2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

An improved neurogenetic model for recognition of 3D kinetic data of human extracted from the Vicon Robot system

Ivan V. Stepanyan*[1,2], Safa A. Hameed[2]

[1]Mechanical Engineering Research Institute of the Russian Academy of Sciences (IMASH RAN), Moscow, the Russian Federation.
[2]Department of Mechanics and Control Processes, Academy of Engineering, Peoples' Friendship University of Russia (RUDN University), Moscow, Russian Federation
*Corresponding Author.

Received 17/05/2023, Revised 25/08/2023, Accepted 27/08/2023, Published 05/12/2023

Abstract

These days, it is crucial to discern between different types of human behavior, and artificial intelligence techniques play a big part in that. The characteristics of the feedforward artificial neural network (FANN) algorithm and the genetic algorithm have been combined to create an important working mechanism that aids in this field. The proposed system can be used for essential tasks in life, such as analysis, automation, control, recognition, and other tasks. Crossover and mutation are the two primary mechanisms used by the genetic algorithm in the proposed system to replace the back propagation process in ANN. While the feedforward artificial neural network technique is focused on input processing, this should be based on the process of breaking the feedforward artificial neural network algorithm. Additionally, the result is computed from each ANN during the breaking up process, which is based on the breaking up of the artificial neural network algorithm into multiple ANNs based on the number of ANN layers, and therefore, each layer in the original artificial neural network algorithm is assessed. The best layers are chosen for the crossover phase after the breakage process, while the other layers go through the mutation process. The output of this generation is then determined by combining the artificial neural networks into a single ANN; the outcome is then checked to see if the process needs to create a new generation. The system performed well and produced accurate findings when it was used with data taken from the Vicon Robot system, which was primarily designed to record human behaviors based on three coordinates and classify them as either normal or aggressive.

Keywords: Breaking-up process, Combining process, Crossover, Feedforward ANN, Mutation, Neuro-Genetic model, Optimization, Recognition, Vicon Robot, 3D data.

Introduction

To tackle complicated problems in our lives, it needs a technique that includes analytical components and is capable of processing and providing potential solutions. In our suggested model, it proposes using an artificial neural network (ANN). The artificial neural network is the cornerstone of artificial intelligence and can solve a wide range of complicated issues that are challenging for people or statistical techniques to handle[1]. Deep learning neural networks have been extensively utilized for classification, detection, and identification[2, 3]. which is a strong tool for data forecasting in a range of systems[4,5]. Additionally, it was suggested to use a genetic algorithm to improve the learning process[6, 7].

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

Choosing a genetic algorithm to enhance work performance is more potent. The efficiency of these evolutionary-based algorithms in resolving optimization issues is well recognized[8, 9]. The goal of our research is to create a system that combines the strengths of the two algorithms, which may then be employed in many practical parts of our lives, such as control, diagnosis, and analysis[10, 11]. The proposed method was utilized to distinguish between normal and aggressive human action by creating an efficient tool that combines two crucial algorithms. Reviewing the earlier studies, the suggested approach put forth in this study has already overcome several challenges by using sufficient data in the proposed model and enhancing the learning process in an artificial neural algorithm with a genetic algorithm by choosing the best layers and enhancing the other layers through crossover and mutation.

Several academics are interested in the process of identifying and recognizing the kinds of human movement because of its significance and requirement for such systems[12-14]. Academics are interested in the usage of wireless equipment networks because of the numerous applications for identifying human physical motions, including healthcare, rehabilitation, athletics, elder monitoring, sports, and human-robot interaction[15-17]. Because human activity is a work sequence that includes temporal information, the hybrid method of assessing activity is preferred[18-20].

Human Activity Recognition (HAR) is a technique used in health care to monitor the actions of patients and the elderly who have illnesses such as dementia, Parkinson's disease, and others. As a result, HAR can aid in spotting abnormalities in routine daily tasks, avoiding any negative consequences. Since HAR systems are utilized in healthcare and elder monitoring, compromising their accuracy is risky[21]. Safety is becoming increasingly critical, and video capture and storage equipment are always evolving to meet this demand. These devices, however, must be supplied with a system capable of consistently recognizing a wide range of anomalous conditions while minimizing human error[22].

The rapid development of artificial intelligence (AI) has paved the way for human activity recognition (HAR). It focuses on learning about a person's individual behavior (or movement) [23]. Several studies on the identification of various human activities have been conducted[24]. End-to-end deep learning models are increasingly being used in safety-critical human activity recognition (HAR) applications, such as healthcare monitoring and smart home control, to minimize developer load while improving prediction model performance and resilience[25]. As people age, they are more likely to have issues including cognitive decline, memory loss, chronic disease, vision, and hearing impairment, as well as many other disorders. They may also find it challenging to interact socially. The patients' capacity to carry out their everyday tasks is hampered by all these challenges, which also lower their quality of life and necessitate their asking for assistance from others[26].

Thus, the ability to recognize human actions is necessary, as is the need to establish natural contact between humans and robots. Among the systems capable of analyzing human action is the Vicon robot system, which serves as a support function as it can read and analyze human movement and allows interaction between humans and robots. As shown in Fig. 1, our proposed method was used to evaluate the 3D data that indicated normal and aggressive human movements acquired from the Vicon robot system[27-29].

The paper is structured in the following sections: The Related Work Studies section describes the review of literature in the past related to the studies related to the proposed work. The dataset section describes the description of using data to apply our model and the features of the data. The neuro-genetic model section describes the mechanism of our proposed model through the algorithm and the flowchart. The method section describes the implementation of our proposed technique. The result section describes the outcomes of the implementation system. The conclusion section describes the summary of the proposed work and its implementation in the paper.

Related work studies

Researchers recently emphasized the effectiveness of hybrid designs that integrate ANN with metaheuristic approaches. In one of the studies[1], the goal was to evaluate the accuracy and consistency of the feed-forward-back propagation neural network

(FFBPNN) approach, improving the performance of the solar transformer-consumption (STC) system. The MATLAB code for the FFBP method was designed to enhance the values of reliability and cost function by reducing the error to 0.0001 accuracy. Its gradient descent learning mechanism may update the neural weights. In the other study[2], the classification framework for potato tuber diseases was provided. The FFNN was built and set up with a sigmoid transfer function. The rate of success for training potato picture identification was 100%. Test results for the batch of potato photos were 91.3% successful.

There were some implementations addressing the problem of prediction by neuro-genetic algorithms; the evolutionary operations investigated in one of studies[10] in interneuron synaptic structure search were performed using a structure synthesis technique designed for a feed-forward multilayer network trained by a teacher. To overbuild networks with one hidden layer, crossover operations over neural network architectures were utilized, where the number of neurons is randomly chosen within a certain range. The created neural network handles diagnostic problems with 72–99% accuracy for the compositions mentioned in the study. In one of researches[11] where the genetic algorithm employed the neural model to imitate the parameters, the computing complexity was less than the other optimization methods, and the neuro-genetic algorithm offered the ideal design for the parallel robot. The purpose of one of the works[12] was to deal with human activity recognition utilizing human gait patterns. Deep neural networks, bidirectional long-short-term memory (BLSTM), convolutional neural networks (CNN), and CNN-LSTM are used to further classify the data. The accuracy of various classifiers' model classifications was stated to be 58%, 84%, 86%, and 90%. While deep learning-based human activity recognition in human-robot cooperation utilizing RGB photos is described in other studies[13]. To anticipate human intention, two deep learning algorithms, ConvLSTM and LRCN, were utilized. The ConvLSTM approach showed 74% prediction accuracy. When compared to ConvLSTM, the LRCN technique showed a 25% worse prediction accuracy.

In one of the researches[14] provided a unique, data-guided paradigm for complicated HAR and automated labeling based on cell phones. The findings reveal that the hierarchical k-medoids (Hk-medoids) method produces high-quality labels (93.89%). Machine and deep learning approaches, as well as other classic dimensionality reduction and TDA feature extraction techniques, are proposed in other works[23] to tackle the HAR problem. The tests are carried out using two publicly available datasets (WISDM and UCI-HAR). The system's accuracy was between 95.34% and 100% when applied to the datasets and the three experiments.

Data Set

The data set in our suggested model was connected to physical activities. The experiment comprised seven male and three female participants (ages 25 to 30) who had witnessed violence in the form of physical fighting. During 20 different studies, each subject was instructed to do 10 normal and 10 aggressive activities. In a 3D intelligent environment, a human is seen undertaking some physical activity. As a perception-to-action unit, two external devices—a 3D tracker (the Vicon system) and a mobile robot (the SCITOS G5)—collaborate to create accomplished activity classification and mechanical attribute-based data[29].

Vicon Robot Module

The 3D data and pictures are obtained in this system by a low-level machine that captures and combines data from nine high-resolution infrared cameras. Then use the motion model extraction module to obtain 3D models based on the alignment of markers on an item. Because the endpoints of the limbs are those that create physical actions, these kinematic models focus largely on the limbs and, more particularly, on the terminal stimuli, and the sampling is repeated to capture the picture per second to construct the time series at the end.

Vicon is a collection of activity information expressed in time series to be supplied to the robot for classification processing and assessment evaluation, ranging from image processing to time series creation. It categorizes the data and replicates a portion of the data to demonstrate the action classification performance.

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

The data set features

The suggested data for the proposed system's implementation are three-dimensional data, which are numerical values that describe a human's physical motions. This information was gathered using the Vicon robot system, which detects bodily motions, and it was preserved and classified for use in training the intended hybrid system.

There were two sorts of physical acts in the data set: normal and aggressive.

Each of these motions has a unique set of qualities, which are represented by ten different forms of movement for each sort of physical action, which are normal and aggressive.

The following is the data that reflects the type of normal physical action:

Normal: bowing, clapping, handshaking, hugging, jumping, running, seating, standing, walking, and waving.

And here are the data that describe the sort of aggressive physical action:

aggressive: elbowing, front kicking, hammering, headering, kneeing, pulling, punching, pushing, side kicking, and slapping.

Each of the ten types of physical movements described the characteristics that are used to train the system and design it for this type of movement.

These characteristics are represented by parts of the body that indicate the type of movement, and their three-dimensional measurements are taken, and each of them is organized and arranged in the manner required to be included in the system. The right arm, the left arm, the right leg, and the left leg—each body part contains a pair of markers

that can be used to give the type of movement (except for the head), and each of them is three-dimensional data.

Arm signs: wrist and elbow

Leg signs: ankle and knee

CRDS: The three coordinates (x, y, and z) that define the position of each 3D marker within space.

x: x coordinate

y: y coordinate

z: z coordinate

Thus, each procedure has nine markers, and with each marker there are three coordinates, as shown in table 1.

Table 1.The data set features

Section	**Head**			**Left Arm**						**Right Arm**						**Left Leg**						**Right Leg**					
Sign				Wrist			elbow			wrist			Elbow			Ankle			knee			ankle			knee		
Crds	x	y	z	x	y	z	x	Y	z	x	y	z	x	y	z	x	y	z	x	y	z	x	y	z	x	y	z

Neuro-Genetic model

In our work, a hybrid smart model was proposed, which is a system consisting of two algorithms, namely the feedforward artificial neural network and the genetic algorithm, to combine the characteristics of the two algorithms utilize their advantages and integrate their work mechanisms to produce an effective system that is more efficient and accurate to be used to give you the value of predicting the required solution. Artificial neural networks are an exact replica of a biological neural network that effectively correlates many parameters with a larger number of unknown data points. And that this algorithm has the characteristics of efficient construction that can be trained based on the input data to adapt to it to produce a system capable of predicting the solution.

The proposed algorithm consists of an input layer, 12 hidden layers (each layer has three neurons), and an output layer. There are two datasets used in the input layer in ANN which are aggressive and normal datasets represented as two neurons, these data are processed through 12 hidden layers. Using the sigmoid function in each neuron, produce the activity type recognition result for the output layer, and give the output as aggressive or normal represented as two neurons, and in the same time we get the result from each layer to keep it for evaluation the layers in the

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

selection stage in GA. Then the result is evaluated by comparing it to the real result of a system check, to be improved after that by means of the genetic algorithm.

The GA algorithm is based on the theory of biological evolution. Survival of the fittest is the main process that is mimicked in GA. And used this algorithm to take advantage of its characteristics in preserving the individuals with good efficiency, which can give the appropriate solution for use in the next generation. It is also possible, using this algorithm, to change the characteristics of the incompetent individuals to make them better at giving the solution and the result.

To do the optimization, it needs to check each layer in the ANN to evaluate the weights of each layer by knowing the error value for each of them. To do this, it proposed to break up the original ANN algorithm into multiple ANNs based on the number of hidden layers, as shown in Fig.3. To consider that after each hidden layer there is an output layer to evaluate the weights in each layer, in this way after each hidden layer it considers new ANN, and find out the result from each ANN, verify it, and calculate the resulting error value, it needs to keep the error values generated from each ANN in a list and consider this list as a fitness list in the genetic stage

In the selection stage, it performed a sorting procedure on the fitness list from low to high, as the lower error value represents the best solution, and this layer can be kept for use in the new generation. This sorting was adopted to arrange the artificial neural networks as shown in Fig.4. The selection process was based on the first values from the list considered to be the best ANN to be used in the next generation.

As shown in Fig.5, the first four items from the sorted fitness list were selected to perform the crossover between the final layers, this is done by exchanging the weights between the final layers in the selected ANNs. It is important for the rest of the artificial neural networks that are inefficient to give the solution, so it took the opportunity to improve their weights through the mutation process.

The mutation process works on neurons as well as on layers, as shown in Figs. 6 and 7, respectively, but this mutation focused on updating the weights in a way that reduces the error value. This is implemented by adjusting the state in the negative direction of the obtained error value; the limits are determined by which weights are randomly generated and whose values minimize the error value.

After completing the necessary steps in GA, then combine the multiple ANNs into one ANN algorithm with multiple hidden layers and check the error generated from them after the merge. If the solution is not satisfactory; and continue the process by creating a new generation until discover a good solution. And that is by breaking up the algorithm again in the next generation and repeating the same operations, and the limits required in the mutation process require that they be set again in proportion to the final error value resulting from the previous generation for the next generation, and the result is calculated, and continue to create other generations until a Satisfying result is obtained. The following algorithm and the pseudocode explain the steps that should follow depending on the Neuro-Genetic model, and in depicted as a flowchart in Fig.1.

Neuro-Genetic Algorithm

Step 1: Send the extracted data from the Vicon robot system from the input layer to the feedforward ANN.

Step 2: Do the sum of products with the required weights, then do the processing within each neuron by sigmoid function in the first layer and send the output to the rest of the hidden layers.

Step 3: Do the sum of the product with the required weights, then do the processing within each neuron by sigmoid function in each layer, then do the Breaking up process on the ANN to multiple ANNs by computing the output and error value from each hidden layer.

Step 4: Keep the error from each ANN in a list, get the output from the output layer, and consider it the result from the original ANN.

Step 5: Determine the outcome of the last ANN and compare it to the intended outcome; if the result does not satisfy the desired result, do the optimization process using genetic algorithm operations.

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

Step 6: Sort the list of ANN errors from low to high to do the selection stage in GA, the enhancing phase, and rearrange the layers according to the sorting list.

Step 7: Do the crossover step in the first four layers after arranging them by exchanging the weights.

Step 8: Give an opportunity to the rest of the hidden layers by doing a mutation on the weights, changing them randomly in the negative direction of error.

Step 9: Combine multiple ANNs into a single ANN and go back to step 3.

The pseudocode

```
data input: X = aggressive data, Y = normal data
W1 = required weights in the first hidden layer
W2 = required weights in 11 hidden layers
W3 = required weights in output layer
N1: size of data set in the first hidden layer
N2: size of input data in 11 hidden layers
num_neurons = 3
num_layers = 11
for i in range(num_neurons):
    for j in range (N1):
            s1 = dot product(X, W)
            s2 = sigmoid(s1)
     s3.append(s2)
N2 = len(s3)
def ANN(W2):
 for j in range(num_layers):
      for j in range(num_neurons):
          for k in range(N2):
              s1 = dot product(s3, W2)
              s2 = sigmoid(s1)
      s4.append(s2)
        s3 = s4
        s4 = []
     res = dot(s3, W3) // the output from each hidden layer
     error = 0.5 * (sqr(res-desired_output)) // error from each hidden layer
     fit_list.append(error) // keep error from each hidden layer in one list
return fit_list, error
a, b = ANN(W2)
def GA(fit_list, error):
     fitness_list =  sort(fit_list)
     for i in range(len(fitness_list)):
          for j in range(len(fitness_list)):
               if  fitness_list[i] == fit_list[j]
               d.append(j)
     for j in range len(d):
        d1 = d[j]
         new_W.append(W2[d1])
     new_W [0] = new_W [1] // crossover
     new_W [2] = new_W [3]
     GA_W.append(new_W)
   if error>0.1:    // mutation
       v1 = 0.1
       v2 = 0.001
 mut_W = random.uniform(v1, v2, size = (num_layers, num_neurons, n2)
GA_W.append(mut_W)
return GA_W
ga_w  = GA(a, b)
a, b = ANN(ga_w)
```

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

Materials and Methods

In our work, it employed the feedforward ANN method (input layer, 12 hidden layers, three neurons in each layer, and an output layer). As depicted in Fig.2, the two datasets (aggressive and normal) feed into ANN through the input layer with two neurons and acquire the output of the neurons using the sigmoid activation function to get the result of aggressive or normal action from the two neurons' output. Then employed the genetic algorithm for the purposes of learning and training it.

To do the implementation, one of the two datasets is fed from the input layer to the first hidden layer in ANN and processed in each of the three neurons, then fed to the rest of the hidden layers in ANN.

Breaking up process

The idea put forth in our hybrid approach's first stage for learning systems is to divide the ANN into multiple ANNs to evaluate the weights in each layer by computing the output from each layer, comparing the result with the actual output, as shown in Fig.3, and computing the error by comparing the result with the actual output. The comparing is done by doing the Mean Squared Error process. So, it assumed that there was an output layer after each layer and assessed each method after breaking it up. Then the error values for each ANN algorithm are recorded in a list.

The genetic algorithm now plays a part. Following the division of an ANN into multiple ANNs and the keeping of the error values in the list, it must carry out the following phases of the genetic algorithm:

The selection

At this stage of the genetic algorithm, it needs to reorder the fitness list of error values obtained from each ANN after the breaking up process from low to high by sorting process, as the lowest value, which is actually the first four values after rearranging the order, is considered the best ANN and has the opportunity to survive in the next generation, and the higher value indicates that these values are the worst and that it cannot be chosen but rather requires a chance to improve in order to be a better ANN. The selection procedure according to Table 2a is shown in Fig.4.

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

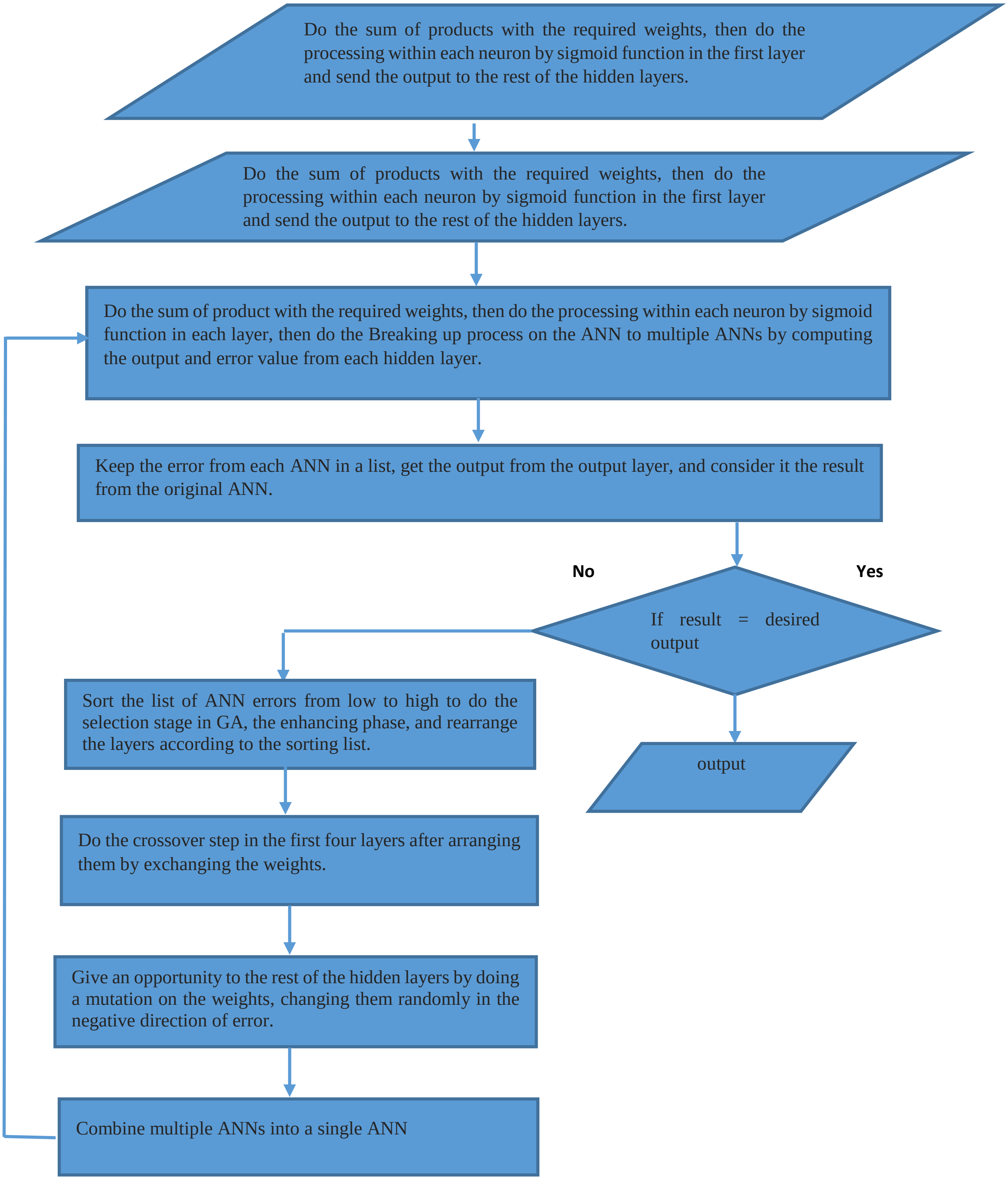

Figure 1. The flowchart of the neuro-genetic model

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

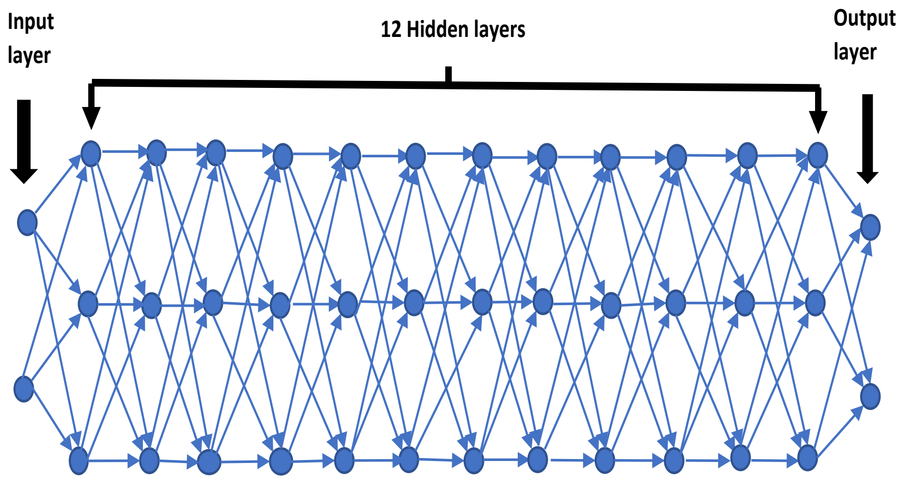

Figure 2. Feedforward Artificial Neural Network (ANN): input layer, 12 hidden layers with three neurons for each layer, and output layer

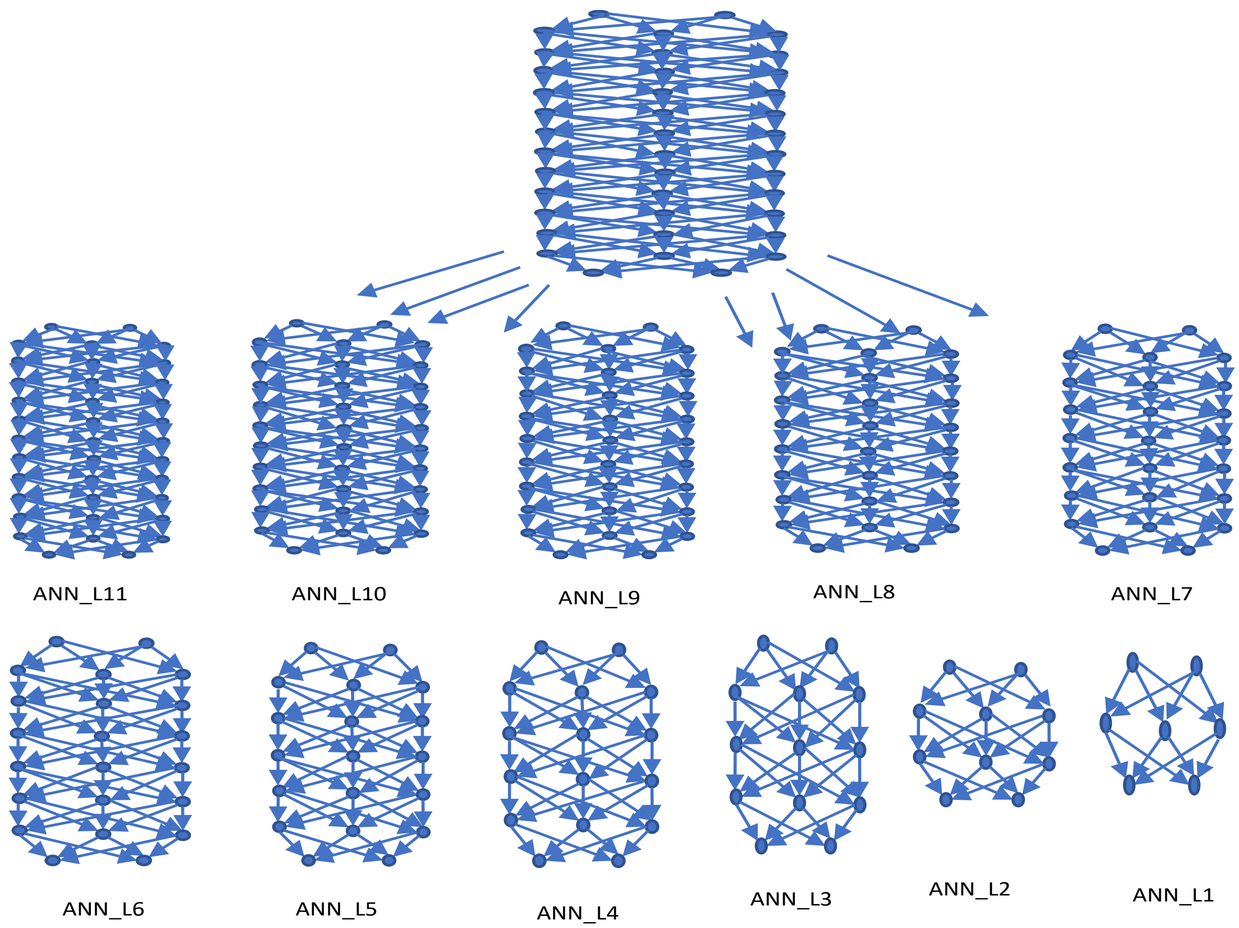

Figure 3. The main ANN (12 layers) breaks into 11 ANN from left to right: 11 layers, 10 layers, 9 layers, 8 layers, 7 layers, 6 layers, 7 layers, 6 layers, 5 layers, 4 layers, 3 layers, 2 layers, and 1 layer

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

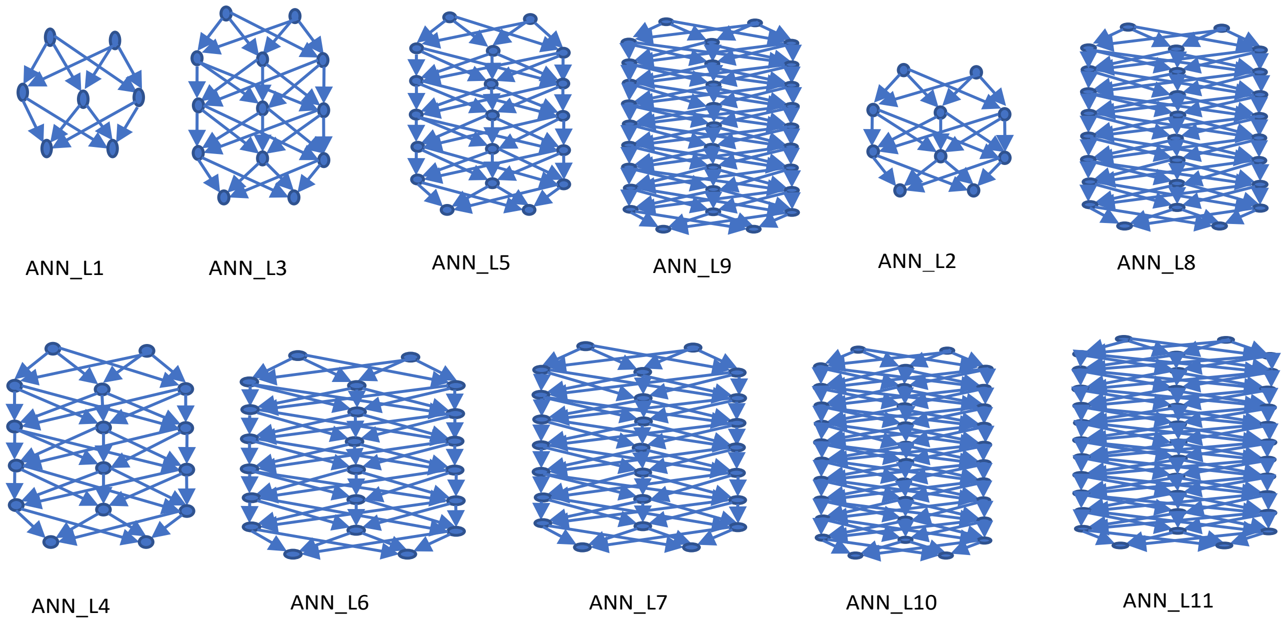

Figure 4. Arrange multiple ANNs from right to left according to the error values, from best to worst

The crossover

In this phase, it operates on layers that have competence and a higher chance of survival and that were selected through the selection process because they had the lowest error value when they were evaluated.
This stage of the genetic algorithm is a crucial one in the optimization process because it depends on the previous stage, which is the selection stage, which is represented by rearranging the errors from low to high, and the rearrangement of the ANNs depends on this sorting.

The top four ANNs have been chosen for the crossover procedure, and at least at this point, they exchange the weights between the final layers, as illustrated in Fig.5, and as shown in Eq. 1, the crossover process depends on the exchange of neurons' weights in the last layer between the two chosen ANNs.

$$Cross\ (N_a, N_b) = f\ (N_a, N_b)$$
$$= f\ (N_a\ (W_{ijk}), N_b\ (G_{ijk}))$$
$$= f\ (N_a\ (G_{ijk}), N_b\ (W_{ijk})) \qquad (1)$$

Whereas:

Na, Nb: The two selected ANNs

W: The coefficient weights for Na

G: The coefficient weights for Nb

i, j, k: The rows and columns for W and G in last layer k

Mutation process

After the two processes of selection and crossover, it performs an important process in the optimization step, which is the mutation process. It performs the mutation process on the rest of the ANNs, which do not have the opportunity to survive in the next generation. The mutation occurs in neurons and layers in ANNs.

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

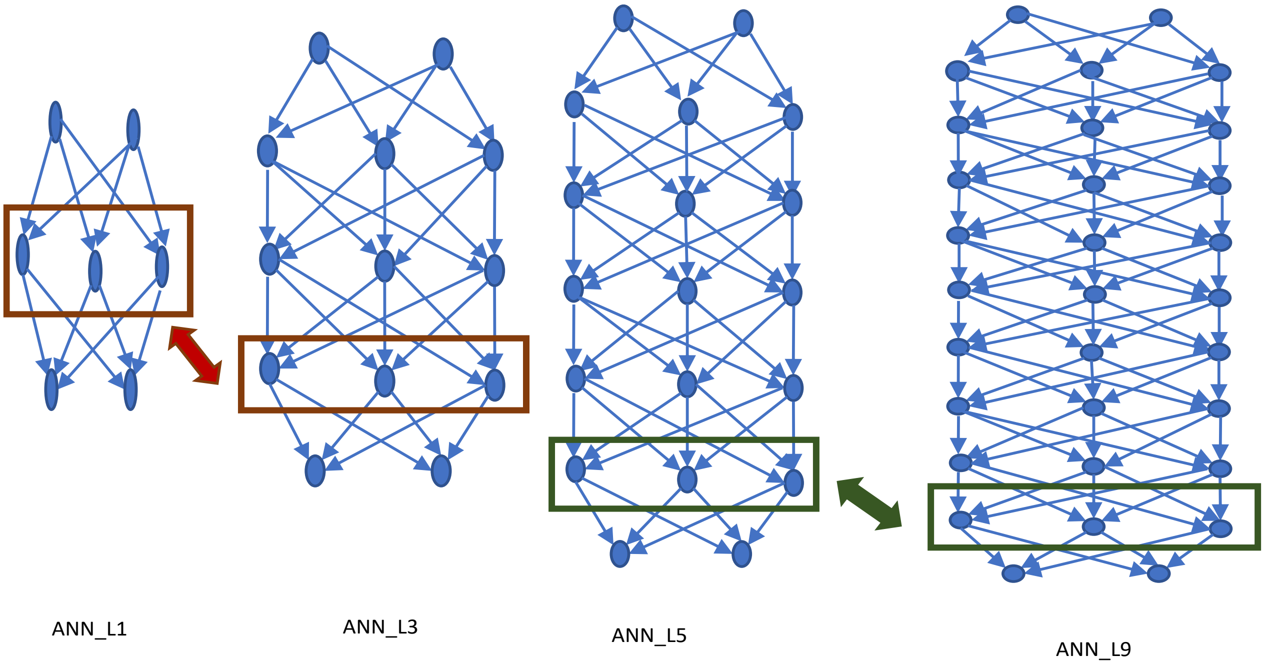

Figure 5. Crossover operation between the top 4 ANN layers in layers

The weights of neurons were randomly chosen to change it. The process of mutation proceeds in a way with the negative direction of error obtained from ANN by setting the limit of the range that decreases the obtained error, then choosing the value of new weights randomly. The range is set to be within the bounds of the limit based on the actual output.

In Figs. 6 a and b, the mutation process occurs on neurons and layers, respectively, and in Fig. 7, the random choice of neurons and layers in ANN is shown.

This method requires preparing the necessary conditions to set the limit of range with a negative direction of error from the previous generation, and this is done by putting a restriction on the choice of the values randomly so that when comparing the result of the current generation with the actual output, the error decreases to be close to zero.

Combining ANNs process

After completing the enhancing phases required in GA, which depend on breaking up the original ANN into multiple ANNs, the combining process takes place on the multiple ANNs to make a single ANN by merging the last layer from each ANN depending on the fitness list order to make one ANN.

The necessary operations of feedforward ANN are performed in the resulting combined ANN, and the algorithm's result is computed and compared with the real output. If it does not achieve a satisfying result, it needs to be generated again with the necessary phases by GA to improve it till it reaches the desired output.

$$Cann = f(Na_1(Gijk)).f(Na_2(Wijk)).f(Na_{12}(Zijk)) \quad (2)$$

Whereas:

Cann: The combination process of ANNs

$Na1 \ldots Na_{12}$: The ANNs

G, W, Z: The coefficient weights for ANNs

i, j, k: The rows and columns for W and G in last layer k

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

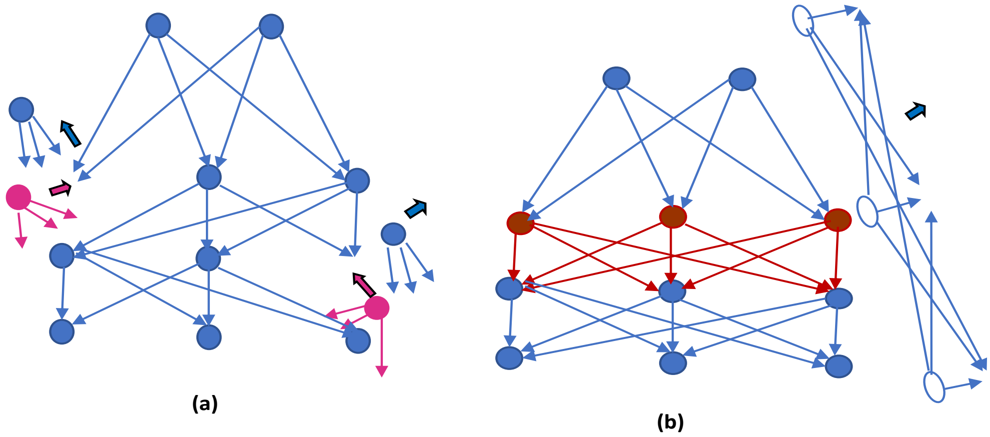

Figure 6. The mutation operation: (a) on the neuron (b) on the layer

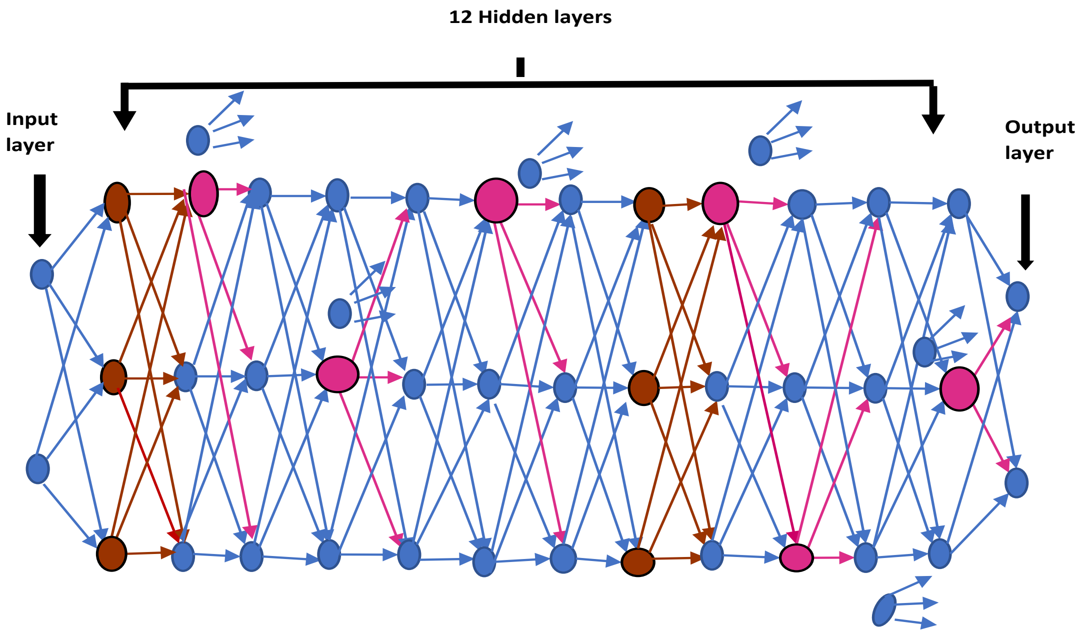

Figure 7. The mutations on the neurons and layers in ANN

Results and Discussion

In our system, 20 different study datasets (10 aggressive datasets and 10 normal datasets) were applied to 20 models.

Each experiment's data were fed into the hybrid model, and the hybrid system was trained by a genetic algorithm to determine the type of activity to be distinguished. It achieved favorable outcomes after optimizing the genetic algorithm via crossover and mutation on the feedforward ANN algorithm. It produced satisfactory results by training the algorithm to acquire the needed outcomes in the following generation after optimization.

Tables 2a and 3a, show the results obtained from the system trained on the first experiment's data type of aggressive activity and normal activity respectively; the data includes 10 types of the intended activity. The model was applied, and a satisfactory result was obtained from the last generation.

In this table, the first column shows the sequence of obtained ANNs algorithms after breaking up the original ANN, and each field in the second column shows the error value obtained corresponding to ANNs. These values are kept in a list to be sorted from low to high, as explained in the method part of the selection phase in GA, and thus it may know the

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

best layers in an ANN algorithm that are at the top of the sorted list, as represented by the third column, and the fourth column indicates the sequence numbers of ANN algorithms that correlate to their error values in the third column. These values in the columns are obtained from the last generation.

after the optimization process by GA using the two processes of crossover and mutation. In this experiment, it obtained a satisfactory result from the last generation, as shown in table 2b and 3b respectively.

Table 2. a) Aggressive model from the first experiment

Aggr-ANNs	Fitness obtained error values	Sorted fitness values	The order
0	-3.8551828911067074e-121	-1.0	9
1	-0.004035434128855457	-0.004035434128855457	1
2	-2.3488706705579096e-18	-2.3488706705579096e-18	2
3	-1.7640841860319345e-80	-3.5970795256464405e-21	6
4	-2.0215166118010636e-63	-2.7248181846321714e-24	7
5	-2.8362560259526247e-26	-2.8362560259526247e-26	5
6	-3.5970795256464405e-21	-2.0215166118010636e-63	4
7	-2.7248181846321714e-24	-5.683620634501858e-71	8
8	-5.683620634501858e-71	-1.7640841860319345e-80	3
9	-1.0	-4.873847001595578e-89	10
10	-4.873847001595578e-89	-3.8551828911067074e-121	0
The Error after optimization		-4.96236099312539e-41	

Table 2. b) The error and output data from the aggressive model's final generation after ten experiments

Agg-Exp	The error	The output
1	0.0	1.0
2	0.0	1.0
3	2.2878728934447336e-17	0.99999999999999997712127106555266
4	0.0	1.0
5	0.0	1.0
6	0.0	1.0
7	0.0	1.0
8	0.0	1.0
9	0.0	1.0
10	0.0	1.0

Table 3. a) Normal model from the first experiment

Norm-ANNs	Fitness obtained error values	Sorted fitness values	The order
0	0.0	0.0	0
1	0.00023252097002035566	0.0	2
2	0.0	0.0	4
3	1.0	0.0	8
4	0.0	0.00023252097002035566	1
5	1.0	0.9989161826888586	7
6	1.0	1.0	3
7	0.9989161826888586	1.0	5
8	0.0	1.0	6
9	1.0	1.0	9
10	1.0	1.0	10
The Error after optimization		0.0	

Table 3. b) The error and output data results from the final generation of the normal model from 10 experiments

Norm-Exp	The error	The output
1	-4.96236099312539e-41	4.96236099312539e-41
2	-1.8602282718359e-71	1.8602282718359e-71
3	2.924145246144702e-133	-2.924145246144702e-133
4	0.0	0.0
5	1.155262893526501e-112	-1.155262893526501e-112
6	4.770020284835052e-193	-4.770020284835052e-193
7	1.894851199121847e-76	-1.894851199121847e-76
8	1.78404991801049e-126	-1.78404991801049e-126
9	2.1957398882818688e-45	-2.1957398882818688e-45
10	4.219016278556796e-71	-4.219016278556796e-71

The output value of the hybrid system designed for aggressive activity is represented by the value 1, while the output value represented by normal activity is 0.

As a discussion of the proposed system, results were acquired in our work to differentiate between aggressive and normal types of activities. When Feedforward ANN was initially applied to the aggressive and normal activity data sets for each of the ten types of activities, it produced results that were unsatisfactory when compared to the actual output. As a result, the algorithm was improved using a genetic algorithm, which took the place of the backpropagation process for learning.

After the optimization process, in which the layers were assessed through the breaking process, it became clear that the best layers were adopted through the crossover process, and through the mutation process, the weights were changed in a way that reduced the error to obtain satisfactory results, and the error values became very close to zero, as shown in the tables.

Conclusion

The method of genetically modifying the feedforward ANN algorithm's structure led to an improvement in learning without the need for the backpropagation operation.

After dividing the ANN algorithm into multiple ANNs, merging them into one ANN, and then reevaluating it, the crossover and mutation processes were used to develop the ANN learning process in our model because it is possible to get better results through the generations produced from each population. The Vicon robot system's 3D data and physical behaviors were analyzed using the model, and the analysis produced a pleasing outcome. The system that is produced can also be applied to automation, control, and analytical systems. In the future, it might be possible to create a method that allows for repeated crossover and selection with better learning, as well as the option of using higher-quality neurons from other layers to get the outcome. It might also be possible to suggest a random mutation based on a mathematical model that considers the inverse relationship between error and mutation.

Authors' Declaration

- Conflicts of Interest: None.
- We hereby confirms that all the Figures and Tables in the manuscript are ours. Besides, the Figures and images, which are not ours, have been given the permission for re-publication attached with the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee in university of Peoples' Friendship University of Russia (RUDN University)

Authors' Contribution Statement

I. V. Stepanyan and S. A. Hameed contributed to the design and implementation of the research, to the analysis of the results and to the writing of the manuscript.

References

1. Gupta R, Ekata, Batra C. Performance Assessment of Solar-Transformer-Consumption System Using Neural Network Approach. Baghdad Sci J. 2022; 19(4): 0865. https://doi.org/10.21123/bsj.2022.19.4.0865
2. Hassoon IM, Qassir SA, Riyadh M. PDCNN: Framework for Potato Diseases Classification Based on Feed Foreword Neural Network. Baghdad Sci J. 2021; 18(2)(Suppl.): 1012. https://doi.org/10.21123/bsj.2021.18.2(Suppl.).1012
3. Asroni A, Ku-Mahamud KR, Damarjati C, Slamat HB. Arabic Speech Classification Method Based on Padding and Deep Learning Neural Network. Baghdad Sci J . 2021; 18(2)(Suppl.). 0925. https://doi.org/10.21123/bsj.2021.18.2(Suppl.).0925
4. Cartwright H, editor. Artificial Neural Networks. Methods Mol Biol. New York, NY: Springer US; 2021. https://doi.org/10.1007/978-1-0716-0826-5
5. Douglass MJJ. Book Review: Hands-on Machine Learning with Scikit-Learn, Keras, and Tensorflow, 2nd edition by Aurélien Géron. Phys Eng Sci Med. 2020; 43: 1135–1136. https://doi.org/10.1007/s13246-020-00913-z.
6. Wirsansky E. Hands-On Genetic Algorithms with Python: Applying genetic algorithms to solve real-world deep learning and artificial intelligence problems . Google Books. Packt Publishing Ltd; 2020. https://books.google.ru/books?hl=ar&lr=&id=A0vODwAAQBAJ&oi=fnd&pg=PP1&dq=3.%09Wirsansky
7. Xiao X, Yan M, Basodi S, Ji C, Pan Y. Efficient Hyperparameter Optimization in Deep Learning Using a Variable Length Genetic Algorithm. arXiv. 2020; 200612703. https://doi.org/10.48550/arXiv.2006.12703
8. Abudalghaffar AN. Darwinian Philosophy as Optimization Method for Design High Reflection Mirror Include New Merit Function. Baghdad Sci J. 2010; 7(1): 90-7. https://doi.org/10.21123/bsj.2010.7.1.90-97
9. Mahmood RAR, Abdi A, Hussin M. Performance Evaluation of Intrusion Detection System using Selected Features and Machine Learning Classifiers. Baghdad Sci J. 2021; 18(2)(Suppl.): 0884. https://doi.org/10.21123/bsj.2021.18.2(Suppl.).0884
10. Stepanyan IV. Evolutionary operations of interneuron synaptic structure for feed-forward multilayer networks. J Mach Manuf Reliab. 2020;49(10):874–7. http://dx.doi.org/10.3103/s105261882010009x.
11. Erick Garcia Lopez, Yu W, Li X. Optimum design of a parallel robot using neuro-genetic algorithm. J Mech Sci Technol 2021; 35(1): 293–305. https://doi.org/10.1007/s12206-020-1229-6
12. Bijalwan V, Semwal VB, Gupta V. Wearable sensor-based pattern mining for human activity recognition: deep learning approach. Ind Rob. 2021; 49(1): 21–33. https://doi.org/10.1108/IR-09-2020-0187
13. Keshinro B, Seong Y, Yi S. Deep Learning-based human activity recognition using RGB images in Human-robot collaboration. Proc Hum Factors Ergon Soc Annu Meet. 2022; 66(1): 1548–53. https://doi.org/10.1177/1071181322661186
14. Qi W, Wang N, Su H, Aliverti A. DCNN based human activity recognition framework with depth vision guiding. Neurocomputing. . 2022; 486: 261–71. http://dx.doi.org/10.1016/j.neucom.2021.11.044
15. Uddin, Z. and Soylu, A. (2021) Human activity recognition using wearable sensors, discriminant analysis, and long short-term memory-based neural structured learning. Sci Rep. 2021; 11: 16455 https://doi.org/10.1038/s41598-021-95947-y
16. Golestani N, Moghaddam M. Human activity recognition using magnetic induction-based motion signals and deep recurrent neural networks. Nat Commun. 2020; 11. https://doi.org/10.1038/s41467-020-15086-2
17. Dirgová Luptáková I, Kubovčík M, Pospíchal J. Wearable Sensor-Based Human Activity Recognition with Transformer Model. Sensors. 2022; 22(5): 1911. https://doi.org/10.3390/s22051911
18. Khan IU, Afzal S, Lee JW. Human Activity Recognition via Hybrid Deep Learning Based Model. Sensors. 2022; 22(1): 323. https://doi.org/10.3390/s22010323
19. Nair R, Ragab M, Mujallid OA, Mohammad KA, Mansour RF, Viju GK. Impact of Wireless Sensor Data Mining with Hybrid Deep Learning for Human Activity Recognition. Rani S, editor. Wirel Commun Mob Comput. 2022;: 1–8. https://doi.org/10.1155/2022/9457536
20. Luwe YJ, Lee CP, Lim KM. Wearable Sensor-Based Human Activity Recognition with Hybrid Deep Learning Model. Inform. 2022; 9(3): 56. https://doi.org/10.3390/informatics9030056

2023, 20(6 Suppl.): 2608-2623
https://dx.doi.org/10.21123/bsj.2023.9087
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

21. Dua N, Singh SN, Semwal VB, Challa SK. Inception inspired CNN-GRU hybrid network for human activity recognition. Multimed Tools Appl. 2023; 82(4): 5369–403. https://doi.org/10.1007/s11042-021-11885-x.
22. Vrskova R, Kamencay P, Hudec R, Sykora P. A New Deep-Learning Method for Human Activity Recognition. Sensors . 2023; 23(5): 2816. https://www.mdpi.com/1424-8220/23/5/2816
23. Balaha HM, Hassan AE-S. Comprehensive machine and deep learning analysis of sensor-based human activity recognition. Neural Comput Appl. 2023; 35(17): 12793–831. http://dx.doi.org/10.1007/s00521-023-08374-7
24. Lee S, Lee DW, Kim MS. A Deep Learning-Based Semantic Segmentation Model Using MCNN and Attention Layer for Human Activity Recognition. Sensors. 2023; 23(4): 2278. https://doi.org/10.3390/s23042278.
25. Jeyakumar JV, Sarker A, Garcia LA, Srivastava M. X-char: A concept-based eXplainable Complex Human Activity Recognition model. Proc ACM Interact Mob Wearable Ubiquitous Technol. 2022; 7(1):1–28. http://dx.doi.org/10.1145/3580804
26. Snoun A, Bouchrika T, Jemai O. Deep-learning-based human activity recognition for Alzheimer's patients' daily life activities assistance. Neural Comput Appl. 2023; 35(2): 1777–802. http://dx.doi.org/10.1007/s00521-022-07883-1
27. Megamsolutions. Research. Iitm.ac.in. 2023. https://rbcdsai.iitm.ac.in/home/research/
28. Vicon . Vicon. 2019. https://www.vicon.com/
29. UCI machine learning repository. Uci.edu. 2023. http://archive.ics.uci.edu/ml

نموذج عصبي محسّن للتعرف على البيانات الحركية ثلاثية الأبعاد للإنسان المستخرجة من نظام فايكن روبوت

إيفان فيكتوروفيتش ستيبانيان[1,2]، صفا عبد الوهاب حميد[2]

[1]معهد بحوث الهندسة الميكانيكية التابع للأكاديمية الروسية للعلوم, موسكو, روسيا.
[2]قسم الميكانيكا وعمليات التحكم ، أكاديمية الهندسة ، جامعة صداقة الشعوب في روسيا, موسكو, روسيا.

الخلاصة

في هذه الأيام ، انه من الضروري التمييز بين نوع السلوك البشري ، تلعب تقنيات الذكاء الاصطناعي دورًا كبيرًا في ذلك المجال. تم دمج خصائص خوارزمية الشبكة العصبية الاصطناعية (FANN) والخوارزمية الجينية لإنشاء آلية عمل مهمة تساعد في هذا المجال. حيث يمكن استخدام النظام المقترح للمهام الأساسية في الحياة ، مثل التحليل والأتمتة والتحكم والتعرف والمهام الأخرى. التقاطع والطفرة هما الآليتان الأساسيتان اللتان تستخدمهما الخوارزمية الجينية في النظام المقترح لاستبدال عملية الانتشار العكسي في ANN. بينما تركز تقنية الشبكة العصبية الاصطناعية المغذية على معالجة المدخلات ، يعتمد هذا العمل على عملية كسر خوارزمية الشبكة العصبية الاصطناعية المغذية. بالإضافة إلى ذلك ، يتم حساب النتيجة من كل ANN أثناء عملية التفكك ، والتي تعتمد على تقسيم خوارزمية الشبكة العصبية الاصطناعية إلى عدة شبكات ANN بناءً على عدد طبقات ANN ، وبالتالي ، كل طبقة في الشبكة العصبية الاصطناعية الأصلية يتم تقييمها. يتم اختيار أفضل الطبقات لمرحلة التقاطع بعد عملية الكسر ، بينما تمر الطبقات الأخرى بعملية الطفرة. ثم يتم تحديد مخرجات هذا الجيل من خلال دمج الشبكات العصبية الاصطناعية في شبكة ANN واحدة ؛ ثم يتم فحص النتيجة لمعرفة ما إذا كانت العملية تحتاج إلى إنشاء جيل جديد. ان أداء النظام جيدًا وأنتج نتائج دقيقة عند استخدامه مع البيانات المأخوذة من نظام Vicon Robot ، والذي تم تصميمه بشكل أساسي لتسجيل السلوكيات البشرية بناءً على بيانات ثلاثية الابعاد وتصنيفها على أنها طبيعية أو عدوانية.

الكلمات المفتاحية: عملية التفكيك ،عملية الجمع ، التقاطع ،خوارزمية الشبكة العصبية الاصطناعية المغذية ،الطفرة ،النموذج الجيني العصبي ،التحسين ،التعرف ، فايكن روبوت و بيانات ثلاثية الأبعاد.