Research Article

A New Embedded Estimation Model for Soil Temperature Prediction

Xuezhi Wang (1), Wenhui Li (1), and Qingliang Li (2,3)

1College of Computer Science and Technology, Jilin University, Changchun 130012, China
2Symbol Computation and Knowledge Engineer of Ministry of Education, Jilin University, Changchun, Jilin Province, China
3School of Computer Science and Technology, Changchun Normal University, Changchun 130032, China

Correspondence should be addressed to Qingliang Li; liqingliang@ccsfu.edu.cn

Received 5 June 2021; Accepted 8 July 2021; Published 17 July 2021

Academic Editor: Yi-Zhang Jiang

Copyright © 2021 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.

With the continuous development of Earth science, soil temperature has received more and more attention in Earth system research as an important parameter. The change of soil temperature ($T_s$) in different regions and related time series is affected by many factors, which bring certain difficulties to the accuracy of soil temperature prediction and the robustness of the algorithm. In this paper, an embedded network prediction model based on the gated recurrent unit (GRU) model is proposed to learn the local and global features of historical temperature for improving the prediction performance of soil temperature. We input different steps into the GRU model, and the output is weighted to obtain the final prediction result. In order to obtain the global characteristics of soil temperature, we connect the previous steps to the output layer directly, and the local characteristics of soil temperature are obtained through the following steps. This paper uses the soil temperature data from two meteorological stations (Laegern and Fluehli) in Switzerland as the input data to predict the soil temperature for different soil depths (5 cm, 10 cm, and 15 cm) at different time points (6 hrs, 12 hrs, and 24 hrs), using RMSE, MAE, MSE, and $R^2$ performance indicators as evaluation criteria to verify the accuracy of prediction. As the experimental results show, our method has the best performance compared to the others (artificial neural networks (ANN), extreme learning machine model (ELM), long short-term memory network (LSTM), gated recurrent unit network (GRU)). In particular, we estimated the soil temperature at the soil depth of 10 cm of the Fluehli station in the coming 6 hrs; our method achieved the best performance; and, meanwhile, our model achieved the maximum value of $R^2$ (0.9914) and the minimum values of RMSE (0.4668), MAE (0.2585), and MSE (0.2214) compared with the other four models. Therefore, our model can not only predict the soil temperature at different depths but also improve the accuracy.

1. Introduction

Geoscience has played an important role in social development and economic construction; soil temperature ($T_s$) and its daily fluctuations are among the most vital meteorological parameters in Earth sciences, such as agriculture, forestry, and geology [1, 2]; and it is an important variable of land-atmosphere interactions [3]. Meanwhile, there are many elements that affect the change of soil temperature; for example, the change of soil depth has a prominent effect on soil temperature. Research has shown that, in the processing of plant growth, shallow soil has a significant impact on seed germination, while deep soil affects root absorption activity [4]. Therefore, the accurate prediction of soil temperature at different depths can be used to guide practical applications in some fields, which is instead of using traditional sensors manually for on-site measurement [5].

Currently, most of the soil temperature prediction methods use environmental factors to estimate [6]. However, the data collected in some regions is unavailable that cannot be used to predict, which will reduce the accuracy of model predictions [7]. Therefore, this paper recommends using time series as the input data for the soil temperature prediction model.

In recent years, researchers usually use methods based on physical models to predict soil temperature through the heat transfer mechanism of the soil itself mainly [6, 8]. However, there are many limitations in practical applications due to the
physical model parameterization and scale issues [9]. With the continuous development of the machine learning method, it has been widely used in Earth sciences [10–12]. Ghorbani et al. proposed a method based on the support vector machine to estimate the soil field capacity and permanent wilting point [13]. And it also plays an important role in the soil temperature field [14]. The extreme learning machine is used to predict the soil temperature for improving the accuracy by Feng et al. [15]. Furthermore, LSTM has also received attention from researchers [16].

For the machine learning method, the artificial neural network (ANN) [2, 17, 18] and ELM [15, 19] can learn the features from the input data without using a physical model, so there is no need to understand their internal physical processes. The artificial neural network has strong self-adaptation and self-learning capabilities and can continuously update the parameters in the model to make the output value closer to the real value, so it is used as a soil temperature prediction model usually [20–22]. Bilgili proposed a method based on the artificial neural network to predict monthly average soil temperature [23]. Mehdizadeh et al. used the model based on feedforward backpropagation neural networks (FFBPNN) and gene expression programming (GEP) to estimate the daily soil temperature at different depths [24]. When the traditional artificial neural network is used to predict the soil temperature, the accuracy of the output results is reduced because the correlation of the time series is not considered. For solving this problem, many researchers merge genetic algorithms into artificial neural networks to optimize neural networks [20, 25, 26]. However, the genetic algorithm is not efficient and prone to premature convergence. Therefore, this method still needs further research.

Deep learning methods are widely used to deal with time series data. Compared with recurrent neural networks (RNN) and LSTM, GRU has a simpler structure and can solve the problem of long-term dependence [27]. Therefore, this paper chooses the model based on the GRU network to predict soil temperature. The model uses hidden states to convey the information and process the relevance of time series data. It is widely used in many fields due to its special network structure. Liu et al. proposed a method, GRU-based nonlinear predictive denoising autoencoders for fault diagnosis of rolling bearing [28]. Miao and Hung designed the Conv-GRU model to estimate the water level, and the results show the effectiveness of the method [29]. Rui et al. combined GRU and LSTM models to predict traffic flow [30]. According to our research, the GRU network had not been used for soil temperature prediction.

The following questions are the focus of this article. The first one is how to choose the input data to $T_s$ estimation model. The estimation of $T_s$ is affected by the past $T_s$. Although the relevant meteorological data have some impact on $T_s$ estimation, the accuracy of the model for $T_s$ estimation will be affected by the errors between the provided data and the real data. Consequently, this paper concentrates on the time series data. The other one is about the network model construction of the method in our paper. In the GRU network, the information is transferred by updating the cell state and the parameters in the hidden state. As the steps of the time series increase, the correlation between the initial data and the output data will be decreased, which will lead to a decrease in prediction accuracy.

The motivation of this paper is to solve the problem of long-term serial dependence of soil temperature data, which leads to a decrease in prediction accuracy. With the goal, this paper proposed a new embedded estimation model based on the GRU network for $T_s$ estimation. In order to obtain the global characteristics of soil temperature, we connect the previous steps to the output layer directly, and the local characteristics of soil temperature are obtained through the following steps. We set different steps to the channels; with the outputs as cells, the estimation result is calculated by fully connecting different cells, using the past $T_s$ data from the Laegern and Fluehli stations in Switzerland from 2006 to 2014 to estimate $T_s$ in the next 6 hrs, 12 hrs, and 24 hrs at different soil depths (5, 10, and 15 cm).

The main contributions of this paper for $T_s$ estimation are listed as follows:

1. According to our research, the GRU network had not been used for soil temperature prediction yet, and the method based on GRU was achieved in this paper for the purpose of estimating soil temperature.

2. In order to obtain the global characteristics of soil temperature, we connect the previous steps to the output layer directly, and the local characteristics of soil temperature are obtained through the following steps.

3. As the results showed, our method has a better performance than the other advanced technology available.

2. Materials and Methods

2.1. The Framework for Soil Temperature Estimation. First of all, we extract the corresponding past $T_s$ data from the Laegern and Fluehli stations on FLUXNET as the input to our model. Meanwhile, we consider several other models based on machine learning technology models (LSTM, BPNN, ELM, and GRU). In our model, we connect the previous steps to the output layer directly to predict $T_s$. Finally, we compare the results by several statistical evaluation criteria (RMSE, MAE, MSE, and $R^2$) to evaluate the performance of the model. Figure 1 shows the overall structure of our study.

2.2. The Structure of GRU. The GRU network has the characteristics of simple structure and fast training speed and can transmit relevant information to the time series for prediction. It is widely applied in many fields due to its advantages precisely. The GUR can solve the time series problem and the gradient problem in backpropagation. The GRU unit structure is shown in Figure 2. It has two gates, which are the reset gate and the update gate. The update gate decides which new information should be discarded and added. The reset gate is used to decide how much past information to forget.
The calculation formulas of the GRU are as follows:

\[
\begin{align*}
    r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]), \\
    z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]), \\
    \tilde{h}_t &= \tanh(W_h \cdot [r_t \ast h_{t-1}, x_t]), \\
    h_t &= (1 - z_t) \ast h_{t-1} + z_t \ast \tilde{h}_t, \\
    y_t &= \sigma(W_o \cdot h_t),
\end{align*}
\]

where \(x(t)\) represents the input value at the current moment and \(h_{t-1}\) is the hidden state of the previous node and uses them to get the gate status. \(r_t\) is the reset gate, \(z_t\) is the reset gate, \(\tilde{h}_t\) represents the new memory, \(h_t\) represents the hidden state, \(y_t\) is the output of the output layer, \(\sigma(\cdot)\) is the sigmoid activation function, and \(\tanh(\cdot)\) is the output tangent function.

2.3. The Structure of Our Model. Through the previous analysis, when there are a large number of cells in the GRU network, the correlation between the features will decrease with the time series extending. Therefore, we proposed the model based on GRU is to solve the problem and improve the accuracy of the estimation model, as the topological structure is shown in Figure 3.

Our model network is composed of the traditional GRU network and the auxiliary networks. The information is updated to the next cell through the parameter back-propagation of the hidden state in the GRU model that would decrease the correlation with the earlier. The traditional GRU network is used as the basic network to obtain local features, and the auxiliary network is composed of the output of different steps to obtain global features, meanwhile, merging the features as the output of the entire network model. Input the past \(T_s\) data into our model to learn the pattern of periodic changes in \(T_s\), which can enhance the correlation between past \(T_s\) data and improve the accuracy for the prediction.

We proposed the final output of our model is \(y_{\text{our method}}(t)\) at the time step \(t\), which combined all the channels to the fully connected layer, as follows:

\[
y_{\text{our method}}(t) = \sum_{ni=0}^{n} W_{ni} y(t - ni + 1)_{ni} + b.
\]

2.4. Objective Function Optimization Algorithm. Adaptive moment estimation algorithm has excellent performance with high computational efficiency and low memory requirements [31]. It can calculate different adaptive learning
rates for different parameters. The method is suitable for processing large-scale data and optimizing parameters, as well as solving sparse gradient problems. The Adam is widely used in the field of deep learning, and it is used to optimize the model in this paper. This article uses the mean square error to optimize our model, and the calculation formula is as follows:

$$L = \frac{1}{N} \sum_{i=1}^{N} (y(t) - \tilde{y}_{\text{our method}}(t))^2,$$

where the observed data from the stations is $y(t)$ and the output of our model is $\tilde{y}_{\text{our method}}(t)$.

Learning rate is an important parameter in deep learning. The value of the learning rate will affect the convergence of the function. When the learning rate is set too small, the convergence will be very slow. Meanwhile, when the value is set too large, the gradient will be affected. We use the method of exponential decay learning rate to improve the convergence of Adam and the method of exponential decay learning rate. By constantly adjusting the learning rate, the step size is set to 100 and the attenuation rate is 0.96; the algorithm is close to the optimal solution.

2.5. Model Training and Test. In this paper, the past $T_s$ data (the Laegern and Fluehli stations in Switzerland from 2006 to 2014) is served as the input to our model for estimating $T_s$ and using TensorFlow backend. The model is tested on Intel Core (TM) i7-5820K, 3.30 GHz CPU, and 64 GB memory running Pycharm 2018. We use three-quarters of all data as training samples (data during 2006.1.1–2013.3.14), and the others were used as testing samples (data during 2013.3.15–2014.12.31). We assume that the value at the $t$ point in the time series is $x_t$, which is predicted by the first $t-1$ elements; use half of the daily soil temperature to predict the values of $T_s$ in the following 6 hrs, 12 hrs, and 24 hrs; and set the value of $t$ to 24. This paper compared our model with the other models (including ANN, LSTM, ELM, and GRU), meanwhile, calculating several evaluation criteria (RMSE, MAE, MSE, and $R^2$) to estimate the model performance, as follows:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (y_i - \tilde{y}_i)^2}{N}},$$

$$\text{MAE} = \frac{\sum_{i=1}^{N} |y_i - \overline{y_i}|}{N},$$

$$\text{MSE} = \frac{\sum_{i=1}^{N} (y_i - \overline{y_i})^2}{N},$$

$$R^2 = \frac{\sum_{i=1}^{N} (y_i - \overline{y})^2 - \sum_{i=1}^{N} (y_i - \tilde{y}_i)^2}{\sum_{i=1}^{N} (y_i - \overline{y})^2},$$

where the total number of data is denoted as $N$, $y_i$ is the observed value of data at the moment, $\tilde{y}_i$ is the predicted value obtained through different methods, $\overline{y}$ is the observed average value of the data. According to our knowledge, we can understand the fitting degree of the model and the accuracy of data prediction through evaluation criteria. With the smaller value of RSME, MAE, and MSE and the larger value of $R^2$, the model will show the best performance.

2.6. Study Area and Field Experiment. This paper studied the data from two stations (Laegern 47.48 N, 8.37 E, Fluehli 46.88 N, 8.01 E) located in Switzerland and downloaded the past $T_s$ data within half an hour on FLUXNET (https://fluxnet.fluxdata.org/) to verify our model. Since these two
stations are located in their domestic ecological nature reserve, $T_s$ has a certain impact on the surrounding ecological environment, such as plant growth, soil fertility, and microbial activities, as shown in Figure 4.

This article takes the past $T_s$ data from the stations as the input. With the data provided by the stations, it can be seen that the depth at the 15 cm soil temperature of the Laegern station is the most stable, achieving the minimum of temperature differences and standard deviation; $x_{\text{min}}$ is the minimum value; $x_{\text{max}}$ is the maximum value; $x_{\text{mean}}$ is the average value; $z_{\text{sd}}$ represents the standard deviation; $z_r$ represents skewness; and $z_v$ represents variation coefficient which is shown in Table 1.

3. Results and Discussions

In this paper, comparing our model with the other four models (BPNN, LSTM, ELM, GRU) for estimating $T_s$ at the two stations’ data, use the Adam to optimize the model and experiment with scikit-learn. For instance, input layer, hidden layer, and output layer consist of the ANN. We set the batch size to 10000, the number of iterations to 100, the learn rate to 0.03, and the number of nodes to 32, and the model get the best performance. We apply the elm function to the ELM model, the activation function in the hidden layer is sigmoid, and the number of nodes is set same to ANN. We Set the same hyperparameters to the GRU and our model so that it can be useful to show the performance of our model.

3.1. Evaluation for the Hyperparameters in Our Model.

According to our research, the value of the hyperparameters has a certain influence on the performance of the model. As the hyperparameters, what we mentioned are the number of channels, iterations, learning rate, and the number of nodes (num\_node\_model). For example, we estimate the soil temperature at the depth of 5 cm for 6 hrs of the Laegern station. As the results showed, num\_node\_model has a certain impact on the fitting of the model and the acquisition of relevant important information. With the learning rate set to 0.03, the number of nodes is 32, the number of channels is set to 4, the number of iterations is set to 100, and our model has the best performance.

When the model is overfitting during the training process, it is not conducive to the model adapting to the changes of the data. In contrast, if underfitting occurs during training, it is not conducive to data mining of related data. Moreover, due to the large learning rate value for optimal weights, the predictive model can easily be trapped into local optimum during the learning process. Then, when the learning rate value is small, it will make the parameters hardly converge to the optimal value for training the predictive model. The results are shown in Table 2, as follows:

3.2. Evaluation for Different Models. Comparing our model with the other four models (BPNN, ELM, LSTM, and GRU) in this part. The inputs to all the predictive models were the past $T_s$ data from the stations. The output of the model was estimated $T_s$ values in the following 6 hrs, 12 hrs, and 24 hrs.

The predicting results of the five models at the depths of 5, 10, and 15 cm in the following 6 hrs, 12 hrs, and 24 hrs are experimented for the Laegern station as shown in Table 3. For the depth of 5 cm results in the coming 6 hrs, our model has better performance than the other models. According to the results, our model is compared with other models, and the percentage of the RMSE has reductions of 35.4% (GRU), 38.9% (LSTM), 39.1% (ELM), and 53.4% (BPNN), respectively; the percentage of the MAE has reductions of 42.9% (GRU), 46.2% (LSTM), 45.4% (ELM), and 58.2% (BPNN); and the percentage of the MSE has reductions of 57.3% (GRU), 58.6% (LSTM), 58.8% (ELM), and 75.9% (BPNN), getting the $R^2$ score amount to 0.9638 (our model) compared to 0.9034 (LSTM), 0.9022 (ELM), 0.8319 (BPNN), and 0.9058 (GRU). Because our model has the lower value of RMSE, MAE, and MSE, meanwhile, having the higher value of $R^2$, it is obvious that our model has a very excellent performance on $T_s$ estimation. However, for the depth of 5 cm results in the following 12 hrs and 24 hrs, we can draw the same conclusion that our model performance is still stronger than the other four models. The accuracy has been continuously improved from 5 cm to 15 cm of the soil depths, but the accuracy in the following 6hrs to 24 hrs has been decreased. It may be caused by systematic errors when the model makes long-term predictions [32]. However, the ELM model gets a superior level of accuracy compared to the others models for the depth of 5 cm in the following 12 hrs, the depth of 10 cm in the following 6 hrs, and the depth of 15 cm. The reason for this might be ELM is a feedforward neural network architecture in which parameters are randomly chosen [19]. Sometimes a nonoptimal solution may be generated, which will affect the performance of the model.

We estimate the soil temperature at the depth of 5 cm, 10 cm, and 15 cm of the Laegern station in the following 6 hrs, 12 hrs, and 24 hrs, respectively. The estimation results are at the depth of 5 cm, 10 cm, and 15 cm of the Laegern station in the following 6 hrs, 12 hrs, and 24 hrs. The linear relationship between the estimated value and the observed value has a certain degree of robustness for long-term estimation.
The frequency plot of the absolute estimation error is shown in Figure 6, and each bar indicates the error percentage. It can be obviously seen that our model has the highest frequency for the smallest error magnitude encountered in estimating $T_s$ of the Laegern station. For example, for the depth of 5 cm in the following 6 hrs, our model has a higher value of frequency (73.8%) compared to the other four models (46.5% (GRU), 45.9% (LSTM), 46.6% (ELM), 36.8% (BPNN)).

Table 1: Statistical properties of the meteorological information and soil temperature data for Laegern and Fluehli stations.

| Station | Variable(°C) | $x_{min}$ | $x_{max}$ | $x_{mean}$ | $z_{sd}$ | $z_s$ | $z_v$ |
|---------|--------------|-----------|-----------|------------|---------|-------|-------|
| Laegern | $T_s$-5 cm   | -1.888    | 26.876    | 10.104     | 6.061   | 0.103 | 0.599 |
|         | $T_s$-10 cm  | -0.181    | 22.193    | 9.726      | 5.435   | -0.031| 0.558 |
|         | $T_s$-15 cm  | 0.16      | 19.394    | 9.010      | 5.025   | -0.068| 0.557 |
| Fluehli | $T_s$-5 cm   | -0.35     | 21.822    | 8.729      | 6.338   | 0.075 | 0.726 |
|         | $T_s$-10 cm  | -0.044    | 21.727    | 8.836      | 6.242   | 0.071 | 0.706 |
|         | $T_s$-15 cm  | 0.432     | 20.826    | 8.813      | 6.023   | 0.062 | 0.683 |

Table 2: Results of Laager’s $T_s$ with different hyperparameters set to our model.

| Learning rate | num_{our model} | $n$ | Iterations | $R^2$   | Estimation time (second) |
|---------------|-----------------|-----|------------|---------|--------------------------|
| 0.03          | 32              | 4   | 100        | 0.9691  | 48.790                   |
| 0.05          | 32              | 4   | 200        | 0.9692  | 99.738                   |
| 0.03          | 32              | 4   | 50         | 0.9612  | 22.534                   |
| 0.03          | 64              | 4   | 100        | 0.9676  | 91.368                   |
| 0.03          | 16              | 4   | 100        | 0.9623  | 30.257                   |
| 0.003         | 32              | 4   | 100        | 0.9615  | 49.391                   |
| 0.3           | 32              | 4   | 100        | 0.9668  | 48.006                   |
| 0.03          | 32              | 3   | 100        | 0.9683  | 41.339                   |
| 0.03          | 32              | 5   | 100        | 0.9679  | 55.641                   |
Using different predictive models to test the data separately from the Fluehli station at the depths of 5 cm, 10 cm, and 15 cm in the following 6 hrs, 12 hrs, and 24 hrs, the results are shown in Table 4. The results noted that our model mainly gets better performance than the others; for example, the proposed model achieved excellent results (RMSE = 0.6534, MAE = 0.3928, MSE = 0.4735, and $R^2 = 0.9860$) compared to the other four models at the depth of 5 cm in the following 6 hrs at Fluehli station. Generally speaking, our model mainly has superior performance for estimating $T_s$ with the experiments in different regions, different times, and different soil depths.

### Table 3: The testing phase results of the different models at Laegern station. Note that the optimal models are boldfaced.

| Depth (cm) | Estimation (hrs) | Method | RMSE   | MAE   | MSE   | $R^2$   |
|------------|-----------------|--------|--------|-------|-------|---------|
| 5          | 6               | BPNN   | 2.3634 | 1.8168| 5.5887| 0.8319  |
|            |                 | ELM    | 1.8075 | 1.3898| 3.2672| 0.9022  |
|            |                 | LSTM   | 1.8037 | 1.4109| 3.2542| 0.9034  |
|            |                 | GRU    | 1.7054 | 1.3285| 3.1537| 0.9058  |
|            |                 | **Our model** | 1.1012 | 0.7586| 1.3458| **0.9638** |
|            |                 | BPNN   | 2.4358 | 1.8543| 5.9368| 0.8224  |
|            |                 | ELM    | 1.5141 | 1.1219| 2.2926| 0.9311  |
|            |                 | LSTM   | 1.8387 | 1.3191| 3.3848| 0.8992  |
|            |                 | GRU    | 1.7321 | 1.2229| 3.2789| 0.9003  |
|            |                 | **Our model** | 1.2576 | **0.9954** | 1.7438 | **0.9539** |
|            |                 | BPNN   | 2.5215 | 1.8938| 6.3525| 0.8078  |
|            |                 | ELM    | 2.4164 | 1.7991| 5.8390| 0.8244  |
|            |                 | LSTM   | 2.2669 | 1.7470| 5.1438| 0.8464  |
|            |                 | GRU    | 2.1569 | 1.6421| 5.0349| 0.8612  |
|            |                 | **Our model** | 1.4221 | 1.0287 | 2.2316 | **0.9302** |
|            | 12              | BPNN   | 1.3643 | 1.1015| 1.8660| 0.9308  |
|            |                 | ELM    | 0.8011 | 0.6077| 0.6418| 0.9764  |
|            |                 | LSTM   | 0.8015 | 0.6268| 0.6482| 0.9743  |
|            |                 | GRU    | 0.7065 | 0.5343| 0.5489| 0.9782  |
|            |                 | **Our model** | 0.4238 | **0.2756** | **0.1714** | **0.9905** |
|            |                 | BPNN   | 1.2214 | 0.8875| 1.5035| 0.9476  |
|            |                 | ELM    | 0.8296 | 0.6273| 0.6883| 0.9747  |
|            |                 | LSTM   | 0.8085 | 0.6134| 0.6551| 0.9773  |
|            |                 | GRU    | 0.7998 | 0.5946| 0.5589| 0.9765  |
|            |                 | **Our model** | 0.6203 | **0.4421** | **0.4212** | **0.9856** |
|            | 24              | BPNN   | 1.3445 | 1.0136| 1.8175| 0.9346  |
|            |                 | ELM    | 1.2105 | 0.9174| 1.4652| 0.9460  |
|            |                 | LSTM   | 1.1412 | 0.8756| 1.2989| 0.9318  |
|            |                 | GRU    | 1.0462 | 0.7216| 1.1236| 0.9563  |
|            |                 | **Our model** | 0.7658 | 0.5554 | 0.6513 | **0.9756** |
|            | 10              | BPNN   | 0.9325 | 0.8367| 0.8734| 0.9659  |
|            |                 | ELM    | 0.5790 | 0.4275| 0.3353| 0.9858  |
|            |                 | LSTM   | 0.5747 | 0.4589| 0.3363| 0.9851  |
|            |                 | GRU    | 0.4825 | 0.3546| 0.2445| 0.9868  |
|            |                 | **Our model** | 0.2754 | **0.1643** | **0.1287** | **0.9951** |
|            | 12              | BPNN   | 0.8812 | 0.7145| 0.7785| 0.9654  |
|            |                 | ELM    | 0.6626 | 0.5008| 0.4390| 0.9814  |
|            |                 | LSTM   | 0.6364 | 0.5175| 0.4225| 0.9868  |
|            |                 | GRU    | 0.5638 | 0.4089| 0.3896| 0.9865  |
|            |                 | **Our model** | 0.4755 | **0.3241** | **0.2114** | **0.9875** |
|            | 24              | BPNN   | 0.9375 | 0.6842| 0.8318| 0.9653  |
|            |                 | ELM    | 0.9024 | 0.6892| 0.8143| 0.9654  |
|            |                 | LSTM   | 0.8984 | 0.7052| 0.8068| 0.9646  |
|            |                 | GRU    | 0.7968 | 0.6068| 0.6994| 0.9689  |
|            |                 | **Our model** | 0.6926 | **0.5435** | **0.4797** | **0.9799** |
Our model in estimating the coming 6 hours

\[ y = 1.0063x + 0.0438 \]
\[ R^2 = 0.9638 \]

Our model in estimating the coming 12 hours

\[ y = 0.9989x + 0.1089 \]
\[ R^2 = 0.9539 \]

Our model in estimating the coming 24 hours

\[ y = 0.9847x + 0.1969 \]
\[ R^2 = 0.9302 \]

GRU model in estimating the coming 6 hours

\[ y = 0.9899x - 0.3265 \]
\[ R^2 = 0.9058 \]

GRU model in estimating the coming 12 hours

\[ y = 1.0789x - 0.4693 \]
\[ R^2 = 0.9003 \]

GRU model in estimating the coming 24 hours

\[ y = 0.981x - 0.1998 \]
\[ R^2 = 0.8612 \]

LSTM model in estimating the coming 6 hours

\[ y = 0.9799x - 0.3569 \]
\[ R^2 = 0.9034 \]

LSTM model in estimating the coming 12 hours

\[ y = 1.1032x - 0.6219 \]
\[ R^2 = 0.8992 \]

LSTM model in estimating the coming 24 hours

\[ y = 0.991x - 0.3152 \]
\[ R^2 = 0.8464 \]

ELM model in estimating the coming 6 hours

\[ y = 0.9687x + 0.3621 \]
\[ R^2 = 0.9311 \]

ELM model in estimating the coming 12 hours

\[ y = 0.979 \]
\[ R^2 = 0.8244 \]

ELM model in estimating the coming 24 hours

\[ y = 0.9589x + 0.4857 \]
\[ R^2 = 0.9022 \]

BPNN model in estimating the coming 6 hours

\[ y = 1.1498x + 0.2447 \]
\[ R^2 = 0.8319 \]

BPNN model in estimating the coming 12 hours

\[ y = 0.8846x + 1.2348 \]
\[ R^2 = 0.8224 \]

BPNN model in estimating the coming 24 hours

\[ y = 1.1235x + 0.5123 \]
\[ R^2 = 0.8244 \]

(a) Figure 5: Continued.
Figure 5: Continued.
Figure 5: The scatterplots of estimated and observed soil temperature using different predictive models for the Laegern station. (a) The depth of 5 cm. (b) The depth of 10 cm. (c) The depth of 15 cm.
Our model in estimating the coming 6 hours

GRU model in estimating the coming 6 hours

LSTM model in estimating the coming 6 hours

ELM model in estimating the coming 6 hours

BPNN model in estimating the coming 6 hours

Our model in estimating the coming 12 hours

GRU model in estimating the coming 12 hours

LSTM model in estimating the coming 12 hours

ELM model in estimating the coming 12 hours

BPNN model in estimating the coming 12 hours

Our model in estimating the coming 24 hours

GRU model in estimating the coming 24 hours

LSTM model in estimating the coming 24 hours

ELM model in estimating the coming 24 hours

BPNN model in estimating the coming 24 hours

Figure 6: Continued.
Figure 6: Continued.
Figure 6: The frequency plot of the absolute estimation error using the predictive models for the Laegern station. (a) The depth of 5 cm. (b) The depth of 10 cm. (c) The depth of 15 cm.
4. Conclusions

Soil temperature ($T_s$) as the important variable is one of the land surface features impacting Earth science, usually used in many research fields; for example, it affects the growth and development of plants and the formation of soil. In this study, we research the performance of backpropagation neural networks (BPNN), gated recurrent unit (GRU), extreme learning machine (ELM), long short-term memory (LSTM) network, and our model for estimating $T_s$ at the depth of 5 cm, 10 cm, and 15 cm in the following 6 hrs, 12 hrs, and 24 hrs over the Switzerland Laegern and Fluehli stations. The statistical results indicated that our model mainly performs better than the other four models on the $T_s$ estimation.

In order to reduce the influence of long-term series on the accuracy of soil temperature estimation and obtain the global characteristics of soil temperature, we connect the previous steps to the output layer directly, and the local characteristics of soil temperature are obtained through the following steps.

The soil temperature is affected by many factors, such as atmospheric temperature and precipitation. EEMD can decompose time series into signals of different frequencies
and display the time series values with distinguishing ability under different frequencies. This method may decompose the complex soil temperature time series into discriminative data. Therefore, we will attempt to integrate EEMD to decompose the soil temperature time series and the features with different frequencies into the model to improve estimation accuracy.

**Data Availability**
The data included in this paper are available without any restriction.

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

**Acknowledgments**
This work was supported by the National Natural Science Foundation of China (NSFC) under Grant no. 51805203, the Science and Technology Development Plan of Jilin Province 20190201023JC, and the Development and Reform Commission of Jilin Province (2019C054-2).

**References**

[1] 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, pp. e48011–9, 2012.

[2] H. Tabari, P. Hosseinzadeh Talaei, and P. Willems, “Short-term forecasting of soil temperature using artificial neural network,” *Meteorological Applications*, vol. 22, no. 3, pp. 576–585, 2014.

[3] 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.

[4] O. Kisi and H. Sanikhani, “Modelling long-term monthly temperatures by several data-driven methods using geographical inputs,” *International Journal of Climatology*, vol. 35, no. 3, pp. 3834–3846, 2015.

[5] M. Saeid, B. Javad, and K. Keivan, “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.

[6] 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: Atmosphere*, vol. 119, no. 2, pp. 407–417, 2014.

[7] S. Meh dizadeh, 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.

[8] J. Qi, S. Li, Q. Li, Z. Xing, C. P.-A. Bourque, and F.-R. Meng, “A new soil-temperature module for SWAT application in regions with seasonal snow cover,” *Journal of Hydrology*, vol. 538, pp. 863–877, 2016.

[9] 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.

[10] J. Shiri, S. Kim, and O. Kisi, “Estimation of daily dew point temperature using genetic programming and neural networks approaches,” *Hydrology Research*, vol. 45, no. 2, pp. 165–181, 2014.

[11] O. Kisi and H. Sanikhani, “Prediction of long-term monthly precipitation using several soft computing methods without climatic data,” *International Journal of Climatology*, vol. 35, no. 14, pp. 4139–4150, 2015.

[12] M. Mohammadi, S. Shamshirband, A. Kamsin, P. C. Lai, and Z. Mansor, “Identifying the most significant input parameters for predicting global solar radiation using an ANFIS selection procedure,” *Renewable and Sustainable Energy Reviews*, vol. 63, pp. 423–434, 2016.

[13] M. A. Ghorbani, S. Shamshirband, D. Zare Haghi, A. Azani, H. Bonakdari, and I. Ebtehaj, “Application of firefly algorithm-based support vector machines for prediction of field capacity and permanent wilting point,” *Soil and Tillage Research*, vol. 172, pp. 32–38, 2017.

[14] M. Delbari, S. Sharifzadari, 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.

[15] 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, no. 2019, pp. 67–77, 2019.

[16] Q. Li, H. H. Yao, Y. Zhao et al., “GANs-LSTM model for soil temperature estimation from meteorological: a new approach,” *IEEE Access*, vol. 8, pp. 59427–59443, 2020.

[17] 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. 2015, pp. 377–387, 2015.

[18] 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.

[19] 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, no. 2018, pp. 52–64, 2018.

[20] 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, no. 2018, pp. 37–50, 2018.

[21] 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.

[22] 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, no. 2019, pp. 73–87, 2019.

[23] M. Bilgici, “The use of artificial neural networks for forecasting the monthly mean soil temperatures in Adana, Turkey,” *Turkish Journal of Agriculture and Forestry*, vol. 35, no. 2011, pp. 83–93, 2011.
S. Mehdizadeh, F. Fathian, M. J. S. Safari, and A. Khosravid, “Developing novel hybrid models for estimation of daily soil temperature at various depths,” Soil and Tillage Research, vol. 197, no. 2020, Article ID 104513, 2020.

A. Danandeh Mehr, E. Kahya, and E. Kahya, “A Pareto-optimal moving average multigene genetic programming model for daily streamflow prediction,” Journal of Hydrology, vol. 549, pp. 603–615, 2017.

S. Bahrami, F. Doulati Ardejani, and E. Baafi, “Application of artificial neural network coupled with genetic algorithm and simulated annealing to solve groundwater inflow problem to an advancing open pit mine,” Journal of Hydrology, vol. 536, pp. 471–484, 2016.

K. Cho, B. V. Merrienboer, C. Gulcehre et al., “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” 2014, https://arxiv.org/abs/1406.1078.

H. Liu, J. Zhou, Y. Zheng, Y. Jiang, and Y. Zhang, “Fault diagnosis of rolling bearings with recurrent neural network-based autoencoders,” ISA Transactions, vol. 77, pp. 167–178, Article ID S0019057818301514, 2018.

S. Miau and W.-H. Hung, “River flooding forecasting and anomaly detection based on deep learning,” IEEE Access, vol. 8, pp. 198384–198402, 2020.

F. Rui, Z. Zuo, and L. Li, “Using LSTM and GRU neural network methods for traffic flow prediction,” in 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), IEEE, Wuhan, Hubei Province, China, November 2016.

D. P. Kingma and J. Ba, “Adam: a method for stochastic optimization,” in Proceedings of the ICLR, San Diego, CA, USA, May 2015.

K. Judd and M. Small, “Towards long-term prediction,” Physica D: Nonlinear Phenomena, vol. 136, no. 1-2, pp. 31–44, 2000.