

## Research Article

# Ensemble Prediction Algorithm of Anomaly Monitoring Based on Big Data Analysis Platform of Open-Pit Mine Slope

Song Jiang <sup>1</sup>, Minjie Lian,<sup>1,2</sup> Caiwu Lu <sup>1</sup>, Qinghua Gu <sup>1</sup>, Shunling Ruan <sup>1</sup>  
and Xuecai Xie <sup>3</sup>

<sup>1</sup>School of Management, Xi'an University of Architecture and Technology, Shaanxi 710055, China

<sup>2</sup>Sinosteel Mining Co. Ltd., Beijing 100080, China

<sup>3</sup>College of Resource and Safety Engineering, China University of Mining and Technology (Beijing), Beijing 100083, China

Correspondence should be addressed to Qinghua Gu; [qinghuagu@126.com](mailto:qinghuagu@126.com)

Received 7 April 2018; Revised 19 May 2018; Accepted 30 May 2018; Published 1 August 2018

Academic Editor: Zhihan Lv

Copyright © 2018 Song Jiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the diversification of pit mine slope monitoring and the development of new technologies such as multisource data flow monitoring, normal alert log processing system cannot fulfil the log analysis expectation at the scale of big data. In order to make up this disadvantage, this research will provide an ensemble prediction algorithm of anomalous system data based on time series and an evaluation system for the algorithm. This algorithm integrates multiple classifier prediction algorithms and proceeds classified forecast for data collected, which can optimize the accuracy in predicting the anomaly data in the system. The algorithm and evaluation system is tested by using the microseismic monitoring data of an open-pit mine slope over 6 months. Testing results illustrate prediction algorithm provided by this research can successfully integrate the advantage of multiple algorithms to increase the accuracy of prediction. In addition, the evaluation system greatly supports the algorithm, which enhances the stability of log analysis platform.

## 1. Introduction

Landslide is one of the most worldwide and dangerous natural disasters, which is a serious life threat for people working at open-pit mine slopes. Because of the complexity and uncertainties of the landslide formation and causing factors, landslide surveillances and predictions are always highly paid attention to by international researchers. With the wild application of systematic science and nonlinear science, researchers realized that landslide body is an open and complex system. By using sensor anomaly monitoring technology, massive amount of alert logs will be generated in the process of microseismic monitoring. These logs contain the time, place, alert level, and other relevant data of anomalous incidents. There are many researchers which have investigated the microseismic monitoring models in recent years [1–3]. Anomaly monitoring techniques are varied based on different types of analysis subjects. Among those, monitoring methods based on anomaly history and system status are most generally applied. Previously, researchers have

focused on factors such as intrusion detection system [4, 5], prediction model [6, 7], and system call [8, 9]. Nohra et al. [10] suggest that near continuous monitoring via infrared spectroscopy is safe and accurate for use in critically ill surgical and trauma patients. Vinoth et al. [11] propose that PC-based microseismic network consisting of geophones, data loggers, GPS synchronization, and Ethernet antennas for wireless communication is employed to study the impact of induced seismicity on the slope failures in real time.

Monitoring method based on anomaly history majorly collects the anomalous data of complex systems (e.g., slope microseismic monitoring alert logs). Machine learning and data mining are used in anomaly analysis in order to discover the potential correlations between history anomaly and future anomaly, which helps the prediction of future landslides. Many methods have been proposed in the literature such as neural network [12, 13]. Based on modified genetic algorithm, back propagation neural network classification algorithm, and the landslide disaster prediction theory, taking into account the rainfall and other uncertainties in

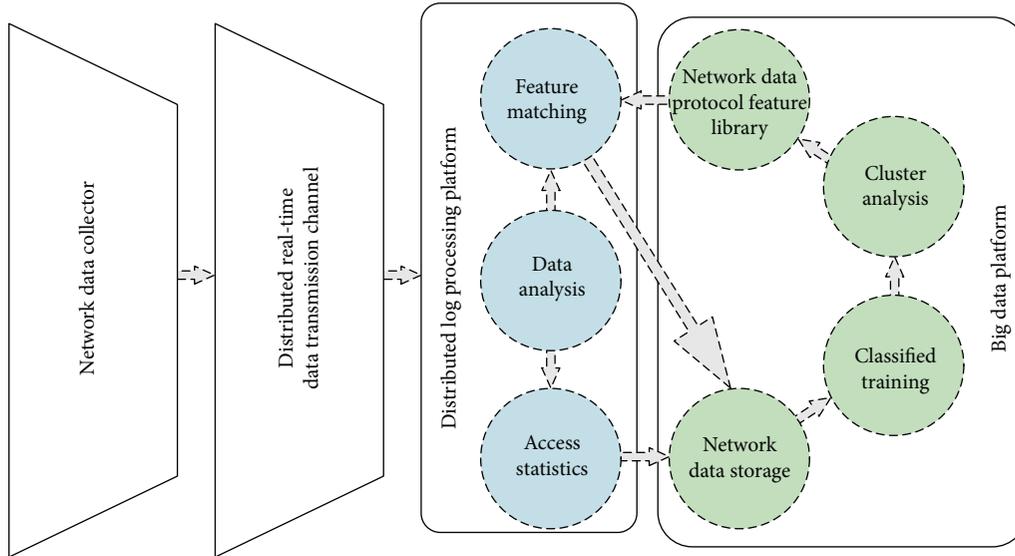


FIGURE 1: System structure.

landslide, this paper proposes the concept of separation of uncertain data/elaborates the processing methods of uncertain property data and builds uncertain genetic neural network and the landslide hazard prediction model. However, the method often results in such problems as local minimization and slow convergence rate. Most of previous researches only employ dichotomy algorithm of SVM [14, 15], which is far from enough for anomaly monitoring system in open-pit mine. There are also methods used in this area, such as grey prediction model [16, 17], fuzzy clustering [18, 19], and numerical simulation [20–22], which also have its respective disadvantages that will not be listed one by one. With the coming of big data age, data collected is growing in an exponential rate. In the areas of big data analysis, Jayasena et al. [23] propose the ant colony optimization (ACO) algorithm for efficient resource allocation in infrastructure layer. Pop et al. [24] orient on computer and information advances aiming to develop and optimize advanced system software, networking, and data management components to cope with big data processing and the introduction of autonomic computing capabilities for the supporting large-scale platforms. Traditional alert log analysis can no longer fulfil the need in discovering the relationship between anomalous data and landslides. Thus, new alert log analysis platforms are invented for alert log data mining at the scale of big data.

## 2. Alert Log Analysis System Based on Big Data Platforms

Not only does the individual system log analysis require distributed data mining techniques but it also has a low recall rate in ineffective predictions. Single system is a system that uses only one processor to process data, while cluster system is a distributed system or concurrent system that comprises of computers connected together. Currently, there is increasing attention paid to cluster system rather than single system [25–27]. Some scholars have focused on distributed storage,

arguing that the MapReduce computing model is effective and easy to process mass data in distributed computing [28–30]; Vahidipour et al. [31] compare several weighted combination methods in which the hierarchical clustering results are used in order to derive a consensus hierarchical clustering. Song et al. [32] propose a two-tiered on-demand resource allocation mechanism consisting of the local and global resource allocation with feedback to provide on-demand capacities to the concurrent applications. Other scholars are interested in the combination of transfer learning and class imbalance learning. For example, Ryu et al. [33] through comparative experiments with the transfer learning and the class imbalance learning techniques, it shows that the proposed model provides significantly higher defect detection accuracy while retaining better overall performance. The system employs distributed processing to achieve real-time monitoring system log and uses distributed storage of big data platform and the capacity of data calculation and analysis to realize the distributed storage of network. The system includes network data collector, distributed real-time data transmission channel, distributed log processing platform, network data protocol feature library, and big data platform. The structure of the system is showed in Figure 1.

The main function and process flow are presented below:

- (1) Data is sent through distributed real-time data transmission channels. Network data collectors will send data packets on each device to distributed log processing platforms.
- (2) Distributed log processing platforms start real-time process for packets acquired and match data characteristics by using network protocol characteristics library.
- (3) Big data platforms perform cluster analysis and classification training to stored log data and update dynamically to network protocol characteristics library described.

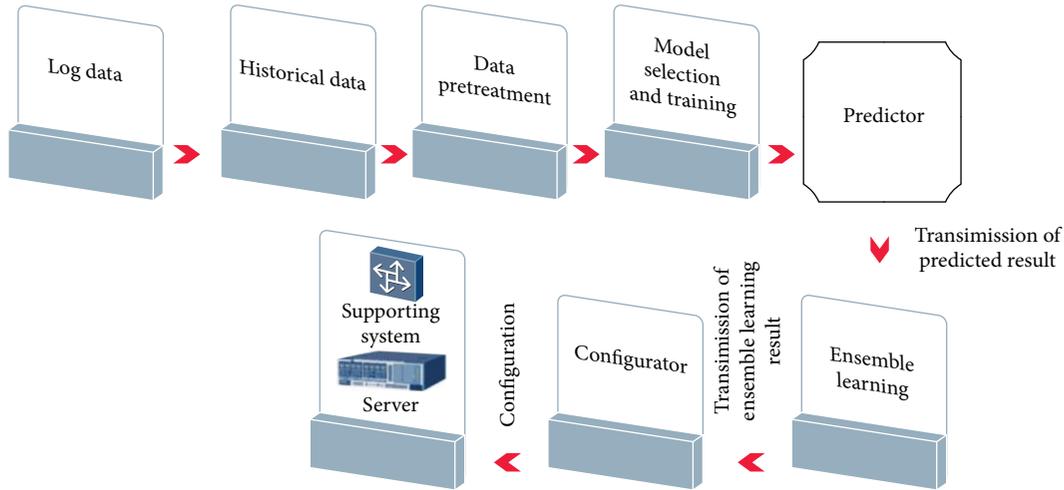


FIGURE 2: Ensemble prediction system structure.

The current cloud calculation model is MapReduce provided by Google laboratory. Besides Hadoop MapReduce and Hive, Map-Only and Iterative are also applied in MapReduce calculation model as discussed by Sobreira et al. [34]. In addition, based on Hadoop and MapReduce, some researchers have also developed new effective programming pattern PDMiner which is a parallel distributed data mining system [35]. The system uses Cloudera CDH as supporting software for big data platform and distributed stream as processing platform Spark Streaming to analyze, match, and count network data packets. A trunking system unit like this is composed by 5 nodes. Each node has equipped 24 CPUs, 127 GB RAM, and 10 hard drives. Data is collected via TcpDump protocol, which uses interchangers to send data from one or multiple ports to other ports for network monitoring without interfering other programs. Apache Kafka is used for real-time data transmission, sending data to Spark Streaming distributed streaming platform for real-time analysis. Results will be stored at big data platforms and provided for systematic training and network protocol data library updates by using support vector machine (SVM) and Bayesian structure classifier with big data modules such as MapReduce, Hive, and Mahout. The log analysis experiment on such platforms indicates that log analysis system has an acceptable monitoring accuracy and time consumption. However, in the circumstances of having anomaly network traffic, the statistics of traffic attributes have a decreasing trend. In order to improve the system log analysis and achieve anomaly detection, this article presents EPABT (ensemble prediction algorithm based on time series) for big data analysis platforms. This system can predict the anomaly quantity of the system in a certain period of time.

### 3. Theoretical Background

This research paper selects multiple EPABT in data mining to establish prediction model and uses two methods for testing. The first method is to separate data set into training set

and testing set. Specifically, the data collected in the last month is used as testing set, the rest data as training set. In this way, the method can achieve anomaly detection via training and testing [36–38]. The second method uses alert logs based on time window to predict anomaly. By using two-stage time window, statics of alert incidents are used to describe network running characteristics, which is the base of decision making features for landslide predictions. The two evaluation systems are correspondingly adopted. The first one is error cost function used in traditional time series, including mean square error (MSE), mean absolute error (MAE), and mean absolute proportional error (MAPE). The other evaluation system uses the precision of prediction, recall rate, and F-Measure as evaluation criteria. New ensemble prediction system structure is depicted in Figure 2.

The workflow of this system includes 3 steps below:

- (1) Analyze, transform, and preprocess historical network system data. Then filter and remove irrelevant data and attributes and transform raw data into a usable format for prediction model.
- (2) Perform model selection and training based on data characteristics. Because of the dynamic attribute of grid systems, model selection and training will be repeated on regular basis with the latest data. System will save all relevant parameters and update existing model after training completed.
- (3) Use individual trained model to predict for future capacity and request. Ensemble learning is used to reduce the variance of predicted results. Ensemble learning uses every available model to predict and determine the weight of each model based on prediction history. In addition, ensemble learning will also use nontime series methods for prediction corrections.

The second testing method needs to set up a time axis and divides it into time windows with different scales. For

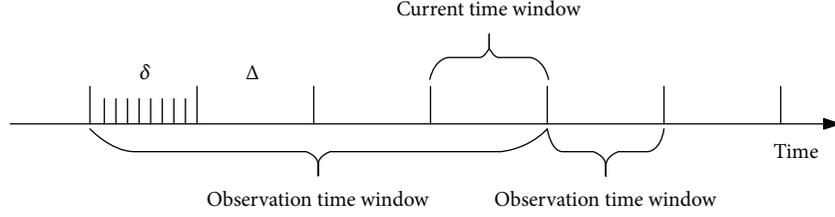


FIGURE 3: Time axis divisions.

microseismic monitoring systems online, system logs and real-time data are considered as a time series. A sliding time window  $\Delta$  is set up. Then finish the prediction with the alert information in the time window and relationship between history alert log and landslides. Time axis divisions are shown in Figure 3.

The definition of time window in this research paper is prediction time window is the time window of predicting the existence of anomaly, with a size of  $\Delta$ . Current time window is defined as the time window before the prediction time window, with a size of  $\Delta$ . Observing time window is the  $n$ th time window before prediction time window, with a size of  $\Delta \times n$ . Sample window is the smaller division of unit time window, with a size of  $\delta$ , number of sample time window in unit time window is defined as  $\Delta/\delta$ . The typical time series analysis prediction methods include average prediction of sliding window, autoregression, artificial neural network, support vector regression, and gene expression programming.

**3.1. Sliding Window Mean Prediction.** Sliding window mean prediction is the simplest method in time series prediction. When using such prediction, a sliding window with a fixed length  $n$  is needed [39]. If we set the prediction time is  $t$ , then the predicted value at time  $t$  is

$$\hat{v}^{(t)} = \text{MA}_{t-1} = \frac{v^{(t-1)} + v^{(t-2)} + \dots + v^{(t-n)}}{n}. \quad (1)$$

$\text{MA}_{t-1}$  denotes the moving average at time  $t$ . By using incremental presentation, average of sliding window can be shown as

$$\hat{v}_{(t)} = \text{MA}_{t-1} \frac{v^{(t-n)}}{n} + \frac{v^{(t-1)}}{n}. \quad (2)$$

The advantage of sliding window mean prediction is its low calculation complexity. However, if time series is unstable, the prediction result will be much less accurate. In addition, it is also difficult in setting a proper sliding window length. If the value of  $n$  is too large, the fluctuation of predicted value will be underestimated. If the value of  $n$  is too small, then the history data cannot be sufficiently used.

**3.2. Autoregression.** Autoregressive (AR) model [40, 41] is another method to predict time series in statistics. The essence of this model is stochastic process. One basic

hypothesis of autoregressive model is to output nonlinear historical value, which can be presented as

$$\hat{v}^{(t)} = c + \varepsilon_t + \sum_{i=1}^n \phi_i v^{(t-i)}. \quad (3)$$

$c$  indicates constant,  $\varepsilon_t$  is random error, and  $\phi_i$  is the parameter of autoregressive model. If the current value has correlation with historical value, the current value can be predicted based on historical data acquired. Thus, the sliding windows mean prediction can be considered as a special case of autoregressive model with all parameter is  $1/n$ . By improving autoregressive model, autoregressive moving average model (ARMA) is also commonly used in such calculations. ARMA model can be presented as  $\text{ARMA}(p,q)$ , in which  $p$  is the number of relevant parameters in AR model,  $q$  is the number of parameters in MA model. For example,  $\text{ARMA}(n,m)$  can be presented as

$$\hat{v}^{(t)} = \sum_{i=1}^n \phi_i v^{(t-i)} + \sum_{i=1}^m \theta_i \varepsilon_{t-i} + c + \varepsilon_t. \quad (4)$$

$\phi_i$  are relevant parameters in AR model ( $n$  terms in total),  $\theta_i$  are the relevant parameters in MA model ( $m$  terms in total).

**3.3. Artificial Neural Network.** If more complex nonlinear relationship exists in data, it is extremely difficult for autoregressive prediction to match the relationship among data. In order to efficiently predict nonlinear data relationships, artificial neural network (ANN) can be inducted [42, 43]. A typical neural network has 3 layers, that is, input layer, hidden layer, and output layer. An artificial neural network can has 0 to multiple hidden layers. An artificial neural network that has 3 layers is presented in Figure 4; one hidden layer is included.

Illustrated in the photo, number of neurons in each layer is 4, 4, and 1. Each input and hidden layer has one deviated neuron which has a fixed value of 1.  $\theta_{ij}^k$  is the weight of neurons from layer  $k$  number  $j$  to layer  $k+1$  number  $i$ .  $a_i^{(k)}$  denotes the value of the  $i$ th neuron at layer  $k$ . If  $k$  is input layer, then the value of neuron can be directly acquired, or the value needs to be calculated from the upper layer (expect deviated neurons). If we set the value of neurons at  $k+1$ th layer as  $a^k = [a_1^{k+1}, \dots, a_m^{k+1}]^T$ , then from (5) we can have

$$a^{k+1} = g(\theta^{(k)} a^{(k)}). \quad (5)$$

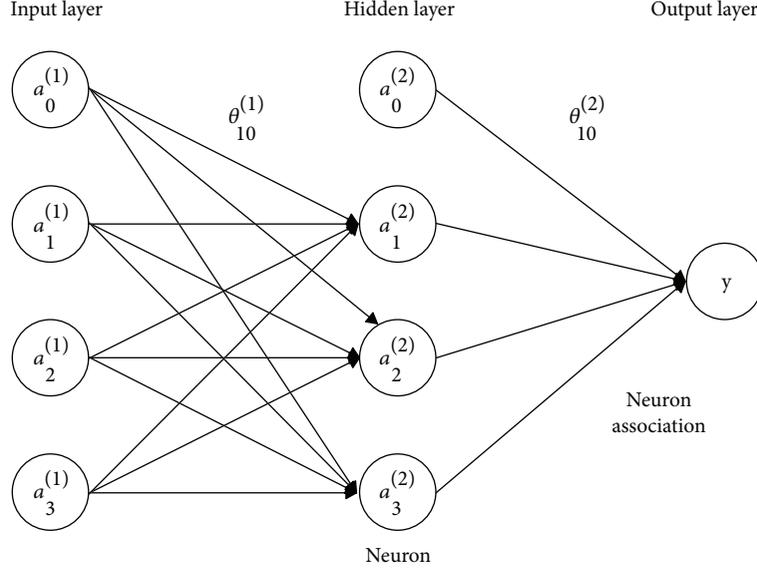


FIGURE 4: Topology structure of artificial neural network.

$\theta_{ij}^k$  is the weigh matrix of neurons from the  $k$ th layer to the  $k+1$ th layer.

$a^{(k)} = [a_0^{(k)}, a_1^{(k)}, \dots, a_n^{(k)}]^T$  is the vector value of all neurons on the  $k$ th layer. For example, the hidden layer neuron in figure four can be presented in 6 as

$$\begin{aligned} a_1^{(2)} &= g\left(\theta_{10}^{(1)} a_0^{(1)} + \theta_{11}^{(1)} a_1^{(1)} + \theta_{12}^{(1)} a_2^{(1)} + \theta_{13}^{(1)} a_3^{(1)}\right), \\ a_2^{(2)} &= g\left(\theta_{20}^{(1)} a_0^{(1)} + \theta_{21}^{(1)} a_1^{(1)} + \theta_{22}^{(1)} a_2^{(1)} + \theta_{23}^{(1)} a_3^{(1)}\right), \\ a_3^{(2)} &= g\left(\theta_{30}^{(1)} a_0^{(1)} + \theta_{31}^{(1)} a_1^{(1)} + \theta_{32}^{(1)} a_2^{(1)} + \theta_{33}^{(1)} a_3^{(1)}\right). \end{aligned} \quad (6)$$

$g(z) = 1/1 + e^{-z}$  denotes the Sigmoid function of variable  $z$ . The purpose of this function is to change to range of  $z$  from the entire real number set to an open interval of  $(-1,1)$ . If Figure 4 is a trained neural network (all neural weight related values  $\theta_{ij}$  can be acquired from training set), the output  $\hat{y}$  can be shown as

$$\hat{y} = g(z) = \frac{1}{1 + e^{-z}} = \frac{1}{1 + e^{-f(a^{(2)})}}. \quad (7)$$

In the formula  $z = f(a^{(2)}) = \theta_{10}^2 a_0^{(2)} + \theta_{11}^2 a_1^{(2)} + \theta_{12}^2 a_2^{(2)} + \theta_{13}^2 a_3^{(2)}$  is to acquire hidden layer output at output layer. The common method used in neural network learning is gradient descent algorithm. If we set  $j(\theta)$  as error functions in a neural network with  $k$  input neurons

$$j(\theta) = -\frac{1}{m} \sum_{i=1}^m (|y - \hat{y}|)^2. \quad (8)$$

In the formula,  $\theta$  indicates the weight matrix, and  $y - \hat{y}$  denotes the difference between actual value and predicted value.

Because artificial neural network update the weight after learning the entire training set and also has a complex topology structure, the gradient descent algorithm will be much less efficient and take much more time in training. In order to make up this disadvantage and increase training speed, stochastic gradient descent algorithm that is a more practical method, can update the weight of make each sample by processing gradient descent.

**3.4. Support Vector Machine.** Support vector machine (SVM) is a supervised learning algorithm for solving classification problems with high accuracy [44]. Because this algorithm always prefers to find the decision hyperplane with the largest margin, the algorithm has great generalization of testing samples. Support vector regressor machine is another variation of support vector machine. Its basic principle is the features of training machine data are mapped from  $n$ -dimension space into a higher  $m$ -dimension space through nonlinear transformation vertebrata (also called kernel function), and then linear regression is preceded in the higher space. The basic idea of regressor can be illustrated as

$$f(x) = \omega \Phi(x) + b \text{ and } \Phi : R \rightarrow \Gamma \omega \in \Gamma. \quad (9)$$

In the formula  $\omega = [\omega_1, \omega_2, \dots, \omega_m]^T$  denotes the weight of each characteristic of original samples in  $m(m > n)$  dimensions after mapping.  $b$  is threshold value. If mapping function  $\Phi$  is deleted in (9), (9) will be a common expression of linear regression.

By setting up a trained support vector regressor machine, the predicted value can be calculated in the time of  $O(m)$ . For example, if the data of a time series is  $v = v^{(t-m)}, v^{(t-n+1)}, \dots, v^{(t)}$  (the range is regulated to  $[-1,1]$ ), then the predicted value at time  $t+1$  is  $v^{(t+1)} = \omega \cdot \Phi(v) + b$ . The purpose of model training is to minimized the error calculated by using (9) after kernel function is given, which is presented in (10)

$$R_{\text{reg}} = \sum_{i=1}^l C(f(x_i - y_i)) + \lambda \|\omega\|^2. \quad (10)$$

In the formula,  $x_i$  denotes the training sample,  $y_i$  denotes the actual value of concentrated training,  $C(\cdot)$  is the cost function, and  $\lambda \|\omega\|^2$  is used to control the complexity of the model. Punishment will be given to weight values which increase the complexity of the model. Quadratic programming can be used to optimize most of the cost functions  $C(\cdot)$ .  $\epsilon$ -intensitive and Huber cost functions are the common cost function candidates.

**3.5. Gene Expression Programming.** Gene expression programming (GEP) is a new type of bionic algorithm originated from biological field [45]. It inherits advantages of traditional genetic algorithm and genetic programming. The reason it can manifest any complicated expression is that it codes according to chromosome expressions. In terms of structure, a chromosome is expressed by an array, but in terms of logic, a chromosome embodies an expression tree.

As shown in Figure 5, there is a simple expression  $\sqrt{(a+b) + (c*d)}$ , its corresponding array and expression tree. For the training of model, gene expression programming optimizes the model through group evolution pattern. A population usually has 40–100 chromosomes whose chromosome expressions have strong diversities. Chromosomes in populations evolve through replication, mutation, recombination, inversion, and other operations under the lead of fitness function, as shown in Figure 6.

## 4. Ensemble Algorithm Based on Time Series and Assessment Standards

**4.1. Ensemble Algorithm.** Since a single prediction model is difficult to be predicted precisely, this article employs the strategy of ensemble learning, combining prediction abilities of different prediction models to present an ensemble prediction algorithm based on time series. The strategy of ensemble learning has already been widely used in classification problems of data mining. In classification problems, data samples are regarded as independent and identically distributed. However, for the problems of time series analysis, strong connections of time dimension are existed between samples. Moreover, since prediction values of time series analysis are consecutive, classified voting mechanism cannot be used to get the final result. For the ensemble learning of time series analysis, its strategy is mainly to update each predictor's weight by predicting assessment standards. The name of each algorithm and related descriptions are shown in Table 1.

In order to achieve the predicted result mentioned above, this article proposes a strategy of weighted linear combination. Assume the predicted result of prediction algorithm  $p \in P$  at moment  $t$  is  $\hat{v}_p^{(t)}$ , and its weight at moment  $t$  is  $w_p^{(t)}$ , so at moment  $t$  the predicted value of some journal is

$$\hat{v}^{(t)} = \sum_p w_p^{(t)} \hat{v}_p^{(t)} \text{ and } \sum_p w_p^{(t)} = 1. \quad (11)$$

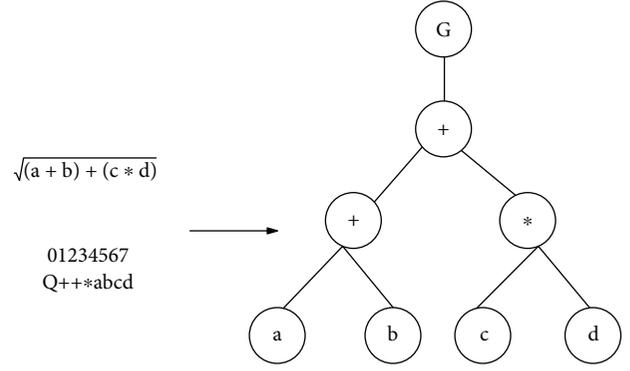


FIGURE 5: Array of gene expression programming and the expression tree.

At initial status,  $t = 0$ , all prediction algorithms have the same contribution to predicted results, like  $w_p^{(t)} = 1/|P|$ . Based on the difference between the update strategy of the weight of ensemble algorithm and the basic strategy of traditional classification, in predicted scene of classification, results can only be indicated as “right” or “wrong,” but the purpose of updating the weight of ensemble algorithm is just to enhance the weight of predictors which have the right result. In prediction scene, predicted results are consecutive values and the weight of prediction algorithm directly affects the result of ensemble algorithm. In order to achieve the update of ensemble prediction algorithm's weight, by using the difference between each predicted value  $\hat{v}_p^t$  and actual value  $v^{(t)}$ , the relative error of prediction algorithm  $i$  at moment  $t$  can be expressed as

$$e_i^{(t)} = \frac{c_i^{(t)}}{\sum_p c_p^{(t)}} w_i^{(t)}. \quad (12)$$

In this equation,  $c_i^{(t)}$  (or  $c_p^{(t)}$ ) shows the prediction algorithm  $i$  or the predicted cost of  $p$  calculated by predicted assessed cost function MAE, LSE, and MAPE. Relative error does not have normalization. Thus, it cannot be used in the updated weight of prediction algorithm. Because the final predicted result is a linear combination of multiple prediction algorithms, the weight is normalized as

$$w_i^{(t+1)} = \frac{e_i^{(t)}}{\sum_p e_p^{(t)}}. \quad (13)$$

According to the strategy of the updated weight, it is guaranteed that the weight of optimal prediction algorithm at each moment increases. In order to face the scene predicted by outlier of slope of open-pit mine's microseismic monitoring, this article proposes asymmetric error cost (AEC) function to evaluate prediction error. AEC is an asymmetric heterogeneous error cost function. The over high prediction and over low prediction are composed by two different costs, manifested as  $R(v^{(s)}, \hat{v}^{(s)})$  and  $P(v^{(s)}, \hat{v}^{(s)})$ , where  $v^{(s)}$  shows the anomaly at future moment  $s$  and  $\hat{v}^{(s)}$

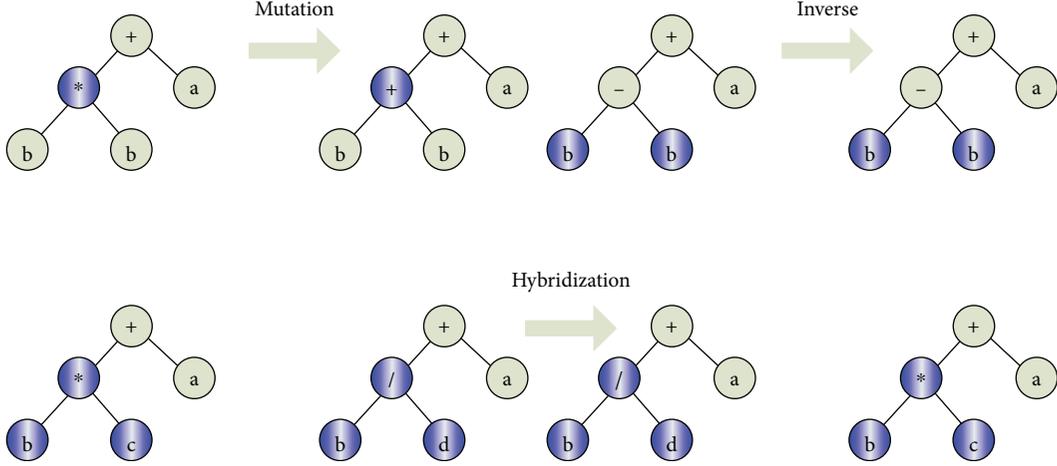


FIGURE 6: The expression change of gene expression programming.

TABLE 1: Characteristics of each time series prediction's algorithm.

Method's names	Method's descriptions	Advantages	Disadvantages
NARMAX	Naive prediction	Eliminate the effect of random fluctuation	Need a large number of previous data records
Autoregressive	Linear regression	Few data needed. Series of independent variables can be used for prediction	Must have autocorrelation
Neural network	Distributed parallel information processing	Functions of self-learning and associative memory	Data are easily lost. Local minimization problem
SVM	Nonlinear kernel Linear learning algorithm	Seize key samples. Eliminate a huge number of redundant samples	Difficult to be implemented for large training samples
GEP	Heuristic algorithm	Use simple code to solve difficult problems	Easily fall into local optimum

shows the number of abnormal prediction at future moment  $s$ . Therefore, the entire cost function can be expressed as

$$A = \beta P(v^{(s)}, \hat{v}^{(s)}) + (1 - \beta)R(v^{(s)}, \hat{v}^{(s)}). \quad (14)$$

In this equation,  $\beta$  is a parameter for adjusting two types of cost weight. Through the change of value  $\beta$ , the cost of weight which is over high predicted or over low predicted can be adjusted manually. AEC is a generalized error cost function, which is able to satisfy the need of application scenario in this article. The expression form of AEC varies according to specific applications. Defined  $P$  function and  $R$  function satisfy two properties: nonnegativity and consistency.

- (1) Nonnegativity: for any nonnegative  $v^{(s)}$  and  $\hat{v}^{(s)}$ , both  $P(v^{(s)}, \hat{v}^{(s)}) \geq 0$ ,  $R(v^{(s)}, \hat{v}^{(s)}) \geq 0$  exist.
- (2) Consistency: if  $v_1(s) - \hat{v}_1(s) \geq v_2(s) - \hat{v}_2(s)$  then  $P(v_1(s), \hat{v}_1(s)) - P(v_2(s), \hat{v}_2(s))$ , negativity and positivity should be consistent; likewise,  $v_1(s) - \hat{v}_1(s) \geq v_2(s) - \hat{v}_2(s)$ ,  $R(v_1(s), \hat{v}_1(s)) - R(v_2(s), \hat{v}_2(s))$  the property of positive and negative should be obeyed as well.

In the prediction process of outlier of slope of open-pit mine's microseismic monitoring, assume the cost of

precise prediction is  $C_{\text{normal}}$ , the cost of over low prediction is  $C_{\text{under}}$ , the cost of over high prediction is  $C_{\text{over}}$ . Often,  $C_{\text{under}}$  is uncertain, but what is certain is that  $C_{\text{under}} \gg C_{\text{normal}}$ , so  $P$  function is AEP cost function can be described as

$$P(v^{(s)}, \hat{v}^{(s)}) = \min(v^{(s)}, \hat{v}^{(s)})C_{\text{normal}} + \max(0, v^{(s)} - \hat{v}^{(s)})C_{\text{under}}. \quad (15)$$

And  $R$  function in AEC cost function is

$$R(v^{(s)}, \hat{v}^{(s)}) = \max(0, \hat{v}^{(s)} - v^{(s)})C_{\text{over}}. \quad (16)$$

Combining (15) and (16), we can get the specific measure of AEC cost function

$$A = f(v^{(s)}, \hat{v}^{(s)}) = \beta P(v^{(s)}, \hat{v}^{(s)}) + (1 - \beta)R(v^{(s)}, \hat{v}^{(s)}) = \begin{cases} \beta v^{(s)}C_{\text{normal}} + (1 - \beta)(\hat{v}^{(s)} - v^{(s)})C_{\text{over}} & \text{当 } \hat{v}^{(s)} \geq v^{(s)} \\ \beta(\hat{v}^{(s)}C_{\text{normal}} + (v^{(s)} - \hat{v}^{(s)})C_{\text{under}}) & \text{当 } \hat{v}^{(s)} < v^{(s)}. \end{cases} \quad (17)$$

4.2. Data Pretreatment. In order to test and verify this model, this article uses big data analysis platform of slope

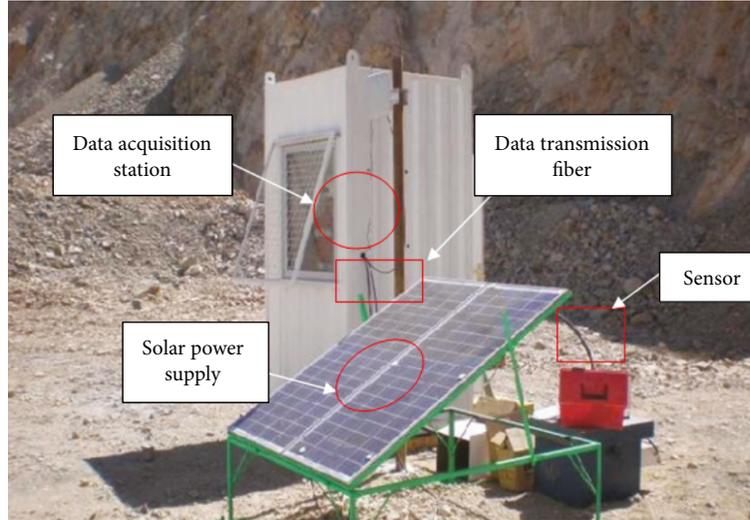


FIGURE 7: The microquake monitoring system.

of open-pit mine's microseismic monitoring [46, 47], collecting monitoring data from Apr. 2017 to Oct. 2017, as shown in Figure 7. Therein, microseismic monitoring uses ARAMIS M/E microseismic monitoring system from Polish accompany EMAG, as shown in Figure 8. Supporting software generates journal document according to shock events. Assume sliding window is  $\Delta = 72$  h, the pace time window slides forwards 3 min, so the system can generate 20 groups of alarm value per hour. Each rating value is calculated based on the monitoring data in recent 72 hours. Through sifting and pretreatment, the effective amount of data is 92752, including the data information of slope monitoring system in a span of 6 months. Alarm report journal data mainly includes

- (1) Level: it means different status of slope abnormal information, which are three statuses in total: NOR, DAN, and CRI.
- (2) Alarm report source: it means the number of specific monitoring sensor which emits alarm report information, starting with 0, which are 0,1,...,100.
- (3) Location information: it describes the location information of the sensor which emits alarm information.
- (4) Alarm time: it indicates the time of recording alarm report information, with the precision of second, such as 02/16/201607:35:06.

Through the analysis of primal logged exception, a huge amount of illegal, redundant records, derivative alarm, noise data like flicker alarm, and so on are detected. Before the prediction, corresponding filtering methods are designed to effectively sift noise data of primal alarm report journal. First is to analyze the characteristics of predicted targets and alarm report journals. Current time window can best represent the current status of the system and is closest to prediction time window. Therefore, each level of alarm report events in current time windows and different types of alarm report events are regarded as features. Extracted features includes

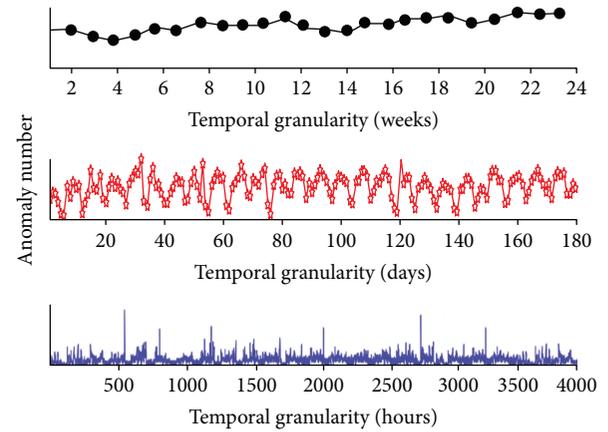


FIGURE 8: The comparison of temporal granularity.

the amount of hint-level alarm report events in current time window, the amount of subordinate-level alarm report events, the amount of important-level alarm report events, the amount of emergency-level alarm report, also failure events, and the amount of every type of alarm report events in current windows.

**4.3. Assessment Standards of the Experiment.** Besides the asymmetric error cost function this article mentions is assessment standard, in order to prove the utility of ensemble prediction algorithm and the accuracy of the assessment standards, this article also adopts other traditional assessment standards [48]. The first prediction pattern adopts normal error cost function as prediction assessment standards. The error cost functions traditional method of time series analysis uses are usually MSE, MAE, MAPE, and so on. Each cost function is

$$\text{MSE} = \frac{1}{n} \sum_{t=1}^n (\text{observed}_t - \text{predicted}_t)^2. \quad (18)$$

MSE can evaluate the variation of data. If the value of MSE is smaller, prediction model has better accuracy describing experimental data, where observed manifests measured value, predicted manifested predicted value. The average absolute value MAE is the average value of absolute error, which can better reflect the reality of predicted value's error. In  $MAE = (1/n) \sum_{i=1}^n |f_i - y_i|$ ,  $f_i$  is predicted value,  $y_i$  is actual value. The greater the MAPE value, the greater the difference between predicted value and original value, meaning the worse the prediction effect. The smaller the MAPE's absolute value, the higher accuracy it is.

$$MAPE = \sum_{t=1}^n \left| \frac{\text{observed}_t - \text{predicted}_t}{\text{observed}_t} \right| \times \frac{100}{n}. \quad (19)$$

The assessment standard the second test mode adopts is Precision, Recall, and F-Measure.

$$\begin{aligned} \text{Precision} &= \frac{TP}{TP + FP}, \\ \text{Recall} &= \frac{TP}{TP + FN}, \\ \text{F-Measure} &= \frac{2 \times TP}{2 \times TP + FP + FN}. \end{aligned} \quad (20)$$

In the equation, TP is the outlier of correct prediction, FP is the outlier of normal prediction (false positives), and FN is outlier that is not predicted (false negatives).

**4.4. Selection of Temporal Granularity.** Since this experiment is based on time series prediction, the selection of temporal granularity is one of the great factors that affect the predicted result [49]. Time series is to cluster the request of original data assembly according to certain time granularity and is gained according to time series. The difference of time granularity selection can bring different difficulties to time series prediction. Figure 8 shows the time series gained by different clustering of temporal granularity, where  $x$ -axis represents temporal granularity,  $y$ -axis represents the outlier. We can know from Figure 8 that the greater the selection of temporal granularity, the greater the outlier at each moment.

As for great temporal granularity, due to the huge base, the predicted absolute value might also be huge even though the prediction bias is small, which causes greater prediction error. Selecting overly small temporal granularity is not suitable for time series analysis either. Overly small temporal granularity makes data at each moment lack statistical significance. In addition, the clustered time series shows irregularity in multiple index. Table 2 provides coefficient of variations, skewness, and kurtosis of different temporal granularity clustering's every time series.

Coefficients of variation measure the mutability of time series. The greater the value, the easier time series changes with time. Skewness measures the asymmetry of time series. The greater the value is, the more asymmetrical the time series is. Kurtosis measures the variance of time series. The greater it is, the more irregular the distribution of time series' values. We can know from Table 2 that when the time granularity is in hours, the irregularity of time series is much

TABLE 2: Metric statistics of different temporal granularity clustering.

Metric	Weeks	Days	Hours
Coefficient of variation	0.1467	0.4254	0.7241
Skewness	-0.5654	-0.2428	2.8424
Kurtosis	2.7564	2.578	21.7721

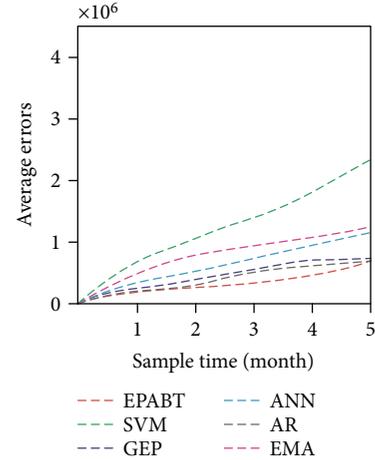


FIGURE 9: Average errors of each prediction method.

greater than the total of time series which have temporal granularity in weeks and days. The author also adjusted the parameters. The results show that the best time granularity of time series is recorded by days. Overall, according to observation and initial analysis, the case adopts time series that has temporal granularity in days.

## 5. Results and Analysis

The first type of monitoring pattern splits data set into training data set and predicted data set. The last month data are used as test set, and the rest are training set. This contrast experiment uses multiple prediction methods to make comparison, including random prediction, NARMAX, autoregressive, neural network, gene expression programme, SVR, and ensemble learning. We can see from Figure 9 that the method of ensemble learning is always optimal or close to the optimal conclusion. Based on the predicted result of data from 1 to 5 months, the winning prediction model in different data set are distinct. However, ensemble learning method is always close to optimal model. This is because the prediction strategy of ensemble learning always tends to make the entire ensemble learning model approach optimal model through first validating the contribution ratio of increasing experience and prediction model which performs the best.

Figures 10–12 are the evaluation result calculated by using mean absolute error, mean square error, and average absolute ratio error, which each corresponding to sliding mean, autoregression model, neural network, gene expression programming, support vector machine, and ensemble learning algorithm. From the figures above, it is obvious that

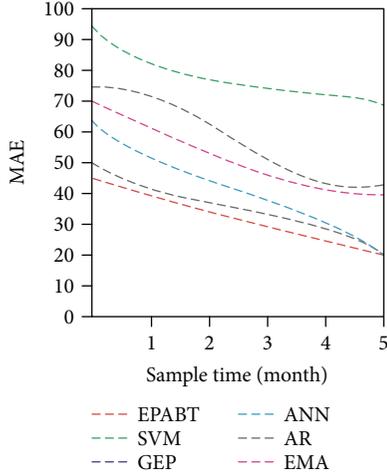


FIGURE 10: Evaluation result of MAE.

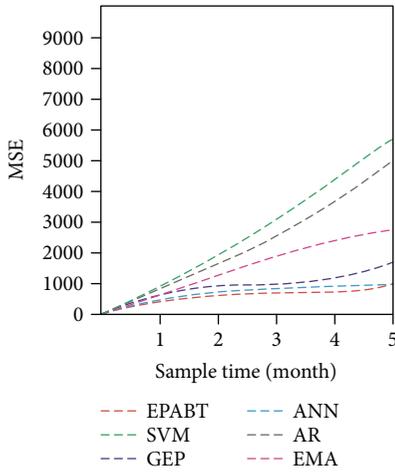


FIGURE 11: Evaluation result of MSE.

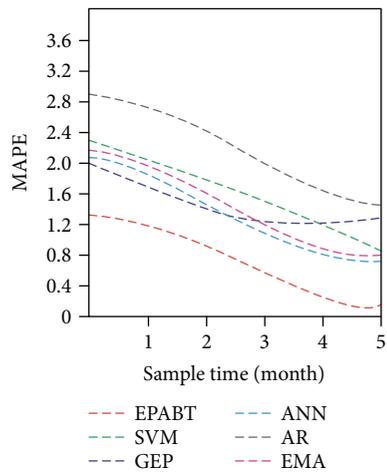


FIGURE 12: Evaluation result of MAPE.

ensemble learning algorithm is always the best algorithm. Overall, ensemble learning algorithm includes all advantages of other algorithm and processes them in the most accurate

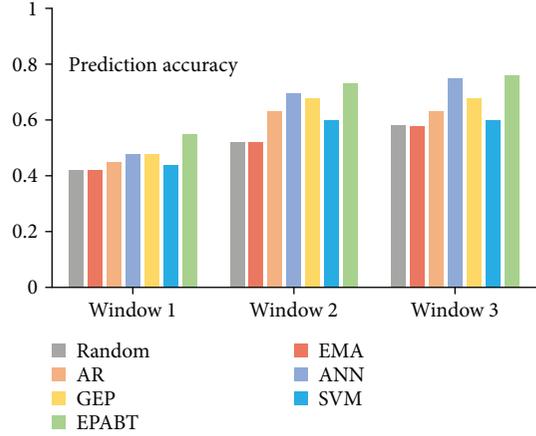


FIGURE 13: Average absolute ratio errors under different time windows.

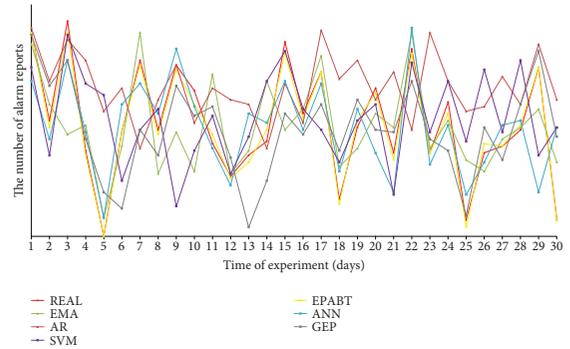


FIGURE 14: The comparison between each algorithm and the actual result.

way. The second monitoring and verification technique uses time window as basic units for error prediction. In the experiment process, time window unit is divided into 1 h, 6 h, 12 h, and 24 h. By using the different division of time window, 3 different experiment data sets can be collected as further bases. Thus, efficiency for multiple prediction models can be evaluated under different time windows, which is presented in Figure 13.

The experiment results illustrate that with the increase of time window, the accuracy of all prediction algorithms shows an ascending trend. In order to test the ensemble algorithm and evaluation system provided in this article, actual anomaly and predicted results in a certain period of time are compared, as presented in Figure 14. From Figure 14, it is not difficult to conclude that gene expression programming has the best outcome, followed by neural network. However, compared to ensemble algorithm, both algorithms cannot optimize the anomaly prediction. Ensemble algorithm has integrated all advantages of other algorithms.

## 6. Discussion and Conclusion

This article designs a diary analysis system based on big data platform for the characteristic of open-pit mine's monitoring system. In analysis process of monitoring system,

disadvantages of diary prediction and the need of predicting anomaly numbers are found, proposing an ensemble prediction algorithm based on time series. Through integrating multiple types of classified prediction algorithm and making classified prediction for diary data collected, this algorithm achieves the purpose of anomaly number of the comprehensive optimal accuracy prediction system. After instantly adjustment of parameters, such as hierarchical structure of neural network, kernel function of SVM, the results turn out that resemble algorithm has higher accuracy. The specific conclusions are as follows:

- (1) Single local diary analysis system not only has a complicated calculation, but takes long time, going through distributed real-time data transmission channel to transmit the data packet to cloud service platform in real-time queuing and adopting Spark Streaming platform to do data analysis, feature matching, and access statistics for network packet. The result shows that the experiment of dairy analysis operated in this platform has good improvement on the accuracy, time consuming and other indexes.
- (2) For the disadvantages of single prediction algorithm, integrating multiple types of prediction algorithm is proposed. The model experiments in training set and test set by using the monitoring data of the slope of open-pit mine in 6 months. The result shows that ensemble prediction algorithm gains better prediction accuracy and effect, whose calculation result is closest to the actual result, and it has a favourable generalization in dealing with homogeneous problems.
- (3) In order to examine the experiment result, two different evaluation systems are compared. All the results show that ensemble algorithm has greater accuracy, which proves more strictly the value of ensemble algorithm mentioned in this article.

In this paper, the ensemble algorithm model is an ideal state for weighted mean. However, in reality, many algorithm cannot be weighted. In our future work, more efforts will be made to study the integrated prediction problem under these contexts.

### Data Availability

The data in this article is a microseismic monitoring data during 6 months in an open-pit slope of molybdenum ore in Henan Province, China. The microseismic monitoring uses ARAMIS M/E microseismic monitoring system from Polish accompany EMAG. The original data is a waveform data, which is pretreated. The author confirms the availability and reliability of the data.

### Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

### Acknowledgments

This work was supported by the Natural Science Foundation of China (51774228; title: "Research on 5D Refined Mining Production Scheduling Model and Collaborative Optimization Method in Metal Open Pit under Constraints of Grade-Price-Cost"), Natural Science Foundation of Shaanxi Province (2017JM7005; title: "Intelligent fusion and early warning of multi-source heterogeneous flow data based on rock failure"), the Key technology projects of safety prevention and control of major accidents in State Administration of Work Safety (2017G-B1-0519; title: "Research on safety monitoring and warning system of ultra deep shaft surrounding rock based on multi-source heterogeneous information fusion"), and the excellent doctorate cultivation fund of Xi'an University Of Architecture And Technology (604031715; title: "Intelligent fusion of multi-source heterogeneous flow data and early warning of rock failure").

### References

- [1] V. Grechka, "Tilted TI models in surface microseismic monitoring," *Geophysics*, vol. 80, no. 6, pp. WC11–WC23, 2015.
- [2] J. S. Yoon, A. Zang, and O. Stephansson, "Numerical investigation on optimized stimulation of intact and naturally fractured deep geothermal reservoirs using hydro-mechanical coupled discrete particles joints model," *Geothermics*, vol. 52, pp. 165–184, 2014.
- [3] J. Rutqvist, P. Jeanne, P. F. Dobson et al., "The Northwest Geysers EGS Demonstration Project, California – part 2: modeling and interpretation," *Geothermics*, vol. 63, pp. 120–138, 2016.
- [4] G. B. White, E. A. Fisch, and U. W. Pooch, "Cooperating security managers: a peer-based intrusion detection system," *IEEE Network*, vol. 10, no. 1, pp. 20–23, 1996.
- [5] O. Depren, M. Topallar, E. Anarim, and M. K. Ciliz, "An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks," *Expert Systems with Applications*, vol. 29, no. 4, pp. 713–722, 2005.
- [6] H. S. Teng, K. Chen, and S. C. Lu, "Adaptive real-time anomaly detection using inductively generated sequential patterns," in *Proceedings. 1990 IEEE Computer Society Symposium on Research in Security and Privacy*, pp. 278–284, Oakland, CA, USA, 1990, IEEE.
- [7] M. Y. Huang, R. J. Jasper, and T. M. Wicks, "A large scale distributed intrusion detection framework based on attack strategy analysis," *Computer Networks*, vol. 31, no. 23–24, pp. 2465–2475, 1999.
- [8] S. Forrest, S. A. Hofmeyr, A. Somayaji, and T. A. Longstaff, "A sense of self for Unix processes," in *Proceedings 1996 IEEE Symposium on Security and Privacy*, pp. 120–128, Oakland, CA, USA, 1996.
- [9] T. A. Syed, R. Ismail, S. Musa, M. Nauman, and S. Khan, "A sense of others: behavioral attestation of UNIX processes on remote platforms," in *Proceedings of the 6th International Conference on Ubiquitous Information Management and Communication - ICUIMC '12*, p. 51, Kuala Lumpur, Malaysia, 2012, ACM.
- [10] E. Nohra, S. Buckman, K. Bochicchio et al., "Results of a near continuous glucose monitoring technology in surgical intensive care and trauma," *Contemporary Clinical Trials*, vol. 50, pp. 1–4, 2016.

- [11] S. Vinoth, L. Ajay Kumar, and E. Kumar, "Slope stability monitoring by quantification and behavior of microseismic events in an opencast coal mine," *Journal of the Geological Society of India*, vol. 85, no. 4, pp. 450–456, 2015.
- [12] A. J. Choobbasti, F. Farrokhzad, and A. Barari, "Prediction of slope stability using artificial neural network (case study: Noabad, Mazandaran, Iran)," *Arabian Journal of Geosciences*, vol. 2, no. 4, pp. 311–319, 2009.
- [13] Z. Liu, K. Cheng, H. Li, G. Cao, D. Wu, and Y. Shi, "Exploring the potential relationship between indoor air quality and the concentration of airborne culturable fungi: a combined experimental and neural network modeling study," *Environmental Science and Pollution Research*, vol. 25, no. 4, pp. 3510–3517, 2018.
- [14] P. Samui and D. P. Kothari, "Utilization of a least square support vector machine (LSSVM) for slope stability analysis," *Scientia Iranica*, vol. 18, no. 1, pp. 53–58, 2011.
- [15] A. Garg, A. Garg, K. Tai, and S. Sreedeeep, "Estimation of factor of safety of rooted slope using an evolutionary approach," *Ecological Engineering*, vol. 64, no. 64, pp. 314–324, 2014.
- [16] P. Lu and M. S. Rosenbaum, "Artificial neural networks and grey systems for the prediction of slope stability," *Natural Hazards*, vol. 30, no. 3, pp. 383–398, 2003.
- [17] L. I. Jialiang, M. A. Donghui, and W. Wei, "Assessment of potential seismic landslide hazard based on evidence theory and entropy weight grey incidence," *Journal of Central South University*, vol. 47, no. 5, pp. 1730–1736, 2016.
- [18] Y. Alimohammadlou, A. Najafi, and C. Gokceoglu, "Estimation of rainfall-induced landslides using ANN and fuzzy clustering methods: a case study in Saen Slope, Azerbaijan province, Iran," *Catena*, vol. 120, pp. 149–162, 2014.
- [19] L. V. Bui, K. Stahr, and G. Clemens, "A fuzzy logic slope-form system for predictive soil mapping of a landscape-scale area with strong relief conditions," *Catena*, vol. 155, pp. 135–146, 2017.
- [20] Z. Liu, D. Wu, H. Yu, W. Ma, and G. Jin, "Field measurement and numerical simulation of combined solar heating operation modes for domestic buildings based on the Qinghai-Tibetan plateau case," *Energy and Buildings*, vol. 167, pp. 312–321, 2018.
- [21] A. Yang, Y. Han, Y. Pan, H. Xing, and J. Li, "Optimum surface roughness prediction for titanium alloy by adopting response surface methodology," *Results in Physics*, vol. 7, pp. 1046–1050, 2017.
- [22] K. Sarkar, T. N. Singh, and A. K. Verma, "A numerical simulation of landslide-prone slope in Himalayan region—a case study," *Arabian Journal of Geosciences*, vol. 5, no. 1, pp. 73–81, 2012.
- [23] K. P. N. Jayasena, L. Li, and Q. Xie, "Multi-modal multimedia big data analyzing architecture and resource allocation on cloud platform," *Neurocomputing*, vol. 253, pp. 135–143, 2017.
- [24] F. Pop, C. Dobre, and A. Costan, "AutoCompBD: autonomic computing and big data platforms," *Soft Computing*, vol. 21, no. 16, pp. 4497–4499, 2017.
- [25] A. Goscinski, M. Hobbs, and J. Silcock, "GENESIS: an efficient, transparent and easy to use cluster operating system," *Parallel Computing*, vol. 28, no. 4, pp. 557–606, 2002.
- [26] S. U. Ahn, I. Y. Yeo, and S. O. Park, "Secure and efficient high-performance PROOF-based cluster system for high-energy physics," *The Journal of Supercomputing*, vol. 70, no. 1, pp. 166–176, 2014.
- [27] R. I. Taniguchi and T. Wada, "Real-time multi-view image processing system on PC-cluster -real-time motion capture and 3D volume reconstruction systems," *Journal of the Robotics Society of Japan*, vol. 19, no. 4, pp. 427–432, 2001.
- [28] A. S. Rawat, D. S. Papailiopoulos, A. G. Dimakis, and S. Vishwanath, "Locality and availability in distributed storage," *IEEE Transactions on Information Theory*, vol. 62, no. 8, pp. 4481–4493, 2016.
- [29] A. Duminuco, E. Biersack, and T. En-Najjary, "Proactive replication in distributed storage systems using machine availability estimation," in *Proceedings of the 2007 ACM CoNEXT conference on - CoNEXT '07*, p. 27, New York, NY, USA, 2016.
- [30] S. Kumar, E. Rosnes, and A. Graell i Amat, "Private information retrieval in distributed storage systems using an arbitrary linear code," in *2017 IEEE International Symposium on Information Theory (ISIT)*, pp. 1421–1425, Aachen, Germany, 2017, IEEE.
- [31] S. M. Vahidipour, A. Mirzaei, and M. Rahmati, "Comparing weighted combination of hierarchical clustering based on cophenetic measure," *Intelligent Data Analysis*, vol. 18, no. 4, pp. 547–559, 2014.
- [32] Y. Song, Y. Sun, and W. Shi, "A two-tiered on-demand resource allocation mechanism for VM-based data centers," *IEEE Transactions on Services Computing*, vol. 6, no. 1, pp. 116–129, 2013.
- [33] D. Ryu, J. I. Jang, and J. Baik, "A transfer cost-sensitive boosting approach for cross-project defect prediction," *Software Quality Journal*, vol. 25, no. 1, pp. 235–272, 2017.
- [34] H. Sobreira, C. M. Costa, I. Sousa et al., "Map-matching algorithms for robot self-localization: a comparison between perfect match, iterative closest point and normal distributions transform," *Journal of Intelligent & Robotic Systems*, vol. 90, no. 2, pp. 1–14, 2018.
- [35] Q. He, F. Zhuang, L. Zeng, W. Zhao, and Q. Tan, "PDMiner: a cloud computing based parallel and distributed data mining toolkit platform," *Scientia Sinica Informationis*, vol. 44, no. 7, p. 871, 2014.
- [36] K. Liu, H. Fu, and H. Chen, "Research on the influencing mechanism of traditional cultural values on citizens' behavior regarding the reuse of recycled water," *Sustainability*, vol. 10, no. 2, p. 165, 2018.
- [37] H. Fu and X. Liu, "Research on the phenomenon of Chinese residents' spiritual contagion for the reuse of recycled water based on SC-IAT," *Water*, vol. 9, no. 11, p. 846, 2017.
- [38] H. Fu and X. Liu, "A study on the impact of environmental education on individuals' behaviors concerning recycled water reuse," *Eurasia Journal of Mathematics, Science and Technology Education*, vol. 13, no. 10, pp. 6715–6724, 2017.
- [39] D. Coyle, G. Prasad, and T. McGinnity, "Extracting features for a brain-computer Interface by self-organising fuzzy neural network-based time series prediction," in *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 4371–4374, San Francisco, CA, USA, 2004, IEEE.
- [40] G. Deodatis and M. Shinozuka, "Auto-regressive model for nonstationary stochastic processes," *Journal of Engineering Mechanics*, vol. 114, no. 11, pp. 1995–2012, 1988.
- [41] M. Khashei, M. Bijari, and G. A. Raissi Ardali, "Improvement of auto-regressive integrated moving average models using fuzzy logic and artificial neural networks (ANNs)," *Neurocomputing*, vol. 72, no. 4–6, pp. 956–967, 2009.

- [42] B. Doucoure, K. Agbossou, and A. Cardenas, "Time series prediction using artificial wavelet neural network and multi-resolution analysis: application to wind speed data," *Renewable Energy*, vol. 92, pp. 202–211, 2016.
- [43] P. Bangalore and L. B. Tjernberg, "An artificial neural network approach for early fault detection of gearbox bearings," *IEEE Transactions on Smart Grid*, vol. 6, no. 2, pp. 980–987, 2015.
- [44] T. Van Gestel, J. A. K. Suykens, B. Baesens et al., "Benchmarking least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.
- [45] C. Zhou, W. Xiao, T. M. Tirpak, and P. C. Nelson, "Evolving accurate and compact classification rules with gene expression programming," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 6, pp. 519–531, 2003.
- [46] M. Salvoni and P. M. Dight, "Rock damage assessment in a large unstable slope from microseismic monitoring - MMG Century mine (Queensland, Australia) case study," *Engineering Geology*, vol. 210, pp. 45–56, 2016.
- [47] M. Fiorucci, R. Iannucci, L. Lenti et al., "Nanoseismic monitoring of gravity-induced slope instabilities for the risk management of an aqueduct infrastructure in Central Apennines (Italy)," *Natural Hazards*, vol. 86, no. S2, pp. 345–362, 2017.
- [48] F. Torfi, R. Z. Farahani, and S. Rezapour, "Fuzzy AHP to determine the relative weights of evaluation criteria and fuzzy TOPSIS to rank the alternatives," *Applied Soft Computing*, vol. 10, no. 2, pp. 520–528, 2010.
- [49] M. Franceschet, A. Montanari, A. Peron, and G. Sciavicco, "Definability and decidability of binary predicates for time granularity," *Journal of Applied Logic*, vol. 4, no. 2, pp. 168–191, 2006.

