NEURAL NETWORK MODELING OF ENGINEERING MATERIALS

Adebayo Y Badmos

Engineering Technology and Materials Science

Black Hawk College

Moline, IL 61265, USA

Abstract

Artificial neural networks are non-linear models used for empirical regression and classification problems. The central idea of the technique consists of training the network to learn an existing relationship between a set of variables and a target in a given dataset and then use the trained network to predict the output of a new data. Their flexibility makes them able to discover more complex relationships in data than the traditional statistical models. A general introduction to the basic principle of neural networks methods is presented and application to engineering materials is illustrated with the analysis of the tensile properties of mechanically alloyed oxide dispersion strengthened steel.

Keywords: Neural network, training, prediction, materials properties.

1. Introduction

Neural network modeling provides a powerful and efficient analytical tool with a wide range of applications in materials development and processing. The technique simplifies the processes involved in alloys development and materials selection, and reduces the time and costs significantly, compared to the traditional methods. Moreover, neural network as a classification model is an efficient process control tool with applications in processes such as welding, heat-treatment, casting etc.

Materials development and processing is very complex. Although scientific investigations on materials have helped greatly in understanding the underlying phenomena, there remain many problems where quantitative treatments are lacking. For example, there is little or no progress in predicting mechanical properties of materials because of their dependence on large numbers of variables. Neural network models are extremely useful in such circumstances, where the complexity of the problem is overwhelming from a fundamental perspective and where simplification is unacceptable.

Neural network [1] is a more general method of regression analysis. As in regression analysis, the input data x_j are multiplied by weights, but the sum of all these products forms the argument of a hyperbolic tangent. The output y is therefore a non-linear function of x_j, the function usually selected being the hyperbolic tangent because of its flexibility. The exact shape of the hyperbolic tangent can be varied by altering the weights (Fig. 1a). Further degrees of non-linearity can be introduced by combining several of these hyperbolic tangents (Fig. 1b), so that the neural network method is able to replicate almost arbitrarily non-linear relationships.

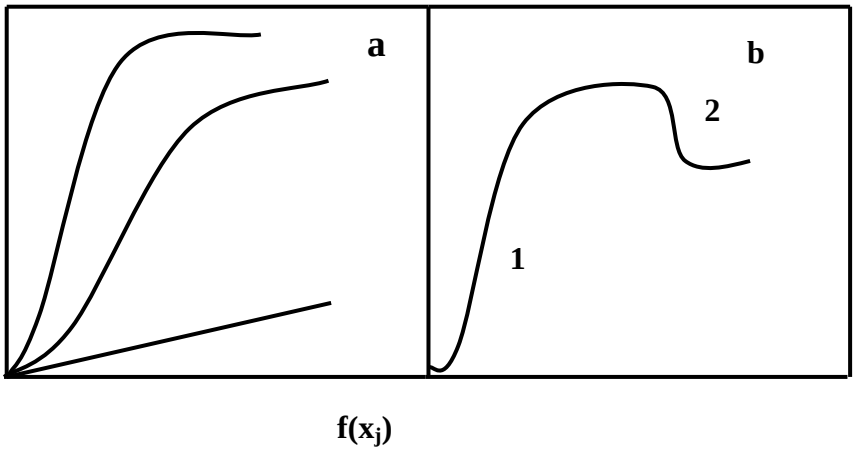

Figure 1: Various shapes and complexity of hyperbolic tangent functions.

The neural network method has recently been applied to many materials problems, for example: the impact toughness of C-Mn steel [2] an analysis of the strength of nickel base superalloys [3]

austenite formation in steels;[4] yield and ultimate tensile strength of steel welds [5] fatigue crack growth rate in nickel base superalloys [6] mechanical properties in the heat affected zone of power plant steels [7] prediction of martensite start temperature [8] prediction of the continuous cooling transformation diagram of some selected steels [9] prediction of the measured temperature after the last finishing stand in hot rolling [10] and analysis of the tensile properties of mechanically alloyed oxide dispersion strengthened iron alloys [11].

This paper discusses the basic principle of a neural network technique and the application to material problems is illustrated with the analysis of the tensile properties of mechanically alloyed iron alloys [11].

2. Neural Network Structure

The general structure of a typical neural network is illustrated in Figure 2. Each network consists of input nodes (one for each variable x), a number of hidden nodes, and an output node.

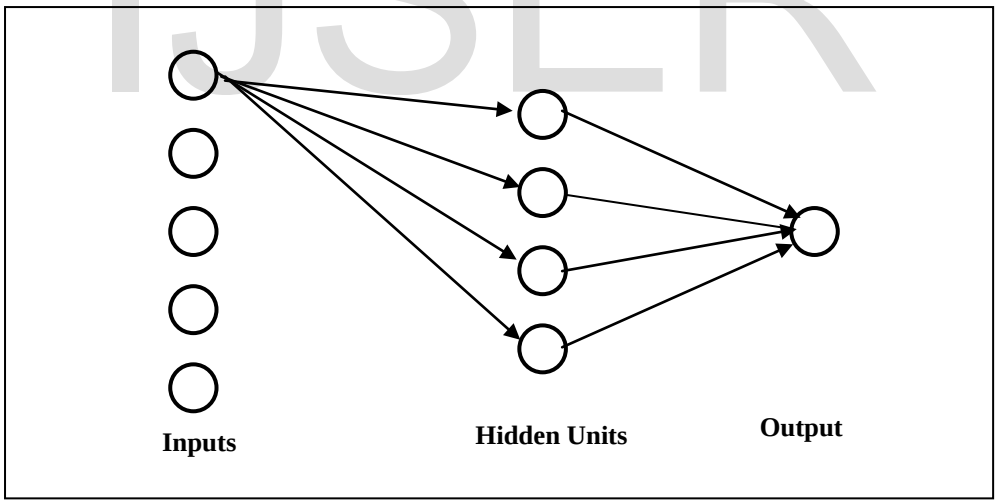

Figure 2: A typical neural network structure

Linear functions of the inputs xj are operated on by a hyperbolic tangent transfer function (equation 1), so that each input contributes to every hidden unit.

$$h_i = tanh\ (\Sigma_j\, w_{ij}^{(1)}\, x_j + \theta_j^{(1)}) \qquad (1)$$

where θ_i and is the bias which is analogous to the constant that appears in linear regression analysis and w_{ij} is the weight which determines the strength of the transfer in each case. The transfer to the output y is linear:

$$y = \Sigma_j w_{ij}^{(2)} h_i + \theta^{(2)} \quad (2)$$

The specification of the network structure, together with set of weights, is a complete description of the formula relating the input to the output. The weights are determined by training the network.

3. Training and Optimization of Neural Networks

A neural network is 'trained' using a set of examples of input and output data. The outcome of the training is a set of coefficients (weights) and a specification of the functions, which in combination with the weight relate the input to the output. The training process involves a search for the optimum non-linear relationship between the input and the output data and is accomplished using a dataset $D = \{x^{(m)}, t^{(m)}\}$ by adjusting the weights w to minimize an error function, e.g.

$$E_D(w) = 0.5 \Sigma_m \Sigma_i [t_i^{(m)} - y_i(x^{(m)};w)]^2 \ldots \quad (3)$$

This objective function is a sum of terms, one for each input-target pair $\{x, t\}$, measuring the degree of correlation between the output $y\{x; w\}$ and the target, t [12]. The parameter m denotes each input-output pair. The minimization is based on the repeated evaluation of the gradient of E_D using 'backpropagation'[1]. The backpropagation algorithm computes for each input-output pair m, the gradient of $\frac{1}{2} [y(x^m;w) - t^m]^2$ by following the 'forward pass' of equations (1-2) by a 'backward pass', in which information about the errors $[y(x^m;w) - t^m]$ propagates back through the network by the chain rule.

The training for each network is started with a variety of random seeds. The value of a term σ_v gives the framework estimate of the overall noise level of the data. The complexity of the model is

controlled by the number of hidden units and the values of the regularization constants, σ_w, one associated with each of the inputs, one for biases, and one for all weights connected to the output.

The noise level decreases monotonically as the number of hidden units increases. However, the complexity of the model also increases with the number of hidden units. A high degree of complexity may not be justified if the model attempts to fit the noise in the experimental data. Mackay [13,14,15,16] has defined a quantity (the 'evidence') which acts as an indicator of the probability of model. In circumstances where two models give similar results for the known dataset, the more probable model would be predicted to be that which is simpler; this simple model would have a higher value of the evidence.

4. Validation of Neural Network

A procedure used to avoid the problem of over-fitting is to divide the training dataset into two equal sets, namely, the training and test datasets. The models are developed using training data only. The unseen test data are then used to assess how well the model generalizes. This process is called validation. A good model would produce similar levels of error in both the test and training data (Fig. 3a) whereas an over-fitted model might accurately predict the training data but badly estimate the unseen test data (Fig. 3b).

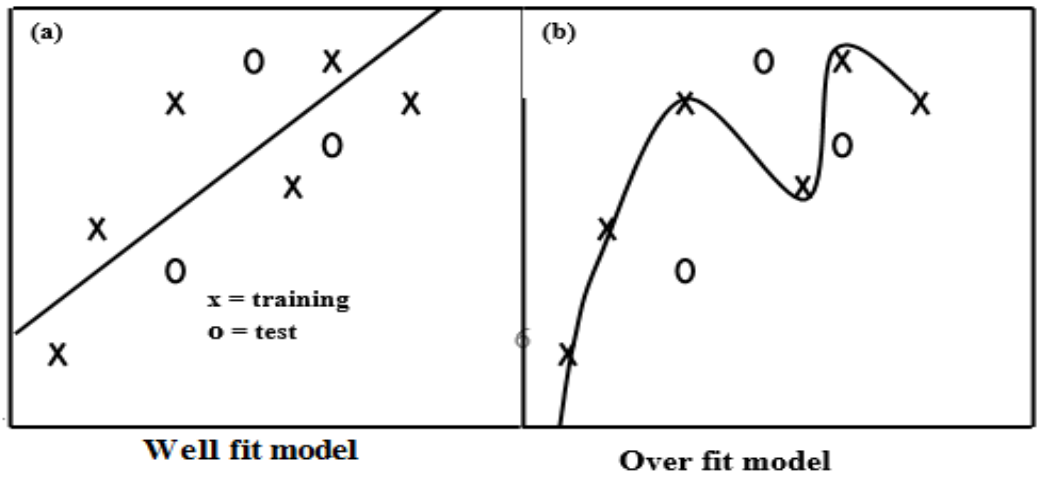

Figure 3: Well-fitted Model and Over-fitted Model.

Once the correct complexity of the model has been determined using this procedure, it can be retrained using all the data with a small but significant reduction in the error. The test error, T_{en}, is a reflection of the ability of the model to predict the target values in the test data.

$$T_{en} = 0.5\ \Sigma_n\ (y_n - t_n)^2 \qquad (4)$$

where y_n is the set of predictions made by the model, and t_n the corresponding target (experimental) values previously unseen by the model.

5. Committee Model

It is common practice in the application of neural networks to train many different candidate networks from the same data by varying either the number of hidden units or starting value**.**

The same data can be modeled in many ways, for example by varying the number of hidden units. The variety of models thus produced can be ranked according to the magnitude of the test error and the best individual model would then have the minimum test error. However, it is possible in principle to improve the performance further by using the average of predictions from a number of good models, i.e. a committee of models. The individual models are first ranked by their test errors and a committee of N models is then formed by combining the best N models. The mean committee prediction is expressed as:

$$y_m = (\Sigma_i\, y_i) / N \qquad (5)$$

where N is the size of the committee and y_i is the estimate of a particular model i.

6. Application to Materials Modeling

The application of neural network analysis to the modeling of engineering materials is illustrated here with the modeling of the mechanical properties of mechanically alloyed oxide dispersion strengthened (MA-ODS) metals. Mechanically alloyed metals are produced through the powder metallurgy method. The alloys possess many unique properties, particularly for high temperature applications; however, a large number of important phenomena are not readily amenable to the general theories. Neural network models were developed for the tensile properties of the alloys as a function of variables known to be important in influencing mechanical properties [11]. The development and application of the model for the ultimate tensile strength are discussed in the following paragraphs.

There were 232 experimental data, 12 input variables, and one output, which is the tensile strength, carefully compiled from literature. The 12 input variables consist of the major alloying elements and processing variables. Both the input and output variables were normalized within the range + 0.5 to – 0.5 to facilitate the subsequent comparison of the significance of each of the variables.

A committee model was used consisting of the top three models whose combination exhibits the least test error as shown in Figure 4.

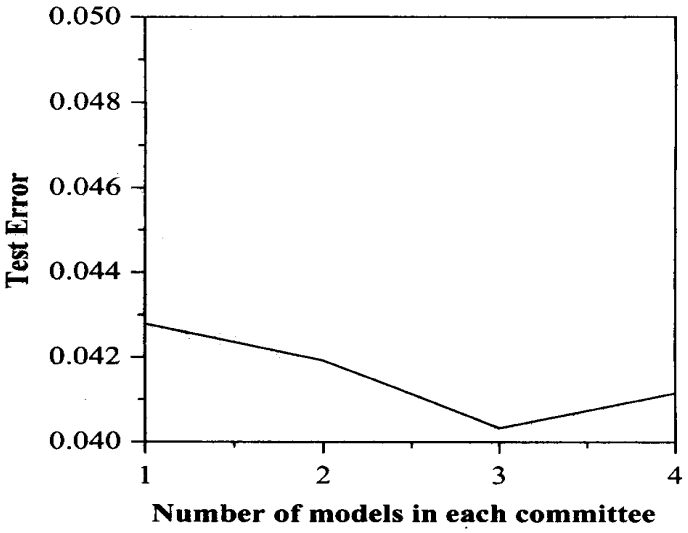

Figure 4: Test errors for committees of models.

The significance of each of the input variables perceived by the various model contained in the committee is shown by a regularization constant, σ_w, in Fig. 5. The test temperature is shown to have the largest σ_w for all the models in the committee. This shows that the models have recognized a pattern correctly because temperature is more widely varied than any other input in the database. Moreover, temperature is known to affect strength very significantly.

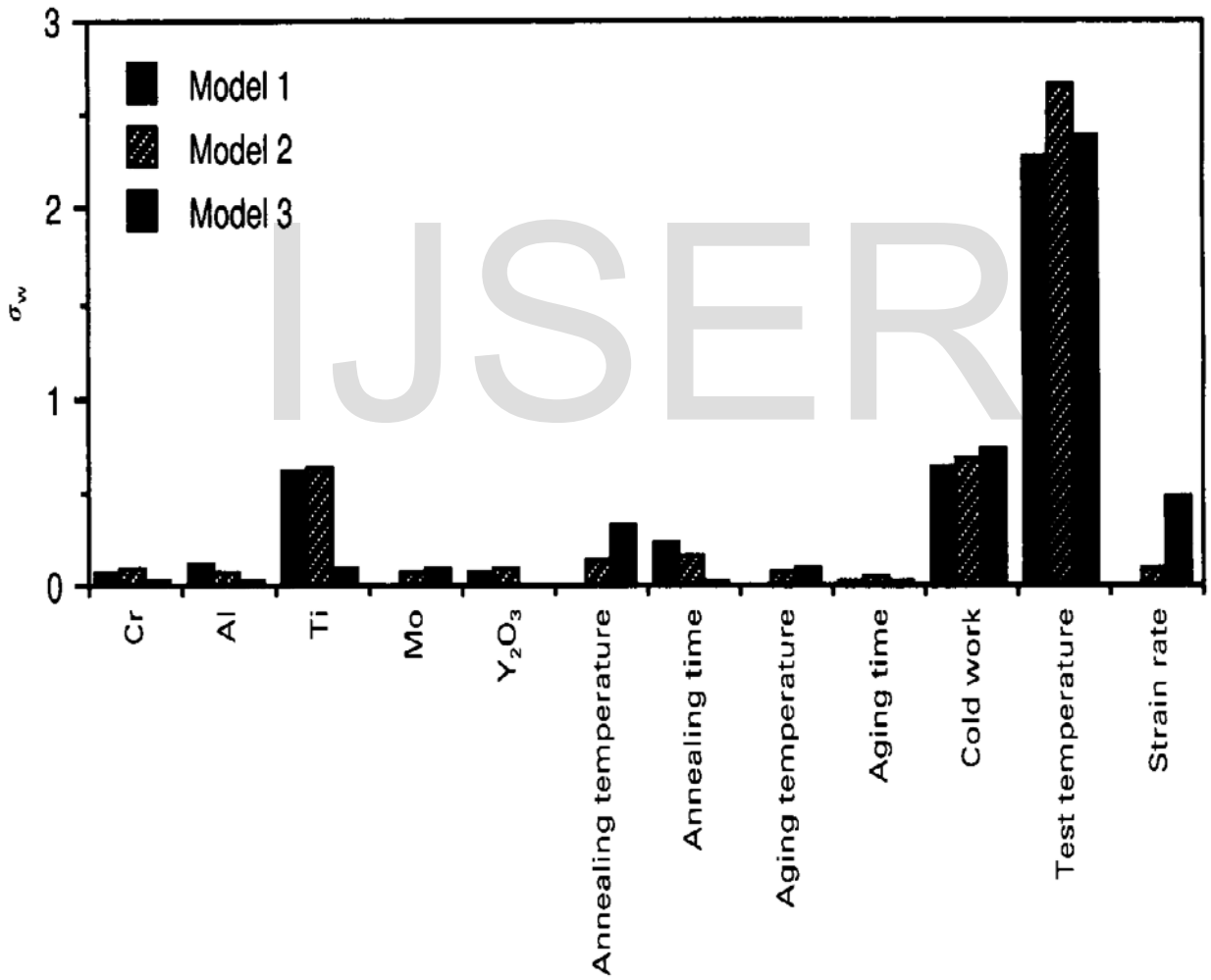

Figure 5: Perceived significance of input parameters by the models in the optimum committee used.

Figure 6 compares the predicted effect of test temperature on the ultimate tensile strength of the recrystallized and unrecrystallized forms of an iron-base MA-ODS alloy. The predicted patterns are reasonable and consistent with experiments. The unrecrystallized form is stronger and, in both conditions, there is no significant reduction in strength until about 500 ^{o}C, when there is a sharp decrease in strength. This is peculiar to MA-ODS alloys.

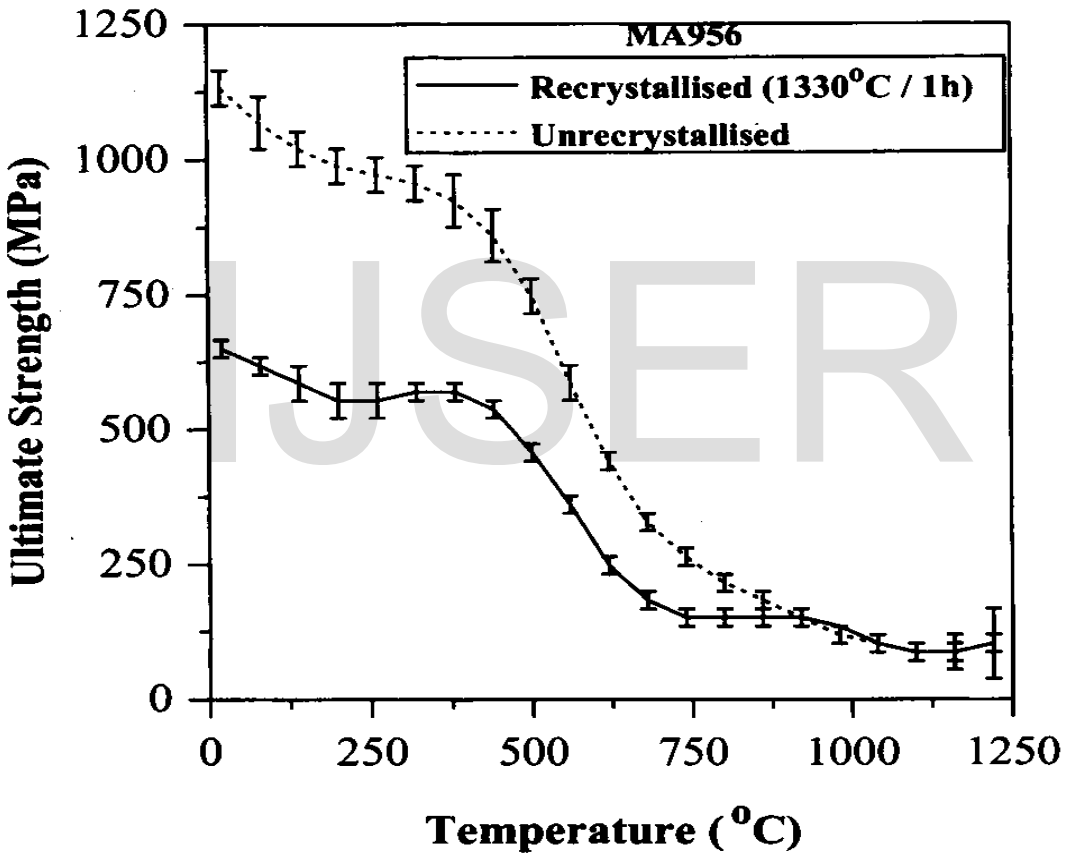

Figure 6: Predicted ultimate tensile strength of some MA-ODS ferritic steels.

Conclusion

A review of the basic concepts of the backpropagation neural network modeling technique has been presented and the application to engineering materials was illustrated with the analysis of the tensile properties of a mechanically alloyed oxide dispersion strengthened ferritic steel in the recrystallized and unrecrystallized conditions.

References

1. Rumelhart D.E., Hinton G.E., & Williams, R.J. (1986). *Nature*, 323: 533-536.
2. Bhadeshia, H.K.D.H., Mackey, D.J.C., & Svensson, L.E. (1995). *Mater. Sci. Technol*., 11(10):1046-1051
3. Jones, J., & Mackey, D.J.C. (1996). Proc. 'Superalloys '96', 417-424, Warendale, PA, TMS.
4. Gavard, L., Bhadeshia, H.K.D.H., Mackey, D.J.C., & Suzuki, S. (1996). *Mater. Sci. Technol*., 1996, 12(6):453-463.
5. Cool, T., Bhadeshia, H.K.D.H., & Mackey, D.J.C. (1997). *Mater. Sci. Eng. A*, 223(1-2):186-200.
6. Fuji, H., Mackey, D.J.C., & Bhadeshia, H.K.D.H. (1996). *ISIJ Int*., 36:1373-1382.
7. Cool, T., & Bhadeshia, H.K.D.H. (1997): '*Mathematical modeling of weld phenomena 3*', (ed. Cerjak, H. & Bhadeshia, H.K.D.H.), 403, London, The Institute of Materials.
8. Vermeulen, W.G., Morris, P.E., de Weijer, A.P., & van der Zwaag, S. (1996). *Ironmaking Steelmaking*, 23(5):433-437.
9. Vermeulen, W.G., van der Zwaag, S., Morris, P.E., & de Weijer, T. (1997). *Steel Res*., (68)72-79.
10. Vermeulen, W.G., Bodin, A., & van der Zwaag, S. (1997). *Steel Res*., 68:20-26.
11 Badmos, A.Y., Bhadeshia, H.K.D.H., & Mackay, D.J.C. (1998). *Mater. Sci. Technol*., 14 (8):793-809.

12. Mackey, D.J.C. (1997). in '*Mathematical modeling of weld phenomena* 3', (ed. Cerjak, H. & Bhadeshia, H.K.D.H.), 403, London, The Institute of Materials.

13. Mackey, D.J.C. (1995). *Network: Comput. Systems*, 6:469-505.