

## Research Article

# Modeling Soil Temperature for Different Days Using Novel Quadruplet Loss-Guided LSTM

Xuezhi Wang <sup>1,2</sup> Wenhui Li <sup>1,2</sup> Qingliang Li <sup>2,3</sup> and Xiaoning Li <sup>1,3</sup>

<sup>1</sup>College of Computer Science and Technology, Jilin University, Changchun 130012, China

<sup>2</sup>Symbol Computation and Knowledge Engineer of Ministry of Education, Jilin University, Changchun, Jilin Province, China

<sup>3</sup>School of Computer Science and Technology, Changchun Normal University, Changchun 130032, China

Correspondence should be addressed to Xiaoning Li; [lixiaoning@ccsfu.edu.cn](mailto:lixiaoning@ccsfu.edu.cn)

Received 6 January 2022; Accepted 22 January 2022; Published 17 February 2022

Academic Editor: Xin Ning

Copyright © 2022 Xuezhi Wang 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.

Soil temperature ( $T_s$ ), a key variable in geosciences study, has generated growing interest among researchers. There are many factors affecting the spatiotemporal variation of  $T_s$ , which poses immense challenges for the  $T_s$  estimation. To enrich processing information on loss function and achieve better performance in estimation, the paper designed a new long short-term memory model using quadruplet loss function as an intelligence tool for data processing (QL-LSTM). The model in this paper combined the traditional squared-error loss function with distance metric learning between the sample features. It can zoom analyze the samples accurately to optimize the estimation accuracy. We applied the meteorological data from Laegern and Fluehli stations at 5, 10, and 15 cm depth on the 1st, 5th, and 15th day separately to verify the performance of the proposed soil temperature estimation model. Meanwhile, this paper inputs the variables into the proposed model including radiation, air temperature, vapor pressure deficit, wind speed, air pressure, and past  $T_s$  data. The performance of the model was tested by several error evaluation indices, including root mean square error (RMSE), mean absolute error (MAE), Nash-Sutcliffe model efficiency coefficient (NS), Willmott Index of Agreement (WI), and Legates and McCabe index (LMI). As the test results at different soil depths show, our model generally outperformed the four existing advanced estimation models, namely, backpropagation neural networks, extreme learning machines, support vector regression, and LSTM. Furthermore, as experiments show, the proposed model achieved the best performance at the 15 cm depth of soil on the 1st day at Laegern station, which achieved higher WI (0.998), NS (0.995), and LMI (0.938) values, and got lower RMSE (0.312) and MAE (0.239) values. Consequently, the QL-LSTM model is recommended to estimate daily  $T_s$  profiles estimation on the 1st, 5th, and 15th days.

## 1. Introduction

Soil temperature ( $T_s$ ) is a main physical variable of the land surface, which has a direct influence on the atmosphere [1]. Relevant fields including geoscience and forestry application aspects have drawn attention from researchers [2, 3]. In principle, all the interactions in terrestrial ecosystems are accompanied by  $T_s$  variations since they involve energy exchanges.  $T_s$  is an essential factor for growing crops that can facilitate the development of the root system by impacting microbial activity, soil decomposition, and fluidity of soil water [4]. In addition, the death of animals and plants produces plenty of carbon substrates and a high volume of

greenhouse gases in the soil. Consequently, it results in an increase in  $T_s$ , thus expediting carbon dioxide emission to the atmosphere [5]. Therefore, accurate  $T_s$  monitoring is crucial for agricultural management and atmosphere environment forecast. However,  $T_s$  data in most areas is still measured by using traditional sensors, and the  $T_s$  data cannot be collected at different depths [6].

Therefore, it can be used to solve some problems in different fields for the study of  $T_s$  estimation. The essential environmental factors have a great influence on the accuracy of  $T_s$  estimation. At present,  $T_s$  is mainly predicted by methods based on physical models and data-driven methods. The physical method is based on the heat

conduction model to estimate  $T_s$  [7]. Meanwhile, the method is greatly affected by related physical parameters and the scale problem [8].

The data-driven methods can explore the internal relationship between  $T_s$  and the surrounding environmental factors for  $T_s$  estimation. At present, several predictive models based on machine learning methods are used for estimating  $T_s$  [9–14]. For example, ANN is composed of a complex network structure that imitates the structure and function of the brain's neural network, and it has powerful data processing capabilities. Bilgili applied the multilayer perceptron (MLP) model to adequately describe  $T_s$  distribution at a monthly temporal scale from meteorological data [15]. Kisi et al. used three machine learning models to estimate monthly  $T_s$  at the soil depth of 5 cm and 10 cm, respectively, and verified the predictive performance of radial basis neural networks performed better than that of generalized regression neural networks and multilayer perceptron models [9]. But generalized regression neural networks had the better performance for deeper depth (50 cm and 100 cm). Kisi et al. applied ANN-based models to predict long-term  $T_s$  at a monthly temporal scale, and they found that genetic programming generated the best performance with the meteorological data [16]. Zeynoddin et al. applied a multilayer perceptron (MLP) model to describe daily  $T_s$  distribution at three soil depths (5, 10, and 20 cm) from past measurements of  $T_s$  [17]. Samadianfard et al. processed the meteorological data such as  $T_a$ ,  $W$ ,  $RH$ ,  $R_s$ ,  $Sh$ , and air pressure ( $Ap$ ) and integrated ANN-based models separately to predict  $T_s$  at a monthly temporal scale [18]. Mehdizadeh et al. noted that machine learning models combined with time series models performed better performance than the predictive models based on the single machine learning method or single time series method for predicting  $T_s$  at a daily temporal scale [19]. Moazenzadeh et al. proposed SVR with krill herd algorithm (SVR-KHA) method in modeling  $T_s$  estimation at different depths (5, 10, 20, 30, 50, and 100 cm), which achieved the best performance, compared to SVR and SVR with firefly algorithm (SVR-FA) [11]. Delbari et al. proposed an SVR-based model to compute daily  $T_s$  at three depths (5, 30, and 100 cm) in Iran [12]. The ELM network featured by a single hidden layer could improve the learning speed and accuracy of the algorithm and can model the accurate  $T_s$ . Nahvi et al. used the improved ELM model on daily  $T_s$  estimation based on the self-adaptive evolutionary method and verified the improved predictive model can estimate the adequate  $T_s$  [20]. Sanikhani et al. tested the data from the Mersin station, and the results that show the performance of ELM has the best predictive performance than other predictive models. [14]. Feng et al. tested the loess plateau data with ELM and random forests (RF) and ANN-based models showed that ELM had the better performance for estimation  $T_s$  of half-hourly at different soil depths [13].

As a time loop neural network containing complex neural network modules, LSTM is used in this paper to solve long-term dependence problems, which can effectively alleviate gradient vanishing through the extraction of required features by the gate control unit. LSTM network [21] can

learn long-term and short-term behaviors, and it has seen application in vast areas. By integrating LSTM and SVR, Guo et al. significantly improved the prediction accuracy of abnormal passenger flow fluctuations [22]. In hydrology, Zhang et al. designed a novel LSTM model with the dropout scheme to estimate the depth of the water table [23]. In the atmosphere field, Qing et al. estimated the solar irradiance based on the LSTM network [24]; the results showed the method could avoid the overfitting of the model. Li et al. designed a new GANs-LSTM model and noted that it could serve as an alternative method to estimate  $T_s$  [25].

This article focuses on the following issues. First, we select the environmental factors which will affect  $T_s$  estimation.  $T_s$  memory can help the predictive model “remember” a warm or cold condition when the anomaly is forgotten by the atmosphere forcing. In addition, recent literature reviews have revealed that the input for prediction models is either the past measurements of  $T_s$  or other meteorological information ( $T_a$ ,  $W$ ,  $RH$ ,  $R_s$ ,  $Sh$ , and  $Ap$ ). Assume that the prediction models are constructed using input combinations of past  $T_s$  and other meteorological information; how does the prediction model performance? The second question is about the construction of loss function in LSTM. The predictive model for  $T_s$  estimation is a regression predictive modeling problem that involves predicting a real-valued quantity. The loss function is crucial for optimizing the predictive model which could express the degree of difference between predicted and observed  $T_s$ ; meanwhile, it can optimize the predictive model by updating the weights. Recently, most previous studies in loss function of regression predictive model mainly focused on the distance metric learning between predicted values and real values [26–29]. However, the distance metric learning between the sample features (environmental factors) is usually ignored which has already been successfully applied to image processing [30–32]. To enrich information processing in loss function and further improve the estimation performance, how can we construct a novel loss function by combinations of distance metric learning? The last question is about timescale evaluating for  $T_s$  estimation. In previous studies, any evaluation at short-term  $T_s$  estimation (half-hourly, hourly, daily timescales) does not consider the timeliness of long-term  $T_s$  estimation. However, any evaluation at long-term  $T_s$  estimation (monthly timescale) does not include the information of  $T_s$  in a small timescale. An ideal decision-support tool for  $T_s$  estimation should provide a multifarious decision-making basis. How can we design a prediction scheme at the same timescale evaluation that provides not only the short-term decision-making basis but also the long-term decision-making basis?

This paper proposed a novel quadruplet loss function based on the LSTM network that combines traditional squared-error loss function with distance metric learning between the sample features, called QL-LSTM. The traditional squared-error loss function is usually applied to the predictive task with great accuracy. The current limitation of this loss function, however, involves the special variation on  $T_s$  based on different predictors. As shown in Figure 1, we have made labels according to  $T_s$

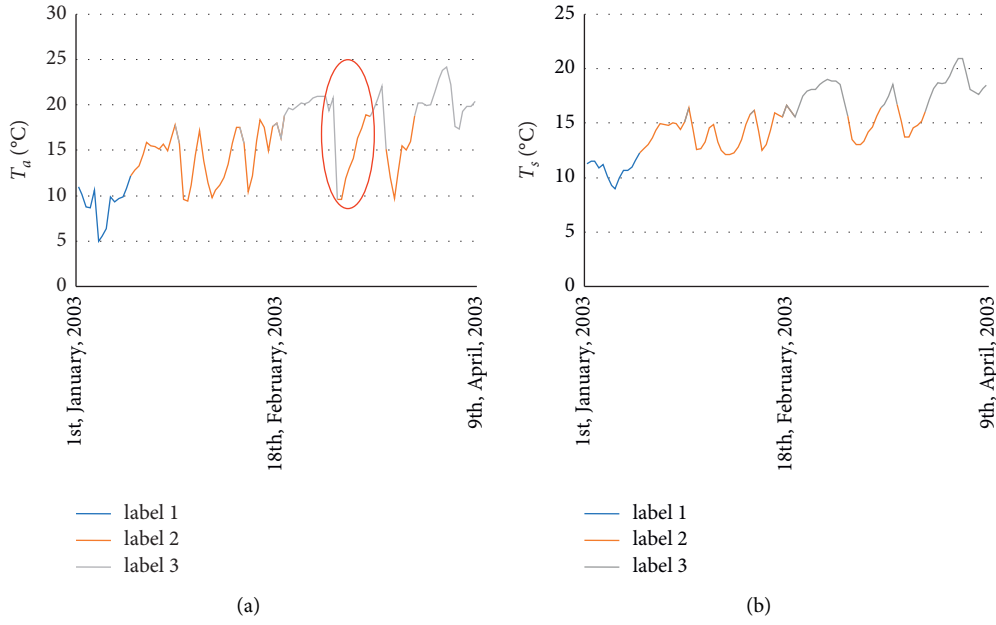


FIGURE 1: Variations of the daily air temperature (a) and soil temperature (b) at Laegern station (located in Switzerland) during 1st, January 2003–9th April 2003 (100 days).

values. The  $T_s$  data which are in the same range are made the same label (the  $T_s$  data which are in the range of 8–12°C are labeled as “1”, the  $T_s$  data which are in the range of 12–16°C are labeled as “2”, and the  $T_s$  data which are in the range of 16–22°C are labeled as “3”). Meanwhile,  $T_a$  data are labeled as the same as  $T_s$  data. In Figure 1(a), we noticed that the  $T_s$  data with the same label are almost within a stable range. However, in the red ellipse, Figure 1(b), we observed that similar  $T_a$  values may have different labels ( $T_s$  data with similar  $T_a$  values may vary considerably). The data with this feature will prevent predictive models from accurately exploring the internal relationship between  $T_s$  and the surrounding environmental factors discovering. To address this problem, the idea of triplet loss [33] is considered in this paper. Triplet loss optimization allows the anchor and positive points to accumulate and therefore prevent the negative points and realize the similarity calculation of samples. This approach can enrich processing information of loss function and overcome the disadvantage of the traditional squared-error loss function and further improve the estimation performance.

The main three contributions of this research paper are summarized as follows:

- (1) As we know, the proposed method that combined traditional squared-error loss function with distance metric learning between the sample features is a new approach to be used for  $T_s$  estimation.
- (2) Daily-scale prediction scheme was designed to provide the multifarious decision-making basis and was used to estimate the  $T_s$  on the next 1st, 5th, and 15th day. To achieve this end, we input the

meteorological and past  $T_s$  data to the estimation model.

- (3) Results showed that our QL-LSTM method outperformed the existing advanced methods in most cases.

## 2. Data and Methods

**2.1. The Framework of Soil Temperature Estimation.** The corresponding meteorological data  $T_s$  as the input of our QL-LSTM model are obtained from FLUXNET at first. In the meantime, several other advanced models based on data-driven technology (SVR, BPNN, ELM, and LSTM) were considered in  $T_s$  estimation. Traditional squared-error loss function and distance metric learning between the sample features were integrated into our model for accurate exploration of the internal relationship between  $T_s$  and the surrounding environmental factors. Finally, the comparison of model performance is reflected by five evaluating indicators (RMSE, MAE, NS, WI, and LMI). Figure 2 denoted the flow chart of soil temperature estimation.

**2.2. Long Short-Term Memory (LSTM) Network.** LSTM can process and learn long-term dependence problems. Due to the characteristics of the LSTM network, we use it to explore the internal relationship between  $T_s$  and the surrounding predictors. LSTM controls the transmission state through the gating state, remembers what needs to be remembered, and forgets unimportant information. Figure 3 shows the internal structure of an LSTM cell, and the calculation formula of the LSTM is as follows:

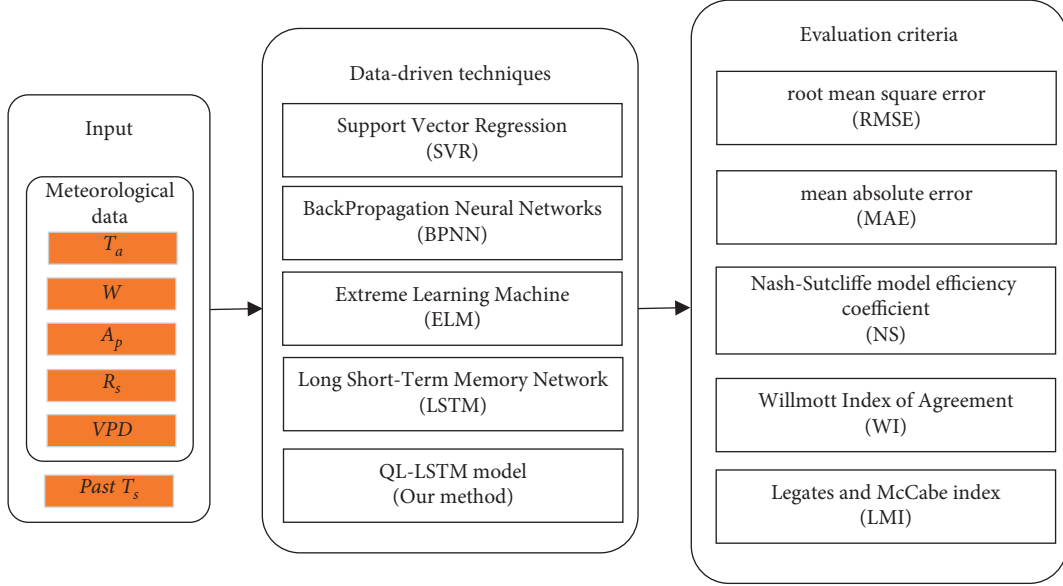


FIGURE 2: The flow chart of soil temperature estimation.

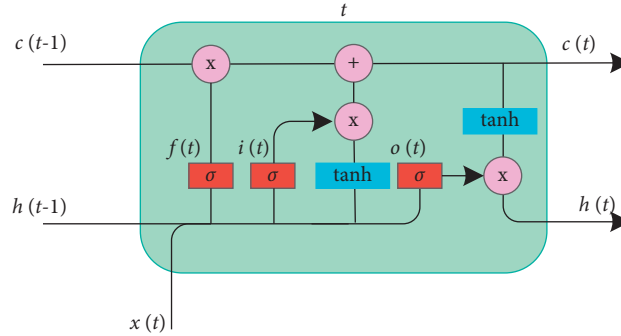


FIGURE 3: The internal structure of an LSTM cell.

$$\begin{aligned}
 i(t) &= \sigma(W_{ih}h(t-1) + W_{ix}x(t) + W_{ic}c(t-1) + b_i), \\
 f(t) &= \sigma(W_{fh}h(t-1) + W_{fx}x(t) + W_{fc}c(t-1) + b_f), \\
 c(t) &= f(t) \otimes c(t-1) + i(t) \otimes \tanh(W_{ch}h(t-1) + W_{cx}x(t) + b_c), \\
 o(t) &= \sigma(W_{oh}h(t-1) + W_{ox}x(t) + W_{oc}c(t) + b_o), \\
 h(t) &= o(t) \otimes \tanh(c(t)), \\
 \hat{y}(t) &= W_{yh}h(t) + b_o,
 \end{aligned} \tag{1}$$

where  $x(t)$  is the input data, and  $\hat{y}(t)$  is the output data;  $i(t)$ ,  $f(t)$ , and  $o(t)$  denote the input gate, forget gate, and output gate;  $c(t)$  represents the unit status at the current moment;  $h(t)$  is the current output value;  $\sigma(\cdot)$  and  $\tanh(\cdot)$  are the activation functions;  $W$  and  $b$  denote the weight matrix and bias term.

**2.3. Triplet Loss.** Triplet loss is a significant “learning criterion” for optimizing the predictive models, which is applied for adjusting the weight parameters of predictive models, including anchor (Anchor) example, positive

(Positive) example, and negative (Negative) example. The similarity calculation of the samples is realized through triplet loss learning, which makes the anchor-to-positive distance smaller than the anchor-to-negative distance. And Figure 4 denoted the visual representation of triplet loss.

Equation (2) expresses the objective function of triplet loss as follows:

$$\sum_i^N \left[ \|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right]_+, \tag{2}$$

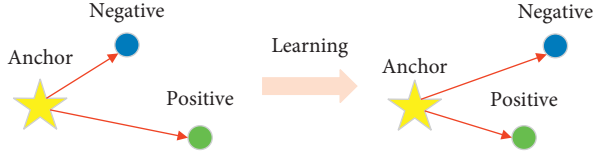


FIGURE 4: A visual representation of triplet loss.

where  $f(x_i^a)$ ,  $f(x_i^p)$ , and  $f(x_i^n)$  are the corresponding feature expression obtained by training a parameter in the triplet;  $\alpha$  represents the minimum interval between the anchor-to-positive distance and the anchor-to-negative distance; the value of  $[\cdot]_+$  defines the degree of loss.

**2.4. QL-LSTM Model.** Previous analysis shows that LSTM with traditional squared-error loss function could not accurately discover the special relationship between  $T_s$  and surrounding predictors. To address this problem, inspired by the study of triplet loss, we combined a predictive model with distance metric learning between the sample features. As far as we know, the method based on distance metric learning between the sample features has not been used to estimate  $T_s$  ever. It must be noted that the distance metric learning between the sample features is first proposed in the field of image processing. However, there is no description of the similarity of samples for  $T_s$  estimation. In this paper, the clustering method is used to label samples; thus, distance metric learning between the sample features could be further applied in  $T_s$  estimation.

The framework of our QL-LSTM is shown in Figure 5. Firstly, for its ability to cluster data efficiently and scalability, the  $T_s$  data were quantized by the clustering method (called Birch) [34]. In the quantization step, any  $T_s$  data quantized to the same label will be defined as similar samples (positive). In contrast, any  $T_s$  data quantized to different labels will be defined as the dissimilar samples (negative). It is worthwhile to observe that the number of labels should be neither too large nor too small [35]. Hence, the Calinski Harabasz Score (CH) and Y\_Silhouette\_score (S) are used to evaluate the quality of the cluster [36]. The larger value of CH or S, the better quality of the clustering results. Second, the labeled data are input into the predictive model (LSTM network). Finally, the weights of the predictive model are updated to reduce the loss based on our quadruplet loss function.

We set  $X = \{(x_i, l_i)\}_{i=1}^N$  as the input data, where  $l_i$  represents  $T_s$  labeled as “ $i$ ” and  $x_i$  represents the labeled environmental factors. Assume  $C$  is the total number of labels, where  $l_i \in [1, 2, 3, \dots, C]$ . Then, we project an instance  $x_i$  onto the estimate  $T_s$  by  $f_{\text{LSTM}}(\cdot; \theta): R^d \rightarrow S^1$ , where  $f_{\text{LSTM}}$  is an LSTM network parameterized by  $\theta$ . Let  $\{X_i^c\}_{i=1}^{N_c}$  be the environmental factors in the  $i$ -th labeled samples.  $N_c$  represents the total number of samples. We evaluate the similarity between samples through cluster analysis and expect the output of the model closer to the true value.

**2.4.1. Hard Sample Mining.** Hard sample mining generally refers to hard negative mining. Adding negative sample sets to participate in model training can improve the

effectiveness of learning and training and mine hard negatives as much as possible [37, 38]. For each fixed picture, the farthest sample picture and the nearest negative sample picture in a training batch are applied to train the network to enhance the generalization ability of the network, so that the network can learn better representations.

Inspired by TriHard loss, we first define  $x_i^c$  as the test sample:  $P_{c,i} = \{x_j^c | j \neq i\}$ ,  $|P_{c,i}| = N_c - 1$  is a collection, which includes the samples with the same label;  $N_{c,i} = \{x_j^k | k \neq c\}$ ,  $|N_{c,i}| = \sum_{k \neq c} N_k$  represents the other samples' collection.  $(x_i^c, y_i^c, P_{c,i}^*, N_{c,i}^*)$  is the quadruplet data set we defined.  $P_{c,i}^*$  is the positive set;  $N_{c,i}^*$  represents the negative set,  $|P_{c,i}^*|$  and  $|N_{c,i}^*|$  represent positive and negative sample pairs, and these tuples form the training sample pairs. The query sample is represented as  $x_i^c$ ; when  $S_{ij}^+$  satisfies the formula (3),  $\{x_i, x_j\}$  is the pair that we need.

$$S_{ij}^+ > \min_{x_k \in P_{c,i}} S_{ik} + \mu, \quad (3)$$

where  $S_{ij}^+ \langle f_{\text{LSTM}}(x_i; \theta), f_{\text{LSTM}}(x_j; \theta) \rangle$  represents the similarity between two samples, where  $\langle \cdot, \cdot \rangle$  represents the calculation of an  $n \times n$  similarity matrix.  $S_{ij}$  is the element in  $S$  at  $(x_i, x_j)$ , and  $\mu$  as a hyperparameter impacts the quadruplet that can control the number of hard positive samples. The condition for selecting a hard and negative pair is the same as above:

$$S_{ij}^- < \max_{x_k \in N_{c,i}} S_{ik} - \mu. \quad (4)$$

**2.4.2. Optimization Objective.** For each test sample  $x_i^c$ , we use the margin  $m$  to make it as close to the positive set  $P_{c,i}$  as possible, and as far away from the negative set  $N_{c,i}$  as possible. All the nontrivial positive points in  $P_{c,i}$  are pulled together by minimizing:

$$L_p(x_i^c; f_{\text{LSTM}}) = \frac{1}{2} \sum_{x_j^c \in P_{c,i}} \left( \|f_{\text{LSTM}}(x_i^c) - f_{\text{LSTM}}(x_j^c)\| - (r - m) \right)_+^2, \quad (5)$$

where  $f_{\text{LSTM}}(x_i^c)$  and  $f_{\text{LSTM}}(x_j^c)$  denote the estimated  $T_s$  of samples  $x_i^c$  and  $x_j^c$ , respectively, and  $\|f_{\text{LSTM}}(x_i^c) - f_{\text{LSTM}}(x_j^c)\|$  is the Euclidean distance between  $f_{\text{LSTM}}(x_i^c)$  and  $f_{\text{LSTM}}(x_j^c)$ . Similarly, all nontrivial negative points in  $N_{c,i}$  need to push out of the boundary  $\tau$ , by minimizing:

$$L_N(x_i^c; f_{\text{LSTM}}) = \frac{1}{2} \sum_{x_j^k \in N_{c,i}^*} \left( \tau - \|f_{\text{LSTM}}(x_i^c) - f_{\text{LSTM}}(x_j^k)\| \right)_+^2. \quad (6)$$

Meanwhile, we applied the squared-error loss function to the LSTM model for  $T_s$  estimation, as follows:

$$L_{\text{MSE}}(x_i^c; f_{\text{LSTM}}) = \frac{1}{2} \sum_{x_i^c \in N_c} (f_{\text{LSTM}}(x_i^c) - y_i^c)^2. \quad (7)$$

In the QL-LSTM, three minimization objectives were put into the model, and they are optimized at the same time:

$$L_{\text{QL-LSTM}}(x_i^c; f_{\text{LSTM}}) = L_p(x_i^c; f_{\text{LSTM}}) + L_N(x_i^c; f_{\text{LSTM}}) + L_{\text{MSE}}(x_i^c; f_{\text{LSTM}}), \quad (8)$$

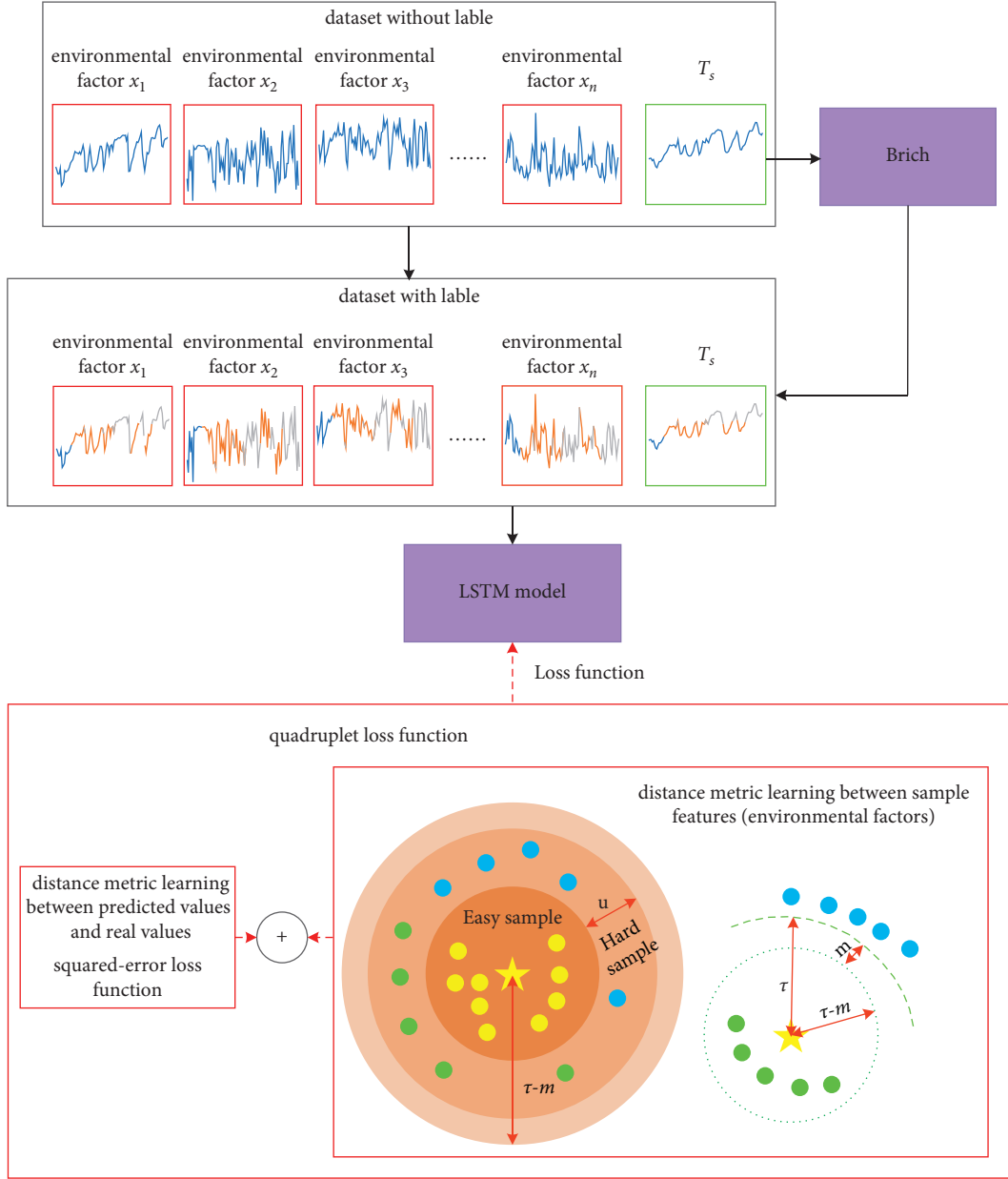


FIGURE 5: The framework of QL-LSTM.

We incorporate stochastic gradient descent and minibatch into the QL-LSTM to optimize the estimation model.

$x_i^c$  is a sample of the minibatch, which is obtained by sampling the labels of the training samples randomly, and serves as an anchor. We represent the QL-LSTM of each minibatch as

$$L_{QL-LSTM}(X; f) = \frac{1}{N} \sum_{\forall c \forall i} L_{QL-LSTM}(x_i^c; f), \quad (9)$$

where  $N$  denotes the batch size. Figure 6 represented the learning procedure of our QL-LSTM model.

**2.5. Model Training and Testing.** The input of our model is the corresponding meteorological data ( $T_a$ ,  $W$ ,  $A_p$ ,  $R_s$ ,  $VPD$ , and  $T_s$ ) from Laegern and Fluehli stations in Switzerland. And we downloaded the data at <https://fluxnet.fluxdata.org/> on FLUXNET with a total of 3,287 patterns from 2006 to 2014. Training datasets had 2465 patterns, and the rest as testing datasets.

Comparing our QL-LSTM model with the other advanced methods (SVR, BPNN, ELM, and LSTM), meanwhile, we calculate several evaluation criteria to analyze the model performance, including model fitting degree and the accuracy of the estimation model, as follows:

$$\begin{aligned}
\text{RMSE} &= \sqrt{\frac{\sum_{n=1}^N (y_i - \hat{y}_i)^2}{N}}, \\
\text{WI} &= 1 - \frac{\sum_{n=1}^N (y_i - \hat{y}_i)^2}{\sum_{n=1}^N (|\hat{y}_i - \bar{y}| + |y_i - \bar{y}|)^2}, \\
\text{NS} &= 1 - \frac{\sum_{n=1}^N (y_i - \hat{y}_i)^2}{\sum_{n=1}^N (y_i - \bar{y})^2}, \\
\text{MAE} &= \frac{\sum_{n=1}^N |y_i - \hat{y}_i|}{N}, \\
\text{LMI} &= 1 - \frac{\sum_{n=1}^N |y_i - \hat{y}_i|}{\sum_{n=1}^N |y_i - \bar{y}|},
\end{aligned} \tag{10}$$

where  $N$  is the number of the whole data,  $y_i$  denotes the observed value,  $\hat{y}_i$  is the predicted value, and  $\bar{y}$  is the average of the true values.

**2.6. Experiments.** The data within half an hour is obtained from two meteorological stations in an ecological nature reserve, located in Switzerland, namely, Legern and Fluley. The corresponding meteorological data ( $T_a$ ,  $W$ ,  $A_p$ ,  $R_s$ ,  $VPD$ , and  $T_s$ ) and past  $T_s$  data were input into the models. Meanwhile, the input variables are normalized to eliminate the dimensional influence between indicators. And the formula is as follows:

$$x_{\text{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \tag{11}$$

where the minimum value of the sample data is represented by  $x_{\min}$ , and the maximum value is represented by  $x_{\max}$ . Moreover, we have conducted research on the influence of the surrounding environmental factors on the model prediction. And we found the value of  $R_s$  in the data of the two stations is low, which is close to the normal distribution.

We conducted a statistical analysis of the data from the two stations. Table 1 listed the details of variables (minimum value ( $x_{\min}$ ), maximum value ( $x_{\max}$ ), average value ( $x_{\text{mean}}$ ), standard deviation ( $z_{sd}$ ), skewness ( $z_s$ ), and variation coefficient ( $z_v$ )). We used the daily data to verify the performance of the model with every half an hour data. The results showed in Table 1 that  $A_p$  had the highest negative skewness and presents a normal distribution at 5 cm depth, which presented similar characteristics in both stations. Meanwhile,  $z_v$  showed the biggest difference between the two stations.  $T_s$  at the 5 cm, 10 cm, and 15 cm depths range  $-1.888$ – $26.876^\circ\text{C}$ ,  $-0.181$ – $22.193^\circ\text{C}$ , and  $0.16$ – $20.826^\circ\text{C}$ , respectively. In summary, results showed that the values of  $z_{sd}$ ,  $z_s$ , and  $z_v$  change very slightly.

### 3. Results and Discussion

For testing the superiority of our QL-LSTM model performance for  $T_s$  estimation using scikit-learn, we compared our test results with those of other advanced models (SVR, BPNN, ELM, and LSTM).

We choose default parameters for the SVR model. For the BPNN model, the square error is used as the loss function, and the optimization is Adam. The number of samples selected for the model is 400, the iteration is set to 500, the learning rate is set to  $5.0\text{e-}4$ , and the size of the nodes is set to 128. The elm function was used to model the ELM model, the sigmoid served to activate the function in the hidden layer, and we set the same size of the nodes to BPNN. Furthermore, we set the hyperparameters of the LSTM to be the same as that of QL-LSTM. As can be seen from Table 2 and 3, the different values of the hyperparameter can generate the different predictive results. When the number of samples selected for the model is set to 400, the iteration to 500, the  $\text{num}_{\text{QL-LSTM}}$  to 128, and set the learning rate to  $1.0\text{e-}3$ , the QL-LSTM model has the best performance.

**3.1. Evaluation for the Hyperparameters in Quadruplet Loss Function.** The quadruplet loss function has five main hyperparameters, which are the total number of labels  $C$ , hyperparameter  $\mu$  in equations (3) and (4), and  $\tau$  and  $m$  in equations (5) and (6). When we evaluate the above hyperparameters in the quadruplet loss function, we set the parameters  $\text{num}_{\text{QL-LSTM}}$  to 128, the learning rate to  $1.0\text{e-}3$ , the iteration time to 500, and the batch size to 400. We first select the best  $C$  based on the Calinski Harabasz Score and  $Y_{\text{Silhouette\_score}}$ . Figure 7 denotes the Calinski Harabasz Score and  $Y_{\text{Silhouette\_score}}$  with different numbers of labels. It is observed that both scores achieve the best result when  $C$  is 25. Then, we gradually tune the hyperparameters,  $\tau$  and  $m$ . Figure 8 represented the results of the estimation model with different  $\mu$ ,  $\tau$ , and  $m$  in Laegern meteorological station. We can see that when we set  $\mu$  to be  $5.0\text{e-}3$ ,  $\tau$  to be  $1.0\text{e-}3$ , and  $m$  to be  $5.0\text{e-}5$ , and our QL-LSTM model could achieve the best estimation performance ( $\text{RMSE} = 0.789$ ,  $\text{MAE} = 0.605$ ,  $\text{NS} = 0.977$ ,  $\text{WI} = 0.994$ , and  $\text{LMI} = 0.865$ ). It is probably because the smaller hyperparameters we set, the less hard samples would be computed. Meanwhile, when we set the larger hyperparameters, the more redundant samples would be computed.

**3.2. The Impact of Different Inputs on the Performance of the Predictive Model.** In this part, we analyzed the environmental factors that may affect our QL-LSTM model for  $T_s$  estimation. Considering that the interaction between different environmental factors would have an impact on the  $T_s$  estimation, we combine the meteorological variables accordingly and input them into the submodels we set as follows:



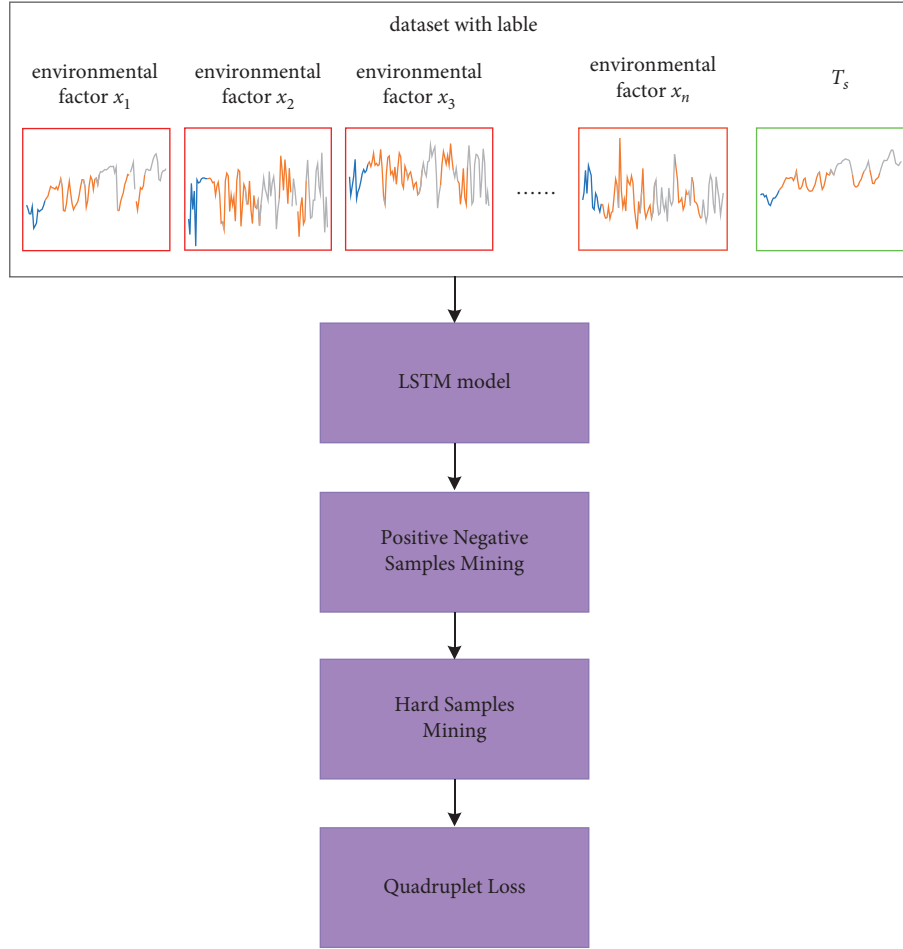


FIGURE 6: The learning procedure of QL-LSTM.

TABLE 1: Statistical results of the applied data for Laegern and Fluehli stations.

Station	Variable	$x_{\min}$	$x_{\max}$	$x_{\text{mean}}$	$z_{s\ d}$	$z_s$	$z_v$
Laegern	$T_a$ (°C)	-14.509	23.646	7.857	7.084	-0.118	0.901
	$R_s$ (W/m <sup>2</sup> )	175.545	379.458	305.081	35.164	-0.418	0.115
	VPD (hpa)	0.54	15.937	3.271	2.334	1.424	0.713
	$W$ (m/s)	0.668	8.025	2.237	1.005	1.411	0.449
	$A_p$ (kpa)	89.876	95.163	93.237	0.714	-0.630	0.007
	$T_s$ -5 cm (°C)	-1.888	26.876	10.104	6.061	0.103	0.599
	$T_s$ -10 cm (°C)	-0.181	22.193	9.726	5.435	-0.031	0.558
	$T_s$ -15 cm (°C)	0.16	19.394	9.010	5.025	-0.068	0.557
Fluehli	$T_a$ (°C)	-14.448	22.877	7.708	6.906	-0.100	0.895
	$R_s$ (W/m <sup>2</sup> )	194.734	377.444	306.015	32.805	-0.327	0.107
	VPD (hpa)	0.416	9.662	2.129	1.543	1.410	0.724
	$W$ (m/s)	0.342	4.636	1.476	0.619	0.894	0.419
	$A_p$ (kpa)	82.308	87.164	85.493	0.711	-0.843	0.008
	$T_s$ -5 cm (°C)	-0.35	21.822	8.729	6.338	0.075	0.726
	$T_s$ -10 cm (°C)	-0.044	21.727	8.836	6.242	0.071	0.706
	$T_s$ -15 cm (°C)	0.432	20.826	8.813	6.023	0.062	0.683



TABLE 2: Predictive performance with different num<sub>QL-LSTM</sub> and learning rates at Laegern station.

Learning rate	num <sub>QL-LSTM</sub>	RMSE	MAE	NS	WI	LMI
1.0e-5	16	4.735	4.059	0.1799	0.325	0.093
	32	2.970	2.508	0.677	0.845	0.439
	64	1.346	1.061	0.933	0.980	0.763
	128	1.365	1.073	0.931	0.982	0.760
	256	1.341	1.049	0.934	0.983	0.765
	16	1.230	0.962	0.944	0.986	0.785
1.0e-4	32	1.175	0.915	0.949	0.987	0.795
	64	1.167	0.903	0.950	0.987	0.798
	128	1.159	0.884	0.950	0.987	0.802
	256	1.148	0.871	0.951	0.987	0.805
	16	1.129	0.855	0.953	0.988	0.808
5.0e-4	32	1.076	0.809	0.957	0.989	0.819
	64	1.001	0.757	0.963	0.990	0.830
	128	0.817	0.629	0.975	0.993	0.859
	256	0.852	0.657	0.973	0.993	0.853
	16	1.069	0.802	0.958	0.989	0.820
1.0e-3	32	0.950	0.716	0.966	0.991	0.839
	64	0.880	0.676	0.971	0.992	0.848
	<b>128</b>	<b>0.817</b>	<b>0.625</b>	<b>0.975</b>	<b>0.993</b>	<b>0.860</b>
	256	0.825	0.634	0.975	0.993	0.858
	16	0.871	0.675	0.972	0.993	0.849
5.0e-3	32	0.858	0.661	0.973	0.993	0.852
	64	0.868	0.661	0.972	0.993	0.852
	128	0.852	0.648	0.973	0.993	0.855
	256	0.854	0.649	0.973	0.993	0.855

TABLE 3: Predictive performance with different numbers of batch size and iterations at Laegern station.

Batch	Iteration	RMSE	MAE	NS	WI	LMI
100	100	0.860	0.666	0.972	0.993	0.851
	200	0.873	0.676	0.972	0.993	0.848
	500	0.880	0.667	0.971	0.992	0.850
	800	1.073	0.825	0.957	0.989	0.815
	100	0.849	0.654	0.973	0.993	0.853
200	200	0.817	0.625	0.975	0.993	0.860
	500	0.843	0.642	0.973	0.993	0.856
	800	0.916	0.692	0.969	0.992	0.845
	100	0.973	0.739	0.965	0.991	0.834
300	200	0.831	0.644	0.974	0.993	0.856
	500	0.817	0.626	0.975	0.993	0.860
	800	0.869	0.664	0.972	0.993	0.851
	100	1.038	0.784	0.960	0.990	0.824
400	200	0.851	0.654	0.973	0.993	0.853
	<b>500</b>	<b>0.809</b>	<b>0.622</b>	<b>0.976</b>	<b>0.993</b>	<b>0.860</b>
	800	0.817	0.628	0.975	0.993	0.859
	100	1.076	0.812	0.957	0.989	0.818
500	200	0.868	0.666	0.972	0.993	0.851
	500	0.809	0.623	0.976	0.993	0.860
	800	0.811	0.626	0.975	0.993	0.860

(i) Input  $I_1$ :  $T_a(d-1)$ (ii) Input  $I_2$ :  $T_a(d-1) + R_s(d-1)$ (iii) Input  $I_3$ :  $T_a(d-1) + R_s(d-1) + VPD(d-1)$ (iv) Input  $I_4$ :  $T_a(d-1) + R_s(d-1) + VPD(d-1) + W(d-1)$ (v) Input  $I_5$ :  $T_a(d-1) + R_s(d-1) + VPD(d-1) + W(d-1) + A_p(d-1)$ (vi) Output:  $T_s(d)$ 

Then, we consider that the past  $T_s$  will continue to have an impact on the future  $T_s$  estimation, so we have carried out lag processing for the past  $T_s$  on different days, as follows:

(i) Input  $I_6$ :  $T_s(d-1)$ (ii) Input  $I_7$ :  $T_s(d-1) + T_s(d-2)$

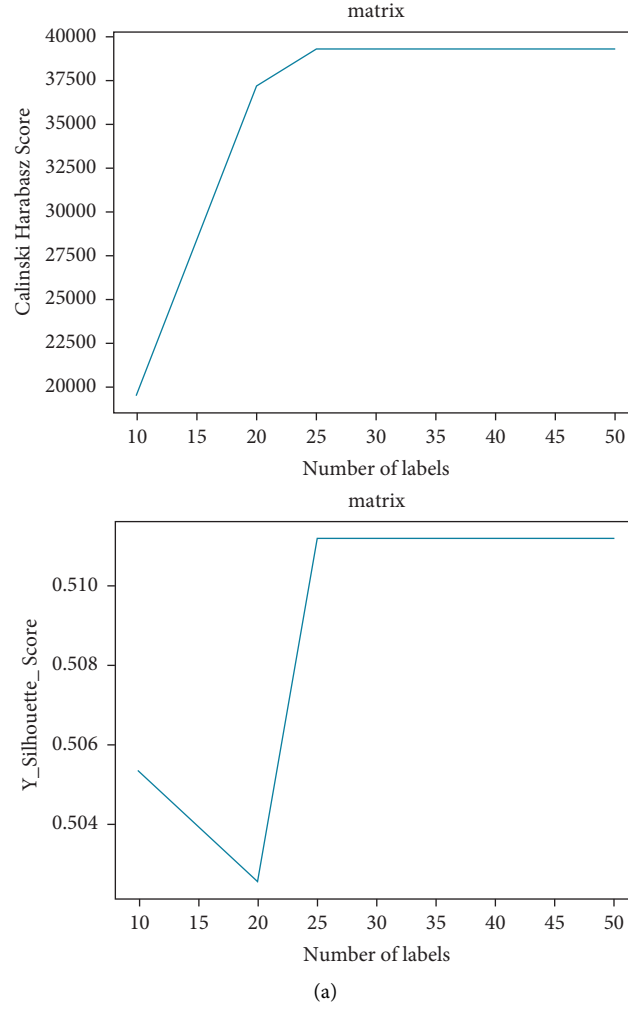


FIGURE 7: The Calinski Harabasz Score and Y\_Silhouette\_score with different numbers of labels.

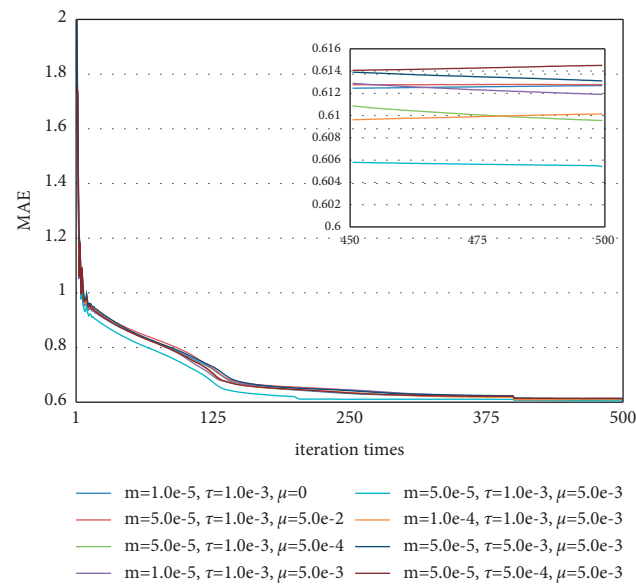


FIGURE 8: The estimation results with different  $\mu$ ,  $\tau$ , and  $m$  at Laegern meteorological station.

TABLE 4: Predictive performance of QL-LSTM at 5 cm depth for the Laegern station.

Method	RMSE	MAE	NS	WI	LMI
QL-LSTM <sub>(I1)</sub>	1.469	1.143	0.921	0.980	0.744
QL-LSTM <sub>(I2)</sub>	1.444	1.113	0.923	0.981	0.751
QL-LSTM <sub>(I3)</sub>	<b>1.221</b>	<b>0.937</b>	<b>0.945</b>	<b>0.986</b>	<b>0.790</b>
QL-LSTM <sub>(I4)</sub>	1.396	1.078	0.928	0.982	0.759
QL-LSTM <sub>(I5)</sub>	1.454	1.143	0.922	0.980	0.744
QL-LSTM <sub>(I6)</sub>	1.143	0.866	0.952	0.987	0.806
QL-LSTM <sub>(I7)</sub>	1.095	0.823	0.956	0.988	0.815
QL-LSTM <sub>(I8)</sub>	<b>1.077</b>	<b>0.815</b>	<b>0.957</b>	<b>0.989</b>	<b>0.817</b>
QL-LSTM <sub>(I9)</sub>	1.096	0.834	0.956	0.988	0.813
QL-LSTM <sub>(I10)</sub>	1.115	0.842	0.954	0.988	0.811

(iii) Input  $I_8$ :  $T_s(d-1) + T_s(d-2) + T_s(d-3)$

(iv) Input  $I_9$ :  $T_s(d-1) + T_s(d-2) + T_s(d-3) + T_s(d-4)$

(v) Input  $I_{10}$ :  $T_s(d-1) + T_s(d-2) + T_s(d-3) + T_s(d-4) + T_s(d-5)$

(vi) Output:  $T_s(d)$

We input what we specified above into QL-LSTM to predict the  $T_s(d)$  at the 5 cm depth of the Laegern station. For our model, we first selected the hyperparameters  $\mu$  as  $5.0\text{e-}3$ ,  $\tau$  as  $1.0\text{e-}3$ ,  $m$  as  $5.0\text{e-}5$ ,  $C$  as 25,  $\text{num}_{\text{QL-LSTM}}$  as 128, learning rate as  $1.0\text{e-}3$ , iteration time as 500, and batch size as 400, and the results are presented in Table 4. Obviously, the methods of QL-LSTM<sub>(I3)</sub> and QL-LSTM<sub>(I8)</sub> are better than the others, respectively. Meanwhile, we could conclude that  $W(d-1)$ ,  $A_p(d-1)$ ,  $T_s(d-4)$ , and  $T_s(d-5)$  all have an influence on the performance of the predictive model. In addition, by comparing the estimation results between meteorological variables input and past  $T_s$  input, we found that our model with past  $T_s$  could achieve greater accuracy in modeling than the one with meteorological variables. The reason may be that the predictive model with past  $T_s$  input has stronger memory for  $T_s$  variables. The estimation of the future  $T_s$  should make the best use of its continuity; in this way, we can make a reliable  $T_s$  estimation, which not only continues its historical tendency but also conforms to its actual performance. Hence, we construct the predictive model (QL-LSTM<sub>(I11)</sub>) by combining the environmental factors ( $T_a(d-1)$ ,  $R_s(d-1)$ ,  $VPD(d-1)$ ) with past  $T_s$  ( $T_s(d-1)$ ,  $T_s(d-2)$ ,  $T_s(d-3)$ ), which is also considered in estimating the  $T_s(d)$  at the 5 cm depth of the Laegern station. Experiment results prove that it could achieve the best estimation performance (RMSE = 0.789, LMI = 0.865, WI = 0.994, NS = 0.977, and MAE = 0.605). Hence, the final input for the predictive models is the environmental factors ( $T_a(d-1)$ ,  $R_s(d-1)$ ,  $VPD(d-1)$ ) and the past  $T_s$  ( $T_s(d-1)$ ,  $T_s(d-2)$ ,  $T_s(d-3)$ ).

The three methods (QL-LSTM<sub>(I3)</sub>, QL-LSTM<sub>(I8)</sub>, and QL-LSTM<sub>(I11)</sub>) are used to test the data of the Laegern station. Figure 9 shows the linear relationship between the predicted value and the observed value. The QL-LSTM<sub>(I11)</sub> model gets the best predictive performance with  $y = 0.9899x + 0.3022$  and the higher  $R^2$  (0.9792) compared with the others. In the frequency diagram (Figure 10) of the models (QL-LSTM<sub>(I3)</sub>, QL-LSTM<sub>(I8)</sub>, and QL-LSTM<sub>(I11)</sub>), the QL-LSTM<sub>(I11)</sub> also has a higher frequency (91%) compared to the others. Therefore, we can draw a conclusion

that the predictive model (a combination of the environmental factors and the past  $T_s$ ) normally outperformed the other two (by either past measurements of  $T_s$  or other meteorological information) in the  $T_s$  estimation.

**3.3. Comparison with Different Models.** In this part, our QL-LSTM model was compared with several advanced models, including SVR, BPNN, ELM, and LSTM. The data of  $T_a$ ,  $R_s$ , and  $VPD$  on day “d-1”, and  $T_s$  data on different days were acted as input data to different predictive models, and the output was the predicted value of  $T_s$  on days “d”, “d+5”, and “d+15”. Time steps were in days.

The testing results of five different models at 5, 10, and 15 cm depth on the 1st, 5th, and 15th days of the Laegern station were shown in Table 5. And we can see that our QL-LSTM model performs better than the existing advanced models at the 5 cm depth on the 1st day. Specifically, the value of RMSE is 0.789, which is reduced relative to 13% (LSTM), 22% (ELM), 28% (BPNN), and 22% (SVR), respectively. The MAE values amount to 0.605 (QL-LSTM), and the others are 0.813 (SVR), 0.872 (BPNN), 0.824 (ELM), and 0.821 (LSTM). Meanwhile, the QL-LSTM model achieved a higher value of NS, WI, and LMI. Hence, it is obvious that our model had the best performance in this case. For the results of 5 cm depth on the 15th day, the LSTM achieved a higher WI (0.892) than estimation from other models on the 15th day, but it is similar to the WI (0.891) of our model. For 10 and 15 cm depth results on the 1st, 5th, and 15th days, the performance of our QL-LSTM model remains stable, although our QL-LSTM model has the lower values of WI (0.952 and 0.933) than the values of WI (0.954 and 0.934) on the LSTM model in individual cases. It can be found that the predictive performance will get better as the soil depth decreases (from 5 cm to 15 cm), but it will decrease as time goes on (from 1st to 15th days). The systematic errors caused this phenomenon for long-term estimation [39].

The same strategy was applied in the Fluehli station to further verify the performance of the models, with the results shown in Table 6. And our QL-LSTM model performs better compared with others. However, for 5 cm depth on the 15th day and 15 cm depth on the 5th day, the BPNN model performs better with the results of RMSE = 2.081, LMI = 0.584, WI = 0.937, NS = 0.769, and MAE = 2.099, and RMSE = 1.832, LMI = 0.726, WI = 0.973, NS = 0.897, and MAE = 1.352 and LMI = 0.726. Our method does not

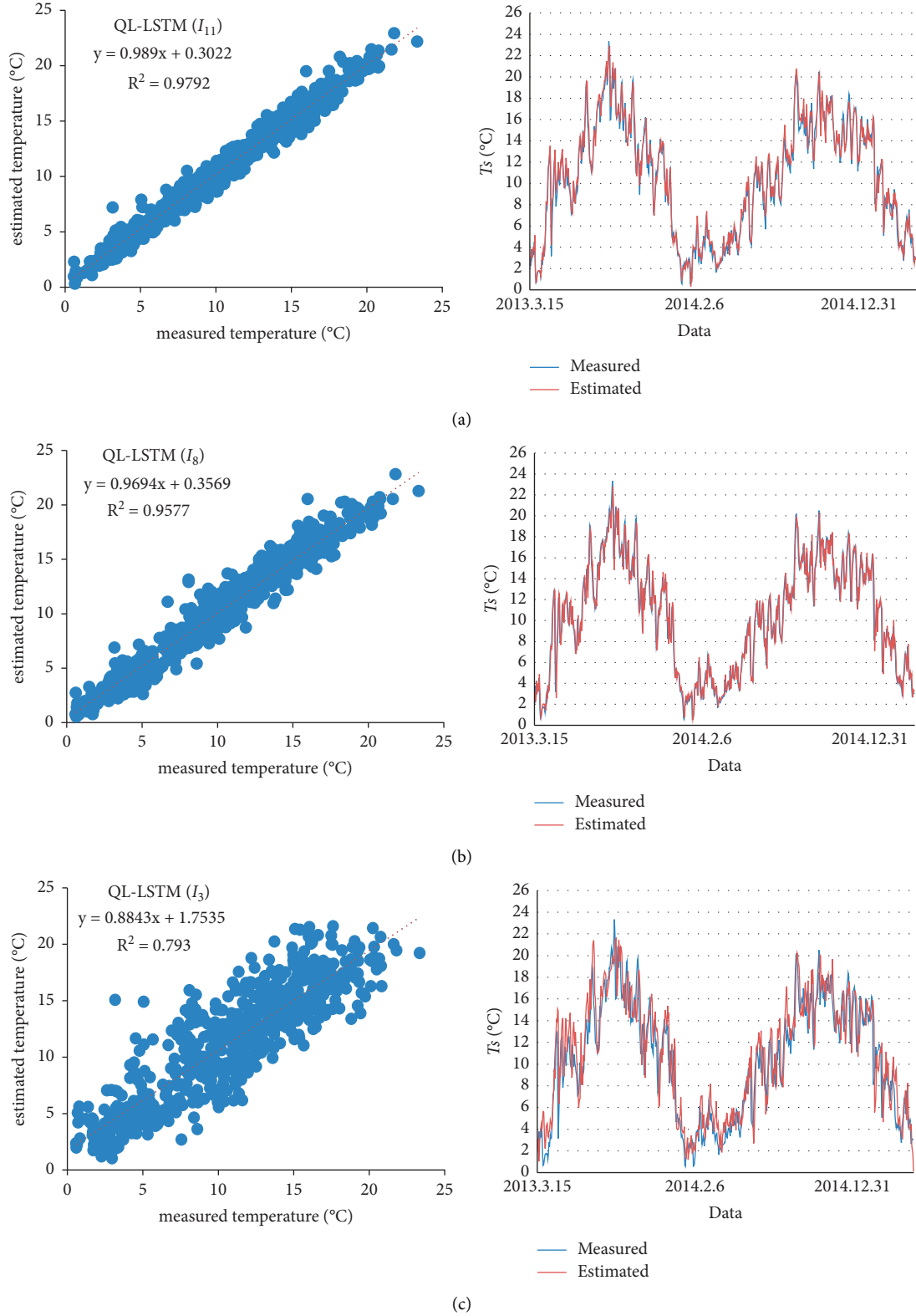


FIGURE 9: The scatterplots of the predictive model testing results (the values of estimated and observed) for the Laegern station. (a) QL-LSTM( $I_{11}$ ), (b) QL-LSTM( $I_8$ ) model, and (c) QL-LSTM( $I_3$ ) model.

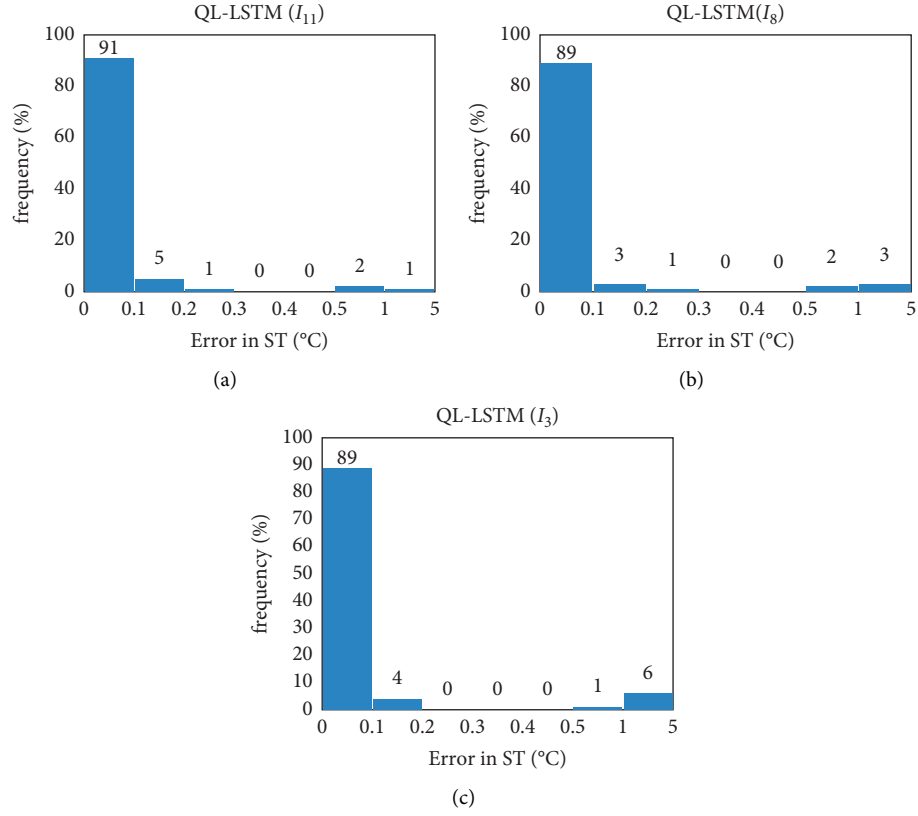


FIGURE 10: The frequency plot of the predictive models (absolute estimation error) for the Laegern station. (a) QL-LSTM( $I_{11}$ ), (b) QL-LSTM( $I_8$ ) model, and (c) QL-LSTM( $I_3$ ) model.

TABLE 5: The predictive performance with different models at the Laegern station.

Depth (cm)	Day	Method	RMSE	MAE	NS	WI	LMI
5	d	SVR	1.007	0.813	-1315.5	0.412	-882.6
		BPNN	1.101	0.872	0.930	0.982	0.838
		ELM	1.016	0.824	0.925	0.983	0.849
		LSTM	0.910	0.821	0.926	0.983	0.850
		QL-LSTM	<b>0.789</b>	<b>0.605</b>	<b>0.977</b>	<b>0.994</b>	<b>0.865</b>
	d + 5	SVR	2.665	2.133	-1205.3	0.419	-845.1
		BPNN	2.812	2.230	0.700	0.919	0.535
		ELM	2.832	2.261	0.693	0.916	0.528
		LSTM	2.643	2.113	0.730	0.926	0.560
		QL-LSTM	<b>2.436</b>	<b>1.908</b>	<b>0.782</b>	<b>0.937</b>	<b>0.573</b>
	d + 15	SVR	3.322	2.680	-1215.2	0.422	-846.9
		BPNN	3.271	2.657	0.601	0.886	0.438
		ELM	3.403	2.723	0.572	0.881	0.424
		LSTM	3.278	2.652	0.601	<b>0.892</b>	0.439
		QL-LSTM	<b>3.078</b>	<b>2.428</b>	<b>0.651</b>	0.891	<b>0.455</b>

TABLE 5: Continued.

Depth (cm)	Day	Method	RMSE	MAE	NS	WI	LMI
10	d	SVR	1.011	0.850	-1291.5	0.414	-870.2
		BPNN	0.986	0.831	0.923	0.983	0.837
		ELM	1.090	0.862	0.916	0.981	0.830
		LSTM	0.980	0.824	0.923	0.983	0.839
		QL-LSTM	<b>0.761</b>	<b>0.605</b>	<b>0.975</b>	<b>0.993</b>	<b>0.854</b>
	d + 5	SVR	2.173	1.752	-1211.9	0.417	-842.1
		BPNN	2.188	1.751	0.774	0.935	0.608
		ELM	2.291	1.833	0.762	0.938	0.596
		LSTM	2.181	1.745	0.781	<b>0.954</b>	0.615
		QL-LSTM	<b>1.973</b>	<b>1.545</b>	<b>0.833</b>	0.952	<b>0.628</b>
	d + 15	SVR	2.794	2.220	-1244.5	0.431	-850.8
		BPNN	2.831	2.263	0.652	0.909	0.493
		ELM	2.920	2.300	0.632	0.899	0.484
		LSTM	2.763	2.131	0.667	0.907	0.509
		QL-LSTM	<b>2.538</b>	<b>1.975</b>	<b>0.724</b>	<b>0.917</b>	<b>0.524</b>
15	d	SVR	0.531	0.450	-1304.5	0.424	-873.5
		BPNN	0.525	0.442	0.942	0.987	0.921
		ELM	0.637	0.488	0.940	0.987	0.913
		LSTM	0.512	0.436	0.944	0.988	0.927
		QL-LSTM	<b>0.312</b>	<b>0.239</b>	<b>0.995</b>	<b>0.998</b>	<b>0.938</b>
	d + 5	SVR	1.761	1.400	-1262.3	0.426	-857.9
		BPNN	1.773	1.441	0.826	0.957	0.667
		ELM	1.921	1.532	0.800	0.950	0.643
		LSTM	1.742	1.408	0.831	0.958	0.675
		QL-LSTM	<b>1.533</b>	<b>1.203</b>	<b>0.882</b>	<b>0.968</b>	<b>0.687</b>
	d + 15	SVR	2.396	1.937	-1266.3	0.431	-857.4
		BPNN	2.406	1.926	0.707	0.933	0.537
		ELM	2.508	2.008	0.674	0.925	0.518
		LSTM	2.401	1.918	0.707	<b>0.934</b>	0.539
		QL-LSTM	<b>2.189</b>	<b>1.719</b>	<b>0.760</b>	<b>0.933</b>	<b>0.553</b>

TABLE 6: The predictive performance with different models at the Fluehli station.

Depth (cm)	Day	Method	RMSE	MAE	NS	WI	LMI
5	d	SVR	0.691	0.549	-1311.5	0.429	-868.5
		BPNN	0.718	0.550	0.942	0.918	0.916
		ELM	0.693	0.538	0.942	0.988	0.918
		LSTM	0.723	0.549	0.941	0.987	0.916
		QL-LSTM	<b>0.492</b>	<b>0.352</b>	<b>0.992</b>	<b>0.998</b>	<b>0.930</b>
	d + 5	SVR	2.316	1.745	-1266.6	0.431	-851.7
		BPNN	2.297	1.716	0.820	0.955	0.687
		ELM	2.463	1.880	0.799	0.949	0.654
		LSTM	2.291	1.718	0.821	0.956	0.686
		QL-LSTM	<b>2.084</b>	<b>1.526</b>	<b>0.872</b>	<b>0.966</b>	<b>0.698</b>
	d + 15	SVR	3.023	2.316	-1257.5	0.433	-847.4
		BPNN	<b>2.801</b>	<b>2.099</b>	<b>0.769</b>	<b>0.937</b>	<b>0.584</b>
		ELM	3.143	2.400	0.694	0.918	0.534
		LSTM	2.971	2.353	0.723	0.922	0.557
		QL-LSTM	2.815	2.165	0.766	0.933	0.570

TABLE 6: Continued.

Depth (cm)	Day	Method	RMSE	MAE	NS	WI	LMI
10	d	SVR	0.628	0.518	-1310.1	0.428	-870.4
		BPNN	0.635	0.508	0.944	0.988	0.926
		ELM	0.621	0.510	0.944	0.988	0.926
		LSTM	0.610	0.500	0.944	0.988	0.928
		QL-LSTM	<b>0.381</b>	<b>0.286</b>	<b>0.995</b>	<b>0.998</b>	<b>0.941</b>
	d + 5	SVR	2.043	1.529	-1274.5	0.430	-856.9
		BPNN	2.021	1.510	0.846	0.963	0.721
		ELM	2.153	1.600	0.830	0.958	0.697
		LSTM	<b>1.777</b>	1.506	<b>0.901</b>	<b>0.973</b>	<b>0.732</b>
		QL-LSTM	1.794	<b>1.307</b>	0.899	0.973	0.732
	d + 15	SVR	2.813	2.143	-1271.7	0.432	-854.43
		BPNN	2.792	2.112	0.738	0.933	0.594
		ELM	2.926	2.300	0.717	0.926	0.558
		LSTM	2.802	2.108	0.736	0.932	0.597
		QL-LSTM	<b>2.601</b>	<b>1.906</b>	<b>0.789</b>	<b>0.948</b>	<b>0.612</b>
15	d	SVR	0.631	0.506	-1291.9	0.430	-862.9
		BPNN	0.642	0.500	0.943	0.988	0.926
		ELM	<b>0.407</b>	0.468	0.944	0.988	0.930
		LSTM	0.618	0.465	0.944	0.988	0.931
		QL-LSTM	0.409	<b>0.290</b>	<b>0.994</b>	<b>0.998</b>	<b>0.941</b>
	d + 5	SVR	2.028	1.551	-1269.3	0.429	-853.7
		BPNN	<b>1.832</b>	<b>1.352</b>	<b>0.897</b>	<b>0.973</b>	<b>0.726</b>
		ELM	2.173	1.638	0.830	0.958	0.691
		LSTM	2.042	1.555	0.846	0.962	0.715
		QL-LSTM	1.835	1.354	0.896	0.972	0.725
	d + 15	SVR	2.771	2.156	-1257.7	0.433	-848.4
		BPNN	2.762	2.134	0.748	0.935	0.595
		ELM	2.802	2.200	0.741	0.931	0.579
		LSTM	2.795	2.112	0.742	0.933	0.597
		QL-LSTM	<b>2.552</b>	<b>1.931</b>	<b>0.805</b>	<b>0.949</b>	<b>0.608</b>

perform well in some cases probably because the weights of the LSTM model are randomly selected to generate the nonoptimal solution. Meanwhile, our novel loss function (quadruplet loss) is applied based on the LSTM model; it only improved estimation performance to a certain extent against the LSTM model. All in all, the results of our testing on the data of different regions show that the performance of our QL-LSTM model is usually better for  $T_s$  prediction with different depths and days.

#### 4. Conclusions

Soil temperature ( $T_s$ ) is a main physical variable of the land surface, which has an impact on many aspects, such as the growth and yield of crops. Therefore, how to predict  $T_s$  accurately is very important. This paper proposed the QL-LSTM model and compared it with the state-of-the-art predictive models to use the meteorological data and past  $T_s$  of the Laegern and Fluehli stations (Switzerland) for daily  $T_s$  estimation at 5, 10, and 15 cm depth on the 1st, 5th, and 15th days. The experiment results showed that the QL-LSTM model performed better than the existing advanced models for  $T_s$  estimation in multifarious cases.

In addition, to enrich processing information in loss function and further improve estimation performance, we attempt to design the novel quadruplet loss function that combines the traditional squared-error loss function with

distance metric learning between the sample features. Similar samples can be zoomed and the dissimilar samples can be pushed.

The distance metric learning between the sample features is combined with the squared-error loss function, which could improve the estimation performance to a certain extent. However, the many hyperparameters in our method may cause sensitivity issues in estimation, which may lead to poor generalization ability of other estimations. In the future, the parametric adaptive method will be explored for a new loss function in the follow-up study.

#### Data Availability

The experimental data are available without any restriction.

#### Conflicts of Interest

The authors declare that there are no conflicts of interest.

#### Acknowledgments

This work was sponsored by the National Natural Science Foundation of China (NSFC) under Grant nos. 51805203, 42105144, and U1811464, the Science and Technology Development Plan of Jilin Province (20190201023JC), and the Development and Reform Commission of Jilin Province (2019C054-2).



## References

- [1] A. Keshavarzi, F. Sarmadian, E. S. E. Omran, and M. Iqbal, "A neural network model for estimating soil phosphorus using terrain analysis," *The Egyptian Journal of Remote Sensing and Space Science*, vol. 18, no. 2, pp. 127–135, 2015.
- [2] L. Lai, X. Zhao, L. Jiang et al., "Soil respiration in different agricultural and natural ecosystems in an arid region," *PLoS One*, vol. 7, no. 10, Article ID e48011, 2012.
- [3] H. Tabari, P. Hosseinzadeh Talaee, and P. Willems, "Short-term forecasting of soil temperature using artificial neural network," *Meteorological Applications*, vol. 22, no. 3, pp. 576–585, 2014.
- [4] W. Wang, K. Akhtar, G. Ren, G. Yang, Y. Feng, and L. Yuan, "Impact of straw management on seasonal soil carbon dioxide emissions, soil water content, and temperature in a semi-arid region of China," *The Science of the Total Environment*, vol. 652, pp. 471–482, 2019.
- [5] B. d. O. Silva, M. R. Moitinho, G. A. d. A. Santos, D. D. B. Teixeira, C. Fernandes, and N. La Scala, "Soil CO<sub>2</sub> emission and short-term soil pore class distribution after tillage operations," *Soil and Tillage Research*, vol. 186, pp. 224–232, 2019.
- [6] S. Mehdizadeh, J. Behmanesh, and K. Khalili, "Evaluating the performance of artificial intelligence methods for estimation of monthly mean soil temperature without using meteorological data," *Environmental Earth Sciences*, vol. 76, no. 8, p. 325, 2017.
- [7] L. L. Liang, D. A. Riveros-Iregui, R. E. Emanuel, and B. L. McGlynn, "A simple framework to estimate distributed soil temperature from discrete air temperature measurements in data-scarce regions," *Journal of Geophysical Research: Atmospheres*, vol. 119, no. 2, pp. 407–417, 2014.
- [8] Y. Liu, A. H. Weerts, M. Clark et al., "Advancing data assimilation in operational hydrologic forecasting: progresses, challenges, and emerging opportunities," *Hydrology and Earth System Sciences*, vol. 16, no. 10, pp. 3863–3887, 2012.
- [9] O. Kisi, M. Tombul, and M. Z. Kermani, "Modeling soil temperatures at different depths by using three different neural computing techniques," *Theoretical and Applied Climatology*, vol. 121, no. 1–2, pp. 377–387, 2015.
- [10] H. Tabari, A.-A. Sabziparvar, and M. Ahmadi, "Comparison of artificial neural network and multivariate linear regression methods for estimation of daily soil temperature in an arid region," *Meteorology and Atmospheric Physics*, vol. 110, no. 3–4, pp. 135–142, 2011.
- [11] R. Moazenzadeh and B. Mohammadi, "Assessment of bio-inspired metaheuristic optimisation algorithms for estimating soil temperature," *Geoderma*, vol. 353, pp. 152–171, 2019.
- [12] M. Delbari, S. Sharifazari, and E. Mohammadi, "Modeling daily soil temperature over diverse climate conditions in Iran—a comparison of multiple linear regression and support vector regression techniques," *Theoretical and Applied Climatology*, vol. 135, no. 3–4, pp. 991–1001, 2019.
- [13] Y. Feng, N. Cui, W. Hao, L. Gao, and D. Gong, "Estimation of soil temperature from meteorological data using different machine learning models," *Geoderma*, vol. 338, pp. 67–77, 2019.
- [14] H. Sanikhani, R. C. Deo, Z. M. Yaseen, O. Eray, and O. Kisi, "Non-tuned data intelligent model for soil temperature estimation: a new approach," *Geoderma*, vol. 330, pp. 52–64, 2018.
- [15] M. Bilgili, "The use of artificial neural networks for forecasting the monthly mean soil temperatures in Adana, Turkey," *Turkish Journal of Agriculture and Forestry*, vol. 35, pp. 83–93, 2011.
- [16] O. Kisi, H. Sanikhani, and M. Cobaner, "Soil temperature modeling at different depths using neuro-fuzzy, neural network, and genetic programming techniques," *Theoretical and Applied Climatology*, vol. 129, no. 3–4, pp. 833–848, 2017.
- [17] M. Zeynoddin, H. Bonakdari, I. Ebtehaj, F. Esmaeilbeiki, B. Gharabaghi, and D. Zare Haghi, "A reliable linear stochastic daily soil temperature forecast model," *Soil and Tillage Research*, vol. 189, pp. 73–87, 2019.
- [18] S. Samadianfard, E. Asadi, S. Jarhan et al., "Wavelet neural networks and gene expression programming models to predict short-term soil temperature at different depths," *Soil and Tillage Research*, vol. 175, pp. 37–50, 2018.
- [19] S. Mehdizadeh, F. Fathian, M. J. S. Safari, and A. Khosravi, "Developing novel hybrid models for estimation of daily soil temperature at various depths," *Soil and Tillage Research*, vol. 197, Article ID 104513, 2020.
- [20] B. Nahvi, J. Habibi, K. Mohammadi, S. Shamshirband, and O. S. Al Razgan, "Using self-adaptive evolutionary algorithm to improve the performance of an extreme learning machine for estimating soil temperature," *Computers and Electronics in Agriculture*, vol. 124, pp. 150–160, 2016.
- [21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [22] J. Guo, Z. Xie, Y. Qin, L. Jia, and Y. Wang, "Short-term abnormal passenger flow prediction based on the fusion of SVR and LSTM," *IEEE Access*, vol. 7, pp. 42946–42955, 2019.
- [23] J. Zhang, Y. Zhu, X. Zhang, M. Ye, and J. Yang, "Developing a Long Short-Term Memory (LSTM) based model for predicting water table depth in agricultural areas," *Journal of Hydrology*, vol. 561, pp. 918–929, 2018.
- [24] X. Qing and Y. Niu, "Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM," *Energy*, vol. 148, pp. 461–468, 2018.
- [25] Q. Li, H. Hao, Y. Zhao et al., "GANs-LSTM model for soil temperature estimation from meteorological: a new approach," *IEEE Access*, vol. 8, pp. 59427–59443, 2020.
- [26] Y. Wang, D. Gan, M. Sun, N. Zhang, Z. Lu, and C. Kang, "Probabilistic individual load forecasting using pinball loss guided LSTM," *Applied Energy*, vol. 235, pp. 10–20, 2019.
- [27] F. Kratzert, D. Klotz, M. Herrnegger, A. K. Sampson, S. Hochreiter, and G. S. Nearing, "Toward improved predictions in ungauged basins: exploiting the power of machine learning," *Water Resources Research*, vol. 55, no. 12, pp. 11344–11354, 2019.
- [28] E. J. Ekpenyong and G. Sunday, "Bayes Estimator of the Shape Parameter of the Burr type XII with generalised squared error and Linex loss functions under some prior distributions," *International Journal of Statistics and Applied Mathematics*, vol. 3, no. 4, pp. 60–64, 2018.
- [29] M. Han, "E-Bayesian estimation and its E-MSE under the scaled squared error loss function, for exponential distribution as example," *Communications in Statistics—Simulation and Computation*, vol. 48, no. 6, pp. 1880–1890, 2019.
- [30] L. Fan, H. Zhao, and H. Zhao, "Distribution consistency loss for large-scale remote sensing image retrieval," *Remote Sensing*, vol. 12, no. 1, p. 175, 2020.
- [31] Q. Li, R. Xu, H. Zhao, L. Xu, X. Shan, and P. Gong, "Computer-aided diagnosis of mammographic masses using local geometric constraint image retrieval," *Optik*, vol. 171, pp. 757–767, 2018.
- [32] L. Fan, H. Zhao, H. Zhao, P. Liu, and H. Hu, "Image retrieval based on learning to rank and multiple loss," *ISPRS International Journal of Geo-Information*, vol. 8, no. 9, p. 393, 2019.

- [33] A. Hermans, L. Beyer, and B. Leibe, "In Defense of the triplet loss for person Re-identification. Computer vision and pattern recognition (CVPR)," 2017, <https://arxiv.org/abs/1703.07737>.
- [34] T. Zhang, R. Ramakrishnan, and M. Livny, "A new data clustering algorithm and its applications," *Data Mining and Knowledge Discovery*, vol. 1, no. 2, pp. 141–182, 1997.
- [35] H. Zhao, Q. Li, and P. Liu, "Hierarchical geometry verification via maximum entropy saliency in image retrieval," *Entropy*, vol. 16, no. 7, pp. 3848–3865, 2014.
- [36] R. Xu, J. Xu, and D. C. Wunsch, "A comparison study of validity indices on swarm-intelligence-based clustering," *IEEE Transactions on Systems Man & Cybernetics Part B*, vol. 42, no. 4, pp. 1243–1255, 2014.
- [37] F. Schro, D. Kalenichenko, and J. Philbin, "A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823, Boston, MA, USA, June 2015.
- [38] H. Liu, J. Cheng, and F. Wang, "Sequential subspace clustering via temporal smoothness for sequential data segmentation," *IEEE Transactions on Image Processing*, vol. 27, no. 2, pp. 866–878, 2018.
- [39] K. Judd and M. Small, "Towards long-term prediction," *Physica D: Nonlinear Phenomena*, vol. 136, no. 1-2, pp. 31–44, 2000.