Comparing empirical model and Neural Networks for the determination of the coal abrasiveness Index

John Kabuba* and AF Mulaba-Bafubiandi

Abstract—Many empirical models have been developed to predict the abrasiveness of thermal coal. These empirical models are based on the mineral inclusion in a coal. In this work, two conventional empirical models were compared with a new approach, based on neural networks (NN). Back-propagation neural network was employed, it provided a much better fit to observed data than conventional models do and it seem a very promising tool for the modelling. The configuration of the back-propagation NN giving the smallest mean square errors (MSE) was three-layer NN with tangent sigmoid transfer function at hidden layer with 4 neurons, linear transfer function at output layer and Levenberg-Marquardt back-propagation training algorithm.

Keywords- Abrasive index, coal, neural network, empirical model

I. INTRODUCTION

WHEN coal is used for electricity generation, it is usually pulverized to an efficient burnable size in a mill and then combusted in a furnace with a boiler. The abrasive properties of coal can lead to serious operational problems at a power plant. Wear of pulverising mill can reduce the size control of the pulverised coal and may lead to mechanical failure of mill components. Coal characterisation techniques allow the mineral inclusions in a coal that are responsible for the abrasive nature of the coal to be accurately characterised hence, there is scope to make improved prediction of wear based on a detailed knowledge of the mineral matter in a particular coal. It is important, however, to understand the nature and properties of the mineral matters in a coal that would contribute to abrasive wear. Predictions of wear rate of components in grinding mills at pulverized coal-fired power stations are currently made using empirical relationships [1]. Most empirical models available in literature [1] - [4] for predicting Abrasive Index of coal are linear, since they assume that primary production is a linear function either of a few independent variables. To achieve this, it was suggested a method to compute analysis based on the proximate analysis information using Neural Network (NN). The NN is an information processing tool that is capable of establishing an input–output relationship by extracting controlling features from a database presented to the network and NN based predictive models are powerful in terms of learning the nonlinear relationships to understand and solve, and thereby achieving ability to predict accurately. The objective of this study is to compare empirical model and NN for the prediction of abrasiveness characteristics of thermal coal.

II. NEURAL NETWORK

A. Basic concepts

Neural network (NN) is an information processing system that is inspired by the way such as biological nervous systems e.g. brain. The objective of a NN is to compute output values from input values by some internal calculations [5]. The basic component of a NN is the neuron, also called “node”. Fig. 1 illustrates a single node of a NN.

Inputs are represented by \(a_1, a_2, ..., a_n\), and the output by \(O_j\). There can be many input signals to a node. The node manipulates these inputs to give a single output signal [6]. The values \(w_{ij}, w_{2j}, ..., w_{nj}\), are weight factors associated with the inputs to the node. Weights are adaptive coefficients within the network that determine the intensity of the input signal. Every input \(a_1, a_2, ..., an\) is multiplied by its corresponding weight factor and the node uses summation of these weighted inputs \(w_{ij}a_1, w_{2j}a_2, ..., w_{nj}\) to estimate an output signal using a transfer function. The other input to the node, \(b_j\) is the node’s internal threshold, also called bias. This is a randomly chosen value that governs the node’s net input through (1):

\[
u_j = \sum_{i=1}^{n} (w_{ij}a_i) - b_j
\]

(1)

Node’s output is determined using a mathematical operation on the node’s net input. This operation is called a transfer function.
The transfer function can transform the node’s net input in a linear or nonlinear manner. Three types of commonly used transfer functions are follows:

- **Sigmoid transfer function**
  \[ f(x) = \frac{1}{1 + e^{-x}}, \quad 0 \leq f(x) \leq 1 \]  

- **Hyperbolic tangent transfer function**
  \[ f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \quad -1 \leq f(x) \leq 1 \]

- **Linear transfer function**
  \[ f(x) = x, \quad -\infty \leq f(x) \leq +\infty \]

The neuron’s output, \( O_j \), is found by performing one of these functions on the neuron’s net input, \( u_j \). NN are made of several neurons that perform in parallel or in sequence. The weight factors and thresholds are adjusted in training process. Training is the process by which the weights are adjusted systematically so that the network can predict the correct outputs for a given set of inputs [8].

**B. Back-propagation algorithm (BPNN)**

If you modify More than 50 NN models have been devised so far, and it has been found that the back-propagation learning algorithm based on the generalized delta rule is one of the most efficient learning procedures for multi-layer NN [9]. This technique generally consists of many sets of nodes arranged in layers (e.g. input, hidden and output layers). The output signals from one layer are transmitted to the subsequent layer through links that amplify or attenuate or inhibit the signals using weighting factors. Except for the nodes in the input layer, the net input to each node is the sum of the weighted outputs of the nodes in the previous layer. An activation function, such as the sigmoid logistic function, is used to calculate the output of the nodes in the hidden and output layers. In the calculation, both input and output are usually normalized to give a value between 0 and 1 incorporating various mapping schemes. This depends on adopted activation functions. The number of hidden layers and the number of nodes in each layer depend on the complexity of the patterns and the nature of the problem to be solved. As mentioned above, it is obvious that because predictability of NN is subject to adopted training schemes, attempts to find adapted training environments for given data may be necessary [10].

**C. Performance of neural networks**

The network performance may be evaluated quantitatively in terms of coefficient of correlation (\( R^2 \)), mean summer square of errors (MSSE), and the error rate. Error rate was defined by Yeh et al. [11] as:

\[ \text{Error rate} = \frac{1}{N_p} \sum_{i=1}^{N_p} \text{error}_i \]  

Where
\[ \text{error}_i = \sqrt{\frac{1}{N_{out}} \sum_{j=1}^{N_{out}} (T_j - O_j)^2} \]

Where \( T \) and \( O \) are target and network output values, respectively; \( N_p \) is the number of input patterns; \( N_{out} \) is the number of output neurons, for a network with one output neuron, as in this study.

**III. METHODOLOGY**

In order to compare the performance of NN with that of other empirical models, BPNN was employed to assess the Abrasive Index of coals to mineral. In this type of NN, from the entire 91 sets of data, 50 sets were used for training of the NN, 21 sets were used to test the NN and the remaining 20 sets were used for validation. The BPNN have six input neurons and one output neuron. The six input neurons represent \( \text{H}_2\text{O}, \text{ash}, \text{SiO}_2, \text{pyrite}, \text{CaCO}_3 \) and mineral matter (MM). The output layers had a single neuron to represent AI.

**A. Back-propagation neural network**

In the proposed BPNN shown in Fig. 2, the training process was stopped after 360 epochs. A full connection between nodes in adjacent layers was selected.

![BPNN structure uses in this study](image)

Log-sigmoid and linear were used as a transfer function in
the hidden layers and output layer respectively. The best network geometry according to the highest correlation and the lowest RMS error of the testing tests. The error rates of proposed network were 2.12, 3.72 and 1.85 for training, testing and validation data, respectively. The correlation coefficient ($R^2$) of proposed network was 0.84, 0.59 and 0.94 for training, testing and validation data respectively. All input and output data in the training phase were scaled so that they changed in the range of 0 and 1. The network was trained to predict the AI using a total of 50 training datasets, 21 testing datasets and 20 validation datasets. Identifying of the optimal network architecture was studied by changing the number of layers, nodes, transfer function, and iteration number. The prediction of BPNN versus actual measure for all (training, testing and validation) is shown in Fig. 3.

\[ AI = \sum_{j} W_j H_j \]

Where $W_j$ and $H_j$ are the concentration and hardness, respectively of mineral $j$ in the coal, respectively and $PC\gamma$ and $B\gamma$ are respectively the grindability factor and bulk density of the coal. This equation requires many inputs for prediction and suffers from the drawback that particle size and shape are not included [4]. In this study, the two following empirical equations proposed by Raask [1]: $AI = Qc + 0.5Pc + 0.2Ac$ (8) and Bandopadhyay [5]: $AI = 1.00Ac + 1.35Qc$ (9) were compared with models based on NN. The main reason for this choice was that these equations have been useful in the design of boilers for power stations [3], [4]. In Equations (8) and (9); $Ac$, $Pc$ and $Qc$ are respectively the mass percentages of quartz, pyrite and ash in coal. Since the main concern of this study is to point out a possible improvement in empirical models of AI rather than in NN works, very basic network and training algorithms were selected. Five different networks were trained, the number of hidden layer nodes was determined by comparing the performance of different networks, with 1 to 12 nodes in the hidden layer the lowest mean squared errors were obtained by network with 5 to 8 nodes in the hidden layer, depending on the input variables and on the number of input nodes. The network will be cited according to their structure 6-4-1 network. In order to be used as training patterns for NN, all non-binary raw data were scaled into a [0, 1] interval. The NN training was carried out according to the following scheme:

- 12 patterns were randomly selected out on the 50 samples in the data set
- After 360 epochs, training was stopped and the weights corresponding to the lowest mean squared error were saved.

This training procedure avoided overfitting, i.e. undesired reproduction of data set. Finally, it should be stressed that the performances of the NN that are presented in this study are to be considered as minimal estimates.

IV. RESULTS AND DISCUSSION

A. Comparison with an empirical equation

Although there are several newer empirical linear equations in literature for prediction AI within narrow rank range [4], [3], the Equations (8) and (9) have more compatibility in variables to our proposed approach as discussed in the previous section. Therefore, we used these empirical equations for comparison. It is clear from Fig. 4, $R^2 = 0.4515$ and comparison of $R^2$ of NN-predicted in Figs 5 and 6 the empirical equations have significantly lower accuracy compared to NN results. The prediction was based on complex set of coals. The empirical model presented by Raask [1] provided a fairly good fit to the observed data set, even though they had to sacrifice its simplicity to achieve this result. AI values predicted by NN are plotted against observed AI values in Fig 4. The performances of these networks should be compared, respectively, with the Raask equation which is applied to US and UK coal with proportion of pyrite [4] (see Fig. 5) and with the Bandopadhyay which is applied to India coal with low proportion of pyrite (see Fig. 4). The NN provided a better fit than the best linear model ($R^2 = 0.950$).
The contents NN based empirical models of AI are far more effective than linear empirical models and the higher $R^2$ values bear witness to their superiority. However, even though $R^2$ is an important criterion, it is obvious that a non-linear model provides an improvement in data fitting in comparison with a linear model. The distribution of the predicted value errors is probably the most relevant, because a good model should not provide the mean error close to 0 and the error distribution should be as symmetrical as possible.

NN based models achieved good results even in this respect. In fact, their mean error was very small if compared with the mean error of the composite linear model and negligible if compared with the null mean error of single linear models. Moreover, NN always had a better error distribution than conventional models.

The main processes that determine AI can be approximated by linear or simple non-linear function only to a limited extent. Therefore, such models are not able to reproduce the behavior of real systems when very low or high values of the individual variable are considered. NN can model non-linear systems independently of their complexity. Of course, complex system need complex network, adequate training and a large data set to be modeled.

Fig 7 represents the percentage relative error between NN testing (91 runs) and experimental data. The ultimate mean square error of the NN based model is satisfactory and is found to be 0.002139579 which is within ±1% error range.

V. CONCLUSION

The developed NN model for predicting the abrasiveness characteristic of thermal coal gave a better performance in predicting the value when compared with empirical formulae.

The following inputs were used to train the neural network: $H_2O\%$, Ash %, Free $SiO_2\%$, Pyrite %, $CaCO_3\%$, Remain MM %. The individual network for predicting the abrasiveness of coal, $k$ takes the following form: $k = f(H_2O, \text{Ash, } SiO_2, \text{Pyrite, } CaCO_3, \text{MM})$

Here, $f(.)$ is the underlying non-linear functions defined by the networks. To avoid saturation of the neurons, all the input and output values were normalized to numbers between 0 and 1. BPNN was used to develop NN models for estimation of abrasiveness characteristics of thermal coal on abrasive index. 91 set of data obtained from different collieries were divided into three groups, each containing 50, 20, 21 batches for training, validation and test, respectively. The network giving the least error on the test data was selected. The average mean square error is 0.002139579, which is sufficient to have error within ±1%. During network training, the training algorithm continuously checks the network error on the test data set. The training was terminated at the point where the network error on the test data was a minimum. The present work suggested that NN can be used as an effective technique in modeling, estimation and prediction of AI. Also, NN can be considered as an effective supplement for the conventional and complicated mathematical in the prediction of AI.

REFERENCES


