2, 52.5 - 96.2, 20.7 - 95.6 (accuracy)
[23]	Original PSO	85	1000	14.4 - 2.49
[29]	Original PSO	100	3000	0.0267
[32]	Original PSO	25	300	0.082 - 0.273
[33]	Original PSO		1000	0.6391
[35]	Original PSO			0.515 - 1.264
[48]	Original PSO	50	500	0.89 (AUC)
[39]	Original PSO	30	50	98.0 (accuracy)
[40]	Original PSO	20	1000	0.03162
[42]	Original PSO	10	500	0.1911, 0.2032
[50]	Original PSO	30	300	95.0 (accuracy)
[44]	Original PSO, WOA	50	500	2.456, 2.281
[45]	Original PSO, Cooperative Quantum PSO (CQPSO)	10	100	2.0 - 3.0, 1.0 - 1.75 (Average error)
[24]	Hybrid PSO	50	50	97.2 - 99.5 (accuracy)
[30]	PSOCoG	20	100	1.195E-05 - 0.225 (Mean Absolute Error)
[51]	Modified PSO (MPSO)	30	200	0.0164 - 0.0616
[34]	Parallel PSO (PPSO)			635.5932 - 1501.1942
[36]	DPSO-PSO	80	100	1.59E-02 - 7.78E-02
[38]	HPSO-CA	186-438	30	0.09798 - 0.4472
[21]	Original DA			85 (accuracy)
[25]	Original DA	208	250	92.2 - 96.7 (accuracy)
[26]	Original DA, HHO	10-100	1000	0.3422, 0.3674
[46]	Optimized DA	5-10	10-20	0.0991 - 0.78259
[20]	HAD	50	100	1.292E-22 - 2.356E-02
[27]	INMDA		200	2.341E-16 - 0.3382
[47]	ABC-ISB	100	3000	4.0130E-06 - 0.002317
[31]	Original HHO		1000	0.4444
[28]	Original GOA, GWO	10-500	1000	2.4087 - 2.4459, 2.2899 - 2.2929
[52]	Original CSO	30	3000	5.693E-03 - 8.149E-03
[37]	Modified ChOA		300	0.01414 - 0.5226
[49]	Hybrid Bat	40	100	96.5 (accuracy)
[41]	Original BBO	20	500	0.1584
[43]	Hybrid Salp Swarm	50, 200	250	2.381E-10 - 0.5617

B. RECURRENT NEURAL NETWORK

The taxonomy in Fig. 5 shows the works in which swarm intelligence algorithms have been used for optimizing recurrent neural networks.

In [55] and [61], the original PSO is utilized for optimizing recurrent neural networks. In [54], [59], [64], [65], and [69] hybrids of PSO have been used. In [54], a modified PSO which uses a new technique for selecting the global best particle is used. In [69], a hybrid of PSO and bat algorithm, called BAPSO is used for optimizing a recurrent neural network. The BAPSO algorithm includes new techniques for updating the inertia weight and the initial velocity of the particles. In [59], a modified PSO with a re-starting strategy, an early stopping rule, linearly updated coefficients and inertia weight, and bounded velocities, is employed. In [64], a hybrid PSO with an adaptive learning strategy is used for optimizing a recurrent neural network. In [65], a hybrid of PSO and the cuckoo search algorithm, called PSO-CS is used for training a recurrent neural network.

In [53], the original Continuous Flock of Starling Optimization (CFSO) is used for optimizing a recurrent neural network. The CFSO algorithm is based on PSO but with a topological component, that is the particles are attracted to the personal and global best positions, and the velocity is updated based on a subset of particles. In [58], a hybrid of the ternary bees algorithm with improved exploitation and exploration phases, called the enhanced ternary bees algorithm, is used. The Ternary bees algorithm is one that is inspired by the foraging behaviour of honey bees. In [60], ACO is used for optimizing a recurrent neural network. In [56], [61], [62], and [63], the original Flower Pollination Algorithm (FPA), grey wolf optimizer whale optimization algorithm, and artificial bee colony are employed respectively to optimize recurrent neural networks. The flower pollination algorithm is based on the cross-pollination and self-pollination processes of flowers which are used as the exploration and exploitation phases of the algorithm. In [57], a hybrid of the cuckoo search algorithm is employed. The cuckoo search algorithm is inspired by the parasitic behaviour of cuckoo birds.

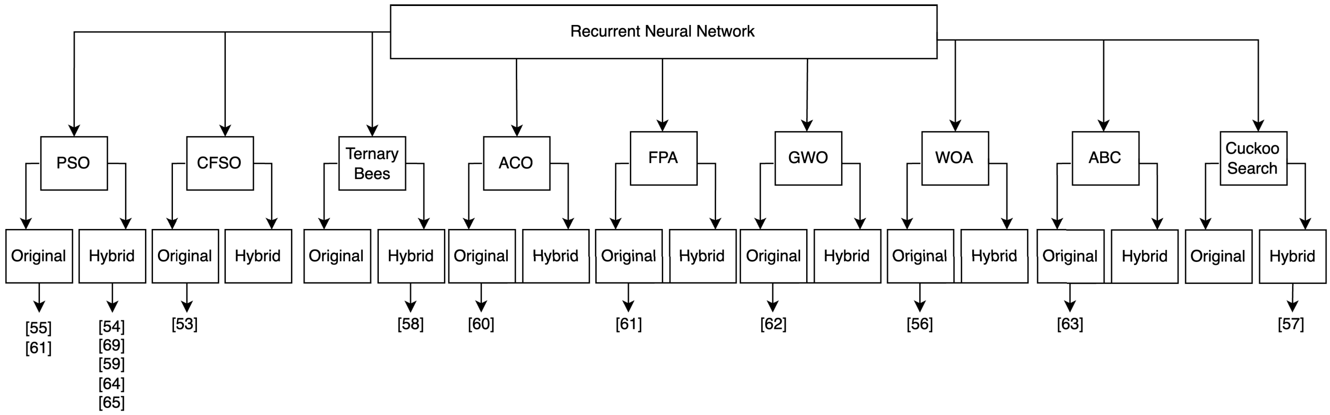

FIGURE 5. Taxonomy classifying the RNNs trained by swarm intelligence algorithms based on the algorithm used.

Table 2 shows a comparison of the different swarm intelligence algorithms employed for training recurrent neural networks in terms of the population size and maximum number of iterations used and the final RMSE of the trained neural network.

From Table 2, it can be deduced that recurrent neural have been optimized using hybrids of PSO more than the original PSO. Only two RNNs have been optimized using the original PSO while five RNNs have been trained using five different hybrids of PSO. However, for the other swarm intelligence algorithms employed, mostly the original ones have been used for optimizing RNNs.

Similar to FNNs, different population sizes and number of iterations are used by the swarm intelligence algorithms for optimizing the RNNs. Furthermore, the most used performance metric is RMSE of the RNN, followed by the accuracy of the RNN.

C. CONVOLUTIONAL NEURAL NETWORK

The taxonomy in Fig. 6 shows the works in which swarm intelligence algorithms have been used for optimizing convolutional neural networks.

In [66] and [67], the original PSO is used for optimizing convolutional neural networks. In [68], a hybrid of PSO called the Orthogonal Learning PSO (OLPSO), which incorporates the use of the orthogonal diagonalization process in PSO, is used. In [70], a hybrid of PSO called Improved PSO (IPSO) which incorporates a decreasing time function inertia weight is employed. In [72], a modified PSO, which excludes the use of the global best particle and includes a list of friends for each particle, is used for optimizing a CNN.

In [71], the original ACO is employed for optimizing a CNN. In [67], in addition to the original PSO, the original ABC, Bacterial Foraging Optimization (BFO), firefly algorithm, Firework Optimization Algorithm (FOA), Harmony Search Algorithm (HSA), and Gravitational Search Algorithm (GSA) have been used for optimizing CNNs. BFO is based on the behaviour of certain bacteria as they move closer to nutrients by detecting chemical gradients. FOA is based on the explosion manner of fireworks. HSA is based on how Jazz musicians adapt the sound to discover the correct harmony of a song. GSA is inspired by the laws of gravitation and mass attraction.

Table 3 shows a comparison of the different swarm intelligence algorithms employed for training convolutional neural networks in terms of the population size and maximum number of iterations used and the final RMSE of the trained neural network.

From Table 3, it can be seen three works have employed the original versions of swarm intelligence algorithms for optimizing CNNS, and three have employed hybrid algorithms. The accuracy of the CNN is most widely used to evaluate its performance, followed by the error rate of the CNN.

VI. OPTIMIZING NEURAL NETWORKS FOR DIFFERENT SUPERVISED LEARNING TASKS

Artificial neural networks can be used for either supervised, unsupervised or reinforcement learning tasks. In supervised learning, labelled datasets consisting of inputs and the correct outputs are utilized for training the neural network. Conversely, in unsupervised learning, unlabelled datasets are utilized for training the neural networks, which try to discover patterns from the given data. In reinforcement learning, the neural network is trained to take a sequence of actions which leads to the highest cumulative reward by considering several outputs for each input data.

However, we have found that the works, which have employed swarm intelligence algorithms for optimizing neural networks, have only used the ANNs for supervised learning. There are two types of supervised learning tasks, namely classification and regression. In classification, the output is a predicted discrete labelled class while in regression the output is a predicted continuous value.

The taxonomy in Fig. 7 shows the works in which swarm intelligence algorithms have been used for optimizing ANNs by means of the type of supervised learning task that it has

TABLE 2. Comparison of Swarm Intelligence Algorithms used to train Recurrent Neural Networks.

Paper	Algorithm	Population size	Num of iterations	Range of RMSE of ANN
[55]	Original PSO	30		96.7 - 97.5 (accuracy)
[61]	Original PSO, FPA	25-40	50-500	83 - 86, 84 - 88 (accuracy)
[54]	Hybrid PSO	30	100	2.170E-09 - 0.04243
[69]	BAPSO	4-6	100,000	0.05 - 0.1253
[59]	Modified PSO			102.90 - 441.94
[64]	Hybrid PSO	30	300	0.0237 - 0.0359
[65]	PSO-CS		1000	1.48E-03 - 0.0173
[53]	Original CFSO	2-7	300	0.05412 - 2.31560
[58]	Enhanced Ternary Bees	5	100	81.0 - 99.0 (accuracy)
[60]	Original ACO	5,10,25	100	94.90 (accuracy)
[62]	Original GWO	8, 15	20	145 - 162 (perplexity)
[56]	Original WOA			0.31 - 0.32
[63]	Original ABC			25.95 - 1.64E02
[57]	Hybrid Cuckoo Search		1000	0.0224 - 0.197

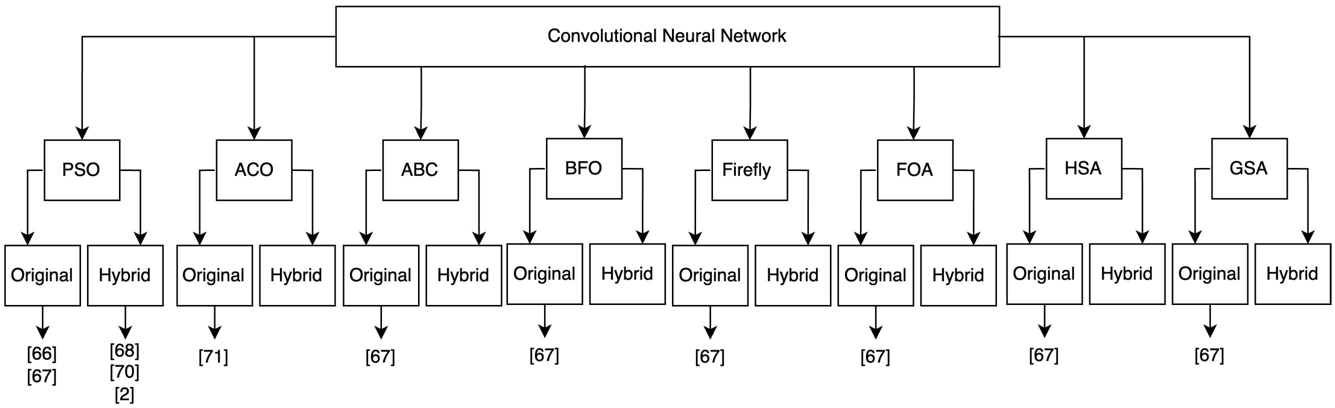

FIGURE 6. Taxonomy classifying the CNNs trained by swarm intelligence algorithms based on the algorithm used.

TABLE 3. Comparison of Swarm Intelligence Algorithms used to train Convolutional Neural Networks.

Paper	Algorithm	Population size	Num of iterations	Range of RMSE of ANN
[66]	Original PSO	6	5	71.15 - 95.16 (accuracy)
[67]	Original PSO, ABC, BFO, Firefly, FOA, HSA, GSA	300	300	92.27 - 93.71, 91.63 - 93.02, 90.96 - 92.51, 89.10 - 91.56, 90.27 - 91.06, 92.22 - 93.11, 92.34 - 93.43 (accuracy)
[68]	OLPSO	5	6	97.7 - 98.2 (accuracy)
[70]	IPSO	50	1000	98.85 (accuracy)
[72]	Modified PSO			0.18 - 16.38 (error rate)
[71]	Original ACO	1-16	50	0.39 - 12.70 (error rate)

been applied to, that is either classification or regression, and the main type of neural network used.

In [20], [21], [24], [25], [27], [37], [39], [41], [43], [46], [47], [48], [49], and [50] the feedforward neural networks trained by swarm intelligence algorithms have been applied to classification problems. In [20], the ANN has been used for some benchmark classification problems from the UCI machine learning repository, namely the 3-bit XOR, iris, balloon, breast cancer, glass, and heart classification problems. In [21], [24], and [25], the ANNs have been used for MRI brain image classification, detection of lung lesions from computed tomography images, and sonar targets classification respectively. In [27] and [47], the ANNs have been applied to benchmark classification problems; in [27], the XOR, balloon, and heart benchmark classification problems are used, and in [47], the 3-bit parity, XOR6, XOR9, XOR13, and 4-bit encoder/decoder classification problems are used. In [37], the ANN is used for the classification of the type of dolphins based on their whistles' signals. In [48], the ANN is employed for the gully erosion susceptibility of a

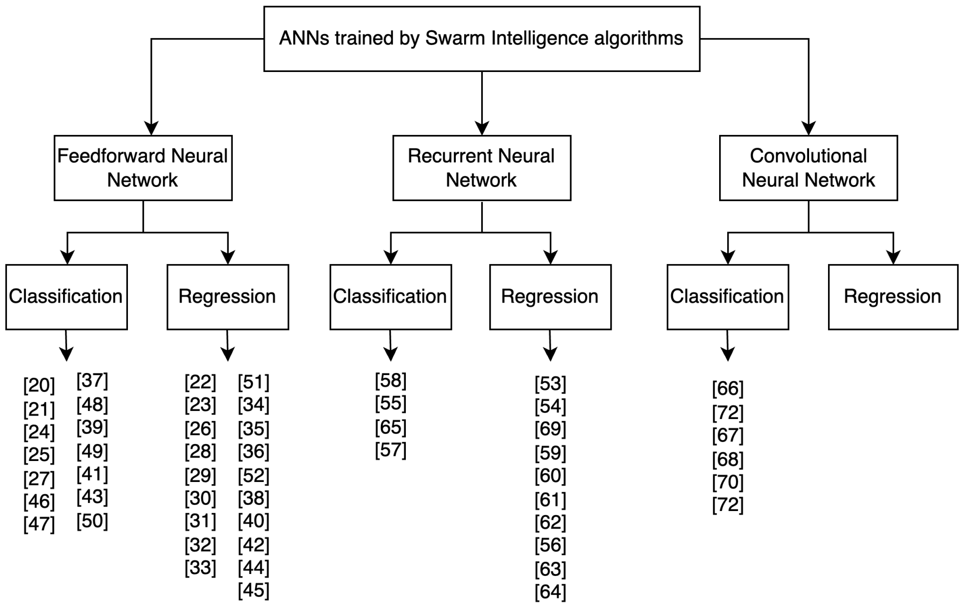

FIGURE 7. Taxonomy classifying the ANNs trained by swarm intelligence algorithms based on the type of the applied supervised learning task.

region. In [39], [41], and [49], the ANNs are employed for the classification of activities in an intrusion detection system, for the detection of phishing websites, and for the classification of fruits respectively. In [43], the ANN is applied to benchmark classification problems, namely the balloon, 3-bit XOR, hepatitis, iris, breast cancer, wine, monk, heart, blood, and Australian classification problems. In [50], the ANN is employed for digital modulation recognition in wireless communication systems. In [46], the ANN is applied to the glass, iris, balloon, and breast cancer benchmark classification problems from the UCI machine learning repository.

In [22], [23], [26], [28], [29], [30], [31], [32], [33], [34], [35], [36], [38], [40], [42], [44], [45], [51], and [52], the feedforward neural networks trained by swarm intelligence algorithms have been used for regression problems. In [22], and [23], the ANNs are employed for fault prediction in object-oriented systems and behaviour prediction of channel connectors embedded in normal and high-strength concrete respectively. In [26], [28], and [29], the ANNs are employed to predict the bearing capacity of footings over two-layer foundation soils, the heating load of residential buildings, and the soil compression coefficient in a housing construction project respectively. In [30], [36], and [40], the ANNs are used for stock market prediction. In [31], [32], [33], and [51], the ANNs are employed for predicting the landslide susceptibility of a region, the concentration of heavy metals in groundwater resources, the spring fatigue life of vehicles, and the bearing ratio of soils for the subgrade layers of railway tracks respectively. In [34], it is used for hydrological forecasting and in [52] for solar power generation prediction. In [35], it is used for predicting the damage level of rubble mound breakwater of the tandem breakwater. In [38], the ANN is applied to the following benchmark regression problems from the UCI machine learning repository: Airfoil Self-Noise, Concrete Slump Test (Slump), Istanbul Stock Exchange, Computer Hardware, Auto MPG, Forest Fires, Breast Cancer Wisconsin (Prognostic), Combined Cycle Power Plant, Energy efficiency (Heating Load), Challenger USA Space Shuttle O-Ring, Fertility, Concrete Compressive Strength, Concrete Slump Test (Strength), Concrete Slump Test (Flow), and Energy efficiency (Cooling Load). In [42], the ANN is used for predicting the change in solar irradiance. In [44], it is used to predict the carbonation depth for recycled aggregate concrete and in [45] for time-series forecasting.

In [55], [57], [58], and [65], the recurrent neural networks trained by swarm intelligence algorithms have been used for classification problems. In [58], the RNN is used for sentiment classification, and in [55] it is used for voltage instability prediction. In [65], the RNN is employed for the iris, glass, and diabetes benchmark classification problems form the UCI machine learning repository, and in [57], it is employed for the iris, Wisconsin breast cancer, thyroid, diabetes, glass, Australian credit card approval, and 7-bit parity benchmark classification problems.

In [53], [54], [56], [59], [60], [61], [62], [63], [64], and [69], the recurrent neural networks trained by swarm intelligence algorithms have been employed for regression problems. In [53], the RNN is used for a benchmark regression problem called the double-tank model. In [54], the RNN is employed for impedance identification in power electronic systems. In [69], it is used for the reverse engineering of gene regulatory networks from temporal genetic expression. In [56], [59], [60], and [61], the RNNs are used for

TABLE 4. Comparison of Applications of ANNs trained by Swarm Intelligence Algorithms.

Paper	Application	ANN used	ANN RMSE
[20]	3-bit XOR, iris, balloon, breast cancer, glass, heart benchmark classification problems	MLP	1.292E-22 - 2.356E-02
[21]	MRI brain image classification	MLP	85.0 (accuracy)
[24]	Lung lesion detection	MLP	97.2 - 99.5 (accuracy)
[25]	Sonar targets classification	MLP	92.2 - 96.7 (accuracy)
[27]	XOR, balloon, heart benchmark classification problems	MLP	2.341E-16 - 0.3382
[47]	XOR6, XOR9, XOR13, 3-bit parity and 4-bit encoder/decoder classification problems	Deep FNN	4.0130E-06 - 0.002317
[37]	Dolphin whistles' classification	MLP	0.01414 - 0.5226
[48]	Gully erosion susceptibility	Deep FNN	0.89 (AUC)
[39]	Intrusion detection system activities' classification	MLP	98.0 (accuracy)
[49]	Phishing websites detection	Deep FNN	96.5 (accuracy)
[41]	Fruit classification	MLP	0.1584
[43]	Benchmark classification problems	MLP	2.381E-10 - 0.5617
[50]	Digital modulation recognition	Deep FNN	95.0 (accuracy)
[46]	Iris, balloon, glass, breast cancer benchmark classification problem	MLP	0.0991 - 0.78259
[22]	Fault prediction in object-oriented systems	MLP	20.7 - 96.2 (accuracy)
[23]	Channel connectors behaviour prediction	MLP	14.4 - 2.49
[26]	Footings' bearing capacity prediction	MLP	0.3422, 0.3674
[28]	Heating load prediction	MLP	2.2899 - 2.4459
[29]	Soil compression coefficient prediction	MLP	0.0267
[30]	Stock market prediction	MLP	1.195E-05 - 0.225 (Mean Absolute Error)
[31]	Landslide susceptibility prediction	MLP	0.4444
[32]	Heavy metals concentration prediction	MLP	0.082 - 0.273
[33]	Spring fatigue life prediction	MLP	0.6391
[51]	Soil bearing ration prediction	ELM	0.6391
[34]	Hydrological forecasting	MLP	635.5932 - 1501.1942
[35]	Rubble mound breakwater damage level prediction	MLP	0.515 - 1.264
[36]	Stock market prediction	MLP	1.59E-02 - 7.78E-02
[52]	Solar power generation prediction	RBF	5.693E-03 - 8.149E-03
[38]	Benchmark regression problem	MLP	0.09798 - 0.4472
[40]	Stock market prediction	MLP	0.03162
[42]	Solar irradiance prediction	MLP	0.1911 - 0.2032
[44]	Carbonation depth prediction	MLP	2.281 - 2.456
[45]	Time-series forecasting	MLP	1.0 - 3.0 (average error)
[58]	Sentiment classification	Deep RNN	81.0 - 99.0 (accuracy)
[55]	Voltage instability prediction	Elman RNN	96.7 - 97.5 (accuracy)
[65]	Iris, glass, diabetes benchmark classification problem	Jordan RNN	1.48E-03 - 0.0173
[57]	Benchmark classification problem	Elman RNN	0.0224 - 0.197
[53]	Double-tank model benchmark regression problem	NARX	0.05412 - 2.31560
[54]	Impedance identification	Elman RNN	2.170E-09 - 0.04243
[69]	Gene regulatory networks' reverse engineering	RNN	0.05 - 0.1253
[59]	Time series forecasting	Deep RNN	102.90 - 441.94
[60]	Airborne pollution prediction	LSTM	94.90 (accuracy)
[61]	Stock market forecasting	LSTM	83 - 88 (accuracy)
[62]	Language modeling	LSTM	145 - 162 (perplexity)
[56]	Agricultural crops yield forecasting	ELman RNN	0.31 - 0.32
[63]	Traffic volume prediction	LSTM	25.95 - 1.64E02
[64]	Shear wave velocity prediction	LSTM	0.0237 - 0.0359
[66]	CIFAR-10, CIFAR100, ILSVRC- 2012 classification dataset	VGGNet, ResNet, GoogLeNet	71.15 - 95.16 (accuracy
[71]	MNIST, Fashion-MNIST, CIFAR-10 classification dataset	CNN	0.39 - 12.70 (error rate)
[67]	Pulmonary cancerous nodules detection	U-Net	89.10 - 93.43 (accuracy)
[68]	Plant disease classification	VGGNet	97.7 - 98.2 (accuracy)
[70]	Multiple sclerosis lesion segmentation	CNN	98.85 (accuracy)
[72]	Image classification	CNN	0.18 - 16.38 (error rate)

time series forecasting, airborne pollution prediction, stock market forecasting, and yield forecasting of agricultural crops respectively. In [62], it is used for language modeling using the Penn Tree Bank dataset. In [63] and [64], the RNNs are employed for traffic volume prediction, and shear wave velocity prediction respectively.

In [66], [67], [68], [70], [71], and [72], the convolutional neural networks trained by swarm intelligence algorithms have been applied to classification problems. There is no CNN trained by a swarm intelligence algorithm which has been employed for a regression problem. In [66] and [71], the CNNs are applied to classification datasets. In [66], the CIFAR-10, CIFAR100, and ILSVRC-2012 datasets are used and in [71], the MNIST, Fashion-MNIST, and CIFAR-10 datasets are used. In [67], the CNN is used in the detection of pulmonary cancerous nodules in computed tomography scans and in [68], it is used for plant disease classification using digital images. In [70], it is applied to multiple sclerosis lesion segmentation in brain MRI images. In [72], the CNN is used for image classification.

Table 4 presents a comparison of the ANNs optimized by swarm intelligence algorithms in terms of the application of the ANN, the type of ANN used, and the performance of the ANN, that is its final RMSE. In the case where the RMSE is not used for evaluating the performance of the neural network, the performance metric used is mentioned within parentheses.

From Table 4, it can be seen that the feedforward neural networks, especially MLPS, optimized by swarm intelligence algorithms have been widely employed for different classification and regression problems. The RNNs optimized by swarm intelligence algorithms have been predominantly utilized for regression problems, and a few classification problems. However, the CNNs optimized by swarm intelligence algorithms have only been utilized for classification problems and they have not been used for any regression problem.

VII. CONCLUSION

Artificial neural networks are becoming prevalent in numerous areas as they are beneficial for various applications. Prior to using an ANN, it is important to determine certain characteristics such as the type of the ANN, and its architecture which comprises mainly of the number of layers and the number of neurons in each layer. Moreover, the ANN needs to be trained whereby the connection weights and biases of the ANN are adjusted so as to find the optimal ones. The main type of ANN to be used is usually chosen based on the nature of the application. However, the architecture is usually determined on a trial-and-error basis, and the training is done using gradient-descent algorithms.

Recently, it has been found that swarm intelligence algorithms are useful and advantageous for either determining the architecture of neural networks or the training of neural networks. This is because instead of determining the architecture of the network by trial and error, a more intelligent method is used. Moreover, for the training of neural networks, the conventional gradient-descent algorithms are often local search algorithms and are incapable of finding the most optimal connection weights and biases for the network. Hence, swarm intelligence algorithms are better options for determining the optimal weights and biases of neural networks owing to both their exploration and exploitation capabilities. A number of swarm intelligence algorithms have been employed for optimizing different neural networks for various applications across a variety of areas. However, to the best of our knowledge, there is no comprehensive survey on the artificial neural networks which have been optimized using swarm intelligence algorithms.

In this paper, we present a review of the different types of ANNs which have been optimized by swarm intelligence algorithms, the way the ANNs have been optimized, the swarm intelligence algorithms used for the optimization of the network, and the applications of the ANNs trained by swarm intelligence algorithms. It has been found that among the three main types of ANNs, feedforward neural networks have been optimized the most using swarm intelligence algorithms while convolutional neural network have been optimized the least using swarm intelligence algorithms. Among the feedforward neural networks, multi-layer perceptrons have been mostly optimized by means of swarm intelligence algorithms. Moreover, it can be seen that for FNNs and RNNs, swarm intelligence algorithms have been most widely employed for optimizing their weights and biases during the training. However, for CNNs, swarm intelligence algorithms have been mostly employed for selecting the optimal structure of the network. Among the different swarm intelligence algorithms, the particle swarm optimization and its hybrids have been mostly for optimizing ANNs. In terms of applications, the optimized ANNs have only been employed for supervised learning tasks, that is either classification or regression. The optimized FNNs and RNNs have been used for different classification and regression applications. However, the optimized CNNs have only been used for classification applications.

For future work, the performance of ANNs optimized using different swarm intelligence algorithms can be compared by using the ANNs for the same application, more novel and higher-performing swarm intelligence algorithms can be utilized for optimizing different types of neural networks, and the optimized ANNs can be used for other applications.

REFERENCES

[1] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. Cambridge, MA, USA: MIT Press, 2018.

[2] N. Shahid, T. Rappon, and W. Berta, "Applications of artificial neural networks in health care organizational decision-making: A scoping review," *PLoS ONE*, vol. 14, no. 2, Feb. 2019, Art. no. e0212356.

[3] J. Runge and R. Zmeureanu, "Forecasting energy use in buildings using artificial neural networks: A review," *Energies*, vol. 12, no. 17, p. 3254, Aug. 2019.

[4] A. Escamilla-García, G. M. Soto-Zarazúa, M. Toledano-Ayala, E. Rivas-Araiza, and A. Gastélum-Barrios, "Applications of artificial neural networks in greenhouse technology and overview for smart agriculture development," *Appl. Sci.*, vol. 10, no. 11, p. 3835, 2020.

[5] B. Li, C. Delpha, D. Diallo, and A. Migan-Dubois, "Application of artificial neural networks to photovoltaic fault detection and diagnosis: A review," *Renew. Sustain. Energy Rev.*, vol. 138, Mar. 2021, Art. no. 110512.

[6] A. R. Pazikadin, D. Rifai, K. Ali, M. Z. Malik, A. N. Abdalla, and M. A. Faraj, "Solar irradiance measurement instrumentation and power solar generation forecasting based on artificial neural networks (ANN): A review of five years research trend," *Sci. Total Environ.*, vol. 715, May 2020, Art. no. 136848.

[7] V. K. Ojha, A. Abraham, and V. Snášel, "Metaheuristic design of feedforward neural networks: A review of two decades of research," *Eng. Appl. Artif. Intell.*, vol. 60, pp. 97–116, Apr. 2017.

[8] M. J. Reddy and D. N. Kumar, "Evolutionary algorithms, swarm intelligence methods, and their applications in water resources engineering: A state-of-the-art review," *H2Open J.*, vol. 3, no. 1, pp. 135–188, Jan. 2020.

[9] Q.-V. Pham, D. C. Nguyen, S. Mirjalili, D. T. Hoang, D. N. Nguyen, P. N. Pathirana, and W.-J. Hwang, "Swarm intelligence for next-generation wireless networks: Recent advances and applications," 2020, *arXiv:2007.15221*.

[10] H. R. Boveiri, R. Khayami, M. Elhoseny, and M. Gunasekaran, "An efficient swarm-intelligence approach for task scheduling in cloud-based Internet of Things applications," *J. Ambient Intell. Hum. Comput.*, vol. 10, no. 9, pp. 3469–3479, 2019.

[11] B. A. S. Emambocus, M. B. Jasser, and A. Amphawan, "Towards an optimized channel estimation in optical spatial multiplexing systems via swarm intelligence algorithms," in *Proc. IEEE 13th Control Syst. Graduate Res. Colloq. (ICSGRC)*, Jul. 2022, pp. 77–82.

[12] L. Peška, T. M. Tashu, and T. Horváth, "Swarm intelligence techniques in recommender systems—A review of recent research," *Swarm Evol. Comput.*, vol. 48, pp. 201–219, Aug. 2019.

[13] S. Mishra, R. Sagban, A. Yakoob, and N. Gandhi, "Swarm intelligence in anomaly detection systems: An overview," *Int. J. Comput. Appl.*, vol. 43, no. 2, pp. 109–118, Feb. 2021.

[14] G. Soni, V. Jain, F. T. S. Chan, B. Niu, and S. Prakash, "Swarm intelligence approaches in supply chain management: Potentials, challenges and future research directions," *Supply Chain Manag., Int. J.*, vol. 24, no. 1, pp. 107–123, Jan. 2019.

[15] E. Figueiredo, M. Macedo, H. V. Siqueira, C. J. Santana, A. Gokhale, and C. J. A. Bastos-Filho, "Swarm intelligence for clustering—A systematic review with new perspectives on data mining," *Eng. Appl. Artif. Intell.*, vol. 82, pp. 313–329, Jun. 2019.

[16] L. Brezočnik, I. Fister, and V. Podgorelec, "Swarm intelligence algorithms for feature selection: A review," *Appl. Sci.*, vol. 8, no. 9, p. 1521, Sep. 2018.

[17] B. A. S. Emambocus, M. B. Jasser, M. Hamzah, A. Mustapha, and A. Amphawan, "An enhanced swap sequence-based particle swarm optimization algorithm to solve TSP," *IEEE Access*, vol. 9, pp. 164820–164836, 2021.

[18] B. A. S. Emambocus, M. B. Jasser, A. Amphawan, and A. W. Mohamed, "An optimized discrete dragonfly algorithm tackling the low exploitation problem for solving TSP," *Mathematics*, vol. 10, no. 19, p. 3647, Oct. 2022.

[19] B. A. S. Emambocus, M. B. Jasser, A. Mustapha, and A. Amphawan, "Dragonfly algorithm and its hybrids: A survey on performance, objectives and applications," *Sensors*, vol. 21, no. 22, p. 7542, 2021.

[20] W. A. H. M. Ghanem and A. Jantan, "A cognitively inspired hybridization of artificial bee colony and dragonfly algorithms for training multi-layer perceptrons," *Cogn. Comput.*, vol. 10, no. 6, pp. 1096–1134, Dec. 2018.

[21] A. T. Abdulameer, "An improvement of MRI brain images classification using dragonfly algorithm as trainer of artificial neural network," *Ibn AL-Haitham J. Pure Appl. Sci.*, vol. 31, no. 1, pp. 268–276, May 2018.

[22] N. Kayarvizhy, S. Kanmani, and V. Uthariaraj, "Ann models optimized using swarm intelligence algorithms," *WSEAS Trans. Comput.*, vol. 13, pp. 501–519, Jan. 2014.

[23] M. Shariati, M. S. Mafipour, P. Mehrabi, A. Bahadori, Y. Zandi, M. N. A. Salih, H. Nguyen, J. Dou, X. Song, and S. Poi-Ngian, "Application of a hybrid artificial neural network-particle swarm optimization (ANN-PSO) model in behavior prediction of channel shear connectors embedded in normal and high-strength concrete," *Appl. Sci.*, vol. 9, no. 24, p. 5534, Dec. 2019.

[24] E. E. Nithila and S. S. Kumar, "Automatic detection of solitary pulmonary nodules using swarm intelligence optimized neural networks on CT images," *Eng. Sci. Technol., Int. J.*, vol. 20, no. 3, pp. 1192–1202, 2017.

[25] M. Khishe and A. Safari, "Classification of sonar targets using an MLP neural network trained by dragonfly algorithm," *Wireless Pers. Commun.*, vol. 108, no. 4, pp. 2241–2260, Oct. 2019.

[26] H. Moayedi, M. M. Abdullahi, H. Nguyen, and A. S. A. Rashid, "Comparison of dragonfly algorithm and Harris hawks optimization evolutionary data mining techniques for the assessment of bearing capacity of footings over two-layer foundation soils," *Eng. Comput.*, vol. 37, no. 1, pp. 437–447, Jan. 2021.

[27] J. Xu and F. Yan, "Hybrid Nelder–Mead algorithm and dragonfly algorithm for function optimization and the training of a multilayer perceptron," *Arabian J. Sci. Eng.*, vol. 44, no. 4, pp. 3473–3487, Apr. 2019.

[28] H. Moayedi, H. Nguyen, and L. Foong, "Nonlinear evolutionary swarm intelligence of grasshopper optimization algorithm and gray wolf optimization for weight adjustment of neural network," *Eng. Comput.*, vol. 37, pp. 1265–1275, Apr. 2021.

[29] D. Tien Bui, V.-H. Nhu, and N.-D. Hoang, "Prediction of soil compression coefficient for urban housing project using novel integration machine learning approach of swarm intelligence and multi-layer perceptron neural network," *Adv. Eng. Informat.*, vol. 38, pp. 593–604, Oct. 2018.

[30] R. Jamous, H. ALRahhal, and M. El-Darieby, "A new ANN-particle swarm optimization with center of gravity (ANN-PSOCoG) prediction model for the stock market under the effect of COVID-19," *Sci. Program.*, vol. 2021, pp. 1–17, Apr. 2021.

[31] D. T. Bui, H. Moayedi, B. Kalantar, A. Osouli, B. Pradhan, H. Nguyen, and A. S. A. Rashid, "A novel swarm intelligence—Harris hawks optimization for spatial assessment of landslide susceptibility," *Sensors*, vol. 19, no. 16, p. 3590, 2019.

[32] M. Alizamir and S. Sobhanardakani, "An artificial neural network–particle swarm optimization (ANN- PSO) approach to predict heavy metals contamination in groundwater resources," *Jundishapur J. Health Sci.*, vol. 10, no. 2, Apr. 2018, Art. no. e67544.

[33] Y. S. Kong, S. Abdullah, D. Schramm, M. Z. Omar, and S. M. Haris, "Design of artificial neural network using particle swarm optimisation for automotive spring durability," *J. Mech. Sci. Technol.*, vol. 33, no. 11, pp. 5137–5145, Nov. 2019.

[34] W.-J. Niu, Z.-K. Feng, B.-F. Feng, Y.-S. Xu, and Y.-W. Min, "Parallel computing and swarm intelligence based artificial intelligence model for multi-step-ahead hydrological time series prediction," *Sustain. Cities Soc.*, vol. 66, Mar. 2021, Art. no. 102686.

[35] G. Kuntoji, S. M. Rao, and E. N. B. Reddy, "Prediction of damage level of inner conventional rubble mound breakwater of tandem breakwater using swarm intelligence-based neural network (PSO-ANN) approach," in *Soft Computing for Problem Solving*. Springer, 2019, pp. 441–453.

[36] G. Kumar, U. P. Singh, and S. Jain, "Swarm intelligence based hybrid neural network approach for stock price forecasting," *Comput. Econ.*, vol. 60, no. 3, pp. 991–1039, Oct. 2022.

[37] J. Wu, M. Khishe, M. Mohammadi, S. H. T. Karim, and M. Shams, "Acoustic detection and recognition of dolphins using swarm intelligence neural networks," *Appl. Ocean Res.*, vol. 115, Oct. 2021, Art. no. 102837.

[38] Y. Wang, H. Liu, Z. Yu, and L. Tu, "An improved artificial neural network based on human-behaviour particle swarm optimization and cellular automata," *Expert Syst. Appl.*, vol. 140, Feb. 2020, Art. no. 112862.

[39] S. Norwahidayah, A. A. Noraniah, N. Farahah, A. Amirah, N. Liyana, and N. Suhana, "Performances of artificial neural network (ANN) and particle swarm optimization (PSO) using KDD cup'99 dataset in intrusion detection system (IDS)," *J. Phys., Conf. Ser.*, vol. 1874, no. 1, May 2021, Art. no. 012061.

[40] F. Ghashami, K. Kamyar, and S. A. Riazi, "Prediction of stock market index using a hybrid technique of artificial neural networks and particle swarm optimization," *Appl. Econ. Finance*, vol. 8, no. 3, p. 1, Apr. 2021.

[41] Y. Zhang, P. Phillips, S. Wang, G. Ji, J. Yang, and J. Wu, "Fruit classification by biogeography-based optimization and feedforward neural network," *Expert Syst.*, vol. 33, no. 3, pp. 239–253, 2016.

[42] A. Aljanad, N. M. L. Tan, V. G. Agelidis, and H. Shareef, "Neural network approach for global solar irradiance prediction at extremely short-time-intervals using particle swarm optimization algorithm," *Energies*, vol. 14, no. 4, p. 1213, Feb. 2021.

[43] D. Bairathi and D. Gopalani, "Numerical optimization and feed-forward neural networks training using an improved optimization algorithm: Multiple leader salp swarm algorithm," *Evol. Intell.*, vol. 14, no. 3, pp. 1233–1249, Sep. 2021.

[44] K. Liu, M. S. Alam, J. Zhu, J. Zheng, and L. Chi, "Prediction of carbonation depth for recycled aggregate concrete using ANN hybridized with swarm intelligence algorithms," *Construct. Building Mater.*, vol. 301, Sep. 2021, Art. no. 124382.

[45] S. A. Abdulkarim and A. P. Engelbrecht, "Time series forecasting with feedforward neural networks trained using particle swarm optimizers for dynamic environments," *Neural Comput. Appl.*, vol. 33, no. 7, pp. 2667–2683, Apr. 2021.

[46] B. A. S. Emambocus, M. B. Jasser, and A. Amphawan, "An optimized continuous dragonfly algorithm using Hill climbing local search to tackle the low exploitation problem," *IEEE Access*, vol. 10, pp. 95030–95045, 2022.

[47] F. Xu, C.-M. Pun, H. Li, Y. Zhang, Y. Song, and H. Gao, "Training feed-forward artificial neural networks with a modified artificial bee colony algorithm," *Neurocomputing*, vol. 416, pp. 69–84, Nov. 2020.

[48] S. S. Band, S. Janizadeh, S. Chandra Pal, A. Saha, R. Chakrabortty, M. Shokri, and A. Mosavi, "Novel ensemble approach of deep learning neural network (DLNN) model and particle swarm optimization (PSO) algorithm for prediction of gully erosion susceptibility," *Sensors*, vol. 20, no. 19, p. 5609, Sep. 2020.

[49] G. Vrbančič, I. Fister, and V. Podgorelec, "Swarm intelligence approaches for parameter setting of deep learning neural network: Case study on phishing websites classification," in *Proc. 8th Int. Conf. Web Intell., Mining Semantics*, Jun. 2018, pp. 1–8.

[50] W. Shi, D. Liu, X. Cheng, Y. Li, and Y. Zhao, "Particle swarm optimization-based deep neural network for digital modulation recognition," *IEEE Access*, vol. 7, pp. 104591–104600, 2019.
[51] A. Bardhan, P. Samui, K. Ghosh, A. H. Gandomi, and S. Bhattacharyya, "ELM-based adaptive neuro swarm intelligence techniques for predicting the California bearing ratio of soils in soaked conditions," *Appl. Soft Comput.*, vol. 110, Oct. 2021, Art. no. 107595.
[52] Z. Yang, M. Mourshed, K. Liu, X. Xu, and S. Feng, "A novel competitive swarm optimized RBF neural network model for short-term solar power generation forecasting," *Neurocomputing*, vol. 397, pp. 415–421, Jul. 2020.
[53] G. M. Lozito and A. Salvini, "Swarm intelligence based approach for efficient training of regressive neural networks," *Neural Comput. Appl.*, vol. 32, no. 14, pp. 10693–10704, Jul. 2020.
[54] P. Xiao, G. K. Venayagamoorthy, and K. A. Corzine, "Combined training of recurrent neural networks with particle swarm optimization and backpropagation algorithms for impedance identification," in *Proc. IEEE Swarm Intell. Symp.*, Apr. 2007, pp. 9–15.
[55] A. M. Ibrahim and N. H. El-Amary, "Particle swarm optimization trained recurrent neural network for voltage instability prediction," *J. Electr. Syst. Inf. Technol.*, vol. 5, no. 2, pp. 216–228, 2018.
[56] P. Murali, R. Revathy, S. Balamurali, and A. S. Tayade, "Integration of RNN with GARCH refined by whale optimization algorithm for yield forecasting: A hybrid machine learning approach," *J. Ambient Intell. Hum. Comput.*, vol. 11, pp. 1–13, Apr. 2020.
[57] N. M. Nawi, A. Khan, M. Z. Rehman, H. Chiroma, and T. Herawan, "Weight optimization in recurrent neural networks with hybrid metaheuristic cuckoo search techniques for data classification," *Math. Problems Eng.*, vol. 2015, pp. 1–12, Nov. 2015.
[58] S. Zeybek, D. T. Pham, E. Koç, and A. Seçer, "An improved bees algorithm for training deep recurrent networks for sentiment classification," *Symmetry*, vol. 13, no. 8, p. 1347, Jul. 2021.
[59] E. Bas, E. Egrioglu, and E. Kolemen, "Training simple recurrent deep artificial neural network for forecasting using particle swarm optimization," *Granular Comput.*, vol. 7, no. 2, pp. 411–420, Apr. 2022.
[60] G. J. Kuri-Monge, M. A. Aceves-Fernandez, J. A. Ramírez-Montañez, and J. C. Pedraza-Ortega, "Capability of a recurrent deep neural network optimized by swarm intelligence techniques to predict exceedances of airborne pollution (PMx) in largely populated areas," in *Proc. Int. Conf. Inf. Technol. (ICIT)*, Jul. 2021, pp. 61–68.
[61] K. Kumar and T. U. Haider, "Enhanced prediction of intra-day stock market using metaheuristic optimization on RNN–LSTM network," *New Gener. Comput.*, vol. 39, no. 1, pp. 231–272, Apr. 2021.
[62] B. Z. Aufa, S. Suyanto, and A. Arifianto, "Hyperparameter setting of LSTM-based language model using grey wolf optimizer," in *Proc. Int. Conf. Data Sci. Its Appl. (ICoDSA)*, Aug. 2020, pp. 1–5.
[63] A. Bosire, "Recurrent neural network training using ABC algorithm for traffic volume prediction," *Informatica*, vol. 43, no. 4, pp. 551–559, Dec. 2019.
[64] J. Wang, J. Cao, and S. Yuan, "Shear wave velocity prediction based on adaptive particle swarm optimization optimized recurrent neural network," *J. Petroleum Sci. Eng.*, vol. 194, Nov. 2020, Art. no. 107466.
[65] R. Talal and Z. Tariq, "Training recurrent neural networks by a hybrid PSO-cuckoo search algorithm for problems optimization," *Int. J. Comput. Appl.*, vol. 159, no. 3, pp. 32–38, Feb. 2017.
[66] J. Chang, Y. Lu, P. Xue, Y. Xu, and Z. Wei, "Automatic channel pruning via clustering and swarm intelligence optimization for CNN," *Appl. Intell.*, vol. 52, pp. 17751–17771, Apr. 2022.
[67] C. A. de Pinho Pinheiro, N. Nedjah, and L. de Macedo Mourelle, "Detection and classification of pulmonary nodules using deep learning and swarm intelligence," *Multimedia Tools Appl.*, vol. 79, nos. 21–22, pp. 15437–15465, Jun. 2020.
[68] A. Darwish, D. Ezzat, and A. E. Hassanien, "An optimized model based on convolutional neural networks and orthogonal learning particle swarm optimization algorithm for plant diseases diagnosis," *Swarm Evol. Comput.*, vol. 52, Feb. 2020, Art. no. 100616.
[69] A. Khan, S. Mandal, R. K. Pal, and G. Saha, "Construction of gene regulatory networks using recurrent neural networks and swarm intelligence," *Scientifica*, vol. 2016, pp. 1–14, Jan. 2016.
[70] R. Krishna Priya and S. Chacko, "Improved particle swarm optimized deep convolutional neural network with super-pixel clustering for multiple sclerosis lesion segmentation in brain MRI imaging," *Int. J. Numer. Methods Biomed. Eng.*, vol. 37, no. 9, Sep. 2021, Art. no. e3506.
[71] E. Byla and W. Pang, "DeepSwarm: Optimising convolutional neural networks using swarm intelligence," in *Proc. UK Workshop Comput. Intell.* Cham, Switzerland: Springer, 2019, pp. 119–130.
[72] S. C. Nistor and G. Czibula, "IntelliSwAS: Optimizing deep neural network architectures using a particle swarm-based approach," *Expert Syst. Appl.*, vol. 187, Jan. 2022, Art. no. 115945.

BIBI AAMIRAH SHAFAA EMAMBOCUS received the B.Sc. degree in computer science from Sunway University, Malaysia, in 2020, where she is currently pursuing the M.Sc. degree. Her research interests include swarm intelligence, evolutionary algorithms, artificial intelligence, and machine learning.

MUHAMMED BASHEER JASSER (Member, IEEE) received the master's and Ph.D. degrees in software engineering from University Putra Malaysia (UPM). He is currently a Program Leader and a Senior Lecturer with Sunway University. Prior to that, he was a Research Assistant at UPM and a Lecturer at Aleppo University. He is also working on several fundamental and industrial research projects in the area of artificial intelligence and software engineering funded by companies and universities. His research interests include optimization algorithms, evolutionary computation, model-driven software engineering, formal specification, verification and theorem proving, artificial intelligence, and machine learning. He is also a member of several professional academic bodies, including the Institute of Electronics, Information and Communication Engineers (IEICE) and Formal Methods Europe. He was granted the Malaysian Technical Cooperation Program Scholarship (MTCP) from the Ministry of Higher Education (Malaysia) for his postgraduate studies.

ANGELA AMPHAWAN (Member, IEEE) received the Ph.D. degree in optical engineering from the University of Oxford, U.K. She is currently leading the Photonics Research Laboratory, School of Engineering and Technology, Sunway University. Prior to this, she was the Deputy Vice Chancellor of the University Malaysia of Computer Science and Engineering. Her research projects have been funded by the U.S. Department of States, German Government, Malaysian Ministry of Education, and Telekom Malaysia. She has given keynote addresses at several Fulbright and IEEE events. Her research interests include optical communications and sensing, covering optical fibers, free-space optics, radio-over free space optics, digital imaging, and the Internet of Things. She has won several best paper awards and exhibition medals. She was a recipient of the Fulbright Award at the Research Laboratory of Electronics, Massachusetts Institute of Technology. She serves on the Editorial Board of the *APL Photonics* journal with the American Institute of Physics and is on the National 5G Task Force. She was previously on the Editorial Board of *Transactions on Emerging Telecommunications Technologies* (Wiley) and the Publicity Co-Chair of the IEEE Wireless Communications and Networking Conference.