World Applied Sciences Journal 5 (5): 553-559, 2008
ISSN 1818-4952

Investigating Jordan Oil Shale Properties Using Artificial Neural Network (ANN)

[1]*Jamal M. Nazzal,* [2]*Ibrahim M. El-Emary and* [3]*Salam A. Najim*

[1,3]Al Ahliyya Amman University, P.O. Box 19328, Amman, Jordan
[2]King Abdulaziz University, P.O. Box 18388, Jeddah, King Saudi Arabia

Abstract: The Artificial Neural Network (ANN) is a functional imitation of simplified model of the biological neurons and their goal is to construct useful computers for real-world problems and reproduce intelligent data evaluation techniques like: Pattern recognition, classification and generalization by using simple, distributed and robust processing units called artificial neurons. ANNs are fine-grained parallel implementation of non-linear static-dynamic systems. The intelligence of ANN and its capability to solve hard problems emerges from the high degree of connectivity that gives neurons its high computational power through its massive parallel-distributed structure. The current resurgent of interest in ANN is largely because ANN algorithms and architectures can be implemented in VLSI technology for real time applications. The number of ANN applications has increased dramatically in the last few years, fired by both theoretical and application successes in a variety of disciplines. Among the potential areas of application is how to use ANN in analyzing El-Lajjun oil shale. So the major first objective of this paper is to build a neural network to investigate the effect of composition of EL-lujjum oil shale. Our designed neural network was learned using already collected data from 100 samples of oil shale from El-lajjun deposit in the south of Jordan. The presented neural network in this paper has a capability of predicting the gross calorific value for new samples based on the composition values of: Calcium Carbonate, Organic Carbon and sulfur. The second objective of this paper lies in comparison study between the output results that are obtained from the ANN and that is obtained from some empirical formula, i.e. comparing between the estimated output result that are obtained from ANN with the other similar estimated results obtained using some mathematical formula. The output results show and indicate that ANN outperforms better the mathematical formula with 3%, this means that ANN has a better certainty factor with respect to other traditional approach. An ANN result converges with the experimental results with a percentage of 99.7%.

Key words: Oil shale . calorific values . ANN . neuron . Back Propagation (BP) . SOFM . MLP . RBFN . ART

INTRODUCTION

In recent years there has been a confluence of ideas and methodologies from several different disciplinary areas to give rise to an extremely interesting research area called Artificial Neural Network (ANN) [1]. A neuron is the fundamental building block of nervous system that performs computational and communication function. The ANN is a functional imitation of simplified model of the biological neurons and their goal is to produce intelligent data evaluation techniques like pattern recognition, classification and generalization by using simple, distributed and robust processing units called artificial neurons or processing elements [2]. The artificial neurons were designed to mimic the first-order characteristics of biological neurons. The intelligence of ANN and its capability to solve hard problems emerges from the high degree of connectivity that gives neurons its high computational power (processing capability) through its massive parallel-distributed structure or architecture, each neuron of which performs only very limited operation. Even though individual neuron works very slowly, they can still quickly find a solution by working in parallel.

A major advantage of ANN approach is that the domain knowledge is distributed in the neurons and information processing is carried out in parallel distributed manner [3]. ANNs are highly parallel data processing tools capable of learning functional dependencies of data [2]. Being adaptive units they are able to learn these complex relationships even when no functional model exists. This provides the capability to do "Black Box Modeling" with little or no prior knowledge of the function itself. ANNs are fine-grained parallel implementation of nonlinear static-dynamic

Corresponding Author: Dr. Jamal M. Nazzal, Al Ahliyya Amman University, P.O. Box 19328, Amman, Jordan

systems. They have the ability to properly classify a highly non-linear relationship and once trained. They can classify new data much faster than it would be possible by solving the model analytically from all of the above capabilities of ANN. Our vision of this paper comes to appear in which we think about designing and implementing a neural network system capable of predicting the gross calorific values of: Calcium Carbonate, Organic Carbon and sulfur. Our neural network was learned using already collected experimental data collected from 100 sample of shale. Our proposed neural network was applied on the samples of shale that is collected from Jordan oil shale resources since the EL-lajjun oil-shale deposit has an estimated or over 1.3 billion tones. To achieve the above objectives, we present this paper which is organized form (five) sections. In section (two), we discuss ANN theory and models. Then in section (three), we present ANN application in EL-lujjun oil shale analysis. Section (four) was devoted to comparative study between the results that are obtained from the estimations using Neural Network with that one that is obtained from mathematical formula obtained by Anabtawi and Nazzal [4]. Section (five) deals with conclusion and future works.

ANN THEORY AND MODELS

Different type of Neural Network (NN) has been proposed but all of them have three things in common: the individual neuron, the connection between them (architectures) and the learning algorithm. Each type restricts the kind of connections that are possible. For example, it may specify that if one neuron is connected to another, then the second neuron cannot have another connection towards the first. The type of connection is referred generally to as the architecture or the topology of the neural network [5].

All Neural networks consist of one or more layers of neurons. In large number of NN models, such as perception, linear Associator, Multi-layer feed-forward network with back-propagation (BP) learning, the Boltzmann machine and the gross berg model, the output from the unit from one layer is only allowed to activate neurons in the next layer [6]. However in some models, such as kohenean nets and hopfild model, the signal is allowed to activate neurons in the same layer. In models like Self-Organizing Feature Map (SOFM), the network connects a vector of inputs to a two dimensional grid of output neurons [6-8]. Figure 1 shows a general classification of ANN models

A connection between a pair of neurons has an associated numerical strength called synaptic weight or adaptive coefficient [8]. The strength of

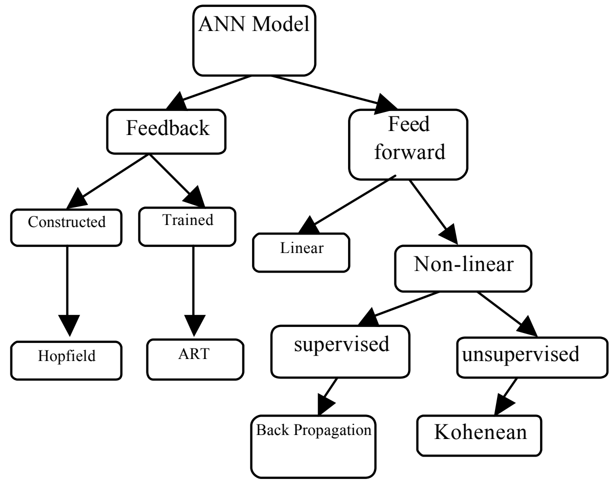

Fig. 1: General classification of ANN models

interconnectivity can be represented as a weight matrix with positive (excitory), negative (inhibitory), or zero (no connection) values [5]. The weight determines from one neuron to another thus coding the knowledge of the network. When the cumulative excitation exceeds the cumulative inhabitation by an amount called threshold (T), typically a value of 40 mv, the neuron fires sending the networks provide instantaneous response. Other networks need time to respond and are characterized by their time domain behavior which we referred to as neural dynamics. The time interval between inputs are applied and neurons give output is called period of latent summation.

A neuron is said to be "trainable" if its threshold and input weights are modifiable. Inputs are presented to the neurons. If the neurons does not give the desired output (determined by us), then it had made a mistake. Then some weight and thresholds have to be changed to compensate for the error. The rules which govern how exactly these changes are to take place is called learning (or training) algorithm. Learning algorithms differ from each other in a way in which the adjustment to synaptic weights of a neuron is formulated.

The weights of the network are incrementally adjusted so as to improve a predefined performance measure over time. The learning process is best viewed as "search" in a multi-dimensional weight space for a solution, which gradually optimizes a prespecified objective function. The NN becomes more knowledgeable about its environment after each iteration of the learning process.

In order for the net to learn, one need to present a number of examples to the net whose attributes are known or are representative for the unknown model [2]. The set of given examples is called the training set or training patterns. After the training period, the network

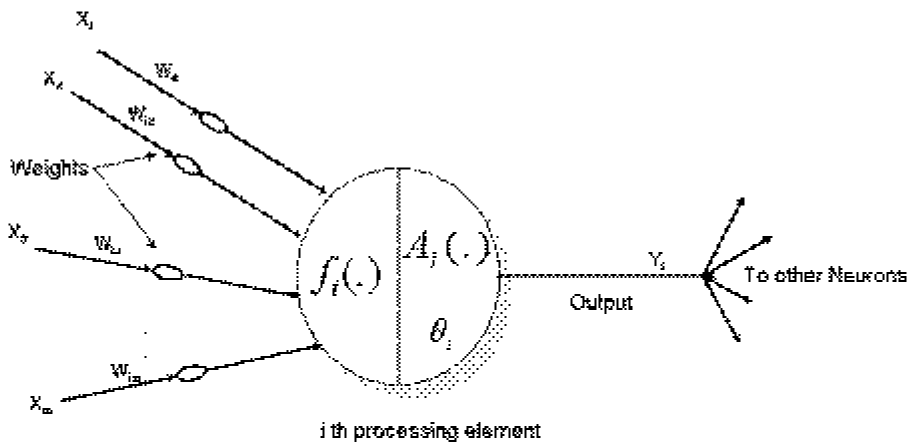

Fig. 2: Artificial neuron

should be able to give correct output for any kind of input. This is called testing. If it was not trained for that input, then it should try to give reasonable output depending on how it was trained. This is called generalization. The actual method of determining the output for a given set of inputs is called the processing algorithm [2]. Different NNs are characterized by different in the architecture, the learning algorithm and the processing algorithm.

Every learning algorithm contains basically a learning rule. There are two main rules available for learning: Hebbiann rule for supervised learning and Delta rule for unsupervised learning. Adaptation of these by simple modifications to suit a particular context generates in many other learning rules. Supervised learning requires the pairing of each input vector with the target vector representing the desired output, together these are called training pair. The desired output represents the optimum action to be performed by the NN [8]. Supervised NN may be feed-forward network such as multi-layer perception (MLP), Functional Link Network (FLN) Radial Basis Function Network (RBFN), Parallel Self-Organizing Hierarchal Neural Network (PSHNN), or a feed-back network such as Hopfield network. Unsupervised learning requires no target vectors for the outputs. The learning algorithm modifies network weights in response to the inputs to produce output vectors that are consistent. Kohonen`s SOFM and Adaptive Resonance Theory (ART) are the examples of unsupervised NN.

Unlike expert systems, NN do not give an explicit set of rules that match the input it receives to the output it is told correct (input-output mapping). This ability to learn by examples is the characteristic of the ANN.

Thus they can modify their behavior in response to their environment. Figure 2 shows the mathematical model of a neuron known as the McCulloch and Pitts neuron. In this model, the i_{th} neuron computes a weighted sum of its inputs and produces an output signal or a zero according to whether this weighted input sum is above or below a certain threshold θi.
The net input to the i_{th} neuron is given by:

$$fi = \sum_{j=1}^{m} W_{ij} X_j(t) - \theta_i \quad (1)$$

The output of the neuron is given by:

$$Y_i\ (t+1) = A(f_i) \quad (2)$$

A popular activation function is known as the logistic sigmoid function (an S-shaped curve) and is defined by:

$$A(f) = \frac{1}{1 + \exp(-f\lambda)} \quad (3)$$

where f is defined as in Eq. (1) and ¸ determines the steepness of the activation function. The choice of λ, depends on the problem and the data being analyzed.

Its output can be fed into other neurons or directly into the environment [2]. The out (output of the neuron) is constructed by talking the weighted sum of the inputs (called NET (net input to a neuron) or combination function (vector-to-scalar function)) transformed by transfer function F (also called activation function (scalar-to-scalar function)). This transfer function introduces nonlinearity into the system. This makes the system so powerful.

APPLICATION OF ANN IN EL-LAJJUM OIL SHALE ANALYSIS

Substantial empirical data exist in disparate data sources concerning product chemicals. However,

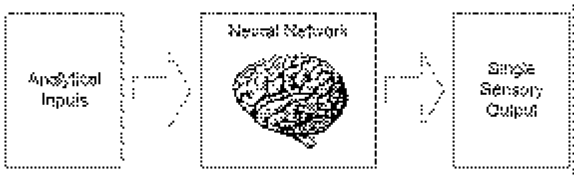

Fig. 3: Simple Intelligent network system

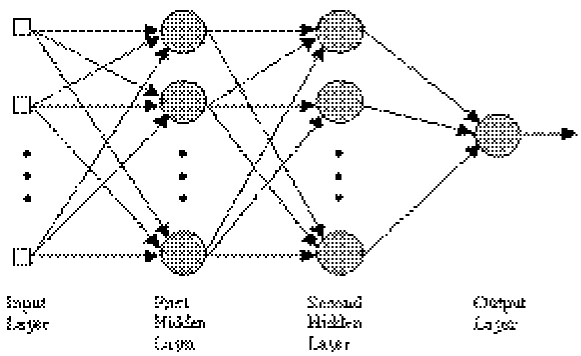

Fig. 4: Implemented Multilayer Perception (MLP)

currently there is no mechanism for linking them to each other. Any such relationships are undoubtedly complex and highly non-linear. So, to identify such relationship, we have focused on using one of modern techniques of soft computing especially neural networks.

Implemented ANN: Attempts to modeling the relationship between the data are restricted to a single quantity and calorific value and focuses on mapping all available inputs through a single neural network as shown in Fig. 3.

The available data consists of 100 records which are divided into 80% (for training) and 20% (for cross validation). The neural network proposed in this paper is based on multilayer perception (MLP) architecture with two hidden layers as shown in Fig. 4.

All data was normalized within the network thereby enabling the results for the various sensory outputs to be compared. Training was terminated automatically when no improvement in the network error was observed during the last five hundred epochs. In all cases, training was carried out fifty times to ensure that a significant mean network error could be calculated for comparison purposes. Prior to each training run, the source data records were randomized to ensure a different training and cross validation data set was presented there by moving any bias. Running of our designed neural network was based on a package solution supplied by Neuro Dimension (www.nd.com).

Training algorithm and parameters: A standard back-propagation algorithm was used to train the feed forward type of neural network. The algorithm is summarized in the steps given below. The mathematical formulations (A), (B), (C), (D), (E) and (T) are described later [10].

Step 1: Set the initial weights and thresholds for all inputs using random numbers between 0.0 and 1.0 and set the number of iterations N to 0.

Step 2: Read inputs and desired output in appropriate form (A).

Step 3: Find the output of each layer and then the final output (B) using the threshold function (T) and increment the number of iterations.

Step 4: Find the error E_l, that is the deviation (C) from the desired output. If $E_l > E_0$ or the number of iterations $N > N_0$, stop processing and repeat the loop a fixed number of times to ensure better learning, else go to step 5 (here E_0 is the maximum permissible error and error and N is the maximum number of iterations permitted).

Step 5: Calculate error functions d_k and d_j for the output layer and other hidden layers using (D).

Step 6: Modify the weights and thresholds using (E). Go to step 3.

These operations are to be performed for every set of input data over which we want to train the neural network. When the training with the first set of inputs is over, we present the new set and desired output for that input set and so on. After all the input sets are entered, the training is repeated with fewer interactions to improve the performance and the fault tolerance of the neural network model [10].

The mathematical formulations used above are as follows:

(T) Threshold function: sigmoid function

$$f(x) = \frac{1}{1+\exp(-(x-xa)/\theta} \qquad (4)$$

where x is the variable value, x_0 is the threshold and θ is the slope of the sigmoid function.

For finite x_0,

$$F(x) \begin{cases} = 0 \text{ as } x \to \text{negative infinity} \\ = 1 \text{ as } x \to \text{positive infinity} \\ = \text{some value between 0 and 1 for any other } x \end{cases} \quad (5)$$

- The inputs are taken in normalized form between 0,1 0.9 using the formula:-

$$x' = \frac{x - MIN}{MAX - MIN} * 0.8 + 0.1 \quad (6)$$

where MAX and MIN are the global maximum and minimum respectively in the input set.

- The output of any layer j is defined as:

$$O_j = f(\Sigma_i w_{ij} X_i - X_0) \quad (7)$$

where f() is the threshold function described above, Wij is the weight between node *i* of the input and node *j* of the next layer, xi is the input to node I and X_0 is the threshold set by the auxiliary node connection between the auxiliary node and the node in which we are interested.

- The mean error criterion is used to measure the deviation of the neural network's output from the desired output:

$$\text{Error} = \frac{1}{2} \Sigma_i \, (\text{Outputi} - \text{DesiredOutput}_i)^2 \quad (8)$$

- Error functions

The error for the output layer: for the K_{th} node in the output layer the error function used is d_k, which is represented as:

$$d_K = (t_k - O_k)\, O_k \,(1 - O_k) \quad (9)$$

where O_k is the actual output and tk the desired output.

The error for the other layers: since the output of the hidden layer is not known, a different error function is used to evaluate the error for each hidden layer node as given below:

$$dj = O_j\,(1 - O_j)\, \Sigma\, W_{jk} .\, d_k \quad (10)$$

where Oj is the output of node j, sk the error of node k in the output layer and Wjk the weight between node j and node k as described above.

- The weights of the neural net were modified using the error functions and the weight change as follows:

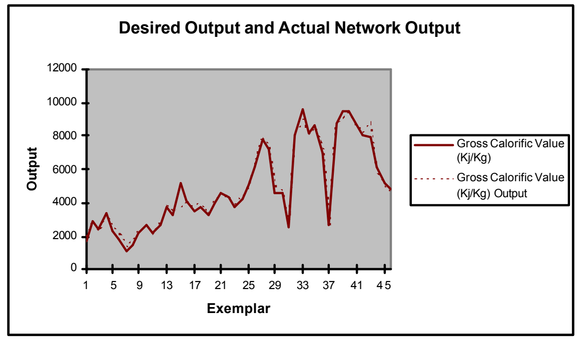

Fig. 5: ANN testing results

$$W_{ij}\,(\text{new}) = W_{ij}\,(\text{old}) + \eta\, d_j\, i_i + a\,[W_{ij}\,(\text{oild}) - W_{ij}\,(*)] \quad (11)$$

where η and a are acceleration functions and Wij (*) is the value of the weight two steps before the current one.

SIMULATION RESULTS OF ANN VERSUS EMPIRICAL FORMULA OF EL-LAJJUN OIL SHALE

In this section, we will focus on some types of comparison between the estimated results obtained from ANN and the corresponding one obtained from empirical formula obtained by Anabtawi and Nazzal [4], in which the effect of bore depth, calcium carbonate, organic Carbon and sulfur content on the calorific value were studied. Results obtained from the empirical formula were well correlated by the following formula:

$$\text{Calorific Value} = 352.44\,(CaCo3)^{-0.066}\,(S)^{0.297}\,(C_{org})^{1.141} \quad (12)$$

With correlation coefficient of 0.983 and with an average standard error of 2.63%.

To obtain the ANN system, measurements of the real calorific values were taken as the values of the training set. During the training, information about the Calcium Carbonate, Organic Carbon and sulfur were provided to the neural networks, contrasting the output achieved with the real value measured and back propagating the error. Figure 5 shows the testing results of the implemented neural network.

In Table 1 summary of testing the implemented ANN, The total training was made with 47 samples.

The results achieved with ANN and the mathematical formula [4] compared with the actual experimental results, are shown in Fig. 6-8.

Figure 6: The results of the calorific value achieved with

Table 1: Summary of testing the implemented ANN

Performance	Gross calorific value (Kj/Kg)
MSE	110652.3766
NMSE	0.018051314
MAE	212.0446838
Min Abs Error	22.83845779
Max Abs Error	1527.321424
r	0.990933855

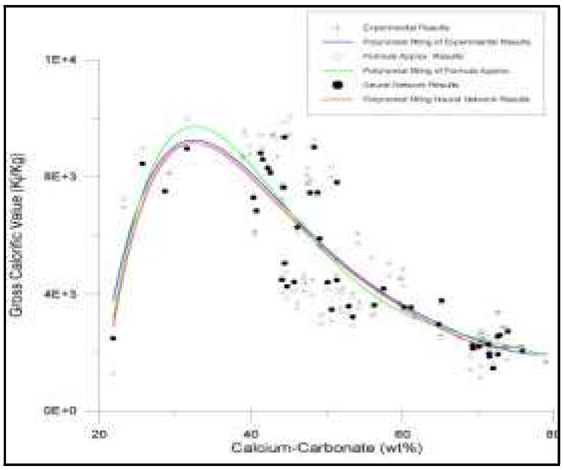

Fig. 6: CaCo3 versus gross calorific value

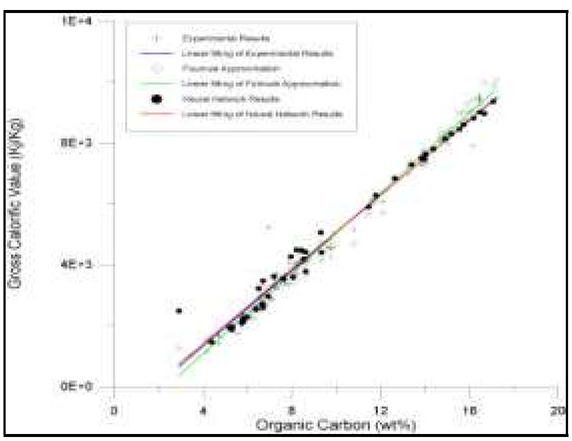

Fig. 7: C_{org} versus gross calorific value

ANN, the mathematical formula and the actual experimental results in relation with the calcium carbonate $CaCO_3$.

Figure 7: The results of the calorific value achieved with ANN, the mathematical formula and the actual experimental results in relation with the Organic Carbon C_{org}.

Figure 8: The results of the calorific value achieved with ANN, the mathematical formula and the actual experimental results in relation with the Sulfur content S.

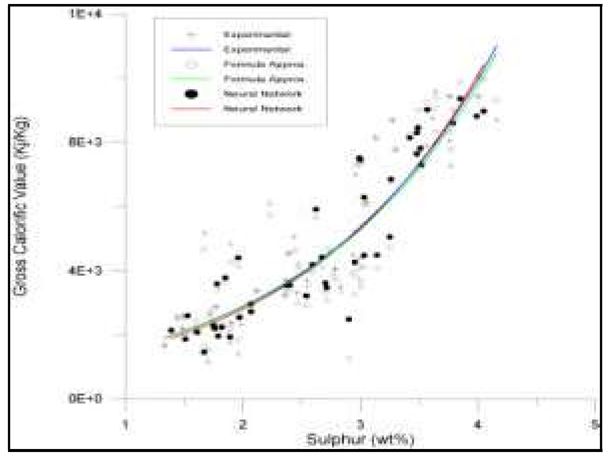

Fig. 8: S versus gross calorific value

From these fgures, it is clear that the estimated results from ANN is approximately 99.7% of the real or/and the actual experimental values of the calorific value; while the empirical formula gives o/p of 97% of the actual value. So, it is better to use ANN in estimating the calorific values specially when these is a relation between this dependent parameter and the other three independent parameters, Calcium Carbonate, Organic Carbon and sulfur content.

From the comparison between the Gross calorific value estimated by the neural network and the mathematical formulas, against the real measurements, we can notice that:

The utilization of the simple mathematical formulas does not provide a satisfactory result. The resulting curve presents values well below the actual real measurements.

The neural network results are more realistic in the sense that the average square error is much smaller compared to the mathematical formula case where the average square error value is used as the threshold to determine the end of the training phase, which also affects the performance of the neural network in the determination of the gross calorific value.

The more sets of values included in the training phase resulting in a better estimation of the neural network.

The achieved results varied according to the number of hidden units in the neural networks. In some cases, the units of-2 and 4 produce a no satisfactory results.

CONCLUSIONS AND FUTURE WORK

Oil shale of the El-lajjun deposit in Jordan was chemically analyzed using various Techniques. The oil was found to consist of: Organic matter, biogenic apatite, detrital clay mineral and quartz and calcite. This paper was intended in studying the effect of composition of El-lajjum oil shale on its calorific value using a new approach called ANN and comparing between the O/P of this ANN with a mathematical approach previously used. From the facts exposed formerly, we conclude that:

The Mathematical formula used has to be complex enough and consider the necessary variables in order to achieve a satisfactory estimation. Since the complexity of the mathematical model could turn development time extremely long which make this solution is far from being a cost-effective one.

The usage of neural network results in a good approximation to reality. Nevertheless, it requires a careful training and test with different configurations which should take into consideration: long training phase and CPU consumption while monitoring.

Finally, even the though both methods (the mathematical and the neural) can be improved, in their simplest expression the back propagation neural network gives a better estimation than the classic mathematical formula.

As a future work, we recommend to use some type of integration of AU based systems and neural networks which may address the following: the possibility of exchange of knowledge between an AI based system and neural network, a knowledge based system learning from neural network performance, division of knowledge between neural network and knowledge based systems and finally creation of learning rules for AI based system from neural networks and vica versa.

REFERENCES

1. Manjaree Pandit, Laxmi Srivastava and Jaydev Sharma, 2003. Fast voltage Contingency selection using fuzzy self-organizing Hierarchical Neural Netwrok. IEEE Trans. On Power systems, 18: 657-664.
2. Task force 38-06-06 of study committee 38, 1995. Artificial Neural Networks for Power Systems. Electra No.159, pp: 78-101.
3. Narendranath Udapa A., D. Thukaram and K. Parthasarathy, 1997. An ANN based Approach for voltage stability Assessment. International conf. on Computer Applications in electrical engineering recent advances, pp: 666-670.
4. Anabtawi, M.Z. and J.M. Nazzal, 1994. Effect of El-Lajjin Oil Shale on its Calorific value. Journal of Testing and Evaluation, JTEVA, 22 (2): 175-178.
5. Nitin Malik, 2005. Artificial Neural Networks and their Applications. National conference on Unearthing Technological Developments & their Transfer for serving Masses, GLA ITM, Mathura, India 17-18.
6. Dillo, T.S., 1991. Artificial Neural Network Applications to power systems and their relationships to symbolic methods. Electrical Power && Energy Systems, 13: 66-72.
7. Khanna, T., 1990. Foundations of Neural Networks. Addison-Wesley, Reading, MA.
8. Free Man, J.A. and D.M. S.K. Apura, 1991. Neural Networks: Algorithms, Applications and programming Techniques, Addison-Wesley, Reading, MA.
9. Srivastava, L., S.N. Singh and J. sharma, 1997. ANN Applications in power systems an overview and key Issues. International conf. on computer Applications in electrical engineering, recent advances, pp: 397-403.
10. Yegnanarayana, B., 1999. Artificial Neural Network, PH, New Delhi.
11. Karla, P.K. *et al*., 1992. Possible Applications of Neural Networks to power systems Operation and Control. Electric Power Systems Research, 25: 83-90.