Article

A Novel Computational Intelligence Approach for Coal Consumption Forecasting in Iran

Mahdis sadat Jalaee [1], Alireza Shakibaei [1], Amin GhasemiNejad [1], Sayyed Abdolmajid Jalaee [1] and Reza Derakhshani [2,3,*]

[1] Department of Economics, Faculty of Management and Economics, Shahid Bahonar University of Kerman, Kerman 76169-13439, Iran; mahdissadat@gmail.com (M.s.J.); ashakibai@uk.ac.ir (A.S.); ghaseminejad@aem.uk.ac.ir (A.G.); jalaee@uk.ac.ir (S.A.J.)

[2] Department of Geology, Shahid Bahonar University of Kerman, Kerman 76169-13439, Iran

[3] Department of Earth Sciences, Utrecht University, 3584 CB Utrecht, The Netherlands

* Correspondence: r.derakhshani@uu.nl

Abstract: Coal as a fossil and non-renewable fuel is one of the most valuable energy minerals in the world with the largest volume reserves. Artificial neural networks (ANN), despite being one of the highest breakthroughs in the field of computational intelligence, has some significant disadvantages, such as slow training, susceptibility to falling into a local optimal points, sensitivity of initial weights, and bias. To overcome these shortcomings, this study presents an improved ANN structure, that is optimized by a proposed hybrid method. The aim of this study is to propose a novel hybrid method for predicting coal consumption in Iran based on socio-economic variables using the bat and grey wolf optimization algorithm with an artificial neural network (BGWAN). For this purpose, data from 1981 to 2019 have been used for modelling and testing the method. The available data are partly used to find the optimal or near-optimal values of the weighting parameters (1980–2014) and partly to test the model (2015–2019). The performance of the BGWAN is evaluated by mean squared error (MSE), mean absolute error (MAE), root mean squared error (RMSE), standard deviation error (STD), and correlation coefficient (R^2) between the output of the method and the actual dataset. The result of this study showed that BGWAN performance was excellent and proved its efficiency as a useful and reliable tool for monitoring coal consumption or energy demand in Iran.

Keywords: coal consumption; computational intelligence; optimization; socio-economic variables

1. Introduction

Rising economic growth in developing countries and continued growth in industrialized countries have increased energy demand. Naturally, various energy sources are used in power plants, which can include fossil fuels (oil, gas, coal) or new energy sources (solar, wind, geothermal, etc.). The increasing dependence on energy has made this sector more interactive with other economic sectors and has made the pace of economic growth and development dependent on energy consumption.

Coal, as one of the main sources of energy, is a non-renewable fossil fuel that has been used since ancient times. Given that most of the first-grade coals have already been mined, the coal that remains for future use is mostly low-grade with high ash and moisture content. In addition, environmental issues related to its extraction, processing, and combustion, threaten the sustainability of coal use. Today, environmental issues, including the factors that have led to climate change through the emission of carbon dioxide, are of global concern, and this is one of the serious problems that coal will face in the future and threaten its sustainability [1,2]. Although coal will continue to be a major player in the global energy spectrum for at least the next two to three decades, coal's sustainable future still depends on reducing its pollution.

Iran has an average level of coal reserves. This valuable fuel is used in many industries, including power plants, steel plants, cement production, chemical and pharmaceutical industries, ammonia gas recovery to produce chemical fertilizers, paper making, and liquid fuel as the main element in production. The lack of coal-fired power plants in Iran reduces the consumption of thermal coal and just a part of this is exported. On the other hand, due to the consumption of coking coal concentrate in steel plants, the shortage of this product in Iran, and the difference in the quality of domestic and foreign coking coal, the tendency to import this fossil fuel has increased. Lack of advanced equipment has reduced efficiency and increased costs, and the lack of a proper strategy in the mining sector has reduced the attractiveness for investors, especially in coal mines. As a result, the cost of domestic coal has increased compared to foreign coal. This has reduced Iran's competitiveness with producers in the global market.

To manage the energy supplement, it is necessary to understand the factors that affect the amount of energy needed in a region. Since the energy demand procedure and the factors that influence it follow a hazy and complicated pattern, efficient tools are needed for efficient use of energy [3]. So, it is necessary to determine effective tools to determine energy demands precisely.

Non-classical approaches have been prolonged to detect and predict complex system-based problems. Despite numerous methods for predicting natural phenomena all over the world, accurately forecasting the events is still difficult. Several non-linear systems exist in the real world, some with dynamic behaviours based on their current state. For such an assessment, different statistical demonstration approaches have been used based on socioeconomic indicators. These models can be appropriately forecasted considering birds, fireflies, bats, and other living organisms utilizing intelligent optimization techniques.

It is believed that artificial neural network (ANN) is one of the highest breakthroughs in the field of computational intelligence. Its application is supported in various fields, which assist in taking further steps to realize the dream of machines capable of thinking and learning in a similar way to the human brain. To model ANN, biological nervous systems have been used (the human brain). Despite processing information, compared to the human brain, it is relatively simple and easy to operate. The ANNs are a model for processing the information created by the imitation of biological neural networks such as the human brain. It should be noted that the novel structure for the information processing system is a critical element of this model, comprising numerous elements (neurons) with robust internal connections, working towards solving some definite problems.

Such neuron/node layers comprise a neural net, with a structure starting with the first layer as the input and ending by the end layer at the output. The layers existing within these two are called hidden layers. An activation value for each node is determined from the previous layers nodes' activation values. The factor is determined by the connection weights or parameters, where each node in the former layer has a role in the subsequent layer nodes' value. Determining the ultimate values for these parameters is an optimization problem, which should be stated.

There have been numerous attempts to determine the connection weights' optimum value within a Multi-layer Perceptron neural network. Generally, the trainers are classified into two groups of gradient-based and stochastic search algorithms. A backpropagation [4] algorithm and its variants are included in gradient-based training algorithms representing the eye-catching advantages of the convergence's high speed. However, they tend to rely heavily on the initial solution, which possibly causes them to become trapped in a local optimal. Nevertheless, stochastic algorithms have gained popularity since they can solve such disadvantages. In these algorithms, the primary solution is made or selected randomly, therefore, they do not end up in local optimal entrapments.

Another method for solving optimization problems is meta-heuristic algorithms related to the class of stochastic optimization algorithms. They aim to solve optimization problems in a "fairly good" way, though they do not always achieve the best results. Nature-oriented algorithms are a kind of meta-heuristic algorithm, inspired by biological

systems present in the environment, such as the grey wolf optimization, which imitates the hunting pattern and hierarchical structure of wolves, bat algorithm, which imitates the bats' hunting behaviour by utilizing echolocation, and the genetic algorithm, which is based on Darwin's theory of "survival of the fittest" and next generation evolution.

Bio-inspired algorithms were formerly utilized in various fields for addressing application-based optimization problems. Inconsistency with the no-free-lunch theorem, as the definite algorithms present better results for only definite specific applications. No optimization algorithm exists that works for all fields universally. An enhanced version of the Binary Bat algorithm was proposed by Gupta [5] to solve the optimization problem of selecting the features on the white blood cells dataset. Different approaches to nature-inspired algorithms were discussed by Tiwari and Bansal [6] on industrial applications. Suguna et al. [7] performed medical image processing (classification and segmentation), using bio-inspired algorithms, such as Lion Optimization Algorithm (LOA) and Monkey Search Optimization Algorithm (MSO).

Related Works

The scope of using nature-based algorithms was examined and run to train an MLP neural network. The grey wolf optimizer was similarly employed by Mirjalili [8] and its performance was compared on multiple benchmark data-sets, versus some other recognized evolutionary trainers, such as genetic algorithm (GA), ant colony optimization (ACO), population-based incremental learning (PBIL), evolution strategy (ES) and particle swarm optimization (PSO). Aljarah et al. [9], working along similar lines to solve this optimization problem, used whale optimization algorithm, which is another nature-based algorithm. Various variants of bat-inspired algorithms were employed by Jaddi et al. [10] as a new method to represent optimizing both the ANNs framework and connecting weights. The application of grey wolf optimization (GWO) was extended by Faris et al. [11], providing a new hybrid encoding scheme to automatically select hidden neurons and weights within a neural network. A multi-verse optimizer was tested and compared with other present meta-heuristic evolutionary algorithms as feed-forward neural network trainers for binary classification problems [12].

A bio-inspired firefly algorithm was analysed by Nandy et al. [13] to train a back-propagation neural networks. Ojha et al. [14] investigated the progress within the meta-heuristic methods utilized for designing feed-forward neural networks over the years. An emergent stochastic training algorithm was studied by Heidari et al. [15] and proposed to train an MLP.

Such optimized neural networks are applied in different fields and over various data sets to solve day-to-day problems. A PSO trained ANN was employed by Blum and Socha [16] to detect the possibility of failures in a multi-storeyed reinforced concrete building structure. Neural network optimization was used to analyse rain patterns and discover wet and drought year alarms by Valipour [17]. The use of artificial neural networks was reviewed by Villarrubia et al. [18] to approximate an objective function and solve optimization problems. An innovative Elman neural network was suggested by Ruiz et al. [19] to estimate energy use in public buildings and obtain energy savings through a genetic algorithm and optimize the connection weights. An ANN model was compared by Yang et al. [20], both with and without a genetic algorithm to predict and parametrically optimize the ORC system. Numerous bio-inspired algorithms exist, each with its disadvantages and advantages. The multiple meta-heuristic algorithms fusion was introduced via hybridization to highlight the best in each one and cover each lacunae.

Several studies have suggested various models for managing energy demand policy using different methods. Particle swarm optimization (PSO) energy demand models were developed by Unler in Turkey to assess energy demands based on economic indicators [21]. Using the genetic algorithm (GA), Canyurt and Ozturk proposed models for estimating Turkey's fossil fuel consumption [22]. Toksari used ant colony optimization models to estimate Turkey's energy demand [23]. To predict monthly electrical energy usage;

Azadeh et al. used an artificial neural network (ANN) [24]. Azadeh et al. introduced and compared the genetic algorithm, ANN, and Fuzzy Regression Algorithm (FRA) to evaluate improvements in electricity use in developing countries in a separate paper [25]. PSO and GA were used by Amjadi et al. to forecast Iran's electricity demand [26]. To estimate China's transportation energy consumption, Zhang et al. used the Partial Least-Squares Regression (PLSR) approach [27]. In Iran, Assareh et al. established a PSO and GA program to estimate oil demand [28]. The bees algorithm, BA, was used by Behrang et al. to predict Iran's total energy demand [29]. Behrang et al. used PSO and GA to predict electricity demand in Iran's industrial sector [30]. Bahmani et al. forecasted Iran's oil consumption using the BA with ANN [31]. Horák et al. [32], applied an artificial neural network to show how financial and nonfinancial indicators play important roles in the explanation of corporate health and support a company's possibilities of reaching relevant performance levels. Vochozka et al. [33], used artificial neural networks to create a methodology for the prediction of a company failure. The study of [34] investigated the influence of the international price of oil on the value of the EUR/USD exchange rate by using the approach of neural networks. In another study Vochozka et al. [35], used (Long Short-Term Memory) for predicting oil prices.

Vrbka et al. [36], used neural networks to determine value-based drivers for SMEs operating in the rural areas of the Czech Republic. Taner et al. [37–40] carried out optimization, energy-exergy analysis, and techno-economic and cost analysis of a drying plant, model of a sugar factory, and turbine power plant of sugar in Turkey, respectively. In another study, Taner [41] carried out an experimental optimization of a PEM fuel cell to enhance the efficiency and development of the simulations and modelling of a PEM fuel cell. Topal et al. [42,43] presented case studies of the poultry industry and power plants in Turkey, based on the application of trigeneration and exergy analysis of the plant.

This paper aims to present a novel hybrid computational intelligence approach for forecasting coal consumption. The hybrid Grey Wolf-Bat Optimization Algorithm with ANN is used to build a demand prediction model for predicting coal consumption in Iran based on socioeconomic indicators.

2. Materials and Methods

2.1. Bat Algorithm (BA)

The bat algorithm, which is an evolutionary algorithm inspired by the behaviour of natural bats, was introduced by Yang [44] and is used to solve various problems. The bat algorithm (BA) works based on the sound echo and the position of the bat. The bat produces sound pulses and receives their echoes. The bats can detect prey from the barriers based on echoes as the sound echoes produced for different objects vary; their corners use this method to detect positions. BA works based on the following three basic steps:

1. All bats can make a sound and receive an echo. They can distinguish the food source from the barrier according to this ability.
2. The bats fly randomly and have a vi velocity in the xi position and a constant frequency of f_{min} and wavelength λ during flight. They can produce sound pulses between 0 and 1.
3. The loudness of bats can differ from a large positive value A_0 to a small positive value A_{min}.

One can assume that the value of frequency f can differ between f_{min} and f_{max} and the corresponding wavelength from λ $_{min}$ to λ $_{max}$. Wavelength amplitude can vary too. The wavelength should be selected based on the problem search space [45]. X_{best} in BA is considered as the global answer to the problem or the best position of the bats. Equations (1)–(3) show the updated frequency, velocity, and position of the bats, respectively [46].

$$f_i = f_{min} + (f_{min} - f_{max})\beta \tag{1}$$

$$v_i^t = v_i^{t-1} + \left(x_i^{t-1} - x_{best}\right)f_i \tag{2}$$

$$x_i^t = x_i^{t-1} + v_i^t \tag{3}$$

f_i is the frequency of the ith bat, v_i^t new velocity of the ith bat, v_i^{t-1} the previous velocity of the ith bat, x_i^t the new position of the ith bat, x_i^{t-1} the previous position of the ith bat, and β a random vector with an array between 0 and 1. First, a random number between $f_{\min}$ and $f_{\max}$ is assigned to each bat. The bat velocity and position are then updated based on Equations (2) and (3). Then a random number is generated. The local search is performed using the production of a random step based on Equation (4) if the pulse generation rate is less than this random number:

$$x_i^t = x_i^{t-1} + \varepsilon A^t \tag{4}$$

It is a random number and is the average volume. The volume decreases and the pulse rate increases when bats find prey. Volume and pulse output rate are calculated as Equation (5):

$$r_1^{t+1} = r_i^0[1 - \exp(-\gamma t)]A_i^{t+1} = \alpha A_i^t, t - 1, \ldots, T \tag{5}$$

γ and α are constant parameters.

2.2. Grey Wolf Optimization Algorithm (GWO)

There are always signs around for modelling to solve and optimize complex problems. The paper will examine an algorithm that can be used to solve optimization problems by modelling a natural organization. GWO imitates the grey wolf organization for hunting in the wild. The algorithm was presented in 2014 by Mirjalili [47]. Similar to PSO and ACO algorithms, this algorithm is among the collective intelligence algorithms and uses only one (hybrid) operator to determine the position of wolves in the problem-solving space.

Grey wolves are creatures living in a semi-democratic way and the status of each wolf is clear in their community. In this community, the leader wolf, her successors, and other wolves are known too. Grey wolves live and hunt in a pack. In each pack of grey wolves, there are on average between 7 and 12 wolves. These 4 groups are the most dangerous predators. The grey wolves first loop around the prey and surround it to hunt and begin to exhaust the prey by tightening the siege. Then, in turn, at the order of the leader wolf, they attack and finally kill the prey. The following is an example of a wolf attack in the real world.

As already stated, grey wolves live in a hierarchical (very strict) manner. The figure below is the structure of the grey wolf hierarchy. Known as the group leader, the Alpha pair makes decisions about hunting, sleeping location, waking time, and so on. Alpha decisions apply to the whole pack. However, there is a kind of democratic behaviour as well. The second rank in the hierarchy belongs to Beta wolves. Beta wolves help the alpha in group decisions and other activities. These wolves are the best candidates for alpha when the alpha is very old or dies.

The wolves with the lowest status are the Omega wolves. This group of wolves has the role of pre-death in the pack. They have to follow all other wolves and are the last wolves to eat. It seems that Omegas have a low significance in the category, yet it has sometimes been observed that if the Omegas are lost, the whole group suffers from problems and civil wars. The wolves not stated in the above hierarchy are called Delta wolves. Delta wolves are commanded by Alpha and Beta but are superior to Omega. Accordingly, the GWO is explained, and following mathematical equations are used. As already stated in the previous sections, grey wolves surround their prey during the hunting process. The following equations are used to model the hunting mechanism:

$$\vec{D} = \left| \vec{C} . \vec{X_p}(t) - \vec{X}(t) \right| \tag{6}$$

$$\vec{X}(t+1) = \vec{X_p}(t) - \vec{A} . \vec{D} \tag{7}$$

$$\vec{A} = 2\vec{a} . \vec{r_1} - \vec{a} \tag{8}$$

$$\vec{C} = 2.\vec{r_2} \tag{9}$$

In the above equations, t is the algorithm iteration.

Vectors A and C are the prey location vector coefficients and X is the grey wolf location vector.

$\vec{A}$ decreases linearly from 2 to 0 during repetitions. r_1 and r_2 are random vectors in the range [1/0].
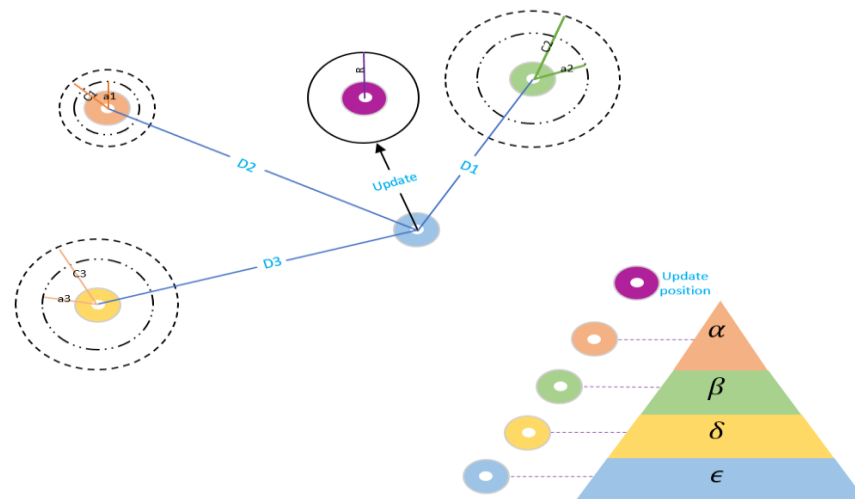
For a two-dimensional problem, the vectors will be as follows (position the vectors in two dimensions and their next position) [47].

Mathematical equations of GWO considering the hierarchy in Figure 1:

$$\vec{D}_\propto = \left| \vec{C}_1. \vec{X}_\propto - \vec{X} \right|, \vec{D}_\beta = \left| \vec{C}_2. \vec{X}_\beta - \vec{X} \right|, \vec{D}_\delta = \left| \vec{C}_3. \vec{X}_\delta - \vec{X} \right| \tag{10}$$

$$\vec{X}_1 = \vec{X}_\propto - \vec{A}_1.\left(\vec{D}_\propto \right), \vec{X}_2 = \vec{X}_\beta - \vec{A}_2.\left(\vec{D}_\beta \right), \vec{X}_3 = \vec{X}_\delta - \vec{A}_3.\left(\vec{D}_\delta \right) \tag{11}$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{12}$$

Figure 1. Hierarchical ordering of wolves.

2.3. Artificial Neural Network (ANN)

An artificial neural network, simply called a neural network, is a computing system inspired by the human brain. This inspiration has developed an artificial neural network to a set of simple and strong computing methods. The characteristics including the power of learning and adaptation, the possibility of underlying information processing, the power of generalization, error tolerance, and uniformity of analysis and design give a strong capability of processing to neural networks and enable them to successfully conduct activities, such as the estimation of nonlinear complex functions and pattern identification and classification. The structure of the neural network is usually a multi-layered network or graph with simple connections among layers. There are one or more computational units in each layer called a node or artificial neuron, which is in fact a simple model of neurons of the human brain. The role of neurons in the neural network is data processing and this is done by a mathematical processor which is an activation function in the artificial neural network. The activation function or operational function considers the problem which is going to be solved by a neural network and selected by the designer. An artificial neural network in the simplest way has an input layer and output layer. But the network with hidden layers has more capabilities. It can be proved that a feed-forward neural network with a hidden layer, sigmoid activation function in the hidden layer, linear activation function

in the output layer, and enough neurons in the hidden layer can estimate every function with sufficient accuracy. The network acts as an input-output system and uses the value of input neurons to compute the value of output neurons. Each neuron connection in different layers has its weights and the network learns the pattern in input and output variables by adjusting these weights during the training phase. According to the learning methods, neural networks can generally be classified into two categories: supervised learning and unsupervised learning [48]. In supervised learning, the correct response (output) for each input pattern is given to the network. Weights are determined in a way that the network creates responses that are close to the correct responses. During repeated weight correction, a network is trained. Repeating the learning process, the network identifies the correct value of weights and lowers the possibility of errors. For a set of identified inputs, the error means the difference between the actual value and the network output. In unsupervised learning, there is no need for an actual response for each input pattern in the training set. In this model of learning, the network discovers the basic structure of the data and the correlations between the patterns in the data and organizes the patterns into appropriate classes. According to the structure, artificial neural networks are divided into two types of feed-forward networks and recurrent networks. The feed-forward network, in which there is no feedback loop, and the recurrent network, which has a feedback loop and neurons in each layer, gives the data from previous and post layers.

The following hypotheses are considered in artificial neural networks:

1. Data processing takes place in simple units called neurons. These artificial neurons (or simply neurons) are a simple model of brain neurons.
2. The data are exchanged in communications between neurons.
3. Each communicative way between neurons has weight.
4. Each neuron uses an operational function (usually non-linear), to apply it to the neuron input (the weighted data) and create a definite output.

The neural network is identified through the communicative pattern between different layers of the network, the number of neurons, the number of layers, the learning algorithm, the operational function of the neuron. But there is no general rule about the standard size of these components for each network and in each application. In most cases, it is an innovative method in which the multi-layered networks with a different number of neurons in each layer, are trained with different learning rates and various activation functions, and then the best network is elected. In the learning phase, the network is trained through weight adjustment to be able to predict or classify target outputs based on a set of inputs.

In Figure 2, an artificial neural network (ANN) is demonstrated with a hidden layer, containing certain weights linking the layers. Through the following steps, the output values will be calculated. Firstly, as follows, weights sum is calculated:

$$S_j = \sum_{i=1}^{n} w_{ij} I_i + \beta_i \tag{13}$$

In the above-mentioned formula, the input variable is I_i, the weight existing between neuron j and the input variable I_i is w_{ij}, and β_i is for the input variable.

Second, using an activation function, values of neurons output in the hidden layers are generated using the weighted summation received values (Equation (13)). A sigmoid function is a popular candidate for such a function, as seen below:

$$f_j(I) = \frac{1}{1 + e^{-S_j}} \tag{14}$$

where f_j signifies the sigmoid function regarding neuron j and S_j refers to the sum of weights.

Lastly, the neuron j output is calculated as seen below:

$$O_j = \sum_{i=1}^{k} w_{ij} f_j + \beta_j \tag{15}$$

where O_j refers to the neuron output j, w_{ij} refers to the weight existing between the output variable neuron j and O_i, f_j refers to the neuron j activation function, and β_i is the bias term for the output variable [39].
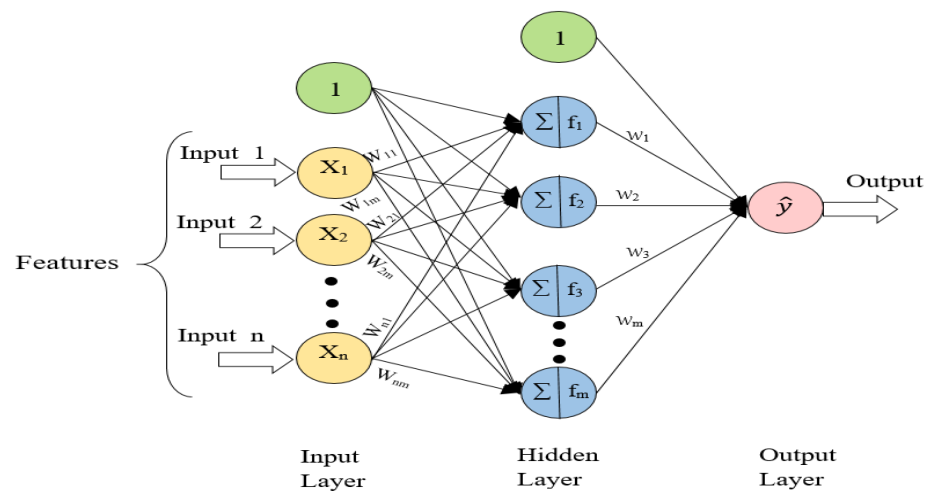

Figure 2. Multi-layer perceptron.

2.4. Proposed Method

To deal with the connection weight optimization problem for MLP training, the presented Hybrid Wolf-Bat Optimization (HWBO) Algorithm combines two well-known nature-inspired algorithms, bat algorithm (BA) and grey wolf optimization (GWO) algorithm. Figure 3 represents the flowchart of the proposed approach. HWBO combines the best features of the two algorithms above: GWO's excellent exploration skills allow for a thorough examination of the whole search space, whereas bat's high exploitative feature allows for better local search capabilities. The MLP network's mean square error is used as the fitness function for both BA and GWO, with the intention of lowering this value. Each hybrid algorithm's iteration consists of one run of a GWO algorithm and an individual run of a BA. The GWO translates to BA as an output of its best three wolves, i.e., α, β, and δ. When the bat population is initialized, three bat posts with the best positions achieved from the GWO are initialized, the rest are randomly initialized. The initialization of three α, β, and δ positions ensures that the whole search area, which was originally lacking in BA, will be covered with a view to finding more optimal solutions [49].

After that, the BA runs as normal, changing the pulse rate and loudness according to equations, and the Global Best solution is revised if it has changed. The α wolf's position is reset to the Global Best outcome from the preceding iteration for the next iteration, and the process begins. Other hybrid algorithm methods were also checked by passing either α or both α and β to BA, while retaining the rest of the configuration the same. When all three, α, β, and δ, were used and moved on to BA, the best results were achieved. Based on these approximations and idealization, the basic steps of the BGWAN can be summarized as the pseudo-code shown in Figure 4.
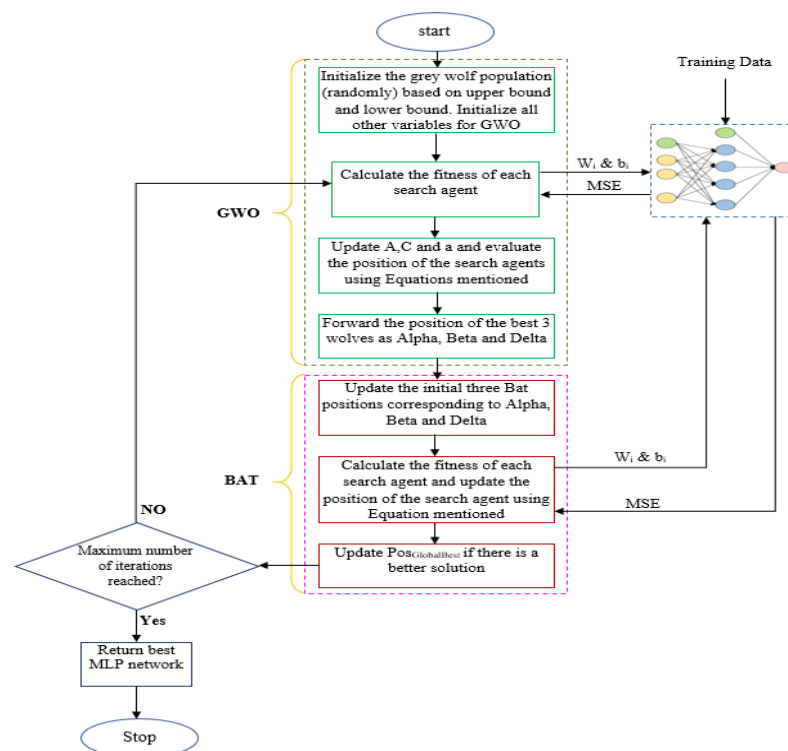
Figure 3. Flowchart of BGWAN.

Pseudo code of BGWAN

1: Initialize grey wolf population (m), let $\overrightarrow{X_i}$ (for i = 1, 2,..., m) be the position of wolves, which is
   randomly initialized based on upper bound *ub* and lower bound *lb* of variables.
2: Initialize A, C and a for wolves.
3: Initialize frequency of pulses f_i, velocity v_i, Pulse emission rate r_i, Loudness A_i, γ (pulse emission rate
   constant) and α (Loudness constant) for bats
4: Assign the MSE of MLP Network as the fitness function, with the objective of the algorithm is to minimize this value.
5: Calculate the fitness value for each search agent as per the fitness function
   Set $\overrightarrow{X_\alpha}, \overrightarrow{X_\beta}$, and $\overrightarrow{X_\delta}$
   Set $fitness_{Alpha}$, $fitness_{Beta}$, and $fitness_{Delta}$ where Alpha, Beta, and Delta are, respectively, the top3
   best wolves.
6: **while** *(t < Max Iterations)* **do**
7: **for** each search agent **do**
8:        Evaluate the value of variables $\overrightarrow{X_1}, \overrightarrow{X_2}$ and $\overrightarrow{X_3}$ using Equations *(11)*,
9:        Update $\overrightarrow{X_i}$ using Equation *(12)*.
10:     **end for**
11:     Update A, C and a.
12:     **Repeat** Step 4.
13:        Initialize bat population such that first three bats are initialized with $\overrightarrow{X_\alpha}, \overrightarrow{X_\beta}$, and $\overrightarrow{X_\delta}$
           while remaining (m-3) bats are initialized randomly based on *ub* and *lb* of variables
14:     For each search agent evaluate the fitness, and select the best agent.
15:     Set $\overrightarrow{X_{best}}$ and $fitness_{Best}$.
16:     **for** each search agent **do**
17:        Render new solutions, using Equations *(1)*, *(2)*, and *(3)*.
18:        **if** *(random > r_i)* **then**
19:           Render a local solution around one of the selected best solution
20:        **end if**
21:        **if** *((random < A_i) & ($fitness_{new}$ < $fitness_i$))* **then**
22:           Update $\overrightarrow{X_i}$ and $fitness_i$ using $\overrightarrow{X_{new}}$ and $fitness_{new}$, respectively.
23:           Update Ai and r_i using Equations *(4)* and *(5)*.
24:        **end if**
25:        **Repeat** step 14.
26:     **end for**
27:     Update t← t+1.
28: **end while**
29: Return $\overrightarrow{X_{best}}$.
30: Test the best agent on validation set.

Figure 4. Pseudo code of BGWAN.

3. Results

In this paper, a novel hybrid computational intelligence approach for predicting coal consumption in Iran is coded with MATLAB 2019 software. Data on Iran's population, GDP, import, export, and coal consumption were collected during the years 1981–2019 [50]. The input parameters of the BGWAN included population, GDP, import, export, and coal consumption were considered as the output parameter. The data on these parameters were divided into training and testing. Of these data, 70% were used for training, 30% of data for validation and testing. To estimate coal consumption in Iran, it is necessary to normalize the data in the first step. Equation (16) was used to normalize the data:

$$Z_N = (Z_R - Z_{min})/(Z_{max} - Z_{min}) \tag{16}$$

where Z_N shows the normalized data and Z_R represents the original data value.

The performance of the BGWAN method is evaluated with mean squared error (MSE), mean absolute error (MAE), root mean squared error (RMSE), error standard deviation (STD), and correlation coefficient (R^2) between the output of the BGWAN and the actual dataset [46,51,52]. These errors are specified in the forms of Equations (17)–(21).

$$MSE = \frac{1}{39} \sum_{i=1}^{39} (Target_i - output_i)^2 \tag{17}$$

$$RMSE = \sqrt{\frac{1}{39} \sum_{i=1}^{39} (Target_i - output_i)^2} \tag{18}$$

$$MAE = \sqrt{\frac{1}{39} \sum_{i=1}^{39} |Target_i - output_i|} \tag{19}$$

$$STD\ Error = \sqrt{\frac{1}{39} \sum_{i=1}^{39} (Target_i - output_{i)}^2} \tag{20}$$

$$R^2 = \left(\frac{\sum (Target_i - \overline{Target}) \times (output_i - \overline{output})}{\sqrt{\sum (Target_i - \overline{Target})^2 \times \sum (output_i - \overline{output})^2}} \right)^2 \tag{21}$$

Figures 5–7 indicate the best Validation Performance Graph and regression plot between actual and predicted data in the BGWAN method. Table 1 and Figure 8 show the performance evaluation of BGWAN outputs.

Figure 9 and Table 2 for the modelling and the testing data show the performance of the BGWAN method.

Table 3 indicates a comparison of the different models introduced in the introduction and present study.
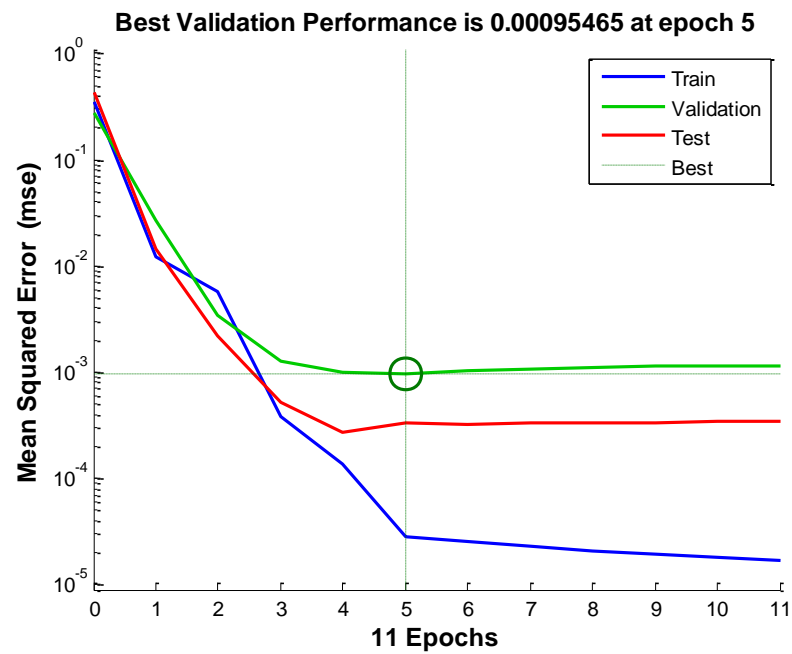
Best Validation Performance is 0.00095465 at epoch 5

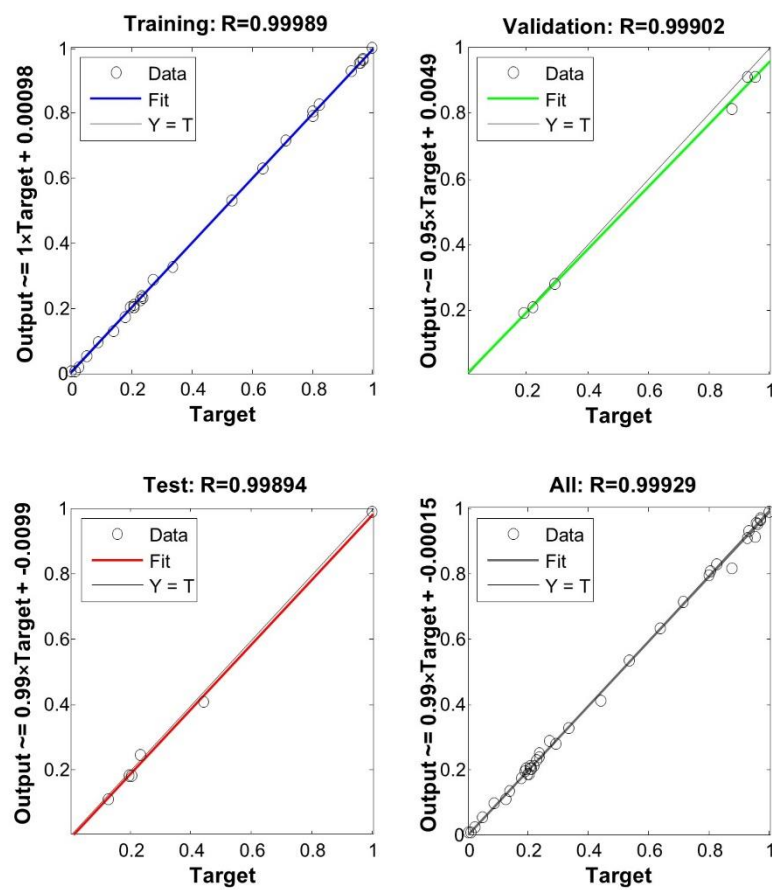**Figure 5.** Validation Performance Graph.
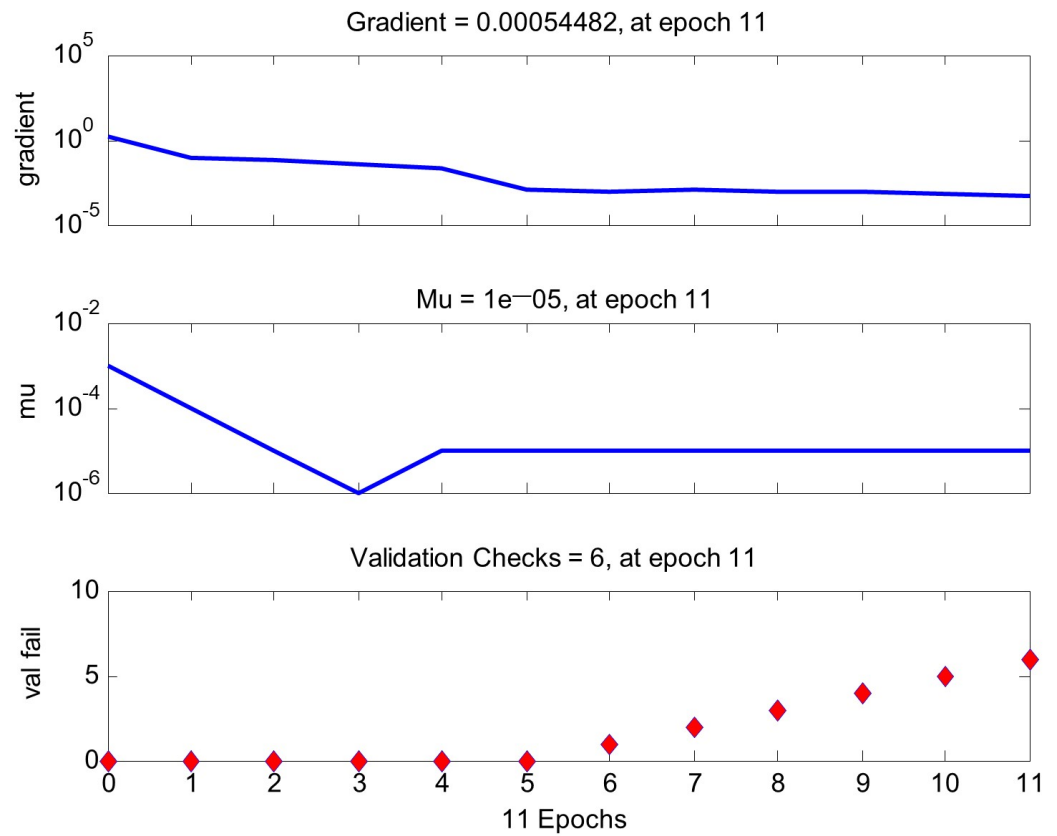

Figure 6. Regression Plot.

Figure 7. Optimized neural network output and model performance criteria for data.

Table 1. BGWAN operation.

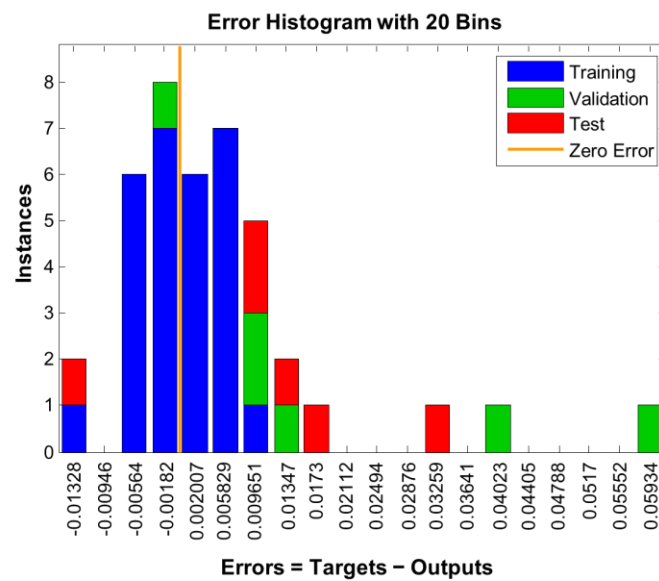| BGWAN Process	Data Usage	MSE	RMSE	MAE	STD Error	R
training	70%	0.00028	0.016	-	-	0.99989
validation	15%	0.00033	0.018	-	-	0.99902
testing	15%	0.00017	0.013	-	-	0.99894
Total	100%	0.0002	0.014	0.011	0.014	0.99929

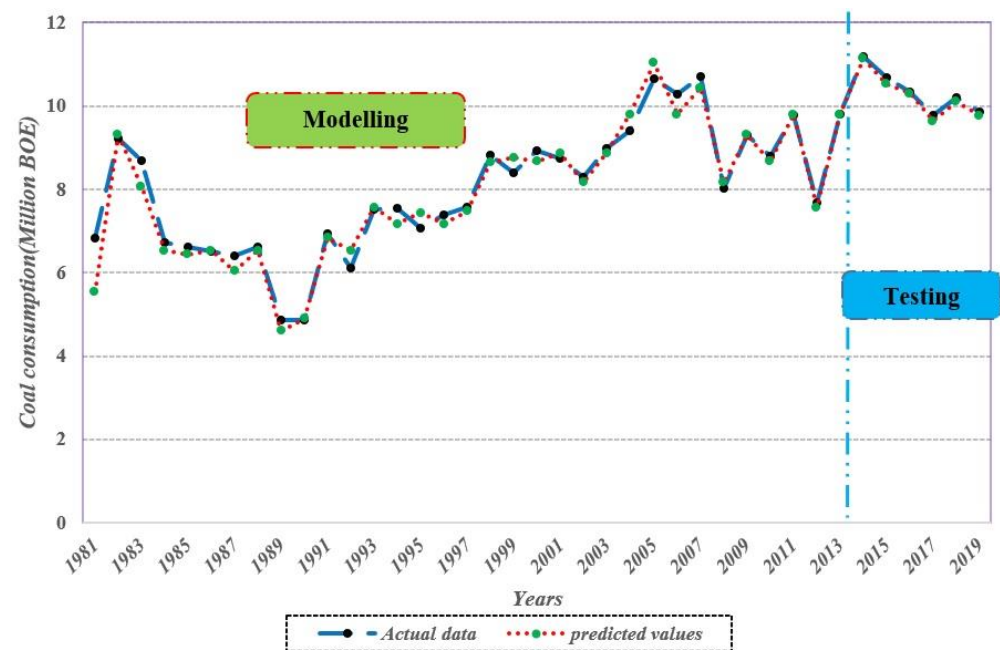**Figure 8.** Error Histogram for BGWAN outputs.

Figure 9. Comparison of Actual coal consumption values and their predicted values with BGWAN.

Table 2. BGWAN operation.

Years	Actual Data	BGWAN Output Predicted	Relative Error
2015	10.68	10.63	0.0041
2016	10.35	10.39	0.0040
2017	9.78	9.76	0.0012
2018	10.19	10.18	0.0015
2019	9.86	9.89	0.0030
Average	-	-	0.0028

Table 3. Comparison of various models introduced in the introduction and present study. Average relative errors are on the testing period of each model.

Source	Method	Target-Country	Average Relative Errors
[19]	Particle Swarm Optimization	Total Energy-Turkey	0.83
[20]	Genetic Algorithm	Oil-Turkey	2.97
	Genetic Algorithm	Natural Gas-Turkey	2.10
	Genetic Algorithm	Coal- Turkey	3.22
[21]	Ant Colony Optimization	Total Energy-Turkey	1.07
[22]	Artificial Neural Networks	Electricity-Iran	1.20
[23]	Genetic Algorithm	Electricity-Iran	1.4
	Artificial Neural Networks	Electricity-Iran	1.56
	Fuzzy regression algorithm	Electricity-Iran	0.82
[24]	Genetic Algorithm	Electricity-Iran	1.36
	Genetic Algorithm	Electricity-Iran	1.51
	Particle Swarm Optimization	Electricity-Iran	3.92
	Particle Swarm Optimization	Electricity-Iran	0.98
[25]	Partial Least Square Regression	Transport Energy-China	2.30

Table 3. *Cont.*

Source	Method	Target-Country	Average Relative Errors
[26]	Genetic Algorithm	Oil-Iran	2.83
	Genetic Algorithm	Oil-Iran	1.72
	Particle Swarm Optimization	Oil-Iran	1.40
	Particle Swarm Optimization	Oil-Iran	1.36
[27]	Bees Algorithm	Total Energy- Iran	1.07
	Bees Algorithm	Total Energy-Iran	1.83
[28]	Genetic Algorithm	Electricity-Iran	1.13
	Genetic Algorithm	Electricity-Iran	1.29
	Particle Swarm Optimization	Electricity-Iran	1.03
	Particle Swarm Optimization	Electricity-Iran	1.69
[29]	Hybrid Bat algorithm with artificial neural network	Oil-Iran	0.0037
Present study	Hybrid Grey Wolf-Bat algorithm with Artificial Neural Network (BGWAN)	Coal-Iran	0.0028

4. Conclusions

A hybrid BGWAN based on the bat algorithm and grey wolf optimization with an artificial neural network is proposed in this study for enhancing the Iran Coal Consumption via investigating the population, GDP, import, export, and coal consumption. Using error standard deviation (STD), mean absolute error (MAE), mean squared error (MSE), root mean squared error (RMSE), and correlation coefficient (R^2) between the actual dataset and the BGWAN output, the BGWAN technique's performance can be evaluated. Using the successful application of the suggested approach, the Iran Coal Consumption was estimated. According to Table 3, the empirical results of Iran's data exhibit that the BGWAN method accuracy was more precise than the other methods. Hence, the findings proved that the recommended model was an appropriate tool for effective coal consumption prediction in Iran. It will provide a level playing field for checking how the energy policy authority impacts on the structure of Iran's energy with high economic interventionism by the government.

The BGWAN success in such a study suggests that it may be applied as a practical instrument for economic analysis in various areas, such as the energy system designs with more theoretical specification complexity. The Iran Coal Consumption prediction approach proposed in this work indicates advantages BGWAN, compared to other mathematical programming models for the easy and simple modelling of linear and nonlinear dependencies between variables, merely from data observed here.

It is also possible to investigate the Iran Coal Consumption forecasting by neural networks or new metaheuristics, such as simulated annealing, harmony search, etc. Comparisons can be made between the results of different approaches and the BGWAN technique. There is a need for more research focused on comparing methods described here with other presentation tools. Moreover, forecasting the Iran Coal Consumption can be studied by krill herd optimization algorithm, electro-magnetism mechanism algorithm, and other intelligent optimization approaches. We can compare the results of applying different methods with BGWAN. According to the proposed model's promising results, it is recommended that future studies apply it to other areas, such as solar radiation and biogas production.

Author Contributions: Conceptualization, A.S. and S.A.J.; methodology, M.s.J., A.G; software, M.s.J., A.G.; validation, A.S., S.A.J.; formal analysis, A.G.; investigation, M.s.J., A.S., A.G.; resources, A.G., R.D.; data curation, A.S.; writing—original draft preparation, A.G., R.D.; writing—review and editing,

References

1. Pudasainee, D.; Kurian, V.; Gupta, R. Coal: Past, Present, and Future Sustainable Use. *Future Energy* **2020**, 21–48. [CrossRef]
2. Derakhshani, R.; Raoof, A.; Mahvi, A.H.; Chatrouz, H. Similarities in the Fingerprints of Coal Mining Activities, High Ground Water Fluoride, and Dental Fluorosis in Zarand District, Kerman Province, Iran. *Fluoride* **2020**, *53*, 257–267.
3. Jalaee, S.A.; GhasemiNejad, A.; Lashkary, M.; Rezaee Jafari, M. Forecasting Iran's Energy Demand Using Cuckoo Optimization Algorithm. *Math. Prob. Eng.* **2019**, *2019*. [CrossRef]
4. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. *Neurocomputing: Foundations of Research*; MIT Press: Cambridge, MA, USA, 1988.
5. Gupta, D.; Arora, J.; Agrawal, U.; Khanna, A.; de Albuquerque, V.H.C. Optimized Binary Bat algorithm for classification of white blood cells. *Measurement* **2019**, *143*, 180–190. [CrossRef]
6. Tiwari, S.P.; Bansal, K. Nature inspired algorithms on Industrial applications: A survey. *Int. J. Appl. Eng. Res.* **2018**, *6*, 4282–4290.
7. Shandilya, S.K.; Shandilya, S.; Nagar, A.K. *Advances in Nature-Inspired Computing and Applications*; Springer: Berlin/Heidelberg, Germany, 2019; Volume 1.
8. Mirjalili, S. How effective is the Grey Wolf optimizer in training multi-layer perceptrons. *Appl. Intell.* **2015**, *43*, 150–161. [CrossRef]
9. Aljarah, I.; Faris, H.; Mirjalili, S. Optimizing connection weights in neural networks using the whale optimization algorithm. *Soft Comput.* **2018**, *22*, 1–15. [CrossRef]
10. Jaddi, N.S.; Abdullah, S.; Hamdan, A.R. Optimization of neural network model using modified bat-inspired algorithm. *Appl. Soft Comput.* **2015**, *37*, 71–86. [CrossRef]
11. Faris, H.; Mirjalili, S.; Aljarah, I. Automatic selection of hidden neurons and weights in neural networks using grey wolf optimizer based on a hybrid encoding scheme. *Int. J. Mach. Learn. Cybern.* **2019**, *10*, 2901–2920. [CrossRef]
12. Faris, H.; Aljarah, I.; Mirjalili, S. Training feedforward neural networks using multi-verse optimizer for binary classification problems. *Appl. Intell.* **2016**, *45*, 322–332. [CrossRef]
13. Nandy, S.; Sarkar, P.P.; Das, A. Analysis of a nature inspired firefly algorithm based back-propagation neural network training. *Int. J. Comput. Appl.* **2012**, *43*, 8–16. [CrossRef]
14. Ojha, V.K.; Abraham, A.; Snášel, V. Metaheuristic design of feedforward neural networks: A review of two decades of research. *Eng. Appl. Artif. Intell.* **2017**, *60*, 97–116. [CrossRef]
15. Heidari, A.A.; Faris, H.; Aljarah, I.; Mirjalili, S. An efficient hybrid multilayer perceptron neural network with grasshopper optimization. *Soft Comput.* **2019**, *23*, 7941–7958. [CrossRef]
16. Socha, K.; Blum, C. An ant colony optimization algorithm for continuous optimization: Application to feed-forward neural network training. *Neural Comput. Appl.* **2007**, *16*, 235–247. [CrossRef]
17. Valipour, M. Optimization of neural networks for precipitation analysis in a humid region to detect drought and wet year alarms. *Meteorol. Appl.* **2016**, *23*, 91–100. [CrossRef]
18. Villarrubia, G.; De Paz, J.F.; Chamoso, P.; De la Prieta, F. Artificial neural networks used in optimization problems. *Neurocomputing* **2018**, *272*, 10–16. [CrossRef]
19. Ruiz, L.G.B.; Rueda, R.; Cuéllar, M.P.; Pegalajar, M. Energy consumption forecasting based on Elman neural networks with evolutive optimization. *Expert Syst. Appl.* **2018**, *92*, 380–389. [CrossRef]
20. Yang, F.; Cho, H.; Zhang, H.; Zhang, J.; Wu, Y. Artificial neural network (ANN) based prediction and optimization of an organic Rankine cycle (ORC) for diesel engine waste heat recovery. *Energy Convers. Manag.* **2018**, *164*, 15–26. [CrossRef]
21. Ünler, A. Improvement of energy demand forecasts using swarm intelligence: The case of Turkey with projections to 2025. *Energy Policy* **2008**, *36*, 1937–1944. [CrossRef]
22. Canyurt, O.E.; Ozturk, H.K. Application of genetic algorithm (GA) technique on demand estimation of fossil fuels in Turkey. *Energy Policy* **2008**, *36*, 2562–2569. [CrossRef]
23. Toksarı, M.D. Ant colony optimization approach to estimate energy demand of Turkey. *Energy Policy* **2007**, *35*, 3984–3990. [CrossRef]
24. Azadeh, A.; Ghaderi, S.; Sohrabkhani, S. A simulated-based neural network algorithm for forecasting electrical energy consumption in Iran. *Energy Policy* **2008**, *36*, 2637–2644. [CrossRef]
25. Azadeh, A.; Saberi, M.; Seraj, O. An integrated fuzzy regression algorithm for energy consumption estimation with non-stationary data: A case study of Iran. *Energy* **2010**, *35*, 2351–2366. [CrossRef]

26. Amjadi, M.; Nezamabadi-Pour, H.; Farsangi, M. Estimation of electricity demand of Iran using two heuristic algorithms. *Energy Convers. Manag.* **2010**, *51*, 493–497. [CrossRef]

27. Zhang, M.; Mu, H.; Li, G.; Ning, Y. Forecasting the transport energy demand based on PLSR method in China. *Energy* **2009**, *34*, 1396–1400. [CrossRef]

28. Assareh, E.; Behrang, M.; Assari, M.; Ghanbarzadeh, A. Application of PSO (particle swarm optimization) and GA (genetic algorithm) techniques on demand estimation of oil in Iran. *Energy* **2010**, *35*, 5223–5229. [CrossRef]

29. Behrang, M.; Assareh, E.; Assari, M.; Ghanbarzadeh, A. Total energy demand estimation in Iran using bees algorithm. *Energy Sources Part. B Econ. Plan. Policy* **2011**, *6*, 294–303. [CrossRef]

30. Behrang, M.A.; Assareh, E.; Assari, M.; Ghanbarzadeh, A. Assessment of electricity demand in Iran's industrial sector using different intelligent optimization techniques. *Appl. Artif. Intell.* **2011**, *25*, 292–304. [CrossRef]

31. Bahmani, M.; Nejati, M.; GhasemiNejad, A.; Nazari Robati, F.; Lashkary, M.; Amani Zarin, N. A Novel Hybrid Approach Based on BAT Algorithm with Artificial Neural Network to Forecast Iran's Oil Consumption. *Math. Prob. Eng.* **2021**, *2021*. [CrossRef]

32. Horak, J.; Krulicky, T.; Rowland, Z.; Machova, V. Creating a Comprehensive Method for the Evaluation of a Company. *Sustainability* **2020**, *12*, 9114. [CrossRef]

33. Vochozka, M.; Vrbka, J.; Suler, P. Bankruptcy or success? the effective prediction of a company's financial development using LSTM. *Sustainability* **2020**, *12*, 7529. [CrossRef]

34. Vochozka, M.; Rowland, Z.; Suler, P.; Marousek, J. The influence of the international price of oil on the value of the EUR/USD exchange rate. *J. Compet.* **2020**, *12*, 167. [CrossRef]

35. Vochozka, M.; Horák, J.; Krulický, T.; Pardal, P. Predicting future Brent oil price on global markets. *Acta Montan. Slovaca* **2020**, *25*. [CrossRef]

36. Vrbka, J. The use of neural networks to determine value based drivers for SMEs operating in the rural areas of the Czech Republic. *Oeconomia Copernic.* **2020**, *11*, 325–346. [CrossRef]

37. Taner, T.; Sivrioglu, M. Energy–exergy analysis and optimisation of a model sugar factory in Turkey. *Energy* **2015**, *93*, 641–654. [CrossRef]

38. Taner, T.; Sivrioglu, M. A techno-economic & cost analysis of a turbine power plant: A case study for sugar plant. *Renew. Sustain. Energy Rev.* **2017**, *78*, 722–730. [CrossRef]

39. Taner, T.; Sivrioğlu, M.; Topal, H.; Dalkılıç, A.S.; Wongwises, S. A model of energy management analysis, case study of a sugar factory in Turkey. *Sādhanā* **2018**, *43*, 1–20. [CrossRef]

40. Taner, T. Optimisation processes of energy efficiency for a drying plant: A case of study for Turkey. *Appl. Therm. Eng.* **2015**, *80*, 247–260. [CrossRef]

41. Taner, T. Energy and exergy analyze of PEM fuel cell: A case study of modeling and simulations. *Energy* **2018**, *143*, 284–294. [CrossRef]

42. Topal, H.; Taner, T.; Altinsoy, Y.; Amirabedin, E. Application of trigeneration with direct co-combustion of poultry waste and coal: A case study in the poultry industry from Turkey. *Therm. Sci.* **2018**, *22*, 3073–3082. [CrossRef]

43. Topal, H.; Taner, T.; Naqvi, S.A.H.; Altınsoy, Y.; Amirabedin, E.; Ozkaymak, M. Exergy analysis of a circulating fluidized bed power plant co-firing with olive pits: A case study of power plant in Turkey. *Energy* **2017**, *140*, 40–46. [CrossRef]

44. Yang, X.-S. A new metaheuristic bat-inspired algorithm. In *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 65–74.

45. Oshaba, A.; Ali, E.; Abd Elazim, S. MPPT control design of PV system supplied SRM using BAT search algorithm. *Sustain. Energy Grids Netw.* **2015**, *2*, 51–60. [CrossRef]

46. Bahmani, M.; GhasemiNejad, A.; Robati, F.N.; Zarin, N.A. A novel approach to forecast global CO2 emission using Bat and Cuckoo optimization algorithms. *MethodsX* **2020**, *7*, 100986. [CrossRef] [PubMed]

47. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]

48. Jalaee, S.A.; Lashkary, M.; GhasemiNejad, A. The Phillips curve in Iran: Econometric versus artificial neural networks. *Heliyon* **2019**, *5*, e02344. [CrossRef] [PubMed]

49. Agrawal, U.; Arora, J.; Singh, R.; Gupta, D.; Khanna, A.; Khamparia, A. Hybrid Wolf-Bat Algorithm for Optimization of Connection Weights in Multi-layer Perceptron. *ACM Trans. Multimed. Comput. Commun. Appl. (TOMM)* **2020**, *16*, 1–20. [CrossRef]

50. BP. Statistical Review of World Energy. Available online: https://www.bp.com/content/dam/bp/business-sites/en/global/corporate/xlsx/energy-economics/statistical-review/bp-stats-review-2019-all-data.xlsx (accessed on 14 March 2021).

51. Jalaee, S.A.; Shakibaei, A.; Akbarifard, H.; Horry, H.R.; GhasemiNejad, A.; Robati, F.N.; Derakhshani, R. A novel hybrid method based on Cuckoo optimization algorithm and artificial neural network to forecast world's carbon dioxide emission. *MethodsX* **2021**, *8*, 101310. [CrossRef]

52. Jalaee, S.A.; Shakibaei, A.; Horry, H.R.; Akbarifard, H.; GhasemiNejad, A.; Robati, F.N.; Zarin, N.A. A new hybrid metaheuristic method based on biogeography-based optimization and particle swarm optimization algorithm to estimate money demand in Iran. *MethodsX* **2021**, *8*, 101226. [CrossRef]