Digital Object Identifier ACCESS. 2023-12478

Analyzing the Performance of Univariate and Multivariate Machine Learning Models in Soil Movement Prediction: A Comparative Study

Praveen Kumar[1], Priyanka P.[1], J. Dhanya[2], Kala Venkata Uday[2], and Varun Dutt[1], Senior Member, IEEE

[1]Applied Cognitive Science Lab, Indian Institute of Technology Mandi, Himachal Pradesh, 175075, India
[2]Geotechnical Engineering Lab, Indian Institute of Technology Mandi, Himachal Pradesh, 175075, India

Corresponding author: Praveen Kumar (e-mail: dr.praveenkumar.ml@gmail.com).

ABSTRACT Movement of soil and associated landslides frequently occur in hilly areas. Regular monitoring, accurate prediction, and timely alerting of people about soil movements on hills susceptible to landslides are essential due to the potential destruction to life and property. A more recent strategy for predicting soil movement is the use of machine learning (ML) models. Different univariate and multivariate ML models have been proposed in the literature. However, evaluating these univariate and multivariate approaches in predicting real-world landslides have received less attention. This paper's primary goal is to develop and compare the ability of univariate and multivariate ML models (Autoregression (AR), Seasonal Autoregressive Integrated Moving Average (SARIMA), Sequential Minimal Optimization regression (SMOreg), Multilayer Perceptron (MLP), and Long-short Term Memory (LSTM)) to predict movements at a real-world landslide. The case study used for the analysis in this paper is the Tangni landslide in India. This study makes use of weekly averaged soil movement data collected from June 2012 to December 2013 (78 weeks) at the Tangni landslide site. The dataset comprises measurements from five sensors. To calibrate the parameters in each model, we divided the collected data into a training dataset (first 62 weeks) and a test dataset (last 16 weeks). Performance analysis of the models utilized Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and R-squared values. The training results revealed that the univariate AR model demonstrated the best performance, achieving an RMSE of 0.149 degrees and an R-squared value of 0.572. The univariate SMOreg model obtained the second-best performance with an RMSE of 0.336 degrees and an R-squared value of 0.582. However, on the test dataset, the multivariate SARIMAX model outperformed the other models, achieving an RMSE of 0.351 degrees and an R-squared value of 0.769. The univariate SARIMA model also performed well with an RMSE of 0.356 degrees and an R-squared value of 0.741. The findings of this study can have significant implications in the field of landslide prediction and prevention. The results indicate that the multivariate SARIMAX model, the most accurate in predicting soil movements, can aid in developing early warning systems against landslides in hilly areas.

INDEX TERMS Autoregression, LSTM, MLP, SARIMA, SMOreg, Tangni landslide.

I. INTRODUCTION

Landslides are frequent in hilly areas [1]. These landslides cause threats to people and infrastructures [2]. As a result, it is advisable to track and predict soil movements and generate alerts on hills vulnerable to landslides [3]. The movements of soil on landslide-prone hills could be predicted using previous values of the soil movements and other critical parameters related to the landslide [4]. Here, the monitoring stations could be installed on landslides to record the soil movements and other vital parameters (e.g., rainfall, weather parameters, moisture in the soil, etc.) [3]. Next, machine learning (ML) models could utilize these historical soil movements and critical parameters to predict the soil [3]. These ML models could take soil movements and other vital parameters as input and predict the soil movements in the future [5]. If future soil movements exceed the predefined thresholds, then timely alerts could be generated for the landslide threat [3].

The ML literature on soil movement prediction emphasizes the use of various models, including statistical approach, support vector machines (SVM), and neural networks (NNs) for predicting soil movements [6]-[19]. Among them, autoregression (AR), autoregressive integrated moving average (ARIMA), and seasonal autoregressive integrated moving average (SARIMA) are the statistical models. These statistical models find the relationship between a time series' current value and past values and could predict the following values using the same relationship [6]-[10]. The statistical models (AR, ARIMA, and SARIMA) require fewer training data to train the model [20]. In contrast, a support vector machine (SVM) uses the support vectors near the decision boundary to make predictions and optimize the decision boundary with a kernel function [11]-[13]. Additionally, researchers have also proposed the multilayer perceptron (MLP) model for soil movement prediction, and the use of recurrent neural networks (RNNs), including the long short-term memory (LSTM) model, has been suggested for the same purpose [10], [14]-[19]. These NNs feed the input in the network, make the predictions, and reduce the error in predictions with the backpropagation techniques.

Depending upon the attributes, there may be two flavors of different machine learning models: univariate and multivariate [21], [22]. Univariate models use the past values of soil movements to predict corresponding future values [21]. However, in multivariate ML models, a time series may likely have two or more time-dependent variables (e.g., soil movements in different co-located drills) [22]. Thus, each variable may depend on its past values and other related variables in multivariate ML models.

Several researchers have explored and developed both univariate and multivariate statistical models to predict soil movements in landslide-prone areas. The ARIMA and dynamic neural network models showed higher accuracy than the generalized autoregressive conditional heteroskedasticity model in predicting landslides. The multivariate ARIMA model better-predicted soil movement rates in landslides. An ensemble model with AR, SVM, and LSTM also accurately predicted landslide movement.

Furthermore, several univariate and multivariate SVM models have been developed by researchers for predicting soil movements in landslides. Wang et al. developed five univariate ML models for predicting soil movement in the TGRA of China, with PSO-LSSVM and PSO-KELM models showing improved accuracy and consistency [11]. Xing et al. created a model that produced high-quality prediction intervals for soil movement's lower and upper bounds using a double DES method [12]. Zhang et al. developed two multivariate SVM models, PSO-SVM and SPA-PSO-SVM, for predicting landslide movement, with the SPA-PSO-SVM model showing potential for use in soil movement prediction [13]. SVM models are a promising option for landslide prediction because they handle complex nonlinear relationships between variables and effectively analyze large amounts of data.

Moreover, researchers have developed univariate and multivariate neural network models to predict landslide soil movements. Modified neural network methods have been used for soil movement prediction, such as the revised Back Propagation Neural Network. At the same time, hybrid models that use ensemble empirical mode decompositions and LSTM models have also been developed. Stacked LSTM models and step-like soil movement characteristics have been considered while predicting soil movements. Multiple geological conditions, rainfall intensity, and human activities have also been considered in multivariate LSTM models. In comparative studies, LSTM models have performed better than SVM and Elman neural network models.

The present work aims to compare univariate and multivariate versions of different statistical, support vector, and neural network models to predict soil movements. The main novelty of this work is the comprehensive assessment of the univariate and multivariate ML models for the soil movement prediction. We draw conclusions based on two years of data from an active landslide site in Chamoli, India (the Tangni landslide). The selected region is crucial as several landslides recently occurred in the Chamoli district [27]. Notably, in 2013, there were 220 landslides, resulting in numerous deaths and extensive infrastructural damages.

To summarize, it was observed that various univariate and multivariate statistical, support vector, and neural network models had been proposed to predict soil movements. However, one is yet to extensively develop and compare the univariate and multivariate versions of statistical, support vector, and neural network models at a single landslide site. This main contribution of this work is to fill this literature gap by developing and comparing univariate and multivariate versions of statistical (AR and SARIMA), support vector machine (SVM), and NN models (MLP and LSTM) at the Tangni landslide in India. The development and comparison of univariate and multivariate models is the first kind of work to predict the soil movement at a single site.

The contributions of this work are highlighted as follows:

- Filling a literature gap by extensively developing and comparing univariate and multivariate versions of statistical (AR and SARIMA), support vector machine (SVM), and NN models (MLP and LSTM) for soil movement prediction.
- Investigating and comparing the performance during training and test of different univariate and multivariate versions of models for predicting soil movements.
- Providing insights into the effectiveness of univariate and multivariate approaches in soil movement and landslide prediction, which can be valuable for future research and practical applications in geotechnical engineering.
- Offering comprehensive analyses of various models, including traditional statistical methods and advanced machine learning techniques, for

predicting soil movements at a real-world landslide site in India.

II. BACKGROUND

The literature study on developing univariate and multivariate statistical models for predicting soil movements in landslides has used different machine learning models such as ARIMA, SVM, and NNs such as LSTM, ELM, and RNN, among others. The results of these studies indicate that multivariate models are generally more accurate than univariate models, and their accuracy may be due to the inclusion of more information in the model.

Several researchers have attempted to use univariate and multivariate statistical models for predicting soil movements [6]-[10]. In that direction, Duan and Niu (2013) predicted the cumulative soil movement of the Bazimen landslide, China, using the ARIMA model [6]. Furthermore, a univariate version of an ARIMA model, a dynamic NN model, and a generalized autoregressive conditional heteroskedasticity model has been proposed [7]. These models were trained using time series data from an active landslide. The result demonstrated that the dynamic neural network model and ARIMA models were applicable and showed higher accuracy than the generalized autoregressive conditional heteroskedasticity model in predicting landslides. Geyuan (2020) developed the multivariate ARIMA model to predict the soil movement rate of the Zhujiadian landslide in China [8]. This experiment's results showed that the multivariate ARIMA better predicted the landslide's soil movement rate. Similarly, Li et al. (2021a) developed a multivariate model with the ensemble of AR, SVM, and LSTM to predict the movement of the Xinming landslide in China [9]. The prediction results showed that the accuracy of the ensembled model was satisfactory. Similarly, Pathania et al. (2022) developed an AR model to make predictions regarding soil movement in the Gharpa landslide area of India [10]. Results revealed that the AR model could be developed for the soil movement prediction in the real-world landslide.

Several researchers have developed the univariate and multivariate SVM models for predicting soil movements [11]-[13]. For example, Wang et al. (2019) developed five univariate ML models for predicting the soil movement of several landslides in the Three Gorges reservoir area (TGRA) of China [11]. The developed univariate models were the PSO–least squares support vector machine (PSO–LSSVM), PSO–SVM, PSO optimized kernel extreme learning machine (ELM) named PSO–KELM, PSO–ELM, and the LSTM model. The results showed that PSO–LSSVM and PSO–KELM models improved their prediction accuracy and consistency. Next, Xing et al. (2021) developed a univariate SVM model to predict the lower and upper bound of the soil movement [12]. At the same time, a double DES method was used to predict nonlinear soil movements. The outcomes showed that the created model may be applied to produce high-quality prediction intervals. Next, Zhang et al. (2021) developed two multivariate SVM models named PSO-SVM and set pair analysis-based PSO-SVM (SPA-PSO-SVM) for the landslide movement prediction [13]. The findings of the research indicated that the SPA-PSO-SVM model might be used to predict soil movement.

Several researchers have developed the univariate and multivariate NN models for the purpose of accurately predicting soil movements [14]-[19]. For instance, Chen and Zeng (2013) developed a univariate modified NN method (revised Back Propagation Neural Network) for soil movement prediction of certain landslides in the TGRA, China [14]. Results revealed that the learning speed of the network and the prediction precision improved upon using the proposed modified model. Furthermore, Xie et al. (2019) proposed a multivariate version of the LSTM for predicting the soil movements in the Laowuji Landslide in China [15]. The model considered multiple geological conditions, rainfall intensity, and human activities. Recently, Xing et al. (2019) compared an SVM, an Elman neural network (ENN), and an LSTM model for soil movement prediction of the landslide [16]. Results showed that the LSTM outperformed the SVM and the Elman network. Next, Li et al. (2021b) developed a univariate LSTM model by stacking multiple LSTM layers (SLSTMs) to predict soil movements [17]. For training the SLSTM model, the time series data of the Xintan landslide in China were used. The accuracy of the developed model was then compared with a univariate SVM model. It was observed that the univariate SLSTM could better predict the soil movements. Next, Niu et al. (2021) proposed a hybrid univariate model, combining ensemble empirical mode decompositions with LSTM (EEMD-LSTM), to predict soil movement in the Muyubao landslide in China [18]. Their findings indicated that the univariate EEMD-LSTM model outperformed other models in terms of predictive accuracy. Similarly, Taorui et al. (2022) developed an LSTM model to forecast the step-like soil movement characteristics observed in the Tangjiao landslide in the Wanzhou district of China [19]. The results demonstrated that the developed model better generalized the step-like elements in the soil movement predictions.

Researchers have also explored the potential of univariate and multivariate models as suitable prediction models in various fields beyond landslides [23]-[26]. For example, Preez and Witt (2003) compared the univariate and multivariate statistical models for forecasting international tourism demand [23]. Similarly, Cadenas et al. (2016) compared a multivariate autoregressive exogenous artificial neural network (NARX) model and a univariate ARIMA model to predict the air speeds [24]. Furthermore, Zhang et al. (2017) compared a univariate single-spectrum analysis (SSA) model and a multivariate vector autoregressive (VAR) statistical model for predicting electric vehicle sales in China [25]. Similarly, Miller and Kim (2021) compared univariate and multivariate models for predicting the price return of the cryptocurrency [26]. In these experiments, researchers [23]-[26] developed ARIMA, LSTM, Holt's exponential smoothing (HES), deep neural networks (DNNs), RNNs, and the ForecastX models and compared the model performance. The results indicate that multivariate models may produce

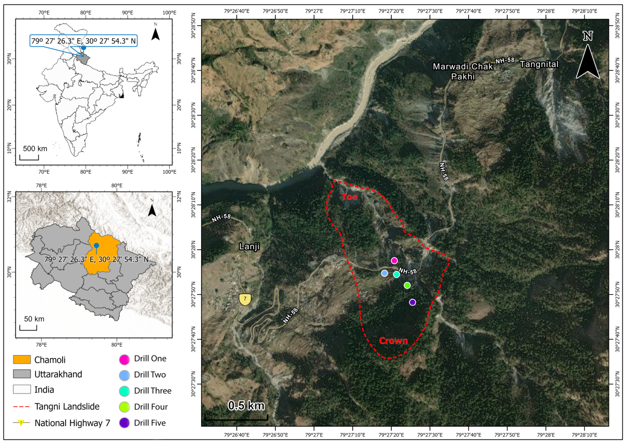

FIGURE 1. **The Tangni landslide is on the Indian map, and © Google Maps 2022 depicts the drill locations at the Tangni landslide.**

more accurate predictions than univariate models [23]-[26]. One of the main reasons for this high accuracy may be the inclusion of more information in the multivariate model. However, Preez and Witt (2003) also pointed out that including multiple parameters in the model might lead to additional sources of error, which may adversely affect the multivariate models' performance [23]. Another likely reason for the adverse performance of multivariate models could be the presence of outliers in data, which may be difficult to spot and control in multivariate models compared to their univariate counterparts [23].

Although univariate and multivariate models have been compared in different domains in the ML literature, the comparison of these models for soil movements prediction are still limited. Overall, one does not know how the univariate and multivariate ML models would perform for soil movement predictions on the same real-world dataset. Thus, it may be worthwhile to compare how univariate and multivariate versions of ML models perform for real-world soil movement predictions.

III. THE TANGNI LANDSLIDE

The study area is in the Chamoli district in the Uttarakhand state of India, having a latitude of 30º 27' 54.3" N and longitude of 79º 27' 26.3" E (Fig. 1). As shown in Fig. 1, the dotted red color polygon represents the study area. National Highway 7 (NH-7) passes through the Tangni landslide, which was previously the NH-58. The NH-7 passes through the landslide at an altitude of 1560 m and connects Fazilka in Punjab with Mana Pass in Uttarakhand. The toe and crown of the landslide are at an altitude of 1230 m and 1890 m, respectively (see Fig. 1). Furthermore, the slope and length of the toe are 42º and 800 m beneath the road level, respectively [28], [29]. Similarly, the slope and length of the crown are 35º and 650 m above the road level, respectively. According to field studies, this landslide consists of the fine to coarse chunks of soils and debris, rock pieces, and various rock blocks [30]. The geography of this site includes slates and dolomite rocks [29]. The site got affected by frequent landslides in 2013, resulting in more economic losses [31]. Thus, to further monitor the Tangni landslide, inclinometers were placed in 5 drills between 2012 and 2013 at Tangni. As depicted in Fig. 1, the sensors were installed in five different drills, with five colors representing each drill location. As shown in Fig. 1, drill one is below the road level, and drills two to five are above the road level. Data collected from the landslide site at Tangni (see Fig. 1) was used for evaluating the univariate and multivariate models considered in this work. The present study uses the time series data from five drills between Jun 2012 to Dec 2013 for the analysis (Fig. 1).

IV. METHODOLOGY

A. DATA PRE-PROCESSING

Twenty-five inclinometer sensors monitored the five drills. Every drill contains contained five sensors from 3 m to 15 m vertical depths to monitor the movement at different depths. Each sensor is located 3 m vertical apart from other. These sensors measured the slope angle in millimeters per meter (mm/m) over time. The working principle of the inclinometer is described in Fig. 2 (a). As shown in Fig. 2 (a), the sensor's length is L, and it has A and B axes. Each sensor axis has positive and negative sides to measure the movement. The positive side measures the uphill displacement, and the negative side measures the downhill displacement of the slope. If the displacement of the one axis is $L\ sin\theta$ in one direction, then θ is the slope angle or movement in that direction. The sensors were set up so that the downward and upward movements towards the highway level were recorded on the negative and positive sides of the A-axis, respectively. Next, the sensors closest to the landslide's sliding surface were projected to produce the maximum movement. Thus, the sensors nearer to the sliding surface will be necessary as these sensors will likely have the maximum soil movements (see Fig. 2 (b)). Overall, based on the data collected from sensors in each drill, we identified one necessary sensor per drill that showed the most significant movement over 2-years. The sensor at 3 m was selected from drill 1, and the sensor at 12 m was chosen from drill 2. Furthermore, the sensor at 6 m was chosen from drill 3, and the sensors at 15 m were chosen from drills 4 and 5. For example, in Fig. 2 (b), the sensor in drill two at 12 m revealed the highest soil movements over two years. As a result, the soil movement time series narrowed down to one critical sensor from each drill.

Furthermore, data pre-processing is a critical step in preparing data for model training, and removing outliers is an essential technique used in data pre-processing [32]. Outliers are data points significantly different from other data points in the dataset and they can skew the statistical analysis and model training results. Therefore, it is essential to identify and remove outliers from the dataset before training the model [32]. In soil movement data, maximum values of soil movement may not be true outliers but instead represent expected behavior over the long term.

In such cases, removing these values as outliers may not be appropriate. An alternative approach to data pre-processing is to average the data over a period of time, such as a week, to reduce noise and highlight trends in the data. We added the soil movements recorded over a week (7-days) per sensor to generate a time series with 78 movement points (one movement point per week) per sensor. Next, we divided these summed soil movements per week per sensor by the number of recorded movement values from the sensor in the week to give us the weekly averaged soil movements. These weekly averaged soil movement time series, one per sensor in each of the five drills at the landslide site, were used further in this study. To compute the average value of the time series over a weekly interval, the following formula was used:

$$Average\ value\ of\ week_i = \frac{1}{n}\sum_{t=(i-1)n+1}^{ni} x_t \qquad (1)$$

where x_t is the value of soil movement recorded by a sensor at time t, n is the total number of movement data points in each weekly interval recorded by the sensor, and i is the index of the weekly interval, respectively.

Fig. 3 shows the five soil movement time series assembled from different drills at Tangni. Fig. 3 shows the upward movement with the positive values and the downward movement with the negative values. For example, for drill one, the movement in the 25th week was -2.7º, and it changed to 1.7º in week 30.

Furthermore, all five drills demonstrated consistent downward soil movements. Drill one showed continuous downward soil movements between week 1 to week 31, whereas the significant movements occurred in week 32. During the first and last weeks, minimal soil movements were recorded in drills two and three, located between the toe and crown. However, drill four and five (installed near the crown) showed movements throughout the time series. For example, drill five had significant movements in the 23rd and 60-78 weeks.

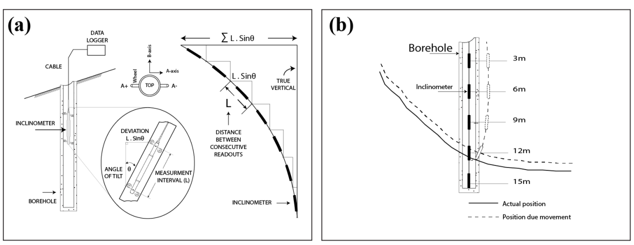

FIGURE 2. (a) Illustration of the working principle of inclinometer in detecting soil movement. (b) A schematic diagram of the critical senor subjected to maximum soil movement during the slope failure. In the representation, the sensor placed at 12 m depth, which lies close to the failure plane, can be identified as the critical one.

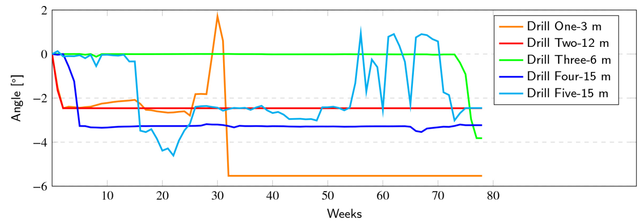

FIGURE 3. The plots of the weekly soil movement across all the drills.

B. DESCRIPTION OF DATASET

The Tangni dataset had three distinct attributes. The first attribute is the measured soil movement. The second attribute is the drill's index from where the soil movement was measured. The third attribute is the depth of the drill where the sensor was installed. Further, we obtained the statistics of the data corresponding to the identified critical time series from different drills. The second column of Table I summarizes the mean value of the data from five drills. Next, the third column summarized the data's standard deviation (SD) per drill. It can be noted that the drill one has the maximum variation in data, having a standard deviation of 1.83º. Furthermore, all the drills are subjected to a mean downward moving trend. Additionally, Table I also details the correlation between the different time series. The corresponding estimates reveal that if two drills are on the same landslide's sliding surface, both will be susceptible to soil movements simultaneously, as evidenced by the high correlation estimates. From Table I, there was a high correlation between drills two and four (0.55), pointing to the possibility that these drills lie on the same sliding surface owing to their alignment and proximity (Fig. 1).

TABLE I

MEAN, SD, AND THE CORRELATION OF THE CRITICAL SOIL MOVEMENT DATA AT FIVE DRILLS IN THE STUDY REGION.

Drill	Mean (°)	SD (°)	Correlation between time series from the drills				
			Drill				
			One	Two	Three	Four	Five
One	-4.1	1.8	1				
Two	-2.5	0.1	0.162	1			
Three	-0.2	0.7	0.174	0.026	1		
Four	-3.1	0.6	0.266	0.552	0.034	1	
Five	-1.7	1.5	0.000	0.142	0.101	0.233	1

C. AUTOCORRELATION ANALYSIS OF TIME SERIES

The autocorrelation function (ACF) shows the relation between a time series' current timestamp value and its prior values [33]. The span up to which the time series data are correlated is typically termed look-back periods or lags. At the same time, partial autocorrelation (PACF) finds the correlation between the current timestamp and residuals of the time series, which were unexplained by the past lags [33]. The ACF value specifies the number of previous residuals needed to predict the next timestamp value. The PACF denotes the number of prior lags needed to predict the present timestamp of the time series. The ACF and PACF were calculated with the ninety five percent confidence interval (CI). The drill one's time series has four significant ACF values and one significant PACF value. While the ACF and PACF values in drill two and three's time series were both zero. Both ACF and PACF values were two in drills four's time series. Furthermore, drill five exhibited three significant ACF values and one significant PACF value in its time series. Based on the range of PACF and ACF values all over the five-time series, we varied the lag values or look-back time from one to five for the developed models.

V. THE UNIVARIATE AND MULTIVARIATE ML MODELS

There may be two flavors of different ML models: univariate and multivariate. The univariate model uses an attribute's historical values to predict the corresponding future values. While, in a multivariate model, the dependent variable depends not only on its historical values but also on other potential variables. In this research, we developed both the univariate and multivariate versions of the ML models for the dataset selected for the study. The literature study found that the univariate and multivariate versions of MLP, LSTM, AR, SVR, and SARIMA models have been used for the soil movement predictions. Hence, we are developing these models in this research. The following section presents a brief discussion on the methodology of the ML models considered in this work.

Figure 4 provides a block diagram of the proposed model used in the experiment. The block diagram shows that the available data are split into the training and test datasets, with an 80:20 ratio. The training dataset is further divided into multiple subsets, which are used to optimize the parameters of the ML model using the walk-forward validation method [34]. This approach helps to ensure that the model's performance is accurately evaluated and fine-tuned based on historical data. The test dataset is used to assess the final performance of the model after the training is complete (best model parameters are determined). By using this approach, the model can be validated and fine-tuned effectively, leading to more accurate and reliable predictions.

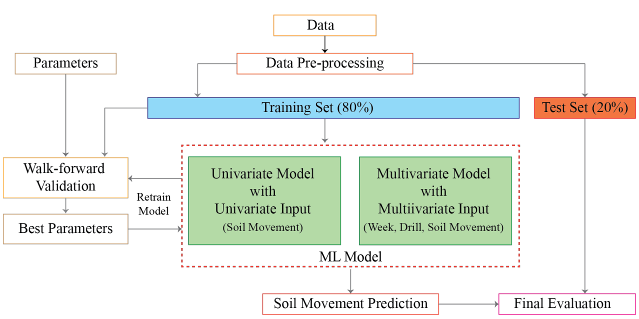

FIGURE 4. The block diagram of the proposed model.

A. UNIVARIATE MODELS

1) MLP

The multilayer perceptron is a popular artificial NN model that is widely used for classification and regression tasks in ML [35]. The architecture of an MLP consists of three layers: the input layer, hidden layer, and output layer. Within each layer, neurons are interconnected, and appropriate activation functions are applied to identify the non-linear patterns present in the dataset. It utilizes the backpropagation technique in training to minimize the error in the prediction [36]. In the architecture of MLP, the number of hidden layer neurons (HLN) and the number of hidden layers could be varied. Once the architecture is finalized, unknowns in the model in terms of weights are optimized based on error minimization between actual and predicted values. We varied the number of hidden layers, a look-back period, and HLN in the MLP model to identify the best model.

The input layer size (ILS) of the MLP model was determined based on the number of attributes and the look-back value considered. Based on the valued of ACF and PACF stated in Section III-C, the look-back periods were modified from 1 to 5. Regarding attributes considered, the univariate version focused on the soil movements, while in the multivariate version, we considered soil movements, depth of the sensor, and drill's index in the dataset. The HLN were changed from 10 to 500. Additionally, we applied different activation functions vis Linear, Rectified Linear Unit (ReLU), and sigmoid to compare the model performances. In our research, we have utilized the one-step-ahead prediction technique to predict soil movements in the upcoming week. Table II summarizes the parameters used for univariate versions of the MLP model. The NN models (MLP, LSTM) randomly initialize the weights in the network at the initial time [37]. The PSO technique was used to find the best initial weights rather than randomly initialize the weights for the MLP model [38], [39].

2) LSTM

RNNs are the model for sequential data which can remember its previous input due to internal memory [40]. The LSTM is another variant of the RNN model, which was developed to learn long-term dependency in the data [41]. The LSTM models can be used for regression tasks, where the goal is to predict a continuous numerical value. LSTMs are particularly useful for this type of task when the data is sequential, such as time series data, and long-term dependencies are present. For example, in a time series dataset, the value of a given time point may depend on the values of previous time points. LSTMs can effectively model these long-term dependencies by introducing memory cells and gates. This feature enables the model to selectively retain or discard information from past time points based on its relevance. To use an LSTM for regression, the output of the LSTM is typically fed through a fully connected layer and then passed through a linear activation function to produce a continuous numerical prediction. LSTMs are effective at regression tasks, mainly when dealing with long-term dependencies in the data.

The input layer characteristics of the LSTM model were taken the same as the MLP models. Further, the number of HLN were changed from 10 to 500. The LSTM's output layer had one OLN with a linear activation function to forecast the one-step-ahead soil movement. The different range of the parameters considered for the univariate LSTM model is shown in Table II. The PSO technique was used to find the best initial weights rather than randomly initialize the weights for the LSTM model.

3) SEQUENTIAL MINIMAL OPTIMIZATION REGRESSION (SMOREG)

The SVM is a well-known model for classification and regression [42]. However, the SVM model suffers from the quadratic programming (QP) problem while optimizing the unknowns in the training phase [43]. A QP problem has a quadratic cost function and linear constraints [44]. When the training data size (N) is large, the worst-case running time complexity of QP is $O(N^3)$. Later, the SMOreg model got developed by incorporating modifications to the SVM model to solve the QP problem [43]. Here, SMOreg needs memory linear to the training dataset of size *N*. SMOreg is an iterative model that works by dividing the training data into two subsets at each iteration, and then solving for the optimal solution for each subset. This process is repeated until convergence, at which point the final set of weights and biases for the SVM model are obtained. One advantage of SMO is that it is relatively efficient and can handle large datasets.

The SMOreg model has two parameters complexity (*C*) and exponent (*e*) of kernel function. The parameter *C* controls the model to choose a soft or hard margin to handle errors. A smaller value of *C* means a softer margin, and a significant value means a hard margin. The kernel function which measures the similarity between two vectors. The linear kernel (LK), polynomial kernel (PK), and radial basis function (RBF) have been used in this experiment [45]. The range of parameters for univariate and multivariate SMOreg is discussed in Table II.

4) AR

The autoregressive model predicts the next value by utilizing the same time series data. Thus, the input in the model constitutes lag variables from previous time steps [46]. The model can be described as a regression problem such that:

$$\hat{y}_t = \beta_0 + \beta_1 x_{t-1} + \cdots + \beta_n x_{t-n} \tag{2}$$

where $x_{t-1}, \ldots, x_{t-n}$ are the previous observations of a 1-dimensional time series, and n represents the lag values or look-back period. The $\beta_0, \beta_1, \ldots, \beta_n$ are the unknown coefficients. Note that the prediction $\hat{y}_t$ at the time step t is made based on the same input variable data at previous time steps. The AR model predicts the soil movements from the corresponding n lag values. Here, the lags were estimated from the ACF and PACF as discussed in Section III-C. The related details are summarized in Table II.

5) SARIMA

The soil movements dataset may contain seasonality in its time series. The SARIMA model finds seasonality and other features in the data. The first step in the SARIMA model is to make the data stationary. The mean, variance, and correlation in the stationary time series do not vary with time. Many statistical forecasting techniques assume that differencing could approximate a stationary state [47]. There is a moving average (MA) factor, an autoregressive (AR) factor, and an order of differencing terms (I) in the seasonal sections of an ARIMA model. These factors work over different lag periods in a season in time series data. The AR and MA factors can be calculated using the (3)-(5).

$$AR(\hat{y}_t) = c + \phi_1 x_{t-1} + \cdots + \phi_p x_{t-p} + \boldsymbol{\epsilon}_t \tag{3}$$

$$MA(\hat{y}_t) = c + \theta_1 \boldsymbol{\varepsilon}_t + \cdots + \theta_q \boldsymbol{\varepsilon}_{t-q} + \boldsymbol{\epsilon}_t \tag{4}$$

$$\hat{y}_t = (x_t - x_{t-1}) - (x_{t-1} - x_{t-2}) - \cdots - (x_{t-d-1} - x_{t-d}) \tag{5}$$

where p and q are the AR and MA trend parameters, respectively. The $x_{t-1}, \ldots, x_{t-\mathrm{p}}$ are the previous p data points. The $\boldsymbol{\epsilon}_t, \ldots, \boldsymbol{\epsilon}_{t-q}$ indicates the error in the previous q data points. The $\boldsymbol{\epsilon}_t$ is the white noise in the data, and c is the constant term, which is the average of the changes between consecutive data points. If c is positive, then x_t will tend to drift upwards. However, if c is negative, x_t will tend to drift downwards. Equation (3) represents the differencing in time series, where the order of differencing parameter is represented by d, which means the d numbers of lag values have been subtracted from the time series. The model includes four variables for the seasonal component: the autoregressive order (AR order) denoted as P, the differencing order for the seasonal component (seasonal I order) represented as D, the moving average order for the seasonal component (seasonal MA order) indicated by Q. Furthermore, the variable m indicated the number of time intervals or time units for a single seasonal cycle. The calculations of the seasonal component of the SARIMA model are like the non-seasonal function in the (3)-(5), but instead of using two consecutive time steps (t and $t-1$), the model usages the seasonal period (m and $m-1$). The seasonal function of the model could be obtained after replacing the term t with m in the (3)-(5).

B. MULTIVARIATE MODELS

1) MULTIVARIATE MLP

The architecture of the multivariate MLP is the same as the univariate version. However, in the multivariate MLP, additional input neurons are added to the input layer to handle the multiple variables included in the dataset. To determine the optimal model, the number of hidden layers, hidden layer neurons, and look-back period are adjusted in a similar manner as the univariate version. Therefore, the main difference between the univariate and multivariate MLP models is the number of input neurons in the input layer. Table II summarizes the parameters used for multivariate versions of the MLP model.

2) MULTIVARIATE LSTM

The architecture of multivariate LSTM is similar to the univariate LSTM, with the only difference being the input layer having multiple neurons to receive the multivariate data. Table II summarizes the parameters used for multivariate versions of the LSTM model.

3) MULTIVARIATE SMOREG

Multivariate SMOreg is a ML model used for classification and regression, which is an extension of the univariate SMOreg model. Unlike the univariate SMOreg model, multivariate SMOreg can handle datasets with multiple input variables, making it suitable for high-dimensional datasets. Table II summarizes the parameters used for multivariate versions of the SMOreg model.

4) MULTIVARIATE AUTOREGRESSION OR VECTOR AUTOREGRESSION (VAR)

The VAR model predicts the next value by utilizing the multiple time series [48]. The vector of all variables' current value in the d-dimensional time series is modeled as a linear sum of previous observations such that

$$\hat{y}_t = \beta_{10} + \beta_{11}x_{1,t-1} + \cdots + \beta_{1n}x_{1,t-n} + \epsilon_1 + \cdots + \beta_{d0} + \beta_{d1}x_{d,t-1} + \cdots + \beta_{dn}x_{d,t-n} + \epsilon_d \quad (6)$$

where $x_{1,t-1}, \ldots, x_{d,t-1}, \ldots, x_{d,t-n}$ are the previous observations of a d-dimensional time series, and n represents the lagged values or look-back period. The $\beta_{10}, \ldots, \beta_{d0}, \ldots, \beta_{dn}$ are the $d \times d$ dimensional matrix of unknown coefficients. The $\epsilon_1, \ldots, \epsilon_2, \ldots, \epsilon_d$ are the white noise in the d-dimensional time series data. The $\hat{y}_t$ is a predicted value of the model. The range of parameters for multivariate VAR is discussed in Table II. The multivariate data from the Tangni had three attributes, out of which drill and sensor depth were constant and showed zero variance. The statistical model could not fit using zero variance attributes [49]. Hence, random noise with 0 mean and 0.001 variances was added to the drill number and sensor depth attributes while modeling.

5) MULTIVARIATE SARIMA

The explanatory variable is inserted in the SARIMA to derive the multivariate SARIMA or multivariate SARIMAX model [50]. Here, while predicting a specific attribute, the time series data from other attributes are included as explanatory variables. The predicted output of the multivariate SARIMAX model could be defined as follows:

$$y_t = c + \phi_1 x_{t-1} \ldots + \phi_p x_{t-p} + \theta_1 \boldsymbol{\varepsilon}_t \ldots + \theta_q \boldsymbol{\varepsilon}_{t-q} + \beta_{1t}x'_{1t} + \beta_{2t}x'_{2t} \ldots, \beta_{jt}x'_{jt} + \boldsymbol{\epsilon}_t \quad (7)$$

where p and q are the AR and MA trend parameters, respectively. The $x_{t-1}, x_{t-2} \ldots, x_{t-p}$ are the data of the previous p data points. The $\varepsilon_t, \varepsilon_{t-1} \ldots, \varepsilon_{t-q}$ indicates the error terms at previous q data points. The $x'_{1t}, x'_{2t} \ldots, x'_{jt}$ are represents the explanatory variables, and $\beta_{1t}, \beta_{2t} \ldots, \beta_{jt}$ are the coefficients of the explanatory variable. The ϵ_t is the white noise in the data. The different range of the parameters considered for the univariate SARIMA and multivariate SARIMAX model is shown in Table II.

TABLE II
PARAMETERS CONSIDERED FOR OPTIMIZATION OF THE UNIVARIATE AND MULTIVARIATE VERSIONS OF MODELS

Parameters	Range of parameters for MLP	
	Univariate	Multivariate
Layers	3	3
ILN	1, 2, 3, 4, 5	3, 6, 9, 12, 15
HLN	10 to 500	10 to 500
OLN	1	1
Batch size	16, 32, 64	16, 32, 64
Look-back period	1 to 5	1 to 5
Optimizer	Adam	Adam
Activation function	ReLU, tanh	ReLU, tanh
Shuffle	Yes/No	Yes/No
	Range of parameters for LSTM	
	Univariate	Multivariate
ILN	1, 2, 3, 4, 5	3, 6, 9, 12, 15
HLN	10 to 500	10 to 500
OLN	1	1
Batch size	16, 32, 64	16, 32, 64
Look-back period	1 to 5	1 to 5
Optimizer	Adam	Adam
Activation function	ReLU, tanh	ReLU, tanh
Shuffle	Yes/No	Yes/No
	Range of parameters for AR	
	Univariate	Multivariate
Lag values	1 to 5	1 to 5
	Range of parameters for SMOreg	
	Univariate	Multivariate
e	1, 2, 3, 4	1, 2, 3, 4
C	0, 1	0, 1
Kernel Function	LK, PK, and RBF	LK, PK, and RBF
	Range of parameters for SARIMA	
	Univariate	Multivariate
AR (p)	0, 1, 2	0, 1, 2
I (d)	0, 1, 2	0, 1, 2
MA (q)	0, 1, 2	0, 1, 2
Trend	n, c, t, ct	n, c, t, ct
AR (P)	0, 1, 2	0, 1, 2
I (D)	0, 1, 2	0, 1, 2
MA (Q)	0, 1, 2	0, 1, 2
Cycle (m)	0, 1, 2	0, 1, 2

C. DROPOUTS IN THE MODELS

The training dataset needs to be sufficiently large to better train the NN models [51]. The time-series data from Tangni has fewer data to train the models. The less data could not fit the actual function, and the parameters of the NN model could overfit during learning. To avoid overfitting, the regularization techniques (dropout) could be applied to the model [52], [53]. In a regularization technique, randomly selected neurons with some probability (p) could be dropout during the training. In this experiment, the dropout was applied to the MLP and LSTM models. The dropout was also used for the recurrent connection of the LSTM model. The p-value of the dropout changes between 0.0 to 0.8 [50].

D. INPUTS AND OUTPUTS IN THE MODELS

The univariate model used only soil movement data for modeling. The input and output vectors were decided based on the lag period. Here, the first 62 weeks of soil movements data are provided for the training and the last 16-week data for testing the models.

The input data was provided as a packet for the multivariate version of the models. As shown in Fig. 5, the input packet is represented as X, and the target value of the corresponding packet is represented by the Y. An input packet contains the soil movement (W), the drill's index, and the sensor's depth in the drill. The length of the packet was the look-back period considered in the experiment. During the model training, the packets could be shuffled or not shuffled (see Fig. 5).

The prediction in both univariate and multivariate models was made sequentially. The first 62-week data were used predict the value for the next 63rd week. Then 64th week's value was predicted using the actual 62-week data and the previously predicted 63rd week's value.

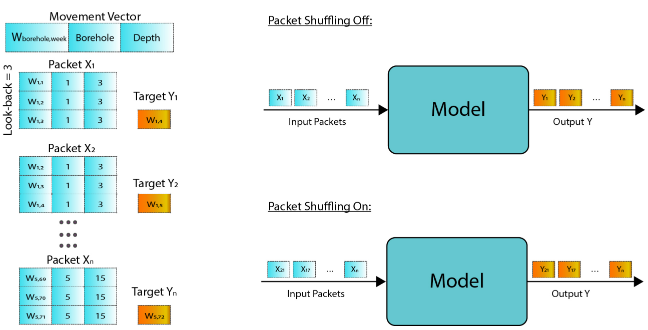

FIGURE 5. **Inputs and outputs in different models.**

E. WALK-FORWARD VALIDATION METHOD

Cross-validation (CV) is a critical technique used in ML to avoid overfitting and enhance the performance of models. When a model becomes very complex and tightly fits the training data, it leads to overfitting, causing inadequate performance when applied to unseen test data. Cross-validation helps to mitigate this issue by testing the model on multiple subsets of the data and measuring its performance. Several researchers used the CV method to avoid overfitting in the training process [54, 55]. However, the expanding window walk-forward validation method is a popular technique used for time-series to evaluate the performance of ML models [34]. As shown in Fig. 6., this method involves training the model on an initial training dataset and then gradually expanding the training dataset by including more recent data points. At each step, the model is evaluated on a fixed validation dataset that includes the next data point in the sequence. This process is repeated until the model has been trained across all available data points. Finally, the model with the minimum validation error was selected and subsequently evaluated on an unseen test dataset. The last 16 weeks of data were chosen as the test dataset in this experiment.

The minimum window size is another important consideration in time series analysis. The minimum window size of six was chosen for the SARIMA and SMOreg models used in this experiment. The minimum window size was equal to the look-back or lag value for other models used in the experiment.

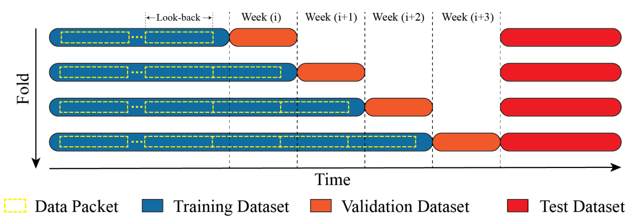

FIGURE 6. **Illustration of expanding window walk-forward validation method.**

F. PERFORMANCE MEASURES

The Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and coefficient of determination (R-squared) are metrics used to evaluate ML model performance. RMSE measures the average difference between predicted and actual values, giving higher weight to significant errors. Smaller RMSE values indicate better performance. MAPE calculates the average percentage difference between predicted and actual values. Smaller MAPE values indicate better performance. R-squared measures the goodness of fit, indicating the proportion of variance explained by the model. Higher R-squared values indicate a better fit, where an R-squared of 1 implies that the independent variables perfectly explain all the variability in the dependent variable. In contrast, an R-squared of 0 signifies no explanatory power. A negative R-squared value indicates that the model performs worse than a constant function that always predicts the mean of the actual data. The formulas for these metrics are defined as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)} \tag{8}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{9}$$

$$R - squared = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{10}$$

where y_i was the actual value, $\hat{y}_t$ was the predicted value, and $\bar{y}$ was the mean value for the soil movement. The value of n was 68 during the training and 16 during the testing.

VI. RESULTS AND DISCUSSION

The different parameters in univariate and multivariate versions of models were optimized using the training data. Table III shows the calibrated values of the ML models' parameters. For example, the univariate MLP model had 50 HLN and one look-back value. The multivariate version of MLP had 200 HLN and one look-back value. Similarly, the univariate LSTM had ten HLN and four look-back values. The multivariate version of LSTM had 200 HLN and three look-back values. The univariate AR model had the lag value of 1, which meant that the model required one past value to predict the next value. A look-back period of four was utilized for the VAR model. For the univariate and multivariate versions of the SMOreg model, the exponent (e) was set as one. The C parameter was selected as one for univariate and zero for the multivariate SMOreg models. The kernel function was chosen as a polynomial for both versions of the SMOreg model. Moreover, the univariate SARIMA and multivariate SARIMAX parameters showed zero value for the trend order parameter (p) signifies no autocorrelation between subsequent movements. Further, set value one for differencing order (d) for both models to make the time series stationarity. Furthermore, the parameter q = 0 for the univariate SARIMA model indicated that the model does not correlate with the previous week's movements to predict the current week's movement. The parameter q = 1 for the multivariate SARIMAX model indicated that the model correlated with the previous week's movements to predict the current week's movement. Similarly, the univariate SARIMA and multivariate SARIMAX models showed zero autoregression (D) value, zero differencing order, and MA window size (Q) was zero in the seasonal component. The univariate SARIMA and multivariate SARIMAX models showed the absence of seasonal components in the data. Both univariate SARIMA and multivariate SARIMAX model did not show any seasonal cycle in the data.

Table IV shows the outcomes of the univariate and multivariate models on the training. The RMSE was used to compare the models, and the R-squared values indicated how well the models fit the training data. In Table IV, the univariate AR model outperformed all others, ranking first in training performance, followed by the univariate SMOreg model in second place. After training, the ML models were tested on the test data from the five drills. The error of the models on the test dataset is presented in Table V. Based on the results of the five drills test, it was determined that the multivariate SARIMAX model exhibited the highest performance. It achieved an average RMSE value of 0.351 degrees (refer to Table V) and the highest R-squared value of 0.769. These findings suggest that approximately 76.9% of the variance in the drilling error can be explained by the independent variables in the multivariate SARIMAX model. These findings highlight the effectiveness of the multivariate SARIMAX model in accurately predicting and explaining the drilling error, showcasing its potential for practical applications in the domain. The best model (multivariate SARIMAX) in the testing is shown in bold font in Table V. The visualization in Fig. 7 depicts the actual and predicted values generated by the multivariate SARIMAX model. It provides a graphical representation of the model's performance over both the training and test datasets. We can visually assess the accuracy and effectiveness of the model's predictions by comparing the actual values with the predicted values (generated by the multivariate SARIMAX model).

TABLE III
OPTIMIZED PARAMETERS OF THE UNIVARIATE AND MULTIVARIATE VERSIONS OF MODELS

Parameters	Best parameters for MLP	
	Univariate	Multivariate
Layers	3	3
ILN	1	3
HLN	50	200
OLN	1	1
Batch size	64	64
Look-back period	1	1
Optimizer	Adam	Adam
Activation function	tanh	tanh
Shuffle	No	No
Dropouts	0	0
Parameters	Best parameters for LSTM	
	Univariate	Multivariate
ILN	4	9
HLN	10	200
OLN	1	1
Batch size	64	64
Look-back period	4	3
Optimizer	Adam	Adam
Activation function	tanh	tanh
Shuffle	No	No
Dropouts	0	0
Parameters	Best parameters for AR	
	Univariate	Multivariate
Lag values	1	4
Parameters	Best parameters for SMOreg	
	Univariate	Multivariate
e	1	1
C	1	0
Kernel Function	Polynomial	Polynomial
Parameters	Best parameters for SARIMA	
	Univariate	Multivariate
AR (p)	0	0
I (d)	1	1
MA (q)	0	1
Trend	n	n
AR (P)	0	0
I (D)	0	0
MA (Q)	0	0
Cycle (m)	0	0

TABLE IV
THE PERFORMANCE OF THE UNIVARIATE AND MULTIVARIATE MODELS IN THE TRAINING

Version	Model	Accuracy Measure	Error in the training dataset (in degrees)					
			Drill					Mean Error
			One-3 m	Two-12 m	Three-6 m	Four-15 m	Five-15 m	
Univariate	MLP	RMSE	1.476	0.114	0.225	0.660	1.486	0.792
		MAPE	0.756	0.030	94.355	0.060	1.746	19.389
		R-squared	0.546	-2.182	-119.90	-0.148	0.522	-24.232
	LSTM	RMSE	0.957	0.004	0.034	0.066	0.956	0.403
		MAPE	0.454	0.001	8.526	0.015	1.686	2.136
		R-squared	0.609	-1.420	-67.118	-93.154	0.450	-32.127
	AR	**RMSE**	**0.221**	**0.103**	**0.016**	**0.172**	**0.235**	**0.149**
		MAPE	0.163	0.008	4.814	4.077	1.120	2.036
		R-squared	0.986	-0.035	-0.006	0.941	0.973	0.572
	SMOreg	RMSE	0.877	0.001	0.017	0.025	0.762	0.336
		MAPE	43.201	0.030	54.667	0.336	63.512	32.349
		R-squared	0.777	0.000	0.909	0.596	0.641	0.584
	SARIMA	RMSE	0.908	0.000	0.023	0.020	0.878	0.366
		MAPE	0.425	0.000	2.524	0.003	0.722	0.735
		R-squared	0.775	1.000	-1.43	0.551	0.595	0.376
Multivariate	MLP	RMSE	0.870	0.108	0.099	0.270	0.706	0.411
		MAPE	0.301	0.035	45.303	0.051	0.821	9.302
		R-squared	0.657	-3.751	-51.809	-0.289	0.669	-10.90
	LSTM	RMSE	0.887	0.026	0.044	0.279	0.745	0.396
		MAPE	0.416	0.008	13.190	0.024	0.646	2.857
		R-squared	0.780	-7.170	-10.323	-86.922	0.644	-20.598
	AR	RMSE	0.761	0.001	0.015	0.179	0.774	0.346
		MAPE	0.504	0.000	2.511	0.037	1.721	0.955
		R-squared	0.852	0.132	0.216	0.221	0.730	0.430
	SMOreg	RMSE	2.764	0.399	0.053	1.605	1.409	1.246
		MAPE	88.552	16.227	2426.725	48.932	556.615	627.410
		R-squared	0.000	1.000	0.000	0.000	0.000	0.200
	SARIMAX	RMSE	0.964	0.044	0.019	0.058	0.881	0.393
		MAPE	0.467	0.009	3.591	0.009	0.804	0.967
		R-squared	0.746	0.000	-0.409	-2.857	0.592	-0.386

Note. The best model (univariate AR) in training is shown with the bold font.

TABLE V
THE PERFORMANCE OF THE UNIVARIATE AND MULTIVARIATE MODELS IN THE TESTING

Version	Model	Accuracy Measure	Error in the testing dataset (in degrees)					
			Drill					Mean Error
			One-3 m	Two-12 m	Three-6 m	Four-15 m	Five-15 m	
Univariate	MLP	RMSE	0.021	0.062	1.448	0.141	1.780	0.690
		MAPE	0.004	0.025	9.851	0.034	1.775	2.338
		R-squared	0.000	0.000	0.445	-0.339	0.182	0.058
	LSTM	RMSE	0.014	0.003	0.912	0.111	1.451	0.498
		MAPE	0.003	0.001	0.548	0.022	1.167	0.348
		R-squared	0.000	0.000	0.418	-0.327	0.220	0.0622
	AR	RMSE	0.744	0.000	1.556	0.096	1.239	0.727
		MAPE	0.125	0.000	0.700	0.021	1.016	0.373
		R-squared	0.000	0.000	-0.302	-0.074	0.316	-0.012
	SMOreg	RMSE	0.115	0.001	0.711	0.092	1.758	0.535
		MAPE	1.782	0.023	37.977	1.496	109.446	30.145
		R-squared	0.000	0.000	0.835	0.516	0.304	0.331
	SARIMA	RMSE	0.000	0.000	0.576	0.068	1.138	0.356
		MAPE	0.000	0.000	0.274	0.012	0.690	0.195
		R-squared	1.000	1.000	0.822	0.458	0.423	0.741
Multivariate	MLP	RMSE	0.946	1.014	2.012	1.172	1.620	1.353
		MAPE	0.134	0.310	7.131	0.301	0.997	1.775
		R-squared	0.000	0.000	-0.012	-132.341	-0.025	-26.476
	LSTM	RMSE	0.153	0.060	1.153	0.795	1.909	0.814
		MAPE	0.020	0.017	9.168	0.223	1.282	2.142
		R-squared	0.000	0.000	-0.088	-14.319	-0.901	-3.062
	AR	RMSE	0.610	0.001	1.424	0.099	1.158	0.658
		MAPE	0.105	0.000	0.548	0.019	0.770	0.288
		R-squared	0.000	-0.285	0.033	-0.015	0.122	-0.029
	SMOreg	RMSE	3.623	0.399	1.532	1.637	1.671	1.772
		MAPE	65.485	16.227	212.067	49.351	185.758	105.778

	R-squared	1.000	0.000	0.000	0.000	0.000	0.2000
SARIMAX	**RMSE**	**0.014**	**0.006**	**0.520**	**0.066**	**1.151**	**0.351**
	MAPE	0.003	0.003	0.276	0.012	0.719	0.202
	R-squared	0.986	0.988	0.822	0.629	0.421	0.769

Note. The best model (multivariate SARIMAX) in the testing is shown in bold font.

VII. CONCLUSIONS

Machine learning models might be used to predict the soil movements for preparedness for possible hazards. This study developed and compared many univariate and multivariate ML models by relying upon data from the Tangni landslide, India. In this experiment, the dataset was divided into training and test sets to evaluate the performance of the models. The evaluation involved analyzing the RMSE, MAPE, and R-squared values to compare the performance of the models. The parameters in the models were chosen for the analysis based on an initial 80% of the data (training) using the expanding window walk-forward validation method and then evaluated using the remaining 20% (testing). The models were made to perform one step ahead prediction. Whereby, given the information on soil movements in the previous weeks, the models predict the possible movement for the following week. Furthermore, the look-back was changed from 1 to 5 based on auto-correlation analysis. Additionally, the PSO technique was used to initialize the weight for the neural network (MLP, LSTM). This technique helped the NNs to find the optimal solution without getting trapped to a local minimum. During the training, the univariate AR model demonstrated the highest performance, while the univariate SMOreg model achieved the second-highest performance. The multivariate SARIMAX model performed better during testing, and the univariate SARIMA model was identified as the second-best model on the test dataset. Our analysis showed that the univariate and multivariate versions of statistical models performed better during the training and testing phase. It was also noted that the time series from the drills show that the training time series has low variance, whereas the testing time series has high variance. Our results pointed to the statistical (SARIMA and SARIMAX) models' ability to handle the high variance in the unseen dataset. It was evident from the high R-squared values achieved by both models, indicating their ability to explain a significant portion of the drilling error variance in the test dataset. The seasonal effects in the movement data could be a possible explanation for our findings. Whereby the movements were noted to be high in the monsoon. For example, the time series from drill five showed significant movement in weeks 55 to 70. Next, our analysis showed that both univariate and multivariate versions of the models were suitable for the soil movement predictions. However, the multivariate variants performed slightly better, probably due to the more parameters that captured generalized patterns.

First, our findings suggested that the univariate and multivariate models (such as multivariate SARIMAX and univariate SARIMA) can capture aggregated soil movements over time at actual landslide sites. Second, the multivariate method outperformed univariate models in the testing phase. For example, the multivariate SARIMAX outperformed the univariate SARIMA model. The reason could be that the multivariate SARIMAX had more parameters to train. Furthermore, the statistical model outperformed the NN models such as MLP and LSTM. The reason could be that the NN models perform better when there is a huge amount of data to train the parameters, while our dataset had only 62 points to train [49]. Statistical models such as univariate models (AR and SARIMA) and multivariate models (VAR and SARIMAX) could train on the small dataset [20].

Moreover, the univariate version of SMOreg, MLP, and LSTM models outperformed the corresponding multivariate variants (Table V). One reason could be that the multivariate data has three attributes, and two of them, drill, and sensor depth, were constant and showed zero variance that misled the models. Furthermore, the multivariate VAR model outperformed the univariate AR model in the test data. We can infer from the results, as mentioned earlier, that the multivariate version of the models could predict the soil movement better than the univariate version of the models. The comparison of univariate and multivariate ML models is the first work on soil movement prediction at the Tangni landslide.

Another observation from this research was that the univariate and multivariate models showed overfitting in the training dataset. Whereby test error was more, and training error was less. Even though we applied dropouts in the models, it could not prevent the models from overfitting, possibly due to the limited number of data points. Another reason could be that the training and testing dataset had different patterns. For example, drill three has no movement in training data but had several significant movements in the testing dataset, resulting in more errors for the testing data. We also observed the larger RMSE values for data with more variability (Sensor at 15 m depth in drill five). We intend to address this issue by implementing more advanced models and incorporating more variables. Thus, we would attempt a hybrid methodology or more advanced models observed during the background study as part of our future research works. To create multivariate, RNN models, we want to extend these investigations to include geotechnical or geological parameters, such as water table elevation, slope structure, material density, shear strength, and rainfall characteristics.

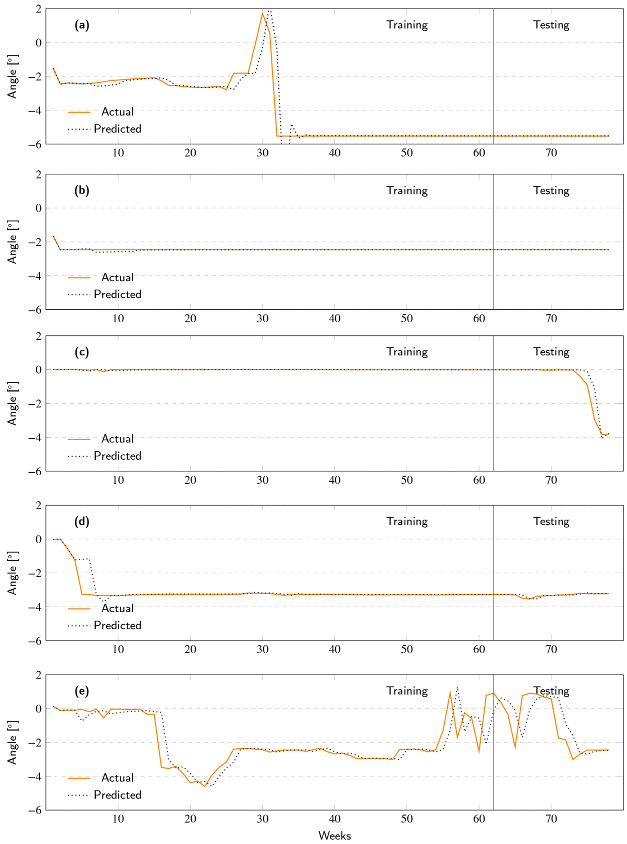

FIGURE 7. The soil movement prediction over weeks from the best performing multivariate SARIMAX in the training and testing data. (a) Drill One-3 m. (b) Drill Two-12 m. (c) Drill Three-6 m. (d) Drill Four-15 m. (e) Drill Five-15 m.

We would also attempt to develop deep learning models such as generative adversarial networks (GAN), variational generative adversarial networks (VGAN), and time-series GAN to generate more synthetic soil movement data.

ACKNOWLEDGMENT

We express our gratitude to the Department of Science and Technology, India, as well as the District Disaster Management Authority of Mandi, Kinnaur, and Kangra for their generous funding support for this research project. Thanks to the director of Defence Geoinformatics Research Establishment (DGRE) DROD for providing the sensor's data from the Tangni landslide for this research. We are also thankful to the IIT Mandi for providing the space and computing facilities for this research work.

CONFLICT OF INTEREST

Authors declare no conflict of interest.

REFERENCES

[1] R. K. Pande, "Landslide problems in Uttaranchal, India: Issues and challenges," *Disaster Prevention and Management: An International Journal*, vol. 15, no. 2, pp. 247–255,2006.
[2] S. Parkash, "Historical records of socio-economically significant landslides in India," *South Asia Disaster Stud*, vol. 4, no. 2, pp. 177–204, 2011.
[3] A. Pathania, P. Kumar, Priyanka, R. Singh, P. Chaturvedi, K. V. Uday, and V. Dutt, "A low cost, sub-surface iot framework for landslide monitoring, warning, and prediction," in *Proc. 2020 International Conference on Advances in Computing, Communication, Embedded and Secure Systems*, 2020.
[4] O. Korup and A. Stolle, "Landslide prediction from machine learning," *Geology Today*, vol. 30, no. 1, pp. 26–33, 2014.
[5] R. Duda, E. H. Peter, and G. S. David, *Pattern Classification (2nd ed.)*. New Jersey, US: John Wiley & Sons, 2012.
[6] G. Duan and R. Niu, "A method of dynamic data mining for landslide monitoring data," *Journal of Yangtze River Scientific Research Institute*, vol. 30, no. 5, pp. 38–41,2013.
[7] A. Aggarwal, M. Alshehri, M. Kumar, O. Alfarraj, P. Sharma, and K. R. Pardasani, "Landslide data analysis using various time-series forecasting models," *Computers & Electrical Engineering*, vol. 88, p. 106858, 2020.
[8] D. Geyuan, "Displacement prediction of the zhujiadian landslide with outliers," *Journal of Engineering Geology*, vol. 28, no. 1, pp. 132–140, 2020.
[9] J. Li, W. Wang, Z. Han, "A variable weight combination model for prediction on landslide displacement using AR model, LSTM model, and SVM model: a case study of the Xinming landslide in China," Environmental Earth Sciences vol. 80, pp. 1–14, 2021a.
[10] A. Pathania, P. Kumar, Priyanka, A. Maurya, M. Kumar, R. Singh, P. Chaturvedi, K. V. Uday, and V. Dutt, "Predictions of soil movements using persistence, auto- regression, and neural network models: A case-study in mandi, India,"*International Journal of Swarm Intelligence*, vol. 7, no. 1, pp. 94–109, 2022.
[11] Y. Wang, H. Tang, T. Wen, and J. Ma, "A hybrid intelligentapproach for constructing landslide displacement prediction intervals," *Applied Soft Computing*, vol. 81, p. 105506,2019.
[12] Y. Xing, J. Yue, C. Chen, D. Cai, J. Hu, and Y. Xiang, "Prediction interval estimation of landslide displacement using adaptive chicken swarm optimization-tuned support vector machines," *Applied Intelligence*, vol. 51, no. 11, pp.8466–8483, 2021.
[13] L. Zhang, B. Shi, H. Zhu, X. B. Yu, H. Han, and X. Fan, "Pso-svm-based deep displacement prediction of majiagoulandslide considering the deformation hysteresis effect," *Landslides*, vol. 18, no. 1, pp. 179–193, 2021.
[14] H. Chen and Z. Zeng, "Deformation prediction of land- slide based on improved backpropagation neural network,"*Cognitive Computation*, vol. 5, no. 1, pp. 52–62, 2013.
[15] P. Xie, A. Zhou, and B. Chai, "The application of long short-term memory (lstm) method on displacement pre- diction of multifactor-induced landslides," *IEEE Access*, vol. 7, pp. 54 305–54 311, 2019.
[16] Y. Xing, J. Yue, and C. Chen, "Interval estimation of landslide displacement prediction based on time series de- composition and long short-term memory network," *IEEE Access*, vol. 8, pp. 3187–3196, 2019.
[17] L.M. Li, M.Y. Zhang, Z.Z. Wen, "Dynamic prediction of landslide displacement using singular spectrum analysis and stack long short-term memory network," Journal of Mountain Science vol. 18, pp. 2597–2611, 2021b.
[18] X. Niu, J. Ma, Y. Wang, J. Zhang, H. Chen, and H. Tang, "A novel decomposition-ensemble learning model based on ensemble empirical mode decomposition and recurrent neural network for landslide displacement prediction," *Ap-plied Sciences*, vol. 11, no. 10, p. 4684, 2021.
[19] Z. Taorui, J. Hongwei, L. Qingli, and Y. Kunlong, "Land- slide displacement prediction based on variational mode decomposition and micgwo-lstm model," *Stochastic Environmental Research and Risk Assessment*, pp. 1–20, 2022.
[20] O. Na, "Generalized information criterion for the ar model," *Journal of the Korean Statistical Society*, vol. 46, no. 1, pp. 146–160, 2017.
[21] G. Papacharalampous, H. Tyralis, and D. Koutsoyiannis, "Univariate time series forecasting of temperature and precipitation with a focus on machine learning algorithms: A multiple-case study from greece," *Water resources man-agement*, vol. 32, no. 15, pp. 5207–5239, 2018.

[22] M. W. Kadous, "Learning comprehensible descriptions of multivariate time series." in *ICML*, vol. 454, 1999, p. 463.
[23] J. D. Preez and S. F. Witt, "Univariate versus multivariate time series forecasting: an application to international tourism demand," *International Journal of Forecasting*,vol. 19, no. 3, pp. 435–451, 2003.
[24] E. Cadenas, W. Rivera, R., Campos-Amezcua, and C. Heard, "Wind speed prediction using a univariate arima model and a multivariate narx model," *Energies*, vol. 9,no. 2, p. 109, 2016.
[25] Y. Zhang, M. Zhong, N. Geng, and Y. Jiang, "Forecasting electric vehicles sales with univariate and multivariate timeseries models: The case of china," *PloS one*, vol. 12, no. 5,p. 0176729, 2017.
[26] D. Miller and J.-M. Kim, "Univariate and multivariate machine learning forecasting models on the price returnsof cryptocurrencies," *Journal of Risk and Financial Man- agement*, vol. 14, no. 10, p. 486, 2021.
[27] S. Khanduri, "Landslide distribution and damages during 2013 deluge: a case study of chamoli district, uttarakhand," *Journal of Geography and Natural Disasters*, vol. 8, no. 2,pp. 1–10, 2018.
[28] P. Chaturvedi, B. Jaiswal, S. Sharma, and N. Tyagi, "Instru-mentation based dynamics study of tangni landslide near chamoli, uttrakhand," *International Journal of Research inAdvent Technology*, vol. 2, no. 10, pp. 127–132, 2014.
[29] S. Sharma, S. Dhar, S. P. Mishra, P. Chaturvedi, and B. Jaiswal, "Instrumentation and slope stability analysis of tangni landslide," *Journal of Engineering Geology*, vol. 40,no. 1, pp. 145–157, 2015.
[30] THDC, "Baseline environment, impacts and mitigation measures," Tehri Hydro Development Corporation Limited, Uttarakhand, Tech. Rep. 2008026/EC/Final Report, Nov. 2009.
[31] IndiaTvNews. (2013) Landslides near Badrinath in uttarakhand. [Online]. Available: https://rb.gy/dn9u3b
[32] E. Lee, J. Khan, J. W. Son, & K. Kim. "An Efficient Feature Augmentation and LSTM-Based Method to Predict Maritime Traffic Conditions," *Applied Sciences*, vol. 13(4), 2556, 2023.
[33] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.
[34] H. S. Yoon, "Time Series Data Analysis using WaveNet and Walk Forward Validation," in *Journal of the Korea Society for Simulation*, 30(4), 1-8, 2021.
[35] F. Rosenblatt, "Principles of neurodynamics. perceptrons and the theory of brain mechanisms," Cornell AeronauticalLab Inc Buffalo NY, Tech. Rep., 1961.
[36] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learn-ing internal representations by error propagation, parallel distributed processing: Explorations in the microstructure of cognition, mass," *Foundations. MIT Press, Cambridge*, vol. 1, pp. 318–362, 1986.
[37] Z. Lin, H. Yanwen, X. Jie, F. Xiong, L. Qiaomin, and W. Ruchuan, "Trust evaluation model based on pso and lstm for huge information environments," *Chinese Journalof Electronics*, vol. 30, no. 1, pp. 92–101, 2021.
[38] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference onneural networks*, vol. 4. IEEE, 1995, pp. 1942–1948.
[39] Y. Shi *et al.*, "Particle swarm optimization: developments, applications and resources," in *Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546)*, vol. 1. IEEE, 2001, pp. 81–86.
[40] L. C. Jain and L. R. Medsker, *Recurrent neural networks: design and applications*. CRC Press, Inc., 1999.
[41] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.
[42] V. Cortes, "Cortes c., vapnik v," *Support-vector networks, Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.
[43] J. Platt, "Sequential minimal optimization: A fast algorithmfor training support vector machines," 1998.
[44] E. Ruth, A. J. Sørensen, and T. Perez, "Thrust allocation with linear constrained quadratic cost function," *IFAC Proceedings Volumes*, vol. 40, no. 17, pp. 337–342, 2007.
[45] A. Patle & D.S. Chouhan, "SVM kernel functions for classification. In 2013 International Conference on Advances in Technology and Engineering (ICATE)," IEEE, pp. 1-9, 2013.
[46] E. J. Hannan and B. G. Quinn, "The determination of the order of an autoregression," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 41, no. 2, pp. 190–195, 1979.D.
[47] Asteriou and S. G. Hall, "Arima models and the box– jenkins methodology," *Applied Econometrics*, vol. 2, no. 2, pp. 265–286, 2011.
[48] B. G. Quinn, "Order determination for a multivariate autoregression," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 42, no. 2, pp. 182–185, 1980.
[49] M. Kuhn, K. Johnson et al., in *Applied predictive modeling.* Vol 26, 2013, New York, Springer.
[50] M. Cools, E. Moons, and G. Wets, "Investigating the variability in daily traffic counts through use of arimax and sarimax models: assessing the effect of holidays on two site locations," *Transportation research record,* vol. 2136, no. 1, pp. 57–66, 2009.
[51] A. Alwosheel, S. van Cranenburgh, and C. G. Chorus,"Is your dataset big enough? sample size requirementswhen using artificial neural networks for discrete choice analysis," *Journal of choice modelling*, vol. 28, pp. 167– 182, 2018.
[52] Y. Gal and Z. Ghahramani, "A theoretically grounded appli- cation of dropout in recurrent neural networks," *Advances in neural information processing systems*, vol. 29, 2016.
[53] V. Pham, T. Bluche, C. Kermorvant, and J. Louradour, "Dropout improves recurrent neural networks for handwrit-ing recognition," in *2014 14th international conference on frontiers in handwriting recognition*. IEEE, 2014, pp. 285–290.
[54] J. Khan, M. Fayaz, A. Hussain, S. Khalid, W. K. Mashwani and J. Gwak, "An Improved Alpha Beta Filter Using a Deep Extreme Learning Machine," in *IEEE Access*, vol. 9, pp. 61548-61564, 2021, doi: 10.1109/ACCESS.2021.3073876.
[55] J., Khan, E., Lee, & K. Kim, "A higher prediction accuracy–based alpha–beta filter algorithm using the feedforward artificial neural network," in *CAAI Transactions on Intelligence Technology*, 2022. https://doi.org/10.1049/cit2.12148

Praveen Kumar is a Ph.D. student at the Indian Institute of Technology Mandi (IIT Mandi), School of Computing & Electrical Engineering. He has been involved in creating an IoT framework based on MEMS that helps to track landslides. Internet of Things, Machine Learning, and Deep Learning are his areas of concern.

Priyanka P. is an M.S. (by research) student at the Indian Institute of Technology Mandi (IIT Mandi), School of Computing & Electrical Engineering. She is working in the area of landslide prediction, warning systems, and InSAR analysis of the landslide affected zone in the Himalayan regions.

J. Dhanya serves as Project Asso- ciate at the School of Engineering, IIT Mandi. She attained Ph.D. from IIT Madras for her thesis titled" Engineer- ing Models for Earthquake slip field and Indo-Gangetic Basin" (September- 2019). She works in the interdisci- plinary area of engineering seismology. She focuses her research on earthquake source characterization, earth medium effects on ground motion, random field modeling, Seismic data modeling, prediction models for seismic activity, and Hazard estimations

K. V. Uday serves as an Assistant Professor at the IIT Mandi, in the School of Engineering. In 2013, he earned his Ph.D. degree from the Depart- ment of Civil Engineering, IIT Bom- bay. Geotechnical engineering, slope stabilisation, landslide (early notice, planning and mitigation), structure re- tention, geosynthetics, environmental geotechnics are his studies.

Varun Dutt is an Associate Professor at the IIT Mandi, in the School of Com- puter and Electrical Engineering. In 2011, he earned his Ph.D. degree from Carnegie Mellon University (CMU) in Engineering and Public Policy. His research focuses on the intersection of computer science, economics, and decision-making, with a specific em- phasis on how to use computational and experimental models to take decisions on management, environmental, and socioeconomic issues.