Research Article

An Efficient Approach for Real-Time Prediction of Rate of Penetration in Offshore Drilling

Xian Shi,[1] Gang Liu,[1] Xiaoling Gong,[2] Jialin Zhang,[1] Jian Wang,[3] and Hongning Zhang[1]

[1]*College of Petroleum Engineering, China University of Petroleum (Huadong), Qingdao, Shandong 266580, China*
[2]*College of Information and Control Engineering, China University of Petroleum (Huadong), Qingdao, Shandong 266580, China*
[3]*College of Science, China University of Petroleum (Huadong), Qingdao, Shandong 266580, China*

Correspondence should be addressed to Gang Liu; lg196009091@sina.com

Predicting the rate of penetration (ROP) is critical for drilling optimization because maximization of ROP can greatly reduce expensive drilling costs. In this work, the typical extreme learning machine (ELM) and an efficient learning model, upper-layer-solution-aware (USA), have been used in ROP prediction. Because formation type, rock mechanical properties, hydraulics, bit type and properties (weight on the bit and rotary speed), and mud properties are the most important parameters that affect ROP, they have been considered to be the input parameters to predict ROP. The prediction model has been constructed using industrial reservoir data sets that are collected from an oil reservoir at the Bohai Bay, China. The prediction accuracy of the model has been evaluated and compared with the commonly used conventional artificial neural network (ANN). The results indicate that ANN, ELM, and USA models are all competent for ROP prediction, while both of the ELM and USA models have the advantage of faster learning speed and better generalization performance. The simulation results have shown a promising prospect for ELM and USA in the field of ROP prediction in new oil and gas exploration in general, as they outperform the ANN model. Meanwhile, this work provides drilling engineers with more choices for ROP prediction according to their computation and accuracy demand.

1. Introduction

Drilling is an expensive and necessary operation for petroleum and gas exploration. The ultimate aim in drilling operations is to increase drilling speed with less cost while maintaining safety. However, because of the geological uncertainty and many uncontrolled operational factors, the optimization of drilling is still a large challenge for oil and gas industries [1, 2]. In offshore drilling, rate of penetration (ROP) is the key parameter to optimize total rig hours. Because it can be used for formation drillability estimation, reliable and fast prediction of rate of penetration (ROP) is highly desirable.

In existing literature, some direct and indirect methods are designed to evaluate the ROP. ROP mathematical models can be used to outline changes of rock mechanical properties and drilling parameters, bit types, and design on ROP. Bourgoyne and Young (B&Y) proposed a ROP calculating model considering eight functions, whereas the relevant parameters need to be obtained by multiple regression for each data entry,

and this equation is only adequate for roller-type rock bits [3]. In addition to B&Y's equation, many authors try to find a simple link between ROP and rock properties because the majority of reservoir mechanical properties can be inferred from well logs. Howarth et al. carried out a laboratory test program for correlating the penetration rate with rock properties and reported that penetration rate correlated well with the uniaxial compressive and tensile strength of the rock [4]. Kahraman introduced a strong correlation between penetration rate and brittleness values obtained from uniaxial compressive strength and tensile strength [5]. However, the application of rock mechanical properties for ROP estimation cannot reveal the real-time downhole conditions. Therefore, it is necessary to use drilling data, but many operational parameters of the drilling equipment are found to relate to ROP. Some authors performed statistical analyses using real-time drilling data, such as weight bit on, bit rotational speed, torque, and the effect of mud properties, and some of them obtained

TABLE 1: Summary of ROP prediction models with artificial intelligence.

Reference	Model	Input number	Input layer	Output layer
Zhang et al. [22]	AHP and BPANN	9	UCS, bit size, bit type, drillability coefficient, gross hours drilled, WOB, RPM, drilling mud density, and AV (Apparent Viscosity)	ROP
Jahanbakhshi et al. [13]	ANN	20	Differential pressure, hydraulics, hole depth, pump pressure, density of the overlying rock, equivalent circulating density, hole size, formation drillability, permeability and porosity, drilling fluid type, plastic viscosity of mud, yield point of mud, initial gel strength of mud, 10 min Gel strength of mud bit type and its properties, weight on the bit and rotary speed, bit wear, and bit hydraulic power	ROP
Bahari and Seyed [11]	Fuzzy	4	UCS, rock quality designation, bit Load, and bit rotation	ROP
Amar and Ibrahim [25]	Radial-basis function and ELM	7	Depth, bit weight, rotary speed, tooth wear, Reynolds number function, ECD, and pore gradient	ROP
Bilgesu et al. [12]	ANN	9	Formation drillability, formation abrasiveness, bearing wear, tooth wear, pump rate, rotating time, rotary torque, WOB, and rotary speed	ROP
Bataee and Mohseni [2]	ANN	4	Bit diameter, WOB, RPM, and mud weight	ROP
Moran et al. [26]	ANN	6	Rock strength, rock type, abrasion, WOB, RPM, and mud weight	ROP and wear
Monazami et al. [27]	ANN	13	Drill collar outside diameter, drill collar length, kick of point, azimuth, inclination angle, weight on bit, flow rate, bit rotation speed, mud weight, solid percent, plastic viscosity, yield point, and measured depth	ROP

a certain degree of success [6–9]. As a matter of fact, there is no exact relation between ROP, rock mass properties, and different drilling variables because not only do a large number of uncertain drilling variables influence ROP but also the relationships between ROP and the affecting factors are complex and highly nonlinear. Thus, some soft computing technologies such as neural networks have been applied to ROP prediction in recent years [10]. The flexibility of this method allows engineers to analyze a wide range of information and deliver high-quality ROP prediction. Bahari and Seyed applied GA to determine constant coefficients of Bourgoyne and Young's ROP model [11]. Table 1 shows current ROP prediction models with artificial intelligence. The results revealed that the ANN model exhibited better performance to predict penetration rate than the prediction performance of multiple regression models. In addition, neural networks can yield good correlation coefficients even if fewer data are available from filed measurements [12–15]. However, traditional ANN still has some significant shortcomings, such as a slow training process and the possibility of trapping in local extrema, which lead to dissatisfactory predictions. Therefore, it is necessary to introduce new algorithms that can potentially further improve ROP prediction accuracy.

The extreme learning machine (ELM) proposed by Huang et al. is a fast algorithm for single hidden-layer feedforward neural networks (SLFNs) [16, 17]. The way ELM trains SLFN is that it first randomly generates the weights of the hidden layer and then calculates the weights of the output layer by solving a linear system using the least square method. This learning algorithm is extremely fast and has good prediction accuracy. It has been proved in theory and in practice that this

algorithm can generate good generalization performance in most cases at speeds much faster than traditionally popular feedforward neural network learning algorithms. Until now, ELM has been widely studied and applied extensively by researchers and it has demonstrated good generalization and prediction performance in many real-life applications [18, 19]. Many algorithms such as evolutionary ELM and enhanced random search-based incremental ELM (EI-ELM) have been proposed to optimize the network structure. The above algorithms choose all or part of the hidden-layer weights randomly and select the candidate ones with the LSE (Least Squares Estimation). However, the model parameters in these algorithms are not efficient enough because only the value of the objective function is used in the search process. USA (upper-layer-solution-aware algorithm), an improved version of the ELM algorithm, gives an optimization of the number of hidden-layer nodes and the parameters modeling the problem [20]. Once the weights of the hidden layer are determined, those of the output layer can be determined as a certain function using the closed-form solution. Therefore, what we need to search for at each epoch along the gradient direction is just the weights of the hidden layer. However, the study of ELM and USA in drilling ROP prediction is rare.

This paper concentrates on ROP estimation using bit type and its properties, mud type and mud viscosity, formation parameters such as rock strength, formation drillability, and formation abrasiveness, and some critical drilling equipment operational parameters such as pump pressure, WOB, and rotary speed based on the previous drilled wells data with the ELM and USA model. The developed ELM and USA model are shown to be efficient with respect to accuracy and running

time compared to traditional ANN models. They thus provide a more reliable and faster real-time tool for predicting ROP in new wells.

2. Artificial Neural Networks and Extreme Learning Machines

2.1. Methodology of Artificial Neural Networks. Artificial neural networks (ANNs) are efficient models in approximating the unknown nonlinear functions, seeking to simulate human brain behavior by processing data on a trial-and-error basis. Because of their powerful ability in approximation and generalization, ANNs have been widely used in petroleum engineering, with applications to bit selection, reservoir characterization, EOR (enhanced oil recovery), hydraulic fracturing candidate selection, and so forth [21–24].

The network structures of ANNs are made up of a number of neurons which are distributed in layers based on their different functions. Generally, a complete neural network consists of three different types on layers, namely, an input layer, one or more hidden layers, and an output layer in which each layer includes a preset number of neurons. It has been rigorously proved that ANN can approximate any continuous function with an arbitrary precision. And ANN is thus a universal approximator. The direction of the information transmission in feedforward neural networks is from input neurons through activation of the hidden neurons to the outputs. For supervised learning, the connecting weights of the layers are updated in the training procedure by minimizing the objective function between the networks' actual outputs and the desired value. Actually, there are many different ways to fulfill the training of an ANN model: back-propagation (BP) algorithm is the most popular one. It is a typical type of supervised learning strategy in machine learning field. In general, the BP method uses the following steps: for a given specific input sample, the actual output is obtained through the information transferring on layers one by one. If the error between the produced and the desired outputs is acceptable, then stop training. If the error is not acceptable in the previous step, then the weights are changed on the interconnections that go into the output layer. Next, an error value is calculated for all of the neurons in the hidden layer that is just below the output layer. Then, the weights are adjusted for all interconnections that go into the hidden layer. The process is continued until the last layer of weights has been adjusted.

Some essential shortcomings of BP method are continually encountered in many real applications, for example, slow convergence speed and prone to being stuck in a local minimum. Thus, the standard BP algorithm sometimes shows poor performance on practical applications which mainly stems from employing the standard gradient descent method to adjust the weights. As a result, there are many different variants that have emerged in recent decades. The commonly used variants based on different optimizing strategies are with Levenberg-Marquardt (LM); the conjugate gradient method with Powell-Beale, Fletcher-Reeves, and Polak-Ribiere updates; the gradient descent method with momentum term, penalty term, and adaptive learning rate; Bayesian regulation; and scaled conjugate gradient [28].
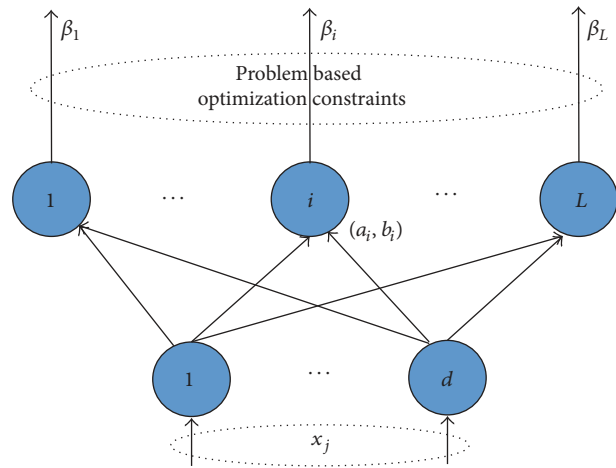

FIGURE 1: Structure and schematic diagram of the ELM method used in this paper.

2.2. Methodology of Extreme Learning Machines

2.2.1. Conventional Extreme Learning Machines. ELM was originally proposed to train a single hidden-layer feedforward neural network (SLFN) and later extended to more generalized SLFNs where the hidden layer may not be made of homogeneous neurons. Some obstacles that back-propagation neural networks faced, such as the slow training speed, the sensitivity to the selection of the parameters, and the high possibility to be trapped in local minima, can be avoided using ELM. Figure 1 illustrates the structure of the ELM method.

Neural networks are used universally for feature extraction, clustering, regression, and classification and require little human intervention in most cases. Inspired by biological learning characteristics, to overcome the challenging problems encountered by BP algorithms, ELM development sets fit hidden-layer neurons while randomly choosing the parameters at the initialization period.

The learning efficiency and effectiveness of ELM were established in 2005, while its universal approximation capability was rigorously proved later. The correspondence between specific biological neural configurations consequently appeared in the literature.

Unlike other randomness-based training methods/network models, the hidden weights can be randomly determined before the learning process, and all of the hidden neurons are independent of the training samples and one another in ELM. Once determined, both the hidden neurons and the hidden weights need not be tuned, although they are important and crucial for training. Moreover, ELM has good generalization ability, as the architecture of ELM is robust enough and has enough hidden neurons for the given problems, which is not the case for conventional learning methods highly dependent on the data.

Because the connecting weights between the input and the hidden layer do not need to be tuned, we use a simple

model with one output node as an example and give the output of the ELM as follows:

$$f_L(x) = \sum_{i=1}^{L} \beta_i h_i(x) = h(x)\beta, \tag{1}$$

in which $\beta = [\beta_1, \ldots, \beta_L]^T$ denotes the output weights connecting the hidden layer and the output layer, and $h(x) = [h_1(x), \ldots, h_L(x)]$ is the hidden-layer output with respect to input x. $h(x)$ is a feature mapping. In fact, if the input is in d-dimensional space, $h(x)$ maps it to L-dimensional hidden-layer feature space. For binary classification applications using ELM, the function of the output is

$$f_L(x) = \text{sign}[h(x)\beta]. \tag{2}$$

ELM often leads to a smaller training error with the smaller norm of weights compared with the traditional learning algorithm. According to Bartlett's theory, the smaller weights make a great contribution to a better generalization performance of neural networks with similar training errors. What makes ELM particularly noteworthy is that the values of the weights need not be adjusted during the training. The values of the input weights are randomly chosen at the beginning of the training procedure and stay fixed, while the weights of the output layer are calculated by searching for the least square solution of the following objective function.

The training error and the norm of the output weights ELM minimization are as follows:

$$\text{Minimize:} \quad \|\mathbf{H}\beta - T\|^2 \tag{3}$$
$$\|\beta\|,$$

where $\mathbf{H}$ is the output matrix of the hidden layer.

The main purpose of ELM is to minimize the training error as well as the norm of connecting output layer weights. According to the theory of linear systems, the weights between the hidden layer and the output layer are calculated using the following equation [17]:

$$\beta = \mathbf{H}^\dagger T, \tag{4}$$

where $\mathbf{H}^\dagger$ is the Moore-Penrose pseudoinverse of matrix $\mathbf{H}$ and $T = [t_1, t_2, \ldots, t_n]^T$.

We can summarize the ELM training algorithm as follows [17]:

(1) Set the hidden node parameters randomly, for example, input weights a_i and biases b_i with the certain hidden nodes, $i = 1, 2, \ldots, L$.

(2) Calculate the hidden-layer output matrix $\mathbf{H}$ through the input and the weights as well as the biases.

(3) Obtain the output weights matrix using (4).

The orthogonalization method, the iterative method, singular value decomposition (SVD), and so forth can be used to evaluate the Moore-Penrose pseudoinverse of a matrix [24]. By the orthogonal projection method, we can obtain

$\mathbf{H}^\dagger = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T$ when $\mathbf{H}^T\mathbf{H}$ is nonsingular and $\mathbf{H}^\dagger = \mathbf{H}^T(\mathbf{H}\mathbf{H}^T)^{-1}$ when $\mathbf{H}\mathbf{H}^T$ is nonsingular. In light of ridge regression theory, we add a positive value to the diagonal of $\mathbf{H}^T\mathbf{H}$ or $\mathbf{H}\mathbf{H}^T$ to make the resultant solution more stable and tend to have better generalization performance.

According to ELM theory, there are many feature mapping functions that $h(x)$ may be designed to incorporate, which gives ELM the ability to approximate any continuous target function. The actual activation functions of human brain systems seem to be more like a nonlinear piecewise continuous stimulation, though the actual functions remain unknown. In fact, the ELM model is based on two major theorems: universal approximation and classification ability.

Theorem 1 (universal approximation theorem; see [16, 29]). *Given any bounded, nonconstant piecewise continuous function as the activation function in hidden neurons, if, by tuning parameters of the hidden neuron activation function, SLFNs can approximate any target continuous function, then, for any continuous target function $f(x)$ and any randomly generated function sequence, $\{h_i(x)\}_{i=1}^{L} \lim_{L\to\infty}\|\sum_{i=1}^{L} \beta_i h_i(x) - f(x)\| = 0$ holds with probability one given the appropriate output weights β.*

According to the nonlinear activation function and piecewise linear combination, ELM can approximate any continuous mapping and divide arbitrary disjoint regions of any shapes with enough randomly hidden neurons based on the nonlinear piecewise continuous activation functions and their linear combinations. In particular, the theory basis for the classification capability of ELM is as follows.

Theorem 2 (classification capability theorem; see [17]). *Given feature mapping $h(x)$, if $h(x)$ is dense in $C(R^d)$ or in $C(M)$, where M is a compact set of R^d, then ELM with random hidden-layer mapping $h(x)$ can separate arbitrary disjoint regions of any shapes in R^d or in M.*

This theorem implies that ELM with bounded nonconstant piecewise continuous functions has universal approximation capability. While it does not require updating the weights of hidden neurons in the training process, it is worth mentioning that ELM is more like the activation of the human brain with the randomly generating hidden weights without tuning.

2.2.2. Upper-Layer-Solution-Aware (USA) Algorithm Model. $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_N]$ is given as the set of input vectors, and each vector is denoted by $\mathbf{x}_i = [x_{1i}, \ldots, x_{ji}, \ldots, x_{Di}]$ in which D is the dimension of the input vector and N is the total sum of training samples. Define L as the number of hidden units and C as the dimension of the output vector. $\mathbf{y}_i = \mathbf{U}^T\mathbf{h}_i$ is the output of the SHLNN. Among them, $\mathbf{h}_i = \sigma(\mathbf{W}^T\mathbf{x}_i)$ is denoted as the hidden-layer output, $\mathbf{U}$ is denoted as weight $L \times C$ matrix at the upper layer, $\mathbf{W}$ is denoted as $D \times L$ weight matrix at the lower layer, and $\sigma(\cdot)$ is denoted as the kernel function.

Note that if $\mathbf{x}_i$ and $\mathbf{h}_i$ are augmented with ones, the bias terms are implicitly represented in the above formulations.

$\mathbf{T} = [\mathbf{t}_1, \ldots, \mathbf{t}_i, \ldots, \mathbf{t}_N]$ is given as the target vectors and each target is denoted by $\mathbf{t}_i = [t_{1i}, \ldots, t_{ji}, \ldots, t_{Ci}]^T$, and the parameters $\mathbf{U}$ and $\mathbf{W}$ are used to minimize the square error.

$$E = \|\mathbf{Y} - \mathbf{T}\|^2 = \text{Tr}\left[(\mathbf{Y} - \mathbf{T})(\mathbf{Y} - \mathbf{T})^T\right], \qquad (5)$$

where $\mathbf{Y} = [y_1, \ldots, y_i, \ldots, y_N]$. Note that the hidden-layer values $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_i, \ldots, \mathbf{h}_N]$ are also determined uniquely if the lower-layer weights $\mathbf{W}$ are fixed. After setting the gradient

$$\begin{aligned}\frac{\partial E}{\partial \mathbf{W}} &= \frac{\partial \text{Tr}\left[\left(\mathbf{U}^T\mathbf{H} - \mathbf{T}\right)\left(\mathbf{U}^T\mathbf{H} - \mathbf{T}\right)^T\right]}{\partial \mathbf{U}} \\ &= 2\mathbf{H}\left(\mathbf{U}^T\mathbf{H} - \mathbf{T}\right)^T \end{aligned} \qquad (6)$$

to zero, the upper-layer weights $\mathbf{U}$ can be determined to the closed-form solution [20]:

$$\mathbf{U} = \left(\mathbf{H}\mathbf{H}^T\right)^{-1}\mathbf{H}\mathbf{T}^T. \qquad (7)$$

Note that (7) defines an implicit constraint between the set of weights $\mathbf{U}$ and the set of weights $\mathbf{W}$, through hidden-layer output $\mathbf{H}$ in the SHLNN. This leads to a structure that our new algorithms will use in optimizing the SHLNN.

Regularization techniques need to be used in actual applications when hidden-layer matrix $\mathbf{H}$ is sometimes ill-conditioned (i.e., $\mathbf{H}\mathbf{H}^T$ is singular) because solving (7) is relatively simple. The popular technique used in this study is based on the ridge regression theory. Even more specifically,

by adding positive value $\mathbf{I}/\mu$ to the diagonal of $\mathbf{H}\mathbf{H}^T$, (7) is converted to

$$\mathbf{U} = \left(\frac{\mathbf{I}}{\mu} + \mathbf{H}\mathbf{H}^T\right)^{-1}\mathbf{H}\mathbf{T}^T, \qquad (8)$$

where $\mathbf{I}$ is the identity matrix and μ is a positive constant (regularization coefficient) that is used to control the degree of regularization. The final solution (8) actually minimizes $\|\mathbf{U}^T\mathbf{H} - \mathbf{T}\|^2 + \mu\|\mathbf{U}\|^2$, in which $\mu\|\mathbf{U}\|^2$ is L_2 regularization term. Positive constant μ is the regularization coefficient. It represents the relative importance of complexity-penalty term $\|\mathbf{U}\|^2$ with respect to empirical risk $\|\mathbf{U}^T\mathbf{H} - \mathbf{T}\|^2$. When μ approaches zero, the model tends to an unconstrained optimal problem, and then the solutions would be completely determined from the training samples. When μ is made infinitely large, this means that the training samples are unreliable, imposing weight $\mathbf{U}$ to be sufficiently of small values. In real applications, regularization parameter μ is set to be a value somewhere between these two cases. The reasonable regularization parameter can efficiently affect both of the learning and prediction performance (generalization ability). Solution (8) shows better stability and better generalization performance than (7). Whenever the pseudoinverse is involved, solution (8) can be used throughout the paper.

The principle of this algorithm is very simple. Once the lower-layer weights are determined, the upper-layer weights can be determined explicitly by using the closed-form solution (8). Based on this solution, at each epoch, all we need to adjust along the gradient direction are the lower-layer weights. Gradient $\partial E/\partial \mathbf{W}$ is obtained by considering upper-layer weights $\mathbf{U}$ under $\mathbf{W}$'s effect and the training objective function is the square error. We can obtain the gradient by treating $\mathbf{U}$ as a function of $\mathbf{W}$ and plugging (7) into criterion (5):

$$\begin{aligned}\frac{\partial E}{\partial \mathbf{W}} &= \frac{\partial \text{Tr}\left[\left(\mathbf{U}^T\mathbf{H} - \mathbf{T}\right)\left(\mathbf{U}^T\mathbf{H} - \mathbf{T}\right)^T\right]}{\partial \mathbf{W}} = \frac{\partial \text{Tr}\left[\left(\left[\left(\mathbf{H}\mathbf{H}^T\right)^{-1}\mathbf{H}\mathbf{T}^T\right]^T\mathbf{H} - \mathbf{T}\right)\left(\left[\left(\mathbf{H}\mathbf{H}^T\right)^{-1}\mathbf{H}\mathbf{T}^T\right]^T\mathbf{H} - \mathbf{T}\right)^T\right]}{\partial \mathbf{W}} \\ &= 2\mathbf{X}\left[\mathbf{H}^T \circ (1 - \mathbf{H})^T \circ \left[\mathbf{H}^\dagger\left(\mathbf{H}\mathbf{T}^T\right)\left(\mathbf{T}\mathbf{H}^\dagger\right) - \mathbf{T}^T\left(\mathbf{T}\mathbf{H}^\dagger\right)\right]\right],\end{aligned} \qquad (9)$$

in which $\mathbf{H}^\dagger = \mathbf{H}^T(\mathbf{H}\mathbf{H}^T)^{-1}$ is the pseudoinverse of $\mathbf{H}$ and $\circ$ is the element-wise production.

In the derivation of (9), we used the fact that $\mathbf{H}\mathbf{H}^T$ and $(\mathbf{H}\mathbf{H}^T)^{-1}$ are symmetric. We also used the fact that

$$\begin{aligned}&\frac{\partial \text{Tr}\left[\left(\mathbf{H}\mathbf{H}^T\right)^{-1}\mathbf{H}\mathbf{T}^T\mathbf{T}\mathbf{H}^T\right]}{\partial \mathbf{H}^T} \\ &= -2\mathbf{H}^T\left(\mathbf{H}\mathbf{H}^T\right)^{-1}\mathbf{H}\mathbf{T}^T\mathbf{T}\mathbf{H}^T\left(\mathbf{H}\mathbf{H}^T\right)^{-1} \\ &+ 2\mathbf{T}^T\mathbf{T}\mathbf{H}^T\left(\mathbf{H}\mathbf{H}^T\right)^{-1}.\end{aligned} \qquad (10)$$

This algorithm updates $\mathbf{W}$ by using the gradient that is defined directly in (9) as

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \rho\frac{\partial E}{\partial \mathbf{W}}, \qquad (11)$$

where ρ is the learning rate. The learning rate here means the updating step which is widely used in solving optimal problems. It shows how far the weights need to be moved in the direction of the given gradient $\partial E/\partial \mathbf{W}$ and how to obtain the new weights which minimizes the objective function. One simple way is to set a positive constant value in the entire training process. In this paper, we select the suitable learning rate by means of many different simulations. This algorithm

uses the closed-form solution (8) to calculate **U** after updating **W**.

The algorithms proposed in this paper achieve significantly better classification accuracy than ELM when the same number of hidden units is used. To achieve the same classification accuracy, the algorithm requires only 1/16 of the model, and, therefore, less test time is needed.

3. Input Data Selection

The target research four wells were drilled at the Bohai Bay basin, which is the largest oil and gas production base offshore in China. The oilfield has experienced many production stages since 1963 and currently many projects are carried out by China National Offshore Oil Corporation (CNOOC) and ConocoPhillips China (COPC). To effectively increase production and decrease drilling cost, drilling speed control is necessary.

Due to drilling requirements and the similarity of wells located close to one another, collecting past data and utilizing the data in a useful manner have an important impact on drilling cost reduction. This means that optimum parameters are always in effect. As a matter of fact, there are various factors than can affect ROP. Previous studies show that the ROP is largely dependent on some critical parameters, which can be classified into three types: rig/bit related parameters, mud related parameters, and formation parameters. The rig/bit and mud related parameters can be manipulated, but the formation parameters have to be handled differently. Table 2 gives a brief classification of some important drilling parameters that affect ROP. To predict and optimize the ROP, both controllable and uncontrollable parameters from drilling reports were collected from daily drilling reports, lab investigations, well log, and geological information.

Because it is impossible to treat all of the relevant drilling parameters in ROP prediction, it is necessary to choose essential data based on literature surveys and statistical analysis. It was concluded that the rock properties, the drilling machine parameters, and mud properties relate to the ROP. In fact, a total of 18 drilling parameters were gathered in this study: depth, torque, rotational per speed (RPM), weight on bit, flow rates, active PVT, pump pressure, hook load, differential pressure, mud weight, mud viscosity, formation drillability, formation abrasiveness, UCS, porosity, bit size, and bit type.

(1) The bit type/size can affect ROP in a given formation. Roller cone bits generally have three cones that rotate around their axis while teeth are milled out of the matrix or inserted. The teeth combine crushing and shearing to fracture the rock. Drag bits usually consist of a fixed cutter mechanism that can be cutting blades, diamond stones, or cutters. Today, polycrystalline diamond compact (PDC) is the most commonly used drag bit with PDC, diamond impregnated, or diamond hybrid cutters mounted on the bit blades/body. Drag bits disintegrate rock primarily through shearing. The difference in the design of roller cone and drag bits and different rock disintegration methods requires different ROP

models. Thus, the effects of the bit type are included in this model.

(2) Operational parameters such as pump pressure and differential pressure are the most important operational hydraulic parameters affecting ROP. Pump pressure is sometimes increased to achieve a higher penetration rate for a certain depth of advance while keeping rotation constant. In addition, the mechanical factors of weight on the bit and rotary speed have linear relationships with ROP. The increase in weight on the bit can push the bit teeth or cutters further down into a formation and disintegrate more rocks, resulting in faster ROP. Bit rotation can be increased to obtain higher penetration rate, keeping the bit load constant.

(3) Formation mechanical characteristics are uncontrollable variables for controlling ROP values. In general, decreasing the penetration rate is greatly attributed to increase of depth because by increasing the depth, rock strength increases and porosity decreases. Moreover, the larger indirect drilling resistance that results from larger UCS and hardness will also limit the depth of cut for a given set of bit design and applied operating parameters. There are two main reasons for selecting rock strength as the representative of rock properties. One is that the rock strength is calculated by a function of depth, density, and porosity; another is that this parameter can be computed by logging data or triaxial compressive experiments and scratch tests in a lab if core samples are available.

(4) The aim of drilling mud is to remove the loose rock chips away from the bit face while the bits cool. Therefore, drilling weight, mud rheology, and solid content are confirmed as an important factor in ROP by some studies. ROP basically decreased by increasing mud viscosity, solid content, and mud weight.

In addition to the above parameters, some comprehensive parameters such as formation drillability and abrasiveness also need to be considered because these parameters can provide some important information sources about ROP prediction that cannot be described by conventional drilling data. More importantly, the model inputs should be easily acquired in real time, recorded from the respective measurement gauges on the rig. Using additional parameters as inputs can result in a large network size and consequently slow down running speed and efficiency. Thus, bit size, bit type, bit wear, UCS, formation drillability, formation abrasiveness, rotational per speed, WOB, pump pressure, mud viscosity, and mud density were selected as 11 inputs for neural network simulation. The data set used in this paper consists of more than 5000 measurements taken from this area at different wells and with different drilling rigs. Figure 2 illustrates three types of PDC bits in the drilling operation in this area.

It should be noted that data can very often be nonnumeric. Because neural networks cannot be trained with nonnumeric data, it is necessary to translate nonnumeric

TABLE 2: ROP related drilling parameters classification.

Rig and bit related parameters	Formation parameters	Drilling fluid properties
Weight on bit (WOB)	Local stresses	Mud weight
Torque	Hardness	Viscosity
Rotary speed (RPM)	Mineralogy	Filtrate loss
Flow rates	Porosity and permeability	Solid content
Pump stroke speed (SPM)	Formation abrasiveness	Gel strength
Pump pressure	Drillability	Mud pH
Hook load	Depth	Yield point
Bit wear	Temperature	
Type of the bit	Unconfined compressive strength (UCS)	

FIGURE 2: Three types of PDC drilling bits.

data into numeric data. In this study, formation drillability, formation abrasiveness, bit size, and bit type are nonnumeric. Numbering classes are used in this study to perform nonnumeric translation. The formation drillability ranged between 30 and 75. The highest number represents the highest drillability and the lower drillability was for hard formations such as shales. The formation abrasiveness ranged between 1 and 8, where 8 signifies the highest abrasive formation. In addition, bit type is assigned between 1 and 12 where 12 is related to the bit type that has highest ROP average value. The bit wear values are determined between 0 and 9, where 9 indicates that a tooth is completely worn out. Table 3 shows the results for the recorded parameters and their ranges.

4. Experimental Design

4.1. Model Architecture. Determining the optimal size of the neural network model is complex. An overly complex network tends to memorize the data without learning to generalize to new situations. It concentrates too much on the data presented for learning and tends toward modeling the noise. The total number of ROP in target wells is summed up to 5500. The training subset constitutes 75% of the total data and testing subsets that include 25% of the total data. As mentioned above, inputs of neural networks include 10

TABLE 3: Summary of results for the recorded parameters.

Parameters	Ranges
Bit size/inch	5.813~8.5
Bit type	1–12
Bit wear	0–9
UCS/psi	2412~21487
Formation drillability	30–75
Formation abrasiveness	1–8
Rotational per speed/rpm	84~126
WOB/10^3 lbs	24~69
Drilling mud type	1~3
Mud viscosity/cp	1~70
Pump pressure/psi	1651–3765
ROP	26.6~120.5

parameters, while the ROP is the only element in the output layer. Figure 3 is the developed neural network architecture.

4.2. Model Performance Indicators. The performances of neural network models are assessed by means of the correction
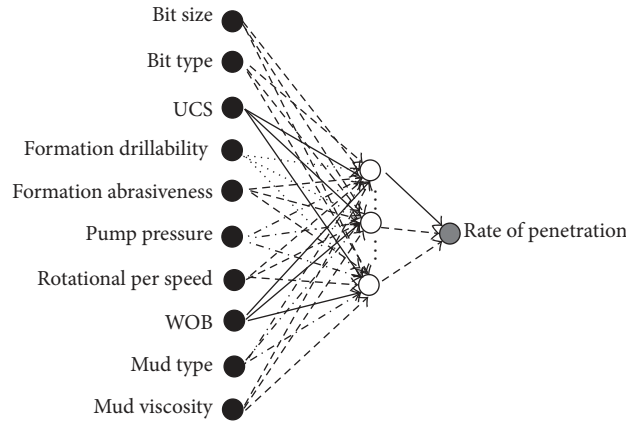
FIGURE 3: Schematic of architecture of the ELM network model. Node number of the input layer is 10, and ROP are the only node at the output layer.

coefficient (R^2), mean absolute error (MAE), root mean square error (RMSE), and variance accounted for (VAF) performance indicators.

R^2 represents the proportion of the overall variance explained by the model, which can be calculated as

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (f(x_i) - y_i)^2}{\sum_{i=1}^{n} f(x_i)^2 - \sum_{i=1}^{n} (y_i)^2 / n}. \tag{12}$$

These performance indicators can give a sense of how good the performance of a predictive model is relative to the actual value.

MSE can be described as

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2. \tag{13}$$

The RMSE is conventionally used as an error function for quality monitoring of the model. Model performance increases as RMSE decreases. RMSE can be calculated by the following equation:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2}. \tag{14}$$

VAF is often used to evaluate the correctness of a model, by comparing the measured values with the estimated output of the model. VAF is computed by the following equation:

$$\text{VAF} = \left(1 - \frac{\text{var}\,[f(x_i) - y_i]}{\text{var}\,f(x_i)} \right) \times 100, \tag{15}$$

where, for (12)–(15), y_i is the measured data and $f(x_i)$ denotes the predicted data, $x_i, \ldots, x_n$ are the input parameters, and n is the total data number.

4.3. Model Parameters Selection. As for the ANN, ELM, or USA models, the kernel function and the number of hidden nodes are critical parameters for neural network accuracy.

The accuracy with different hidden layers was compared in terms of RMSE performance indicators. When the kernel function is determined, then increase the hidden numbers incrementally until 100 is reached. Figure 4(a) illustrates the accuracy comparison with different nodes for the ANN, ELM, and USA models, respectively. As for the ELM model, the MSE between prediction results and measured values decreases as the node number of the hidden layer gradually increases with more than 65 nodes. As for the USA model, the MSE between prediction results and measured values also decreases as the node number of the hidden layer gradually increases, except for individual volatility points. As for the ANN model, the MSE between prediction results and measured values was observed in a fluctuant trend as the node number of the hidden layer gradually increases, which indicates that the choice of hidden nodes can greatly affect final accuracy.

Four types of kernels have been tested using a training data set for the ELM model, and they are the sigmoid function, radial-basis function, hardlim function, and triangular function. Figure 4(b) shows the accuracy comparison with different kernel functions for the ELM model. Among the four kernels for the ELM model, the sigmoid-based model comes to the threshold first when the node number is set to 40, while an overfitting problem appears as the node number exceeds 60. The hardlim-based, triangular-based, and radial-basis based models display a similar trend. However, it seems that the node number might exceed 60 when the minimum errors are close to the threshold for the hardlim-based model. For the triangular-based model, the RMSE reaches the lowest point at 3.11% when the node number is set as 85, while the RMSE reaches the lowest point at 3.12% when the node number is set as 95 for radial-basis based models. Figure 4(c) shows the accuracy comparison with different kernel functions for the USA model. Observe that the radial-basis based model comes to the threshold first when the node number is set as 65. As for the hardlim-based model, it seems that the node number might exceed 60 when the minimum
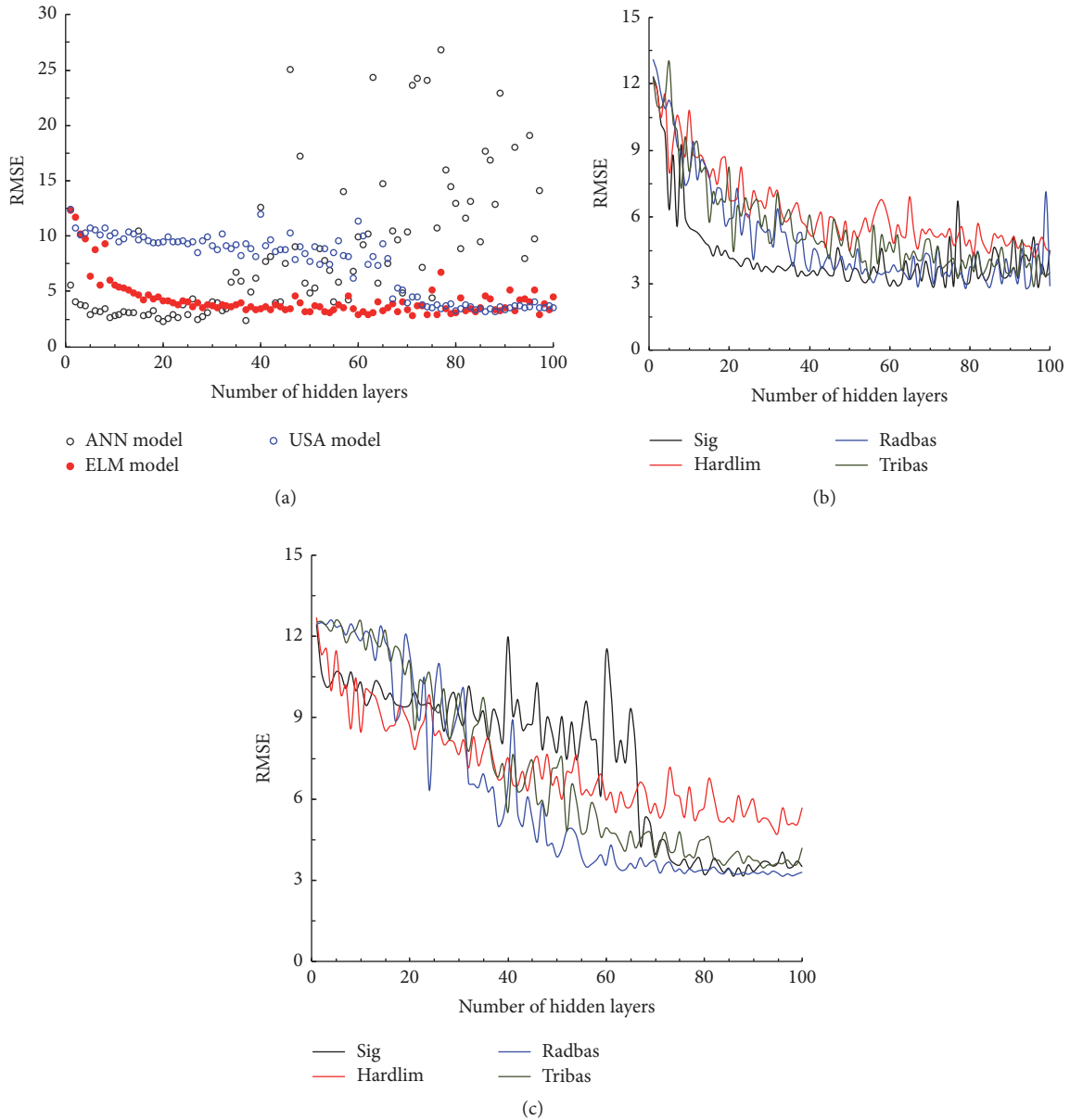
(a)

(b)

(c)

FIGURE 4: (a) The RMSE change with the increase of hidden layers; (b) comparison of RMSE with different kernel functions for the ELM network model; (c) comparison of RMSE with different kernel functions for the USA network model.

errors are close to the threshold, which is similar to the ELM model. The triangular-based and sigmoid-based models display a similar trend: the RMSE reaches the lowest point at 3.11% when the node number is set as 80 for the sigmoid model, while the RMSE reaches the lowest point at 3.48% when the node number is set as 94 for the tribasis based models.

For comparison, the classical ANN model was also used in this paper, and the sigmoid kernel is selected in the ANN model. The applied ANN architecture is a feedforward neural network, which is a network structure in which the information will propagate in one direction from input to output. A back-propagation algorithm with Levenberg-Marquardt training function has been used for training. This algorithm can approximate any nonlinear continuous function to an arbitrary accuracy. Based on the experimental tests, and when the node number of the hidden layer is set as 36, the ANN network model can obtain the best prediction accuracy.

5. Results and Discussion

When the parameters for the neural network model are finally settled, the next step is to validate the model using a testing

TABLE 4: The calculated performance indicators for ANN, ELM, and USA model.

Data set	Performance indicator	ANN	ELM	USA
Training	RMSE	1.51	0.95	0.78
	MAE	7.5	4.2	2.1
	VAF	96.88	100	100
	R^2	0.91	0.93	0.98
Testing	RMSE	3.56	3.11	2.76
	MAE	12.7	9.7	7.65
	VAF	91.88	92.82	94.52
	R^2	0.90	0.92	0.94

data set. In this section, prediction performance is made between ANN, ELM, and USA models, while training time is recorded. The model accuracy can be evaluated by comparing the calculated RMSE, VAF, MAE, and R^2 performance indicators for training and testing data sets presented in Table 3. Due to random subdivision processes, the performances of the models for training and validation data sets are slightly different. As seen from Table 4, the prediction performance in terms of the predefined indices is better for the testing set than for the training set. Overall, R^2 values for the developed models were more than 0.95 at the training phase, while, in the case of the conventional ANN at the testing phase, R^2 was found to be 0.9. The high performance of the models for the testing set can be considered to be an indication of the good generalization capabilities of the models. R^2 value greater than 0.9 indicates a very satisfactory model performance, while R^2 value between 0.8 and 0.9 indicates an acceptable model. Therefore, all of the developed neural networks are competent for ROP prediction. R^2 of 0.99 for both the ELM and USA models in the training phase indicates higher accuracy compared to the ANN model (R^2 of 0.88). Moreover, it also can be observed that the highest VAF of 94.52, the lowest RMSE of 2.76 and MAE of 7.65 belong to the USA model, indicating that the USA based model gives better prediction performance than the other models. In addition, the ELM model can also produce acceptable results, which yielded slightly fewer residual errors than the ANN models. In other words, the deviation from the observed values of penetration rate predicted by ELM is less than the deviation of the ANN model's prediction. Figure 5 shows the regression analysis of the predicted and measured ROP by different models in both training and testing phases. According to the running speed with the increase of hidden numbers in Figure 6, the obvious differences are observed, concluding that the cost time for the ELM model is less than the other two models. To visualize the quality of the prediction, predicted penetration rates and residual errors by different models are compared with the measured ones for the overall data set and shown in Figures 7 and 8, respectively. This low deviation obtained by three models also proves that ANN, ELM, and USA are competent for ROP prediction. Thus, the ELM and USA methods can both achieve high accuracy and maintain high running speeds. This study shows that ELM technology is a promising tool for ROP prediction, and this work can be incorporated into a software system that can be used in drilling optimization guidance.

6. Conclusions

(1) This study investigates the feasibility of ELM and its variation, the USA model, for predicting the ROP of offshore drilling operations using recoded data. This ELM and USA models can successfully predict ROP in real time, and this prediction can be used as a reference range for drilling engineers identifying drilling problems and making decisions.

(2) Choosing the relevant parameters is typically difficult, and previous literature informed input selection. Input rock material properties are the uniaxial compressive strengths of the different rock types, formation drillability, and abrasiveness, which can be calculated from lab cores and well logging data. Moreover, the pump pressure and differential pressure are selected as important hydraulic parameters that affect ROP, while RPM and WOB are treated as the equipment operational parameters. In addition to these parameters, drilling mud properties such as mud viscosity and bit related parameters are also involved in developing neural network models. However, for systems with data values beyond the range of this study or for a new bit, it is necessary to develop new neural networks. Conversely, the wider range of input can enable the neural network models developed to have wider applications in select cases.

(3) The results of the USA model were compared to those of the classical ANN and ELM models. Performance of the models was checked by using R^2, MAE, RMSE, and VAF performance indicators. According to the performance indicators, all of the neural networks are competent for ROP prediction, but the prediction performances of the USA model were found to be better than those of the other two models with respect to accuracy, while the ELM model had the lowest running speed. Compared to conventional ANN models, the result of this work shows the potential of some fast and flexible neural network approaches to model the ROP with high accuracy while maintaining running speed. Therefore, drilling engineers can make better
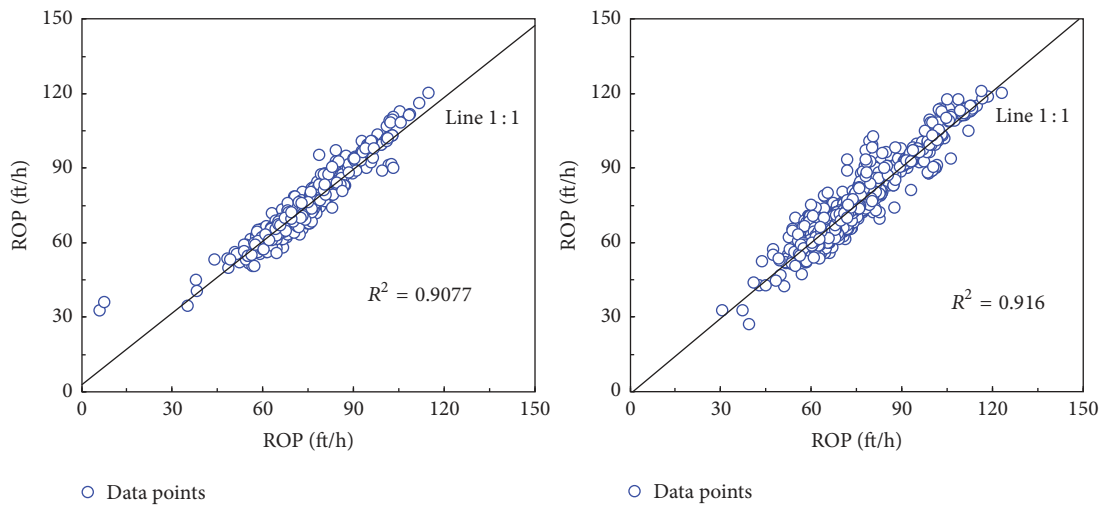
(a) ANN testing and training
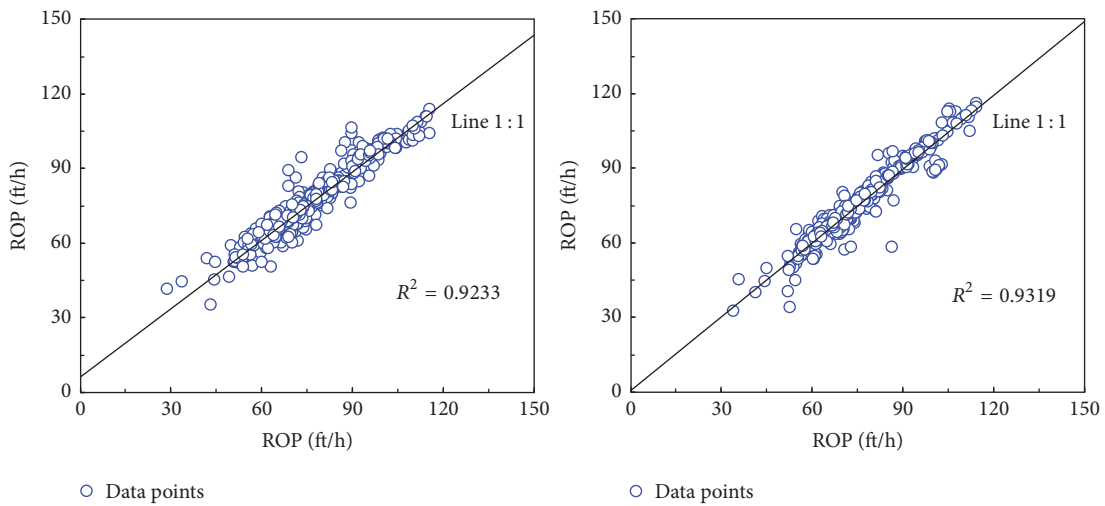
(b) ELM testing and training
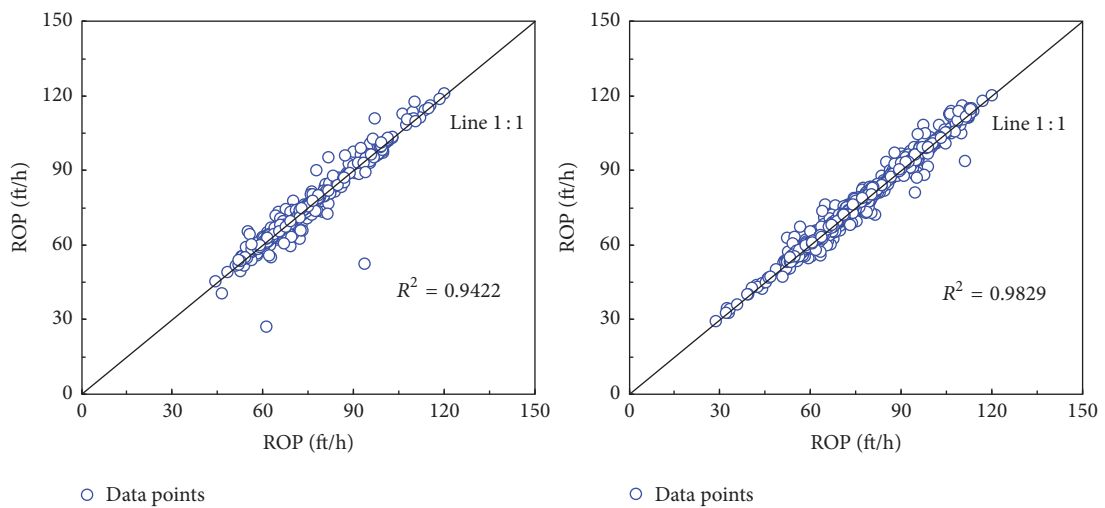
(c) USA testing and training

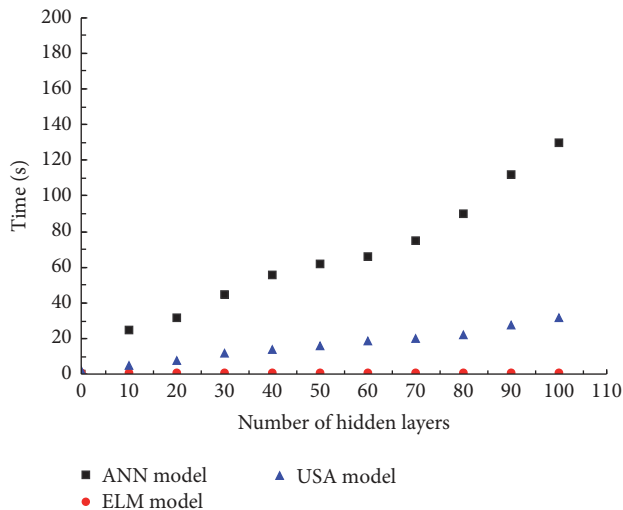FIGURE 5: Regression analysis of the predicted and measured ROP.

FIGURE 6: Time cost with the change of number of hidden layers.
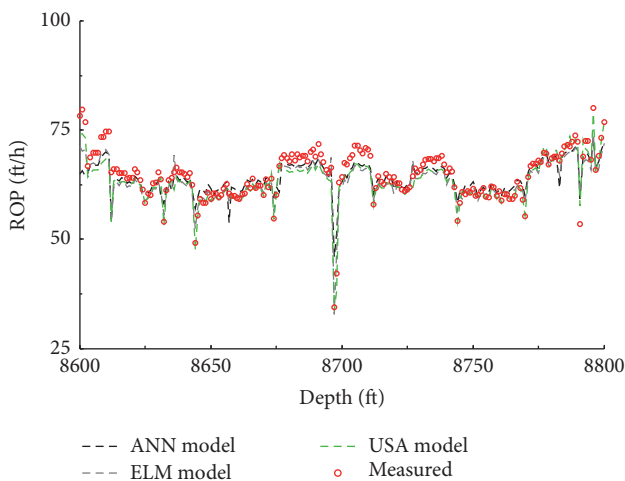

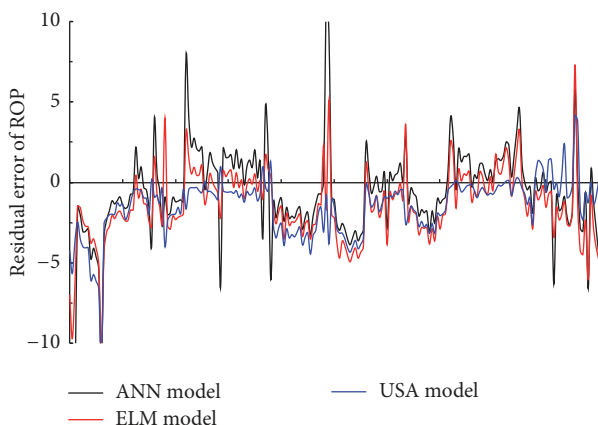FIGURE 7: The predicted and measured ROP along the depth with developed models.

FIGURE 8: Residual errors of predicted ROP by developed models.

choices according to accuracy and computational demand in practical use.

Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.
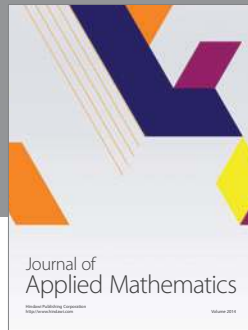
Acknowledgments

References

[1] V. P. Perrin, M. G. Wilmot, and W. L. Alexander, "Drilling index—a new approach to bit performance evaluation," in *Proceedings of the SPE/IADC Drilling Conference*, Paper SPE 37595, pp. 199–205, Amsterdam, The Netherlands, March 1997.

[2] M. Bataee and S. Mohseni, "Application of artificial intelligent systems in ROP optimization: a case study in Shadegan oil field," in *Proceedings of the SPE Middle East Unconventional Gas Conference and Exhibition (UGAS '11)*, pp. 13–22, February 2011.

[3] M. H. Bahari, A. Bahari, F. N. Moharrami, and M. B. N. Sistani, "Determining Bourgoyne and Young model coefficients using genetic algorithm to predict drilling rate," *Journal of Applied Sciences*, vol. 8, no. 17, pp. 3050–3054, 2008.

[4] D. F. Howarth, W. R. Adamson, and J. R. Berndt, "Correlation of model tunnel boring and drilling machine performances with rock properties," *International Journal of Rock Mechanics and Mining Sciences*, vol. 23, no. 2, pp. 171–175, 1986.

[5] S. Kahraman, "Correlation of TBM and drilling machine performances with rock brittleness," *Engineering Geology*, vol. 65, no. 4, pp. 269–283, 2002.

[6] S. Akin and C. Karpuz, "Estimating drilling parameters for diamond bit drilling operations using artificial neural networks," *International Journal of Geomechanics*, vol. 8, no. 1, pp. 68–73, 2008.

[7] V. Uboldi, L. Civolani, and F. Zausa, "Rock strength measurements on cuttings as input data for optimizing drill bit selection," in *Proceedings of the SPE Annual Technical Conference and Exhibition*, Paper SPE 56441, pp. 35–43, Houston, Tex, USA, October 1999.

[8] M. J. Fear, "How to improve rate of penetration in field operations," in *Proceedings of the SPE/IADC Drilling Conference*, Paper SPE 55050, New Orleans, La, USA, 1998.

[9] R. Rommetveit, K. S. Bjørkevoll, G. W. Halsey et al., "Drilltronics: an integrated system for real-time optimization of the drilling process," in *Proceedings of the IADC/SPE Drilling Conference*, vol. 87124, pp. 275–282, SPE, Dallas, Tex, USA, March 2004.

[10] A. H. Paiaman, M. K. Ghassem Al-Askari, B. Salmani, B. D. Al-Anazi, and M. Masihi, "Effect of drilling fluid properties on rate of Penetration," *NAFTA*, vol. 60, no. 3, pp. 129–134, 2009.

[11] A. Bahari and A. B. Seyed, "Trust region approach to find constants of Bourgoyne and Young penetration rate model in Khangiran Iranian gas field," in *Proceedings of the SPE Latin American and Caribean Petroleum Engineering Conference*, Paper SPE 107520, Buenos Aires, Argentina, 2007.

[12] H. I. Bilgesu, L. T. Tetrick, U. Altmis, Sh. Mohaghegh, and S. Ameri, "A new approach for the prediction of penetration (ROP) values," in *Proceedings of the SPE Eastern Regional Meeting*, Paper SPE 39231, Loaington, Kentucky, 1997.

[13] R. Jahanbakhshi, R. Keshavarzi, and A. Jafarnezhad, "Real-time prediction of rate of penetration during drilling operation in oil and gas wells," in *Proceedings of the 46th US Rock Mechanics/Geomechanics Symposium*, pp. 2390–2396, Chicago, Ill, USA, June 2012.

[14] S. Yilmaz, C. Demircioglu, and S. Akin, "Application of artificial neural networks to optimum bit selection," *Computers and Geosciences*, vol. 28, no. 2, pp. 261–269, 2002.

[15] H. Rahimzadeh, M. Mostofi, A. Hashemi, and K. Salahshoor, "Comparison of the penetration rate models using field data for one of the gas fields in Persian Gulf Area," in *Proceedings of the International Oil and Gas Conference and Exhibition*, Paper SPE 131253, Beijing, China, 2010.

[16] G.-B. Huang, D. H. Wang, and Y. Lan, "Extreme learning machines: a survey," *International Journal of Machine Learning and Cybernetics*, vol. 2, no. 2, pp. 107–122, 2011.

[17] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 42, no. 2, pp. 513–529, 2012.

[18] A. Mozaffari and N. L. Azad, "Optimally pruned extreme learning machine with ensemble of regularization techniques and negative correlation penalty applied to automotive engine cold-start hydrocarbon emission identification," *Neurocomputing*, vol. 131, pp. 143–156, 2014.

[19] R. Moreno, F. Corona, A. Lendasse, M. Graña, and L. S. Galvão, "Extreme learning machines for soybean classification in remote sensing hyperspectral images," *Neurocomputing*, vol. 128, pp. 207–216, 2014.

[20] D. Yu and D. Li, "Efficient and effective algorithms for training single-hidden-layer neural networks," *Pattern Recognition Letters*, vol. 33, no. 5, pp. 554–558, 2012.

[21] C. Gokceoglu and K. Zorlu, "A fuzzy model to predict the uniaxial compressive strength and the modulus of elasticity of a problematic rock," *Engineering Applications of Artificial Intelligence*, vol. 17, no. 1, pp. 61–72, 2004.

[22] J. Zhang, J. Li, Y. Hu, and J. Zhou, "The identification method of igneous rock lithology based on data mining technology," *Applied Mechanics and Materials*, vol. 466-467, pp. 65–69, 2012.

[23] J. Cao, J. Yang, Y. Wang, D. Wang, and Y. Shi, "Extreme learning machine for reservoir parameter estimation in heterogeneous sandstone reservoir," *Mathematical Problems in Engineering*, vol. 2015, Article ID 287816, 10 pages, 2015.

[24] E. Khamehchi, I. R. Kivi, and M. Akbari, "A novel approach to sand production prediction using artificial intelligence," *Journal of Petroleum Science and Engineering*, vol. 123, pp. 147–154, 2014.

[25] K. Amar and A. Ibrahim, "Rate of penetration prediction and optimization using advances in artificial neural networks, a comparative study," in *Proceedings of the 4th International Joint Conference on Computational Intelligence (IJCCI '12)*, pp. 647–652, Barcelona, Spain, October 2012.

[26] D. Moran, H. Ibrahim, A. Purwanto, and J. Osmond, "Sophisticated ROP prediction technology based on neural network delivers accurate results," in *Proceedings of the IADC/SPE Asia Pacific Drilling Technology Conference and Exhibition*, SPE 132010, pp. 100–108, Ho Chi Minh City, Vietnam, November 2010.

[27] M. Monazami, A. Hashemi, and M. Shahbazian, "Drilling rate of penetration prediction using artificial nerual network: a case study of one of Iranian southern oil fields," *Journal of Oil and Gas Business*, no. 6, pp. 21–31, 2012.

[28] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.

[29] X. Liu, Y. Chang, H. Lu et al., "Optimizing fracture stimulation in low-permeability oil reservoirs in the ordos basin," in *Proceedings of the SPE Asia Pacific Oil and Gas Conference and Exhibition*, Paper SPE158562, Perth, Australia, 2012.