CHAPTER 7

DEVELOPMENT OF ARTIFICIAL NEURAL NETWORK MODEL

7.1 INTRODUCTION

In the modern world the technique like Neural Networks turn to be a very powerful tool to build inter-relationships between input and output variables of many complicated systems. Generally, system refers to a structure that gets an input data and processing them to provide an output. An Artificial Neural Network is an interconnected group of nodes which is similar to the huge network of neurons in a brain. This is based on mathematical and algorithms called as threshold logic. This model is basic approach to develop the biological processes in the brain and also the application of neural networks to artificial intelligence. The neural networks are mainly based on the large collection of neural units which modeling the way as a biological brain solve problems with large clusters of biological neurons connected by axons. Normally, the neural networks contains multiple layers or a cube design and the signal path pass through from front to back. The forward stimulation is used to reset the weight on the front neural units where the back propagation also presents. But sometimes, it can do in combination with training where the correct result is known. Here, dynamic neural networks are the most advanced networks in which they form new connections based on rules. Neural networks are an information processing system and which are constructed and implemented to model the human brain. For implementation of artificial neural networks, high speed digital computers are used, which makes the simulation of neural process feasible.

Artificial Neural Networks (ANNS) are systems that are deliberately constructed to make use of some organizational principles resembling those of
human brain. The key factors that distinguish Artificial Neural Networks from other computational techniques are

- ANNS are nonlinear; able to classify patterns and capture complex interactions among the input variables in the system.
- ANNS are adaptive: they can take data and learn from it. (i.e.) online training.
- ANNS can generalize: they can correctly process data that broadly resembles the data they were trained originally.
- ANN is a parallel-distributed information processing structure.

7.2 NEED

The neural network solves the many complicated systems and problems and builds inter-relationship between input and output parameters. It acquires the knowledge through training and learning session with examples and providing solutions without any prescribed specific formulae about the nature of the problem.

7.3 WORKING PRINCIPLE OF ANN

First the selected parameters are given as input to the neural network and resultant desired or target response is set as output. An error is calculated from the difference of the desired response and the real system output. The error information is feedback again to the system that makes all adjustments to their parameters in a systematic fashion also referred as the learning rule. This process is repetitive until the desired output is satisfactory. The performance of the system is based on the selection of network topology, the trigger function, learning rule and the criteria for stopping the training phase. It is difficult in identifying the size and parameters of the network since there is no formula or rule to do it. The network consists of one input layer, one or two hidden layers and one output layer of neurons. All the neurons between two
successive layers are fully connected, i.e. each neuron of a layer is connected to each neuron of neighboring layers. However there is no connection between neurons of the same layer. The input layer receives input information and passes it onto the neurons of the hidden layer(s), which in turn pass the information to the output layer. The output from the output layer is the prediction of the net for the corresponding input supplied.

7.3.1 Artificial neuron model

The model of an artificial neuron is shown in figure 7.1. In this model the processing elements (neuron) computes the weighted sum of its inputs and outputs according to whether this weighted input sum is above or below a certain threshold $\theta_k$. The externally applied bias has the effect of lowering the net input of the activation function.

![Artificial neuron model](image)

**Figure 7.1 Artificial neuron models**

7.3.2 Neural network connections

ANNS are weighted directed graphs in which neurons are referred as nodes and directed edges with weights are connected between outputs and inputs. The connection patterns are categorized into
- Feed-forward networks in which graphs have no loops.
- Recurrent or feedback networks with loops.

Feed-forward networks are static as they create only one set of output values for a given input. These networks are often referred as memory-less in the sense that their response to an input is independent of the previous network state. This network projects an output layer of neurons based on an input layer of source codes. The simplest form of single layer feed forward network is presented in Figure 7.2. In a feed forward network the knowledge is stored in a distributed manner in the form of synaptic strengths and thresholds. Thus it can be generalized, i.e. it may be used for the situations where the net has not been trained.

![Single layer feed forward network](image)

**Figure 7.2 Single layer feed forward network**

The interconnection of several layers forms a multilayer feed-forward networks. The typical setup is presented in figure 7.3. The layer between the input and output layer is called a hidden layer and its function is to arbitrate between the external input and output. The hidden layer has no direct contact with the external environment. The radial basis function network is a special class of multilayer feed-forward networks. Feedback networks that have
closed loops are called recurrent networks. In single layer recurrent network, the processing element output is feedback to itself or other processing element or both. In the multilayer recurrent network the neuron output can be directed back to the nodes in the preceding layers.

![Multilayer feed forward network](image)

**Figure 7.3** Multilayer feed forward network

### 7.4 LEARNING IN ANN

Learning is not a unique process; there are different learning processes, each suitable to different species. The concepts of learning processes have been borrowed from the behaviorist’s lab and implemented in electronic circuitry. Learning is a process by which neural network adapts itself to stimulus and eventually (after making the proper parameter adjustments to itself) produces a desired response. Learning is also a continuous classification process of input stimuli. During the process of learning, the network adjusts its parameters based on synaptic weights in response to input stimulus. So that actual output response converges to the derived output response.
7.4.1 Supervised Learning

In supervised learning, learning and solution have been trained with the help of supervision. Each input vector requires a corresponding target vector. It is denoted as the desired output. The input vector along with the target vector is referred as training pair. During training, the input vector is presented to the network which provides the output vector. It is taken as the actual output, which is compared with the desired output vector. If there is any difference between the actual and desired output vectors then the error signal get generated by the network. This error signal is used for the adjustments of weights until the actual output matches the desired output. Weights can be initially randomly set and then adjusted by the network. Hence the next iteration or cycle will produce a closer match between the desired and the actual output. This learning method tries to minimize the current errors of all processing elements. This global error reduction is created over time by modifying the input weights continuously until acceptable network accuracy is reached. When the neural network reaches an user defined performance level then this training is considered as complete. This level ensures that the network has achieved the desired statistical accuracy as it produces the required outputs for a given sequence of inputs.

7.4.2 Unsupervised Learning

In unsupervised learning, the learning is done without the help of a teacher. In unsupervised learning, the network receives the input patterns and organizes these patterns to form clusters. There is no feedback to check the output whether it is correct or not. So the network must discover patterns, regularities, features from the input data and also the relations for the input data over the output. These networks use no external influences to adjust their weights. Instead, they internally monitor their performance. These networks
look for regularities or trends in the input signals, and makes adaptations according to the function of the network.

7.4.3 Reinforcement Learning

The reinforcement learning is same as that of the supervised learning because the network receives some feedback from its environment. In supervised learning, the correct target output values are known for each input pattern. But in some cases, less information might be available. Here, the feedback obtained is only evaluative. The external reinforcement signals are proposed in the critic signal generator then that critic signals are sent to an artificial neural network for adjustment of weights. The reinforcement learning is also called learning with a critic which indicates supervised learning.

7.4.4 Learning algorithms

The learning algorithm is the precise mathematical method being used to update the inter-neuronal synaptic weights by each training iteration. A variety of learning algorithms were used under each learning rule. Selecting a suitable model from the set available model that minimizes the experimentation cost is the real task for the training of the network model. Several algorithms with the use of optimization theories and statistical techniques are readily available. Some algorithms use back propagation to work out the actual gradients with inclusion of gradient descent. The net work parameters and its cost derivatives in a gradient related direction are considered for this algorithm. It includes the factors like steepest descent, quasi-Newton and conjugate gradient. The other methods for training this network include gene expression programming, expectation-maximization, particle swarm optimization, non-parametric methods and simulated annealing.
7.4.4.1 Back propagation learning algorithm

Based on this algorithm, the network learns a distributed associative map between the input and output layers. The process by which the weights are calculated during the learning phase of the network makes this algorithm different than the others. In general, difficulty with multilayer perceptions include calculation of hidden layer weightage in an efficient way that results in the least (or zero) output error. If there are more hidden layers then there are more difficulties will arise. For weights updating, one must calculate an error. The error is the difference between the actual and desired (target) outputs. At the output layer this error can be easily measured. There is no direct observation of the error at the hidden layers. Hence some other techniques must be used to calculate error.

7.4.4.2 Learning with back propagation algorithm

During the training session of the network, a pair of patterns is presented \((x_k, d_k)\), where \(x_k\) is the input pattern and \(d_k\) is the target or desired pattern. The \(x_k\) pattern causes output responses at each neuron in each layer and, hence actual output \(O_k\) at the output layer. At the output layer, the difference between the actual and target outputs yields an error signal. This error signal depends on the values of the weights of the neurons in each layer. This error is minimized, and during this process new values for the weights are obtained. The speed and accuracy of the learning process (i.e., the process of updating the weights also depends on factor known as the learning rate. The basis for this weight update algorithm is just as the gradient – descent method used for simple perceptron with differentiable units. For a given input – output pair \((x_k', d_k')\) the back – propagation algorithm performs two phase of data flow. First, the input pattern \(a_k'\) is propagated from the input layer to the output layer and, as a result of this forward flow of data, it produces an actual output \(y_k\). Then the error signals resulting from the difference between \(d_k\) and \(y_k\) are back -
propagated from the output layer to the previous layers for them to update their weights.

7.4.5 Learning rates.

Learning rate is defined in the perspective of optimization, and minimizing the loss function of a neural network. Perceptibly, a slower learning rate means more time is spent in accomplishing the off-line learning to produce an effectively trained system. Due to faster learning rates, the network may not be able to make the fine possible discriminations with a system that learns more slowly. The learning rate is depending upon the factors like paradigm selection, type of learning rule or rules employed Network complexity, architecture, size and expected accuracy. Generally the value is positive and lies in between zero and one. The network will oscillate if the value is greater than one. Small values will not correct the current error as rapidly, but the best minimum convergence could be arrived if small steps are taken in correcting errors.

7.4.6 Learning laws.

The understanding of neural network and their processes may vary about their perceptions. Some researchers may set the primary objective is modeling of biological learning. Others want to experiment the things for adaptations. However the man's perceptive of how neural processing actually works is very limited. It is the necessity of learning laws in modeling. Hebb's Rule is the oldest learning law and other laws are formed with little variations with this law. Indeed, learning is definitely more complex than the simplifications represented by the learning laws at present. Some major laws are presented as examples.
Hebb's rule states that if a neuron receives an input from another neuron and both are highly active then the weight between the neurons should be strengthened.

Hopfield Law specifies that the magnitude of the strengthening or weakening. It states that whether the output and input are both active and inactive, increment the connection weight by the learning rate, if not decrement the weight by the learning rate.

The most commonly used is the Delta rule that is based on the simple idea of constantly modifying the strengths of the input connections to reduce the difference between the desired output value and the actual output of a processing element. It changes the synaptic weights and further minimizes the mean squared error of the network.

The Gradient Descent Rule is similar to the Delta Rule in that the derivative of the transfer function is still used to modify the delta error before it is applied to the connection weights. However, an additional proportional constant tied to the learning rate is appended to the final modifying factor acting upon the weight.

7.5 APPLICATIONS

Neural networks are performing successfully where other techniques found difficulties in recognizing and matching of incomplete, vague and complicated patterns. The wide variety of problems can be solved by the application of neural networks. The most common use for neural networks is to set the priorities. Essentially, all organizations should have the control of priorities that govern the allotment of their resources. Neural networks also used as a module of knowledge acquisition in expert system in stock market forecasting with amazingly accurate results. Now days it can be astonishingly used for the
prediction of bankruptcy in credit card institutions. The specific advantage of reproducibility in modeling of nonlinear processes found application areas include system identification and control such as vehicle control, natural resources management, process control, trajectory prediction, decision making in games. ANNs have been used in medical fields particularly in diagnose the type of cancers, and to differentiate the quantity level of invasive cancer cell lines. ANNs have been used for ocean modeling and coastal engineering, building black-box models in geosciences, hydrology, and geomorphology etc.

7.6 DESIGN OF NEURAL NETWORK FOR WEAR

The excellent learning ability and relating the inputs and output determines the usage of back propagation type of neural network. Multilayer preceptor type is used for the convenient of the system. It minimizes the mean square error by comparing the actual output and desired output for the given inputs. The feed forward ANN with back propagation algorithm is used to develop the model. The neurons were organized in such a way that the signals are going in forward direction and errors are moving backwards. The supervised learning reduces the errors in the network. The training started with the random weights and the purpose is to reduce the error as low as possible. Hence the relationship between the actual responses and error is always in nonlinear manner. The primary objective of this design is to develop a neural network model to predict the desired specific wear rate from the given input parameters like percentage of reinforcements, applied loads and sliding velocity. MATLAB tool is used to design the network model. It includes training and testing the wear data, simulation and estimation of network and post processing of predicted data for validation. The design and algorithm used for the developed model is presented in Figure 7.4.
7.7 MODEL DEVELOPMENT AND IMPLEMENTATION

The Levenberg–Marquardt algorithm with back propagation is used in the implementation of neural network model. Since this algorithm is fundamentally a non linear least-mean square often connected with Multi-layered type feed forward system. The input neurons are connected with hidden layer neurons and concerned weights are monitored by the network. This weights are linked again with back propagation to reduce the error and kept as minimum. The interconnection between the errors and input signals treated as weights that can be controlled in learning stage. The number of training cycles shall be repeated till the network reaches the significant level of accuracy. The number of neurons presents in the hidden layer and associated transfer function is to be decided by trial and error method depends on the values of mean square error. Titanium dioxide particles in percentage, loads applied and sliding velocity were selected as three input neurons at the
same time the specific wear rate was selected as the neuron at output layer. In order to eliminate the difficulties in the training and fast accessing of the network, transfer function is set within the range of 1 and 0. The experimental data is set as 0 and 1 as their limit. The experimental data used for this model implementation is 100. Among the available data, 70 values were used to train the network. 15 values were used for the validation of the model and remaining 15 data were used to test the performance of the network.

![Figure 7.5 Experimental data analysis](image)

The data allotted to the three modules namely training, validation and testing is presented in Figure 7.5. Also it shows the results of Mean Squared Error (MSE) and regression ‘R’ values. The expected error values must be equal to one. It refers to the squared values of difference between response neurons and input neurons. Since it indicated very low value the selected model is found satisfactory. The regression value is very closer to accuracy. The regression value defines the relationship between the target value and output. The correlation value gave a very close relationship between the target and response.

### 7.8 RESULTS AND DISCUSSION

#### 7.8.1 Regression curve analysis

This network application arbitrarily divides input neuron vectors and target neurons vectors into three sets as follows. 70% are used for training, 15% are
Figure 7.6  ANN model regression curve

used to for validation and to stop the training before over fitting. The balance 15% is used as a entirely self-regulating test of network generalization. The regression curve for the developed model is presented in Figure 7.6.  The correlation coefficient is determined with the use of regression analysis curve. The correlation between the predicted and experimental values and its average value of 100 data is 0.988 for specific wear rate. This value gave a good sign for the model to be accurate. Similarly the system gave the value of 1 for training, 0.999 for validation and 0.999 for testing. The degree of closeness of the plotted points with the solid line would decide the accuracy of the predicted values. The dashed line represents the best linear fit. The results
observed were logical because of the subsequent considerations: The final mean square error was very small and closer to zero. The test and validations set errors had similar characteristics.

7.8.2 Comparison of ANN and experimental values

The predicted values given by the artificial network model and experimental values are compared with the aid of graphs. Randomly 15 samples and its predicted and actual values are compared and presented in the Figure 7.7. Then two set of data from the five different compositions of the composites were selected for the analysis and the result is presented in Figure 7.8. From the interpretation it is found that there is good agreement between the actual and predicted values of wear data.

![Predicted Vs Actual analysis of samples](image)

**Figure 7.7** Predicted Vs Actual analysis of samples
Figure 7.8 Predicted Vs Actual analysis of composites

7.8.3 Fit and validation analysis

Figure 7.9 Data fitting analysis of composites
The correlation coefficient of the network and the percentage of errors are based on the degree of fitness of the data with their allotted functions. Hence it is essential to select the appropriate quantity of data for the training, validating and testing. The target and input values of each module should be evaluated. Also this system is having the facility to evaluate the total targets and output of the signals. The inputs and targets are plotted as a curve and presented in Figure 7.9. The best fit curve is plotted in the Figure 7.10. The network model gave the best performance validation is 0.00054298 and indicated in the validation analysis curve.

7.9 CONFIRMATION TESTS

The validation of the ANN model is purely based on the confirmation tests. The randomly three input variables are selected to conduct the experiments for the specific wear rate. The specific values of load, sliding velocity and percentage of reinforcement are applied in the dry sliding wear behaviour test with the use of computerized pin on disc machine. The wear losses were
measured and documented for determining the actual output. The input values were given to the developed model and final predicted output response is also recorded. With the same procedure 5 different runs were selected and the output parameters were monitored. From the interpretation of predicted and experimental values the deviation is calculated for further analysis. It is observed that the percentage of error is as low as 2%. Hence the developed model is perfect and accurate.

7.10 COMPARISON OF RSM AND ANN

The response surface methodology in the form of Box-Behnken methodology and Mixture design approached predicted the wear data. Hence it is necessary to combine and compare the predicted values of wear output from the tools of statistical quality control and artificial neural network model with that of experimental results. The compared values are presented in Figure 7.11.

![Comparison of Wear data](image)

Figure 7.11 Comparison analysis of ANN with RSM
One set of data from the variation of reinforcements (0, 2.5, 5, 7.5 and 10%) is taken and compared the predicted and actual output values. The result shows that both the technologies are accurate to predict the specific wear rate and can be useful in determination of other parameters related with metal matrix composites.

7.11 SUMMARY

An intelligent approach of Artificial Neural Network model was developed to forecast the specific wear rate of magnesium based titanium dioxide composites. The effects of particles reinforcement percentage, applied load and sliding speed and its effect on tribological behaviour was studied. The specific wear rate of magnesium based titanium dioxide composites with different mixing ratios were predicted by adequately trained neural network based on material composition and testing conditions. The network predicted the wear rate and the value of correlation coefficient is closer to the value of 1. By considering the error analysis, the percentage of average error is within acceptable range. The selected back propagation algorithm and multi-layered perceptron feed forward network ensures then accuracy of network system in metal matrix composites. The investigated values of predicted and experimental values obeyed similar trend for specific wear rate. Thus the developed model can be effectively used to predict the wear rate of Mg-TiO$_2$ composite material at 95% confidence level within the range of investigation.