MDPI

Article

Harnessing Deep Learning and Snow Cover Data for Enhanced Runoff Prediction in Snow-Dominated Watersheds

Rana Muhammad Adnan [1,*] , Wang Mo [1,*] , Ozgur Kisi [2,3,4] , Salim Heddam [5] , Ahmed Mohammed Sami Al-Janabi [6] and Mohammad Zounemat-Kermani [7]

1   College of Architecture and Urban Planning, Guangzhou University, Guangzhou 510006, China
2   Department of Civil Engineering, Lübeck University of Applied Science, 23562 Lübeck, Germany; ozgur.kisi@th-luebeck.de
3   Department of Civil Engineering, School of Technology, Ilia State University, 0162 Tbilisi, Georgia
4   School of Civil, Environmental and Architectural Engineering, Korea University, Seoul 02841, Republic of Korea
5   Hydraulics Division, Agronomy Department, Faculty of Science, University20 Août 1955 Skikda, Route El Hadaik, BP 26, Skikda 21024, Algeria; s.heddam@univ-skikda.dz
6   Department of Civil Engineering, Cihan University-Erbil, Kurdistan Region, Erbil 44001, Iraq; ahmed.mohammedsami@cihanuniversity.edu.iq
7   Department of Water Engineering, Shahid Bahonar University of Kerman, Kerman 76169-14111, Iran; zounemat@uk.ac.ir
*   Correspondence: rana@gzhu.edu.cn (R.M.A.); saupwangmo@gzhu.edu.cn (W.M.)

Abstract: Predicting streamflow is essential for managing water resources, especially in basins and watersheds where snowmelt plays a major role in river discharge. This study evaluates the advanced deep learning models for accurate monthly and peak streamflow forecasting in the Gilgit River Basin. The models utilized were LSTM, BiLSTM, GRU, CNN, and their hybrid combinations (CNN-LSTM, CNN-BiLSTM, CNN-GRU, and CNN-BiGRU). Our research measured the model's accuracy through root mean square error (RMSE), mean absolute error (MAE), Nash–Sutcliffe efficiency (NSE), and the coefficient of determination (R^2). The findings indicated that the hybrid models, especially CNN-BiGRU and CNN-BiLSTM, achieved much better performance than traditional models like LSTM and GRU. For instance, CNN-BiGRU achieved the lowest RMSE (71.6 in training and 95.7 in testing) and the highest R^2 (0.962 in training and 0.929 in testing). A novel aspect of this research was the integration of MODIS-derived snow-covered area (SCA) data, which enhanced model accuracy substantially. When SCA data were included, the CNN-BiLSTM model's RMSE improved from 83.6 to 71.6 during training and from 108.6 to 95.7 during testing. In peak streamflow prediction, CNN-BiGRU outperformed other models with the lowest absolute error (108.4), followed by CNN-BiLSTM (144.1). This study's results reinforce the notion that combining CNN's spatial feature extraction capabilities with the temporal dependencies captured by LSTM or GRU significantly enhances model accuracy. The demonstrated improvements in prediction accuracy, especially for extreme events, highlight the potential for these models to support more informed decision-making in flood risk management and water allocation.

Keywords: streamflow forecasting; long short-term memory; bidirectional long short-term memory; gated recurrent unit; bidirectional gated recurrent unit; convolutional neural network; moderate resolution imaging spectroradiometer; snow-covered area

1. Introduction

Streamflow prediction is a critical component of water resource management, particularly in watersheds and basins where snowmelt significantly contributes to river discharge [1]. The Upper Indus Basin (UIB) of Pakistan, a region dominated by snow and glacier melt, is a critical water resource for agriculture and hydropower generation in the country [2]. Understanding and predicting streamflow in this region is crucial for

effective water resource management, especially in the context of climate change, which significantly impacts snow cover and glacial melt [3]. The Moderate Resolution Imaging Spectroradiometer (MODIS), one of the most recent developments in remote sensing technologies, offers useful information on snow-covered areas (SCAs), facilitating improved monitoring and analysis of these crucial factors [3,4] (Tayyab et al., 2018; Bilal et al., 2019). Globally, MODIS snow cover products have been utilized to monitor snow dynamics in regions like the Tibetan Plateau, where they play a critical role in understanding water availability for millions of people downstream [5]. Additionally, research utilizing MODIS data has provided insights into the impacts of climate variability on snow cover persistence and melt timing in Canada based on surface snow depth observations [6]. Some other applications of MODIS include assessing snow cover in mountainous regions such as Turkey [7], Austria [8], the Colorado Rocky Mountains, the Upper Rio Grande, California's Sierra Nevada, the Nepal Himalaya [9], China [10], and the Moroccan Atlas Mountains [11]. Considering the combination of meteorological, hydrometrical, and remotely sensed data leads to building better and more adequate hydrological models.

Traditionally, hydrological models have relied heavily on statistical [12], empirical [13], Stochastic [14], and physically based [15] models to predict streamflow. However, these models often struggle to capture the complex and non-linear relationships between various hydrological variables [16]. Recent advances in machine learning (ML) have significantly improved the accuracy and reliability of hydrological models. Preliminary results indicate that ML models consistently outperform traditional approaches in terms of NSE and RMSE, demonstrating their superior ability to capture non-linear relationships within hydrological data. For instance, studies have shown that hybrid models incorporating deep learning architectures yield significantly higher NSE values compared to conventional models when applied to similar datasets [17–19].

In this respect, AI-based data-driven techniques and ML models have been successfully employed in modeling and simulating sophisticated hydrological events, such as streamflow prediction [17,18]. Among the developed ML models in hydrological applications, artificial neural networks (ANNs) play an important and dominant role. The earlier employment of ANNs in stream flow and river flow prediction, like shallow neural networks, demonstrates their superior capability in capturing the non-linear nature of surface flow rate compared to conventional conceptual and empirical models due to the availability of authentic datasets (e.g., hydrometric and meteorological data) [19].

Some researchers have applied shallow learning ANNs and tree-based models for modeling and predicating streamflow in the UIB and Himalayan basins. Rahman et al. [20] compared the capability of two different types of hydrological models, such as the empirical soil and water assessment tool (SWAT) and multi-layer perceptron ANN, to simulate streamflow in the UIB. The authors claimed the superiority of the applied ANN model over the SWAT model. In another similar study, Raaj et al. [21] integrated the SWAT model with some ML models (e.g., ANN and XGBoost) for the estimation of peak flow in the Himalayan River basin. The general outcome of the study highlighted the successful integration of SWAT and ML models in achieving promising results for peak flow prediction. Mushtaq et al. [22] utilized three ML models, including CART (classification and regression tree), XGBoost (extreme gradient boosting), and RF (random forest), for 10-daily streamflow prediction in the UIB region. The models were built based on climate data (precipitation, snow water equivalent, temperature, and evapotranspiration). The findings of the study demonstrated the significant ability of ML models to predict streamflow.

Over the course of the last decade, with the advent of deep learning, advanced ANN models such as long short-term memory (LSTM) [23], bidirectional LSTM (BiLSTM) [24], gated recurrent unit (GRU) [25,26] and convolutional neural networks (CNNs) [27] have shown great promise in hydrological time series forecasting due to their ability to learn and model long-term dependencies. Deep learning can be introduced as a subset of ML modeling, which utilizes multi-layered neural networks to automatically extract features from large datasets. By this, in specific cases, deep learning enables ML models to produce more

accurate predictions without extensive feature engineering. This approach is particularly advantageous in hydrology, where complex non-linear relationships exist between various hydrological variables. Several studies have highlighted the effectiveness of deep learning models in capturing these relationships and improving prediction accuracy compared to traditional models. For instance, Imran et al. [28] applied stochastic models (e.g., SARIMA) as well as LSTM models for flood forecasting in the UIB region. It was found that the LSTM model acted better and provided more accurate forecasting performance than the stochastic model.

Studies have demonstrated the efficacy of hybrid models that combine different types of deep learning strategies (i.e., CNN-LSTM, CNN-GRU, CNN-BiLSTM, and CNN-BiGRU) to enhance predictive performance [17,18,29,30]. According to Thapa et al. [1], the integration of CNN with LSTM (CNN-LSTM) has been effective in extracting features from high-resolution hydro-meteorological data for streamflow simulation in mountainous catchments, significantly improving model performance when combined with the Gamma test method to select optimal input variables. Wang et al. [31] developed a hybrid CNN-LSTM model to extract physical and meteorological characteristics from high-resolution data, significantly improving streamflow simulation accuracy in mountainous catchments. Similarly, a hybrid CNN-LSTM model introduced by Zhou et al. [32], which integrates self-attention mechanisms with CNN and LSTM, has shown robust performance in hourly runoff prediction by effectively considering temporal and feature dependencies. The use of bidirectional LSTM (BiLSTM) and bidirectional GRU (BiGRU) models, which process data in both forward and backward directions, could further improve the capture of complex temporal dependencies in streamflow data [33].

The aforementioned research suggests that current developments in machine learning and deep learning have transformed methods for predicting streamflow, moving beyond traditional statistical approaches that often fail to capture the complexities of hydrological processes. Recent studies have shown that hybrid models combining various ML techniques and data extracted from MODIS can significantly enhance the predictive performance of each method [34–37].

According to the studied literature, the integration of snow cover data from MODIS, along with advanced DL models like LSTM, GRU, and CNN, offers a robust framework for accurate and reliable streamflow prediction, essential for effective water resource planning and flood control. In this study, we aim to leverage these advanced deep learning models for monthly streamflow prediction in the Upper Indus Basin located in Pakistan. In the recent literature, CNN-BiGRU and CNN-BiLSTM models have been successfully applied to predict various hydrological variables [38–41]. However, these models have not been extensively evaluated or compared for streamflow prediction. Addressing this research gap motivated us to select CNN-BiLSTM and CNN-BiGRU as the primary deep learning models in this study for streamflow prediction. Lagged streamflow values and snow cover area (SCA) data were chosen as inputs in this study. The autocorrelation function (ACF) feature selection technique was employed to identify key input variables from these datasets. These inputs were selected based on their proven effectiveness in previous studies on streamflow simulation [42–45]. The ACF technique was adopted due to its successful application in identifying influential variables for hydrological modeling [46,47]. By integrating lagged streamflow values identified using autocorrelation analysis with SCA data derived from MODIS, we propose a novel approach to enhance prediction accuracy. Specifically, the MODIS MOD10CM remote sensing product was utilized to extract SCA. This product was selected based on its established effectiveness in deriving SCA for high-altitude regions, as reported in the literature [48–50]. To evaluate the performance of the proposed models, statistical metrics such as root mean square error (RMSE), mean absolute error (MAE), the coefficient of determination (R^2), and Nash–Sutcliffe efficiency (NSE) were used. The selection of these metrics is justified by their frequent and successful application in hydrological modeling studies [51,52]. The specific objectives of this research are as follows:

- To develop and compare the performance of various deep learning models (LSTM, GRU, CNN, BiLSTM, BiGRU, CNN-LSTM, CNN-BiLSTM, CNN-GRU, and CNN-BiGRU) for streamflow prediction in the UIB.
- To assess the impact of including SCA data from MODIS on the prediction accuracy of these models.
- To identify the model that best captures the non-linear relationships between past streamflow values based on the autocorrelation function (ACF) and SCA data, providing the most accurate monthly streamflow predictions.

The novelty of this research lies in the integration of MODIS-derived SCA data with hybrid deep learning models for streamflow prediction, which has not been extensively explored in existing literature. Previous studies have focused primarily on traditional hydrological models or have used individual deep learning models incorporating remote sensing data [34,35]. By combining these advanced methodologies, our research provides a more robust and accurate prediction framework, which can significantly improve water resource management. To our knowledge, no previous research has looked at the combined impact of catchment features and climate, specifically snow conditions, on catchment storage and low flows in the UIB region. In addition, the contributions of this study are threefold: (i) a comprehensive evaluation of various deep learning architectures for streamflow prediction in a snow-dominated basin; (ii) a novel way to incorporate SCA data from MODIS into hybrid deep learning models (e.g., CNN-BiLSTM and CNN-BiGRU) to enhance their prediction capabilities; and (iii) insights into the effectiveness of different model architectures in capturing the complex dynamics of streamflow in the UIB, contributing to the broader field of hydrological modeling and management.

2. Study Area

The Gilgit River basin is selected as a case study region, as shown in Figure 1. The Gilgit River, which flows through the districts of Gupis-Yasin, Ghizer, and Gilgit in Pakistan's Gilgit-Baltistan region, is a tributary of the Upper Indus River. Shandur Lake is the source of the Gilgit River, which flows on to merge with the Indus River close to the villages of Juglot and Bunji. The Hindu Kush, the Himalayas, and the Karakoram are three notable mountain ranges that are thought to meet at this confluence. The rugged and mountainous terrain is traversed by the river. A high range with peaks that include elements of the Himalayas, the Karakoram Range is characterized by its geology. Limestone and sandstone, as well as metamorphic rocks like gneiss and schist, make up the majority of the rocks in the vicinity of the Gilgit River. Giant glacial deposits are also present in the area [53].

River flow rates vary greatly depending on the season because the river is fed by glacier meltwater from the Karakoram Range. Because of the glaciers melting in the summer, the flow rose dramatically, and in the winter, it decreased [54]. Several glaciers and streams nearby have produced tributaries that feed into the river, enhancing its overall flow. The Gilgit River is vital to the hydrology of the area, sustaining local residents' access to water resources and promoting agriculture. Hydroclimatic data from Gilgit station have been collected for a duration of 50 years from WAPDA, Pakistan, to examine the prediction accuracy of machine learning models using only streamflow inputs. For the training dataset, 40 years of monthly climatic data are selected, whereas the remaining 10 years of data are selected as testing datasets. To see the effect of SCA data input, snow cover data from MOD10CM remote sensing snow cover product is extracted for the recent twenty years duration and is utilized with the corresponding streamflow data. SCA data inputs with corresponding streamflow data are partitioned into equal training and testing partitions to see the effect of SCA input on the prediction accuracy of machine learning models.
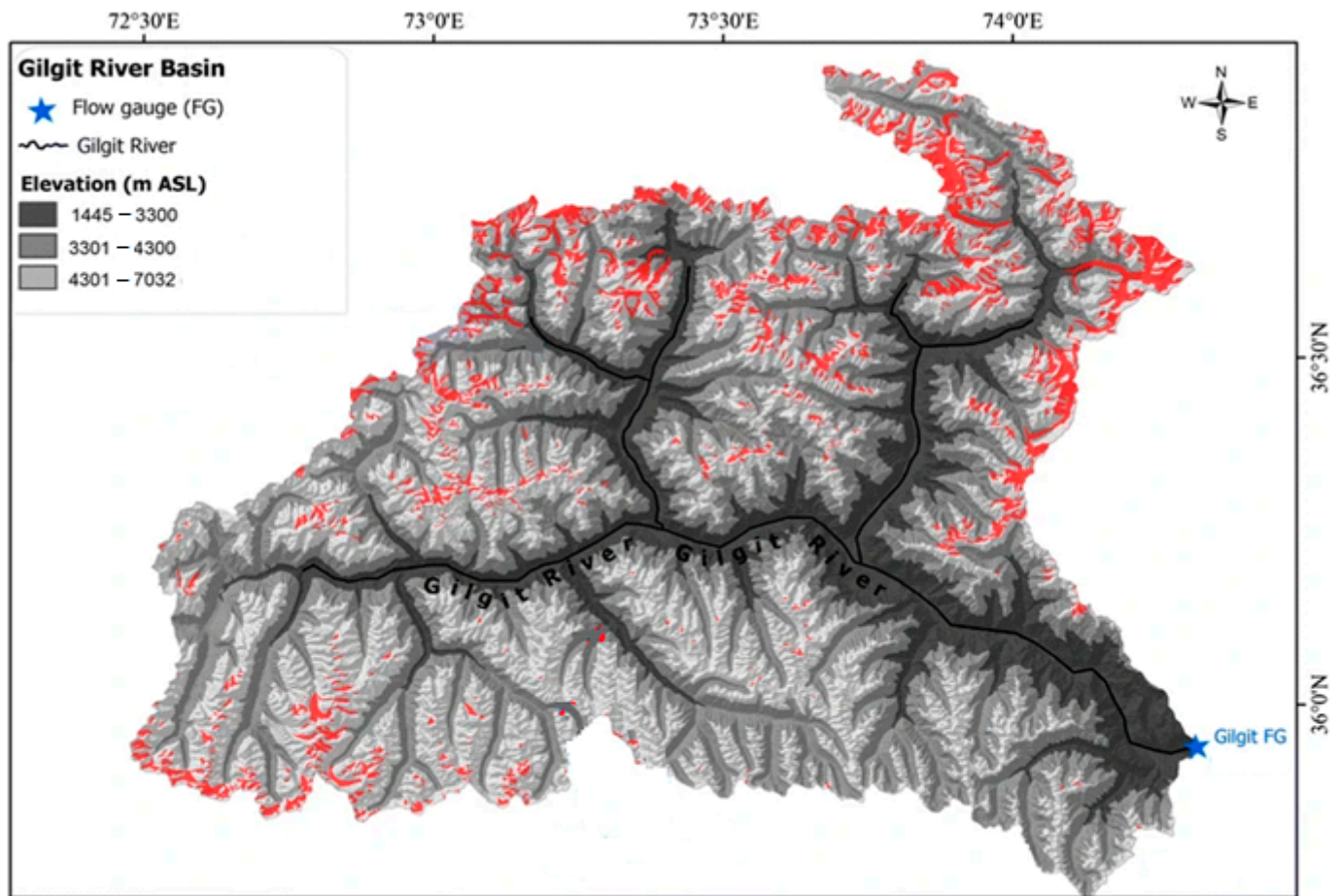
Figure 1. Location map of the study area.

3. Methods

In this study, different standalone improved and hybrid versions of deep learning models are utilized to predict the streamflow. A brief description of these deep learning models is given below.

3.1. Long Short-Term Memory (LSTM)

Based on the idea that recurrent neural networks (RNNs) have the capability to incorporate the information gained from previous time steps and to use it as new input, long short-term memory deep learning (LSTM) was developed to overcome some problems related to the long-term dependencies [55]. The LSTM (Figure 2) can capture the long-term dependencies in sequence data by introducing "*memory units*" and "*output gates*" [56]. The LSTM handles the available information from the input to the output using three different gates: "*forgotten*", "*input*", and "*output gate*". The "*forgotten gate*" works as follows:

$$f_t = \sigma\left[W_f[x_t, h_{t-1}] + b_f\right] \tag{1}$$

In the previous equation, f_t refers to the forgotten gate state at time t, σ is the sigmoidal activation function, W_f is the weight matrix, b_f is the bias matrix, h_{t-1} is the hidden layer state at time $t-1$, and x_t is the input variable [57].

The "*input gate*" uses the *tanh* activation function to provide the candidate values c, formulated as follows:

$$i_t = \sigma[W_i[x_t, h_{t-1}] + b_i] \tag{2}$$

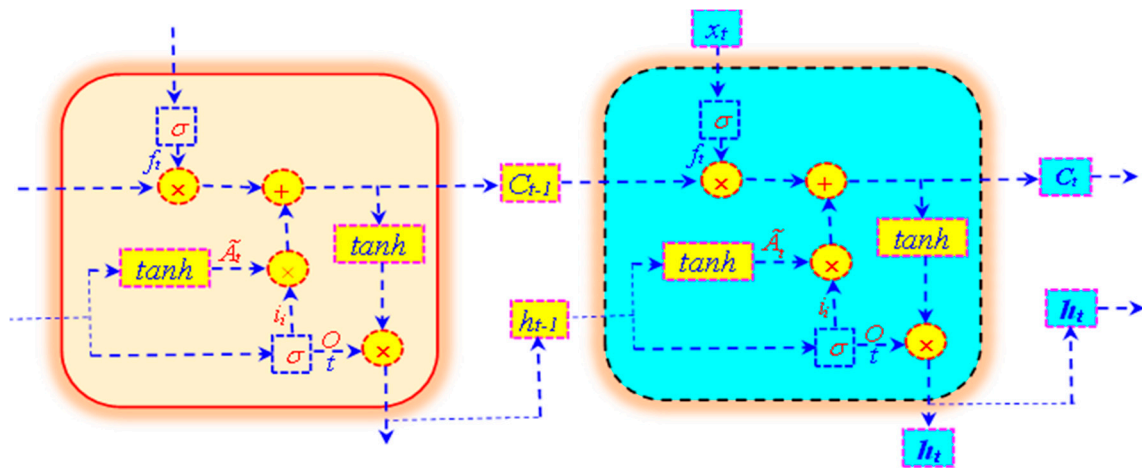$$\widetilde{c}_t = tanh[W_c[x_t, h_{t-1}] + b_c] \tag{3}$$

Figure 2. The long short-term memory (LSTM) architecture.

In the previous equations, i_t and $\widetilde{c}_t$ are the remembered information and the candidate memory unit. Furthermore, W_i is the weight matrix of the input gate, b_i is the bias matrix of the input gate, W_c is the weight matrix of the candidate memory unit, and b_c represents the *bias* of the candidate memory unit [58].

The output of the model is calculated by the output gate as follows:

$$o_t = \sigma[W_0[x_t, h_{t-1}] + b_0] \tag{4}$$

$$h_t = o_t \bullet tanh(c_t) \tag{5}$$

where h_t is the hidden state at time t, c_t is the output of the forgetting gate, the operator $\bullet$ is the Hadamard product, O_t denotes the output gate value, and W_0 and b_0 are the output gate weight and bias [58].

3.2. Bidirectional Long Short-Term Memory (BiLSTM)

According to Figure 3, we can see that the bidirectional long short-term memory model (BiLSTM) is composed of an ensemble of LSTM models with forward and backward stages. Overall, the BiLSTM structure is composed of well-known gates: a forget gate, an output gate, an input gate, and a memory cell [59]. The decision to withhold the information from the previous past calculation is provided by the "*forget gate*", while the "*input gate*" is used for updating the available information, finally, the tanh layer is used for generating new information [44]. The "*cell state*" plays the role of maintaining information within the cell and controlling the information flow across the gates. From a mathematical point of view, the input variable (x_t) is used and combined to provide the "*hidden state*" (h_t), which moves directly to the fully connected layer for providing the final response using the sigmoid activation function [59,60]. The forward and backward stages can be formulated as follows:

$$\overrightarrow{h}_t = LSTM_f\left(x_t, \overrightarrow{h}_{t-1}\right) \tag{6}$$

$$\overrightarrow{h}_t = LSTM_b\left(x_t, \overrightarrow{h}_{t-1}\right) \tag{7}$$

where the $LSTM_f$ and $LSTM_b$ represent the forward LSTM layer and backward LSTM layer; $\overrightarrow{h}_t(i = t-1, t)$ *and* $\overrightarrow{h}_t(j = t, t+1)$ signify the output of the hidden state for the forward LSTM layer and for the backward LSTM layer, respectively [61].
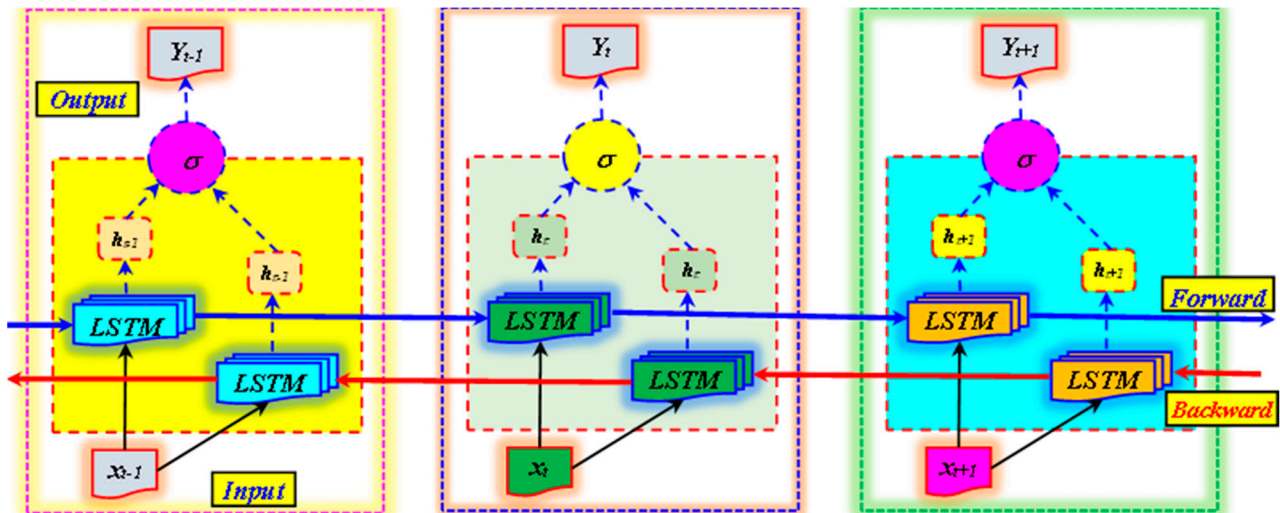
Figure 3. The bidirectional long short-term memory (LSTM).

3.3. Gated Recurrent Unit (GRU)

The gated recurrent unit (GRU) is a variant of the recurrent neural network (RNN). According to Cho et al. [62], the gated recurrent unit (GRU) was proposed for improving the RNN computation and, more precisely, for overcoming the problem of the "*vanishing gradient issue*". Inspired by the original LSTM, for which an individual memory cell is omitted, the GRU adopts two gates (Figure 4), namely the update and the reset gates, which are solely responsible for inputting and updating the information and deciding which information moves to the next stage and which should be stopped [63]. The mathematical formulation of the GRU can be written as follows:

$$R_t = \sigma(W_r x_t \oplus W_r h_{t-1} \oplus b_r) \tag{8}$$

$$Z_t = \sigma(W_Z x_t \oplus W_Z h_{t-1} \oplus b_Z) \tag{9}$$

$$C_t = tanh(W_c x_t \oplus W_c(R \odot h_{t-1}) \oplus b_c) \tag{10}$$

$$o_t = (1 - Z_t) \odot C_{t-1} + Z_t \oplus c_t \tag{11}$$

where W_r, W_z, W_c, b_r, b_z, and b_c are the weights and bias trainable parameters in the reset and update gates, where Z_t and R_t are the outputs of the update gate and reset gate, respectively, x_t is the input vector at the t th time step, h_{t-1} is the available information's at the previous time step, σ is the sigmoid function, $\oplus$ is an elementary addition operation, O_t is the hidden state at current time t, and c_t is the candidate output state [64,65].

3.4. Bidirectional Gated Recurrent Unit (Bi-GRU)

Bi-GRU can be viewed as a combination of an ensemble of single GRU networks having two opposed layers. The first layer has a forward direction, while the second layer has a backward opposed direction [66]. The responses of the two layers are combined together to calculate the final response of the Bi-GRU network (Figure 5). The main idea behind the proposition of the Bi-GRU is to help the single GRU in capturing the maximum of gained information at time step t simultaneously from previous time steps and from the subsequent steps [66]. From a mathematical point of view, after collecting the information from the forward and backward layers, the Bi-GRU provides the final hidden state $\overleftrightarrow{h_t}$ at time t [67] as follows:

$$\overleftrightarrow{h_t} = W_t \overrightarrow{h_t} + \beta_t \overleftarrow{h_t} + b_t \tag{12}$$

where W_t, β_t, and b_t are the weight of the hidden layer in a forward and backward state, and b_t represents the bias of the hidden layer at time t.
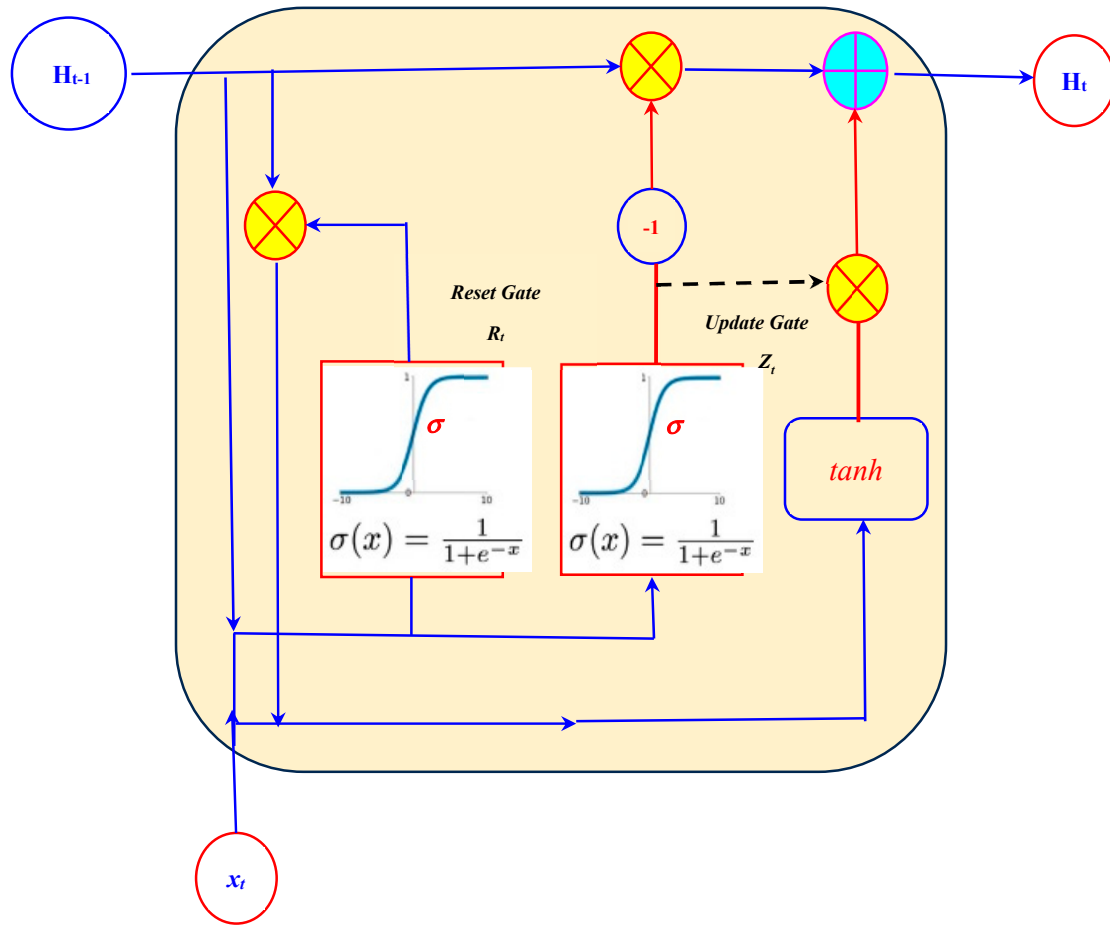
Figure 4. The gated recurrent unit (GRU).
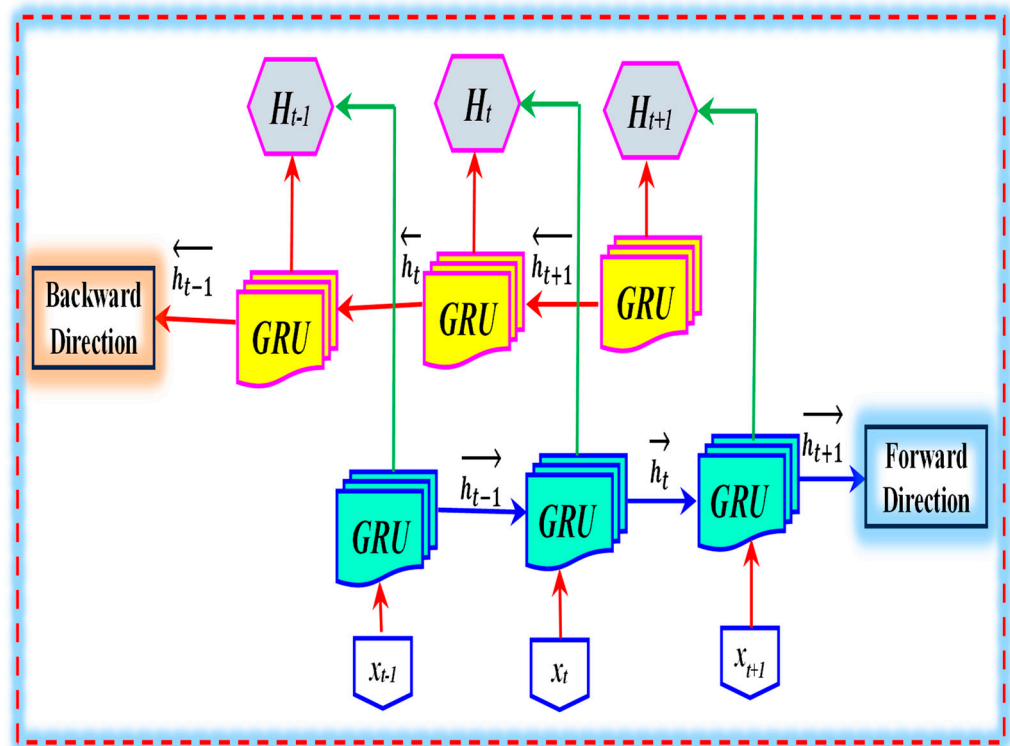
Figure 5. Bidirectional gated recurrent unit (Bi-GRU).

3.5. Convolutional Neural Network (CNN)

A convolutional neural network (CNN) is a kind of deep learning architecture mainly used for solving various kinds of time series, text, audio, and, more importantly, for handling image data [68]. The CNN works in two distinguished stages: feature extraction and classification [69]. The CNN architecture is similar to the standard artificial neural network, for which it is recognized that several layers will be needed to achieve the final task. As shown in Figure 6, there is an input layer, an output layer, and an ensemble of hidden layers. Beyond the input and output layers, the hidden layers are composed of a convolutional layer, a pooling layer, an activation layer, and finally, a fully connected layer [70]. The convolutional layers are used for extracting the probable features from the input space. They are considered critical components in the CNN model, and they work as a filter, also called kernels. The output of the convolutional layer can be activated using various activation functions, i.e., ReLU, Tanh, and sigmoid, for which a high nonlinearity can be gained. The pooling layer was introduced for improving the computational of the CNN by decreasing the number of features maps by retaining only the essential. In this layer, the max pooling is commonly used. The fully connected layer is the last step, and it is at the end of the CNN model and used for the final prediction of the model [71–73].

Figure 6. Blocks diagram of convolutional neural network (CNN)-based LSTM, BiLSTM, GRU, and Bi-GRU deep learning.

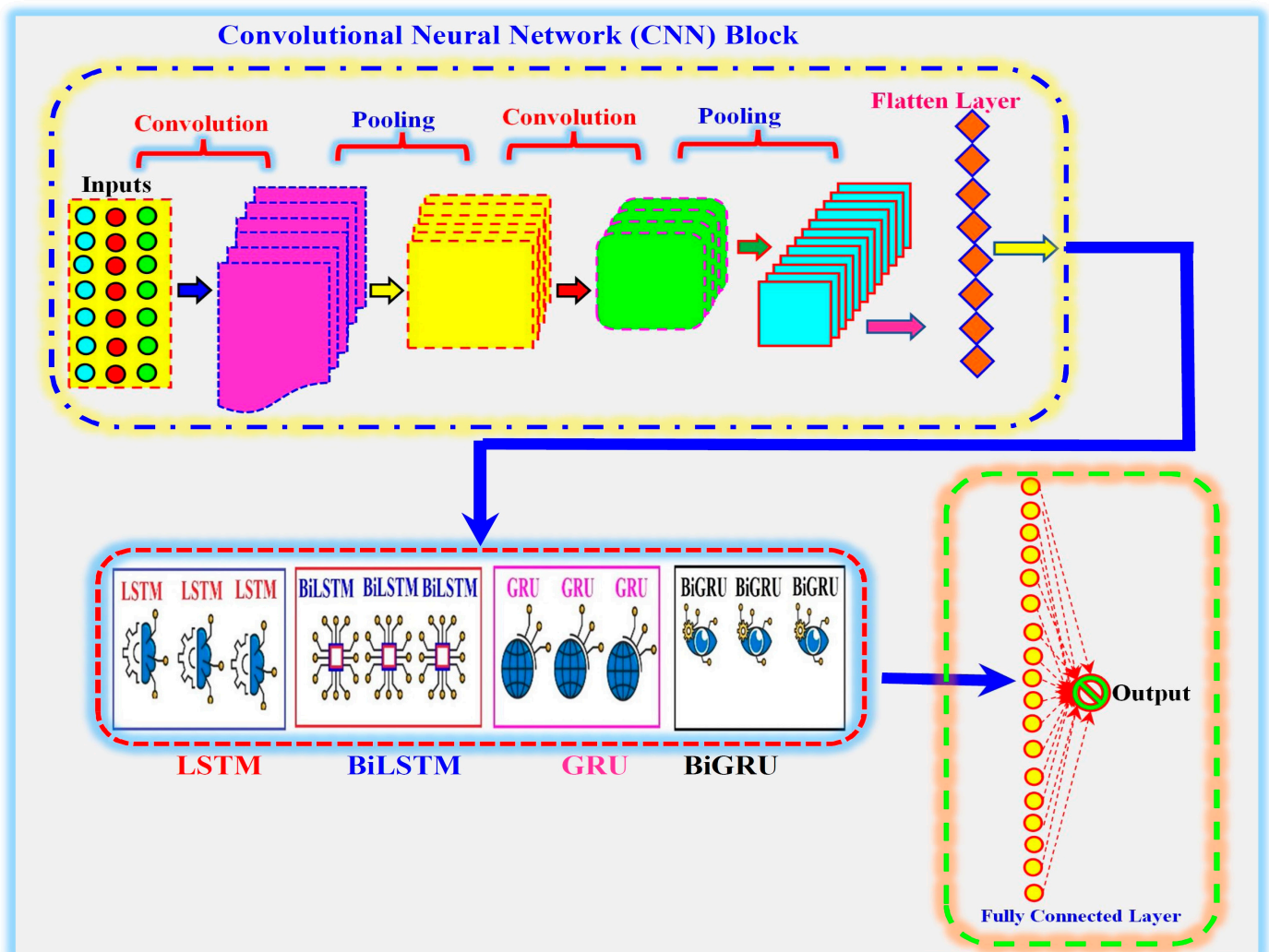*3.6. Convolutional Neural Network-LSTM (CNN-LSTM)*

In the present study, the standard CNN was combined with the LSTM model to improve its performance, and the CNN-LSTM was introduced. According to Figure 5, the CNN-LSTM possesses the same architecture as the CNN with a new layer of LSTM block between the flattened and the fully connected layer [74]. The CNN-LSTM uses the same learning algorithm for which the convolutional and the pooling layers were used for capturing the features from the input space, while the LSTM layer captures the nonlinearity and improves the prediction of the modeled variable [75]. One of the major reasons for developing the CNN-LSTM is that the CNN is a robust tool for extracting the features, but it generally fails in handling sequential data [76].

3.7. Convolutional Neural Network–BiLSTM (CNN-BiLSTM)

CNN-BiLSTM (Figure 5) is a combination of the convolutional neural network and bidirectional long short-term memory. The BiLSTM has a high capability in handling sequential data in comparison to the LSTM by using the combination of two layers, i.e., the backward and the forward hidden layers [77]. Firstly, the CNN-BiLSTM adopts a CNN in the first stage for capturing the features and improving the generalization ability of the model, while in the last stage, the BiLSTM is adopted for improving the accuracy and increasing the speed of the training algorithm [78]. Between different deep learning layers, various dropout mechanisms are integrated, which help in discarding part of the updated parameters at each learning step, which significantly helps in avoiding the overfitting problem, while the CNN significantly decreases the input space by removing the possible redundant features [78]. The overall CNN-BiLSTM process can be summarized as follows: (*i*) splitting the input signal into a one-dimensional dataset to be exploited by the convolutional layer, (*ii*) the application of Padding, Relu, max pooling, and dropout operations for reducing the overfitting problem, (*iii*) normalization and sending the data to the BiLSTM layer, and (*iv*) estimating the output of the model using the fully connected layer [78].

3.8. Convolutional Neural Network-GRU (CNN-GRU)

CNN-GRU (Figure 5) is a combination of the convolutional neural network and the gated recurrent unit. According to Figure 5, the output of the CNN is moved directly to the GRU and presented as the input space, for which the GRU extracts the features and realizes a non-linear mapping between the input variables and the modeled variable [79]. The GRU stores the information about the most relevant features provided by the CNN block; this is achieved by passing the output value of the flattened layer to the gate units for "*tracking*" the state of the sequence [80]. Finally, the importance of combining the CNN with the GRU is that the response of both CNN and GRU algorithms were "*concatenated*" to provide the final response [81,82].

3.9. Convolutional Neural Network-BiGRU (CNN-BiGRU)

CNN-BiGRU is a combination of the CNN and BiGRU. The selected predictors are taken as the input variables of the model and the convolutional layers are used for extracting the features [83]. The pooling layer is used for compressing the high number of parameters, which significantly contributes to the decrease in the data space dimension. In parallel, the dropout layer serves as a tool for handling the overfitting problem by a rational selection of the neurons in the network, taking into account a precise probability. Passing by the BiGRU layer, the filtered data are then captured, reshaped into a "*one-dimensional sequence*", and moved to the fully connected layer to provide the final response [84].

4. Performance Indicators

The accuracies of LSTM, BILSTM, CNN-LSTM, CNN-BILSTM, GRU, BIGRU, CNN-GRU, and CNN-BIGRU are compared in monthly streamflow prediction for a high altitude snow-fed catchment (Gilgit river) of Pakistan using previous values of streamflow selected

based on auto-correlation function. SCA (snow-covered area) data from MODIS MOD10CM remote sensing snow cover product and MN (month number) are also used as model inputs. The outcomes of the benchmark models are compared using the following criteria:

$$RMSE : \; Root\ Mean\ Square\ Error = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left[(Q_0)_i - (Q_C)_i\right]^2} \tag{13}$$

$$MAE : \; Mean\ Absolute\ Error = \frac{1}{N}\sum_{i=1}^{N}\left|(Q_0)_i - (Q_C)_i\right| \tag{14}$$

$$NSE : \; Nash - Suutcliffe = 1 - \frac{\sum_{i=1}^{N}\left[(Q_0)_i - (Q_C)_i\right]^2}{\sum_{i=1}^{N}\left[(Q_0)_i - \overline{Q}_0\right]^2}, \; -\infty < NSE \leq 1 \tag{15}$$

where Q_C, Q_0, $\overline{Q}_0$ are calculated, observed, and average streamflow, respectively, and N is the data quantity.

5. Results

To analyze the performance of LSTM models in streamflow prediction, we compare the results from both the training and testing stages in Table 1. In this table, Qt-1, SCA, and MN refer to the streamflow from the previous month, snow cover area, and month number, respectively. In the training stage, the RMSE values decrease as more input features are added, from 221.3 with just Qt-1 to 104.1 when all features (Qt-1, Qt-11, Qt-12, SCA, MN) are included. Similarly, RMSE values decrease from 234.6 with Qt-1 to 132.5 with all features. The RMSE values are consistently higher in the testing stage than in the training stage, indicating that the model performs better on the training data. However, the trend of improvement with additional inputs is consistent in both stages, suggesting that the model generalizes well. MAE decreases from 164.4 with Qt-1 to 57.4 with all inputs in the training stage, while the MAE of the testing stage decreases from 167.5 with Qt-1 to 74.5 with all inputs. Like RMSE, the MAE values are higher during testing, but the reduction in error with more input features remains consistent across both stages. R^2 in the training stage increases from 0.514 with Qt-1 to 0.896 with all inputs, indicating a better fit as more features are added. The R^2 of the testing stage increases from 0.476 with Qt-1 to 0.841 with all inputs. The R^2 values are slightly lower in the testing stage, reflecting a decrease in model performance on unseen data. However, the overall pattern of improvement with additional inputs is mirrored in both training and testing, showing that the model's ability to explain variance improves with more features. In the training stage, NSE increases from 0.511 with Qt-1 to 0.891 with all inputs, indicating better predictive power. The NSE of the testing stage increases from 0.465 with Qt-1 to 0.838 with all inputs. The NSE values are also lower during testing, similar to R^2, but the upward trend with additional inputs remains. This suggests that while the model is not overfitting, there is a slight drop in performance on testing data. The models consistently improve performance metrics (lower RMSE and MAE, higher R^2 and NSE) as more input features are included in the training and testing phases. This indicates that the additional input variables (SCA, MN) provide valuable information for predicting streamflow. There is a noticeable difference between training and testing results, with training performance being better. This is typical in machine learning, where the model performs slightly better on the data it was trained on. However, the consistency in trends across both datasets suggests that the model generalizes well to unseen data although there is room for improvement.

Table 2 compares the performance of bidirectional long short-term memory (BILSTM) models during both the training and testing stages. The models consistently improve predictive accuracy as more features are added. Training RMSE drops from 217.8 with only Qt-1 to 102.8, while testing RMSE decreases from 225.9 to 125.6. MAE and NSE metrics follow a similar pattern, with MAE decreasing from 153.2 to 54.07 in training and from 161.5 to 70.62 in testing, and NSE increasing from 0.526 to 0.901 in training and from 0.503 to 0.853

in testing. R^2 improves from 0.529 to 0.908 in training and from 0.509 to 0.859 in testing. Including SCA and MN consistently enhances model performance, indicating these features help capture seasonal variations and improve prediction accuracy, even with slightly better results on training data. The performance of CNN-LSTM models during both the training and testing stages of streamflow prediction is compared in Table 3. RMSE decreases from 214.6 to 91.4 in training and from 219.6 to 119.5 in testing as more features are included. MAE follows this trend, dropping from 151.3 to 51.82 in training and from 157.8 to 68.15 in testing. R^2 and NSE also show improvements, with R^2 increasing from 0.549 to 0.926 in training, from 0.526 to 0.87 in testing, and NSE improving from 0.544 to 0.918 in training and from 0.514 to 0.862 in testing. The consistent improvement across all metrics with the inclusion of SCA and MN demonstrates the model's effectiveness in capturing both seasonal and temporal data despite slightly higher testing errors than training.

Table 1. Training and test statistics of the models for streamflow prediction—LSTM.

Model Inputs	Training Period				Test Period			
	RMSE	**MAE**	R^2	**NSE**	**RMSE**	**MAE**	R^2	**NSE**
Qt-1	221.3	164.4	0.514	0.511	234.6	167.5	0.476	0.465
Qt-1, Qt-11	138.3	85.32	0.814	0.805	141.6	87.6	0.807	0.801
Qt-1, Qt-11, Qt-12	113.4	64.8	0.865	0.858	139.3	81.7	0.819	0.815
Qt-1, Qt-11, Qt-12, SCA	107.3	59.6	0.887	0.881	134.4	78.3	0.826	0.819
Qt-1, Qt-11, Qt-12, MN	105.2	58.5	0.892	0.886	133.8	75.76	0.837	0.831
Qt-1, Qt-11, Qt-12, SCA, MN	104.1	57.4	0.896	0.891	132.5	74.5	0.841	0.838
Mean	131.6	81.67	0.811	0.805	152.7	94.227	0.768	0.762

Table 2. Training and test statistics of the models for streamflow prediction—BILSTM.

Model Inputs	Training Period				Test Period			
	RMSE	**MAE**	R^2	**NSE**	**RMSE**	**MAE**	R^2	**NSE**
Qt-1	217.8	153.2	0.529	0.526	225.9	161.5	0.509	0.503
Qt-1, Qt-11	123.6	72.7	0.851	0.845	133.8	81.6	0.835	0.827
Qt-1, Qt-11, Qt-12	106.5	58.6	0.886	0.881	131.6	74.5	0.839	0.831
Qt-1, Qt-11, Qt-12, SCA	105.3	55.32	0.895	0.891	128.5	72.3	0.845	0.837
Qt-1, Qt-11, Qt-12, MN	103.7	54.61	0.902	0.895	127.2	71.6	0.849	0.843
Qt-1, Qt-11, Qt-12, SCA, MN	102.8	54.07	0.908	0.901	125.6	70.62	0.859	0.853
Mean	126.617	74.750	0.829	0.823	145.433	88.687	0.789	0.782

Table 4 summarizes the performance of CNN-BiLSTM models during the training and testing stages. The models show significant improvement in accuracy with more features. Training RMSE decreases from 209.7 to 75.8, while testing RMSE drops from 215.7 to 101.6. MAE and NSE improvements are also noted, with training MAE reducing from 147.3 to 42.3 and testing MAE from 156.2 to 54.9. R^2 increases from 0.565 to 0.952 in training and from 0.543 to 0.905 in testing. Including SCA and MN results in consistent performance improvements across all metrics, suggesting these features effectively capture temporal and seasonal variations, leading to better predictions. Table 5 illustrates the outcomes of GRU (gated recurrent unit) models during both the training and testing stages. The models benefit from additional input features, with training RMSE decreasing from 218.3 to 102.7 and testing RMSE from 226.4 to 127.7. MAE shows similar reductions, from 154.42 to 55.03 in training and 162.3 to 74.6 in testing. R^2 and NSE improvements are also observed, with R^2 increasing from 0.527 to 0.904 in training and from 0.507 to 0.850 in testing, and NSE

from 0.524 to 0.897 in training, from 0.501 to 0.844 in testing. The consistent reduction in errors with the inclusion of SCA and MN highlights their importance in enhancing model performance by capturing seasonal variations.

Table 3. Training and test statistics of the models for streamflow prediction—CNN-LSTM.

Model Inputs	Training Period				Test Period			
	RMSE	MAE	R^2	NSE	RMSE	MAE	R^2	NSE
Qt-1	214.6	151.3	0.549	0.544	219.6	157.8	0.526	0.514
Qt-1, Qt-11	111.5	61.4	0.882	0.877	128.2	76.6	0.851	0.846
Qt-1, Qt-11, Qt-12	99.8	55.8	0.908	0.901	125.4	72.6	0.848	0.841
Qt-1, Qt-11, Qt-12, SCA	97.6	54.43	0.911	0.906	124.6	70.831	0.851	0.845
Qt-1, Qt-11, Qt-12, MN	93.3	52.61	0.917	0.914	122.6	69.2	0.861	0.855
Qt-1, Qt-11, Qt-12, SCA, MN	91.4	51.82	0.926	0.918	119.5	68.15	0.87	0.862
Mean	118.033	71.227	0.849	0.843	139.983	85.864	0.801	0.794

Table 4. Training and test statistics of the models for streamflow prediction—CNN-BILSTM.

Model Inputs	Training Period				Test Period			
	RMSE	MAE	R^2	NSE	RMSE	MAE	R^2	NSE
Qt-1	209.7	147.3	0.565	0.561	215.7	156.2	0.543	0.539
Qt-1, Qt-11	91.2	50.7	0.911	0.906	113.7	68.8	0.868	0.862
Qt-1, Qt-11, Qt-12	86.7	46.7	0.926	0.921	110.2	65.61	0.878	0.873
Qt-1, Qt-11, Qt-12, SCA	82.63	45.7	0.938	0.933	108.3	61.5	0.888	0.883
Qt-1, Qt-11, Qt-12, MN	76.5	44.6	0.946	0.942	104.8	57.2	0.891	0.886
Qt-1, Qt-11, Qt-12, SCA, MN	75.8	42.3	0.952	0.947	101.6	54.9	0.905	0.901
Mean	103.755	62.883	0.873	0.868	125.717	77.368	0.829	0.824

Table 5. Training and test statistics of the models for streamflow prediction—GRU.

Model Inputs	Training Period				Test Period			
	RMSE	MAE	R^2	NSE	RMSE	MAE	R^2	NSE
Qt-1	218.3	154.42	0.527	0.524	226.4	162.3	0.507	0.501
Qt-1, Qt-11	133.7	82.7	0.832	0.821	136.8	84.6	0.825	0.821
Qt-1, Qt-11, Qt-12	109.9	59.7	0.873	0.867	134.3	77.6	0.826	0.821
Qt-1, Qt-11, Qt-12, SCA	106.6	56.5	0.886	0.881	130.2	76.7	0.836	0.832
Qt-1, Qt-11, Qt-12, MN	104.2	55.63	0.898	0.893	128.2	75.1	0.844	0.839
Qt-1, Qt-11, Qt-12, SCA, MN	102.7	55.03	0.904	0.897	127.7	74.6	0.85	0.844
Mean	129.233	77.330	0.820	0.814	147.267	91.817	0.781	0.776

The performance of BiGRU (bidirectional gated recurrent unit) models during both the training and testing stages is reported in Table 6. The models show improved accuracy with added features, with training RMSE decreasing from 215.5 to 97.62 and testing RMSE from 221.3 to 122.8. MAE follows the same trend, decreasing from 152.6 to 52.45 in training and from 159.2 to 69.6 in testing. R^2 and NSE values increase, indicating better model fit, with R^2 improving from 0.545 to 0.917 in training and from 0.516 to 0.862 in testing, and NSE from 0.542 to 0.912 in training and from 0.509 to 0.856 in testing. Including SCA and

MN enhances model performance across all metrics, reflecting the value of incorporating both temporal and seasonal data for accurate streamflow predictions. Table 7 reports the performance of CNN-GRU (convolutional neural network–gated recurrent unit) models in streamflow prediction during the training and testing stages. The models demonstrate significant accuracy gains with additional features. Training RMSE decreases from 212.7 to 80.24, while testing RMSE reduces from 217.3 to 111.2. MAE decreases from 150.8 to 47.03 in training and 157.4 to 66.32 in testing. R^2 and NSE also improve, with R^2 increasing from 0.553 to 0.942 in training, from 0.532 to 0.884 in testing, from 0.551 to 0.936 in training, and from 0.529 to 0.881 in testing. The consistent improvement across all metrics, including SCA and MN, highlights the model's ability to generalize well and accurately predict streamflow variations. Table 8 performs CNN-BiGRU (convolutional neural network–bidirectional gated recurrent unit) models in streamflow prediction during the training and testing stages. The models show substantial improvements in performance with more features. Training RMSE decreases from 207.8 to 71.6, and testing RMSE drops from 213.6 to 95.7. MAE, R^2, and NSE metrics also show consistent improvement, with MAE decreasing from 144.7 to 39.62 in training and from 155.2 to 50.7 in testing, R^2 improving from 0.578 to 0.962 in training and from 0.558 to 0.929 in testing, and NSE increasing from 0.574 to 0.957 in training and from 0.553 to 0.921 in testing. Including SCA and MN significantly enhances model performance across all metrics, demonstrating their importance in capturing the temporal and seasonal dynamics of streamflow.

Table 6. Training and test statistics of the models for streamflow prediction—BIGRU.

Model Inputs	Training Period				Test Period			
	RMSE	MAE	R^2	NSE	RMSE	MAE	R^2	NSE
Qt-1	215.5	152.6	0.545	0.542	221.3	159.2	0.516	0.509
Qt-1, Qt-11	118.5	65.3	0.869	0.861	131.5	79.6	0.842	0.835
Qt-1, Qt-11, Qt-12	104.3	57.4	0.892	0.886	128.3	73.2	0.844	0.837
Qt-1, Qt-11, Qt-12, SCA	101.6	55.62	0.905	0.901	127.1	71.7	0.849	0.843
Qt-1, Qt-11, Qt-12, MN	98.34	53.64	0.911	0.905	125.2	70.21	0.857	0.851
Qt-1, Qt-11, Qt-12, SCA, MN	97.62	52.45	0.917	0.912	122.8	69.6	0.862	0.856
Mean	122.643	72.835	0.840	0.835	142.700	87.252	0.795	0.789

Table 7. Training and test statistics of the models for streamflow prediction—CNN-GRU.

Model Inputs	Training Period				Test Period			
	RMSE	MAE	R^2	NSE	RMSE	MAE	R^2	NSE
Qt-1	212.7	150.8	0.553	0.551	217.3	157.4	0.532	0.529
Qt-1, Qt-11	105.3	57.32	0.891	0.886	123.6	73.21	0.854	0.849
Qt-1, Qt-11, Qt-12	95.27	55.62	0.901	0.895	119.8	71.81	0.859	0.852
Qt-1, Qt-11, Qt-12, SCA	85.85	52.33	0.921	0.918	116.9	69.5	0.862	0.855
Qt-1, Qt-11, Qt-12, MN	83.25	50.41	0.932	0.927	114.5	68.91	0.875	0.867
Qt-1, Qt-11, Qt-12, SCA, MN	80.24	47.03	0.942	0.936	111.2	66.32	0.884	0.881
Mean	110.435	68.918	0.857	0.852	133.883	84.525	0.811	0.806

All models show significant RMSE reductions as more input features are added in the training stage (Tables 1–8). CNN-based models, particularly CNN-BiGRU, achieve the lowest RMSE, indicating superior training data fitting accuracy. The CNN-BiGRU also reduces MAE during training, followed closely by CNN-LSTM and CNN-GRU, showing these models' precision in aligning predictions with observed values. The CNN-BiGRU

and CNN-GRU achieve the highest R^2 and NSE values during training, indicating robust model fits and reliable predictions. The CNN-BiGRU maintains the lowest RMSE and MAE in testing, demonstrating excellent generalization to unseen data. CNN-GRU also performs well, showing consistency in both training and testing. These models (CNN-BiGRU and CNN-GRU) also achieve high R^2 and NSE in testing, suggesting they effectively capture the variance and dynamics of streamflow, even with unseen data. The LSTM and BiLSTM models show solid improvements with added features, particularly in training, but generally perform slightly lower than CNN-integrated models in testing. They offer good baseline performance with the advantage of simpler architectures compared to CNN-based models. GRU models perform comparably to LSTM models, with BiGRU showing slightly better results due to its bidirectional nature. These models effectively handle temporal sequences, making them reliable for streamflow prediction but slightly behind CNN-based models in testing performance. These models benefit from combining CNN's spatial feature extraction and LSTM/GRU's temporal handling. They consistently perform well in both stages, with CNN-GRU slightly outperforming CNN-LSTM in testing, indicating robust generalization and strong predictive power. The CNN-BiGRU model stands out across all metrics in both stages. It combines the strengths of CNN for spatial pattern recognition and BiGRU for capturing temporal dynamics from both directions, leading to the best overall performance in accuracy, precision, and generalization. The CNN-BiGRU and CNN-GRU are the top performers across both training and testing stages, excelling in RMSE, MAE, R^2, and NSE metrics. These models effectively balance spatial and temporal data processing, achieving high predictive accuracy and reliability. For applications requiring high accuracy and generalization, CNN-BiGRU is recommended. For a balance between complexity and performance, CNN-GRU offers strong predictive capabilities with slightly simpler architecture. While simpler models like LSTM, BiLSTM, and GRU are effective and easier to implement, including CNN layers significantly enhances performance, especially in more complex temporal-spatial scenarios like streamflow prediction. The bidirectional capabilities of BiGRU further boosts model performance, making CNN-BiGRU the most robust and reliable option among the models evaluated.

Table 8. Training and test statistics of the models for streamflow prediction—CNN-BIGRU.

Model Inputs	Training Period				Test Period			
	RMSE	MAE	R^2	NSE	RMSE	MAE	R^2	NSE
Qt-1	207.8	144.7	0.578	0.574	213.6	155.2	0.558	0.553
Qt-1, Qt-11	87.2	48.3	0.922	0.915	114.3	64.4	0.875	0.871
Qt-1, Qt-11, Qt-12	83.6	46.8	0.931	0.925	108.6	61.7	0.881	0.873
Qt-1, Qt-11, Qt-12, SCA	76.7	42.7	0.945	0.939	105.2	58.6	0.894	0.887
Qt-1, Qt-11, Qt-12, MN	73.62	40.74	0.956	0.952	101.2	53.6	0.908	0.901
Qt-1, Qt-11, Qt-12, SCA, MN	71.6	39.62	0.962	0.957	95.7	50.7	0.929	0.921
Mean	100.087	60.477	0.882	0.877	123.100	74.033	0.841	0.834

For a comparison of the model's prediction performances, a graphical comparison is also performed. For this purpose, scatter plots, Taylor diagrams, and violin plots are utilized (Figures 7–9). The scatterplots (Figure 7) demonstrate that hybrid models incorporating CNN layers, particularly CNN-BiGRU and CNN-BiLSTM, significantly outperform standalone LSTM, GRU, and BiGRU models. CNN-BiGRU, in particular, shows the highest R^2 value, indicating its strong predictive power and ability to capture complex streamflow dynamics accurately. The Taylor diagram (Figure 8) shows that CNN-BiGRU outperforms other models in correlation, variability, and RMSE, followed closely by CNN-BiLSTM. This diagram confirms that hybrid models incorporating CNN and bidirectional recurrent layers (GRU or LSTM) offer superior performance for streamflow prediction. The other models,

such as LSTM, GRU, and BiLSTM, perform moderately well but fall short of the hybrid CNN models in accurately capturing the observed streamflow characteristics. The violin plots (Figure 9) highlight that hybrid models with CNN layers, particularly CNN-BiGRU and CNN-BiLSTM, provide the most accurate representation of observed streamflow distribution. CNN-BiGRU has the best alignment with the observed distribution, capturing both the variability and extreme values effectively. This visualization reinforces the findings from previous analyses, suggesting that CNN-BiGRU is the most robust model for accurately predicting streamflow, followed closely by CNN-BiLSTM. In contrast, standalone GRU and LSTM models show limitations in capturing the full range of streamflow values, especially for extreme flows.
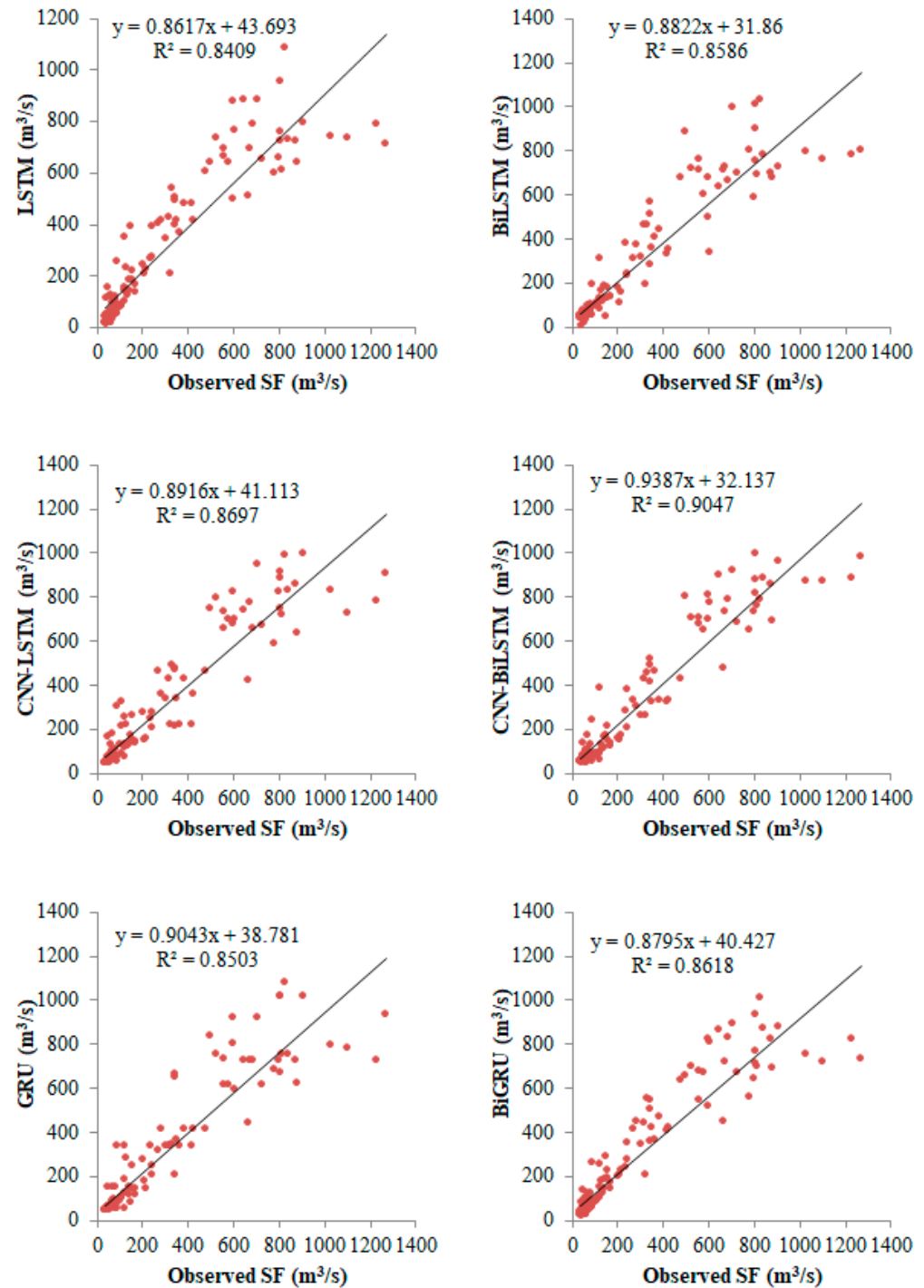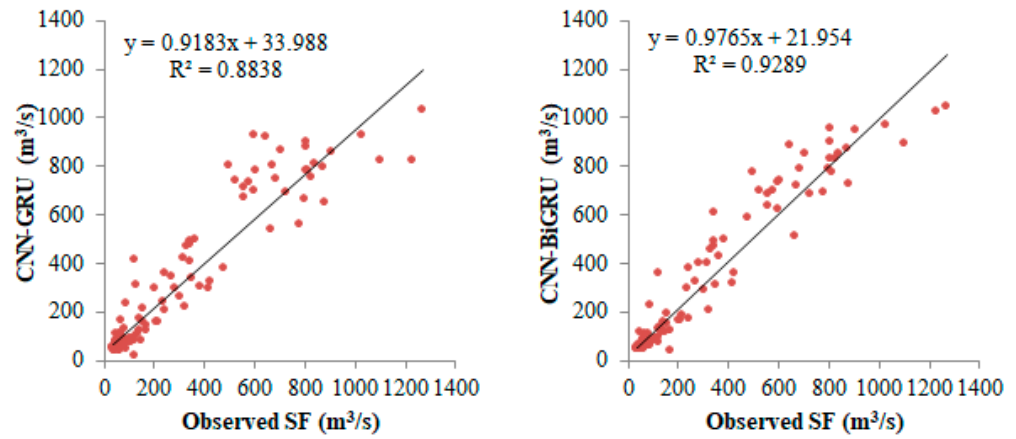

Figure 7. *Cont.*

Figure 7. Scatterplots of the observed and predicted streamflow by different models in the test period using the best input combination.
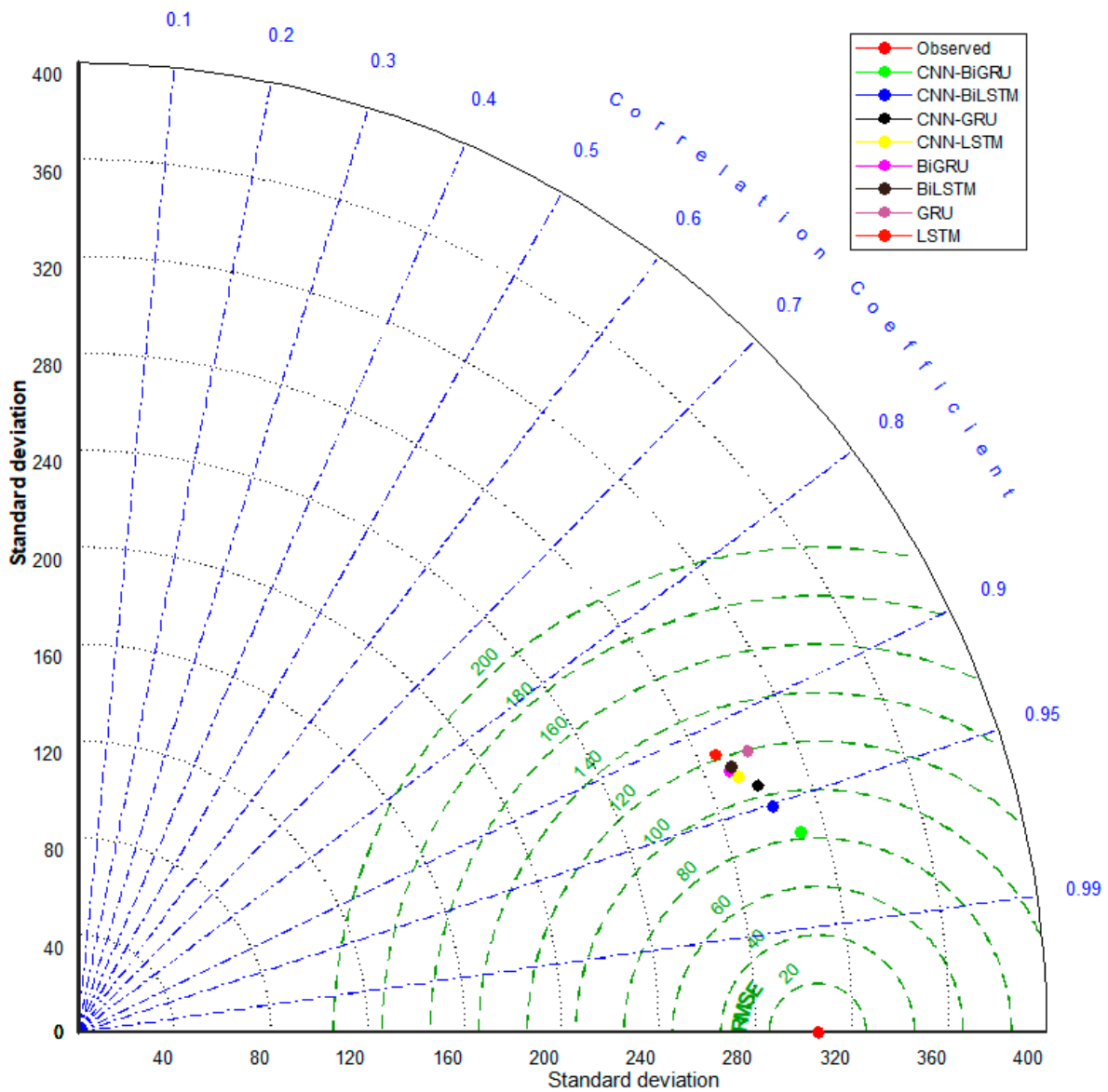

Figure 8. Taylor diagrams of the predicted streamflow by different models in the test period using the best input combination.
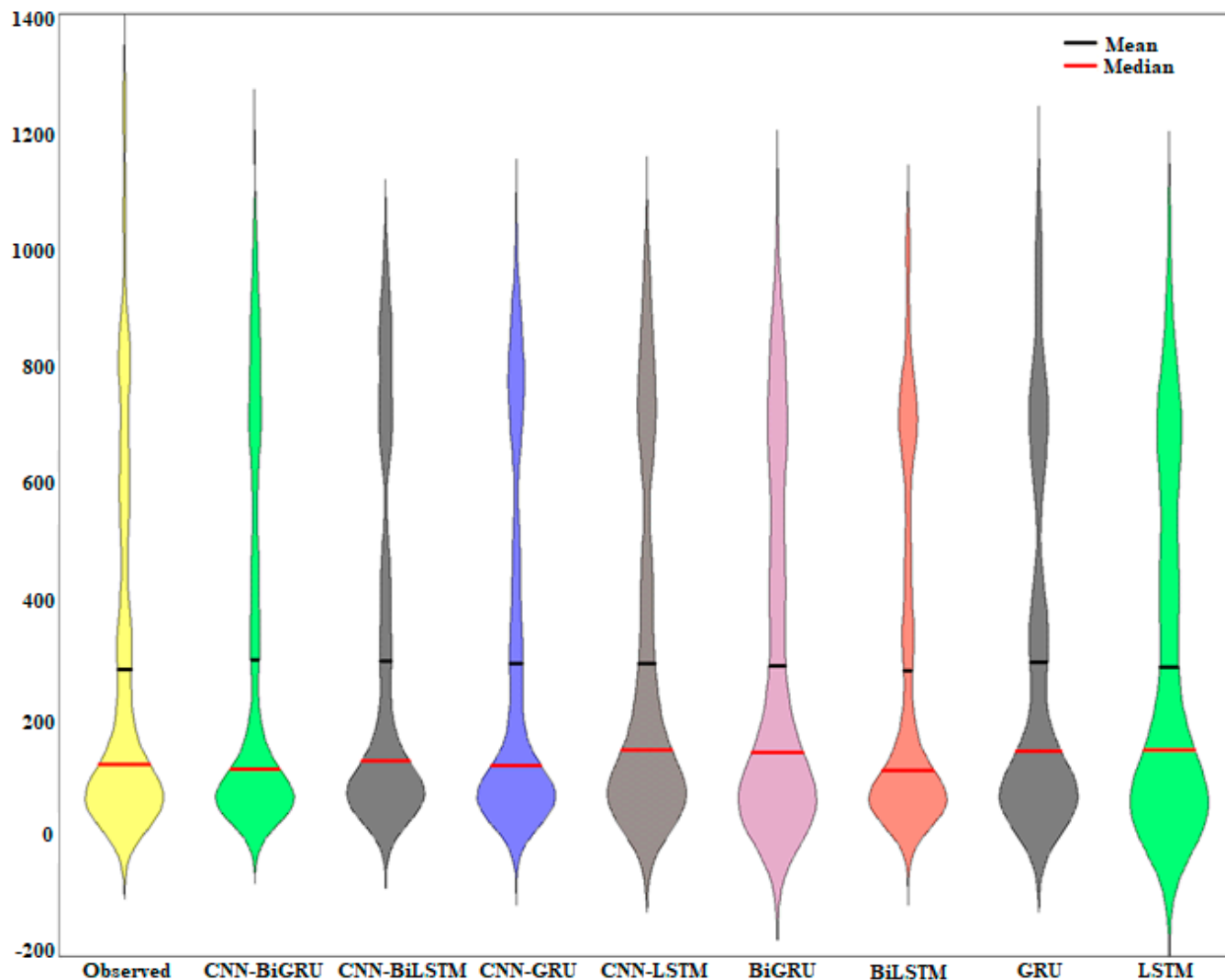
Figure 9. Violin charts of the predicted streamflow by different models in the test period using the best input combination.

Table 9 compares the performance of various models (LSTM, BiLSTM, CNN-LSTM, CNN-BiLSTM, GRU, BiGRU, CNN-GRU, and CNN-BiGRU) in predicting peak streamflow values during the testing stage. The key metric for comparison is the relative prediction error percentage for dates with observed peak streamflow greater than 807. CNN-BiGRU has the lowest absolute error (108.4), indicating the most accurate peak streamflow predictions. CNN-BiLSTM follows with an absolute error of 144.1, showing strong predictive accuracy. CNN-GRU and CNN-LSTM also perform well, with absolute errors of 157.1 and 211.9, respectively. LSTM and GRU models have higher absolute errors, with LSTM showing the highest error at 284.8. The performance varies across dates, but CNN-BiGRU consistently shows lower prediction errors compared to other models, especially in later years (e.g., 2010–2015). CNN-LSTM and CNN-BiLSTM models also perform well but show higher errors in some cases, particularly in earlier years (e.g., 2006–2007). Non-CNN models like LSTM and BiLSTM show higher errors, particularly in cases where observed values are higher. The trend shows that models incorporating CNN layers (especially in combination with BiGRU) tend to outperform others, suggesting that the convolutional layers help capture more intricate patterns in peak streamflow data. CNN-BiGRU is the most effective model for predicting peak streamflow, with the lowest overall prediction error, indicating its strength in handling complex patterns in peak data. CNN-BiLSTM and CNN-GRU also perform well, showing that models combining CNN with either BiLSTM or GRU

can capture and predict peak streamflow more accurately than traditional LSTM or GRU models. LSTM and GRU models, while still useful, show higher errors in peak streamflow prediction, indicating potential limitations in capturing complex temporal patterns without the added CNN layers. The analysis suggests that for peak streamflow prediction, models that integrate CNN with BiGRU or BiLSTM provide the most accurate results.

Table 9. The comparison of different models in peak streamflow prediction for the test period.

Date	Observed Values	Relative Prediction Error Percentage							
	Peaks > 807	LSTM %	BILSTM %	CNN-LSTM %	CNN-BILSTM %	GRU %	BIGRU %	CNN-GRU %	CNN-BIGRU %
7/2006	829.5	−31.1	−24.1	−18.9	4.8	−30.4	−22.1	9.3	0.3
8/2006	872.2	17.2	20.2	2.0	1.8	16.4	5.8	8.7	−0.3
7/2007	905.4	12.0	19.5	−9.7	−6.5	−12.4	3.3	5.6	−4.6
6/2008	810.3	10.5	6.9	8.1	−1.0	17.0	12.0	3.7	−2.2
7/2009	1031.6	28.4	23.0	19.6	15.8	22.6	27.2	10.2	9.8
8/2009	882.9	27.5	23.0	28.1	21.3	29.4	22.0	26.0	17.8
7/2010	842.2	13.3	7.3	1.6	−5.1	10.7	−3.4	3.6	−1.3
8/2010	1233.3	36.3	36.6	36.4	28.2	40.9	33.0	33.0	16.8
8/2013	1102.9	33.3	31.3	34.0	21.2	29.1	34.6	25.4	19.2
7/2014	807.5	6.2	−11.4	−13.0	−9.0	−26.0	5.3	−8.9	−11.3
7/2015	1270.3	44.0	36.6	28.9	22.8	26.3	42.4	19.0	20.0
8/2015	813.5	25.1	15.1	11.6	6.4	7.0	14.2	3.6	4.7

Table 10 presents a seasonal analysis of the streamflow prediction models' performance, comparing different metrics across winter, spring, summer, and autumn. the models include CNN-BiLSTM and CNN-BiGRU, with additional variations incorporating snow-covered area (SCA) data as an input feature. In winter, The CNN-BiGRU model with SCA data performs best, with the lowest RMSE (4.268 in training and 4.417 in testing) and highest R^2 (0.763 in training and 0.742 in testing). This indicates that the inclusion of SCA data helps capture wintertime streamflow patterns, which are heavily influenced by snow conditions. Spring has the highest overall R^2 values, with CNN-BiGRU + SCA again showing the best performance (R^2 of 0.941 in training and 0.937 in testing). The MAE and RMSE values also indicate high accuracy, with CNN-BiGRU + SCA achieving RMSE values of 48.61 in training and 45.64 in testing. This superior performance can be attributed to spring meltwater dynamics, which the models capture more accurately with SCA data integration. Summer shows a slight drop in R^2 and NSE scores compared to spring, likely due to increased variability in streamflow from rapid glacial melt and monsoon impacts. The CNN-BiGRU + SCA model again performs best, with an RMSE of 114.1 in training and 118.9 in testing, indicating it effectively captures the complex summer flow patterns when SCA data are included. The models maintain high accuracy into autumn, with CNN-BiGRU + SCA achieving the best results across all metrics. This model's RMSE values are 40.19 in training and 37.84 in testing, and R^2 reaches 0.914 in training and 0.908 in testing, suggesting a reliable fit. The strong autumn performance may reflect lower seasonal variability in streamflow as snowmelt slows, allowing the model to achieve stable predictions. The integration of SCA data consistently enhances model accuracy across seasons, especially with CNN-BiGRU. The seasonal performance variation indicates that snow cover plays a critical role in predicting streamflow in the Upper Indus Basin, as evidenced by improved RMSE, MAE, R^2, and NSE values with SCA. The CNN-BiGRU model with SCA integration emerges as the most effective model for seasonal streamflow prediction, reflecting its ability to generalize across seasonal patterns with high reliability.

Table 10. Training and test statistics of the best models for streamflow prediction during different seasons.

Season	Model	Training Period				Test Period			
		RMSE	MAE	R^2	NSE	RMSE	MAE	R^2	NSE
Winter	CNN-BiLSTM	5.109	4.022	0.752	0.748	5.263	4.249	0.731	0.728
	CNN-BiGRU	4.942	3.779	0.758	0.755	5.062	3.864	0.735	0.731
	CNN-BiLSTM + SCA	4.626	3.587	0.758	0.753	4.785	3.692	0.738	0.735
	CNN-BiGRU + SCA	4.268	3.184	0.763	0.761	4.417	3.342	0.742	0.739
Spring	CNN-BiLSTM	56.37	42.09	0.929	0.926	51.18	39.85	0.921	0.920
	CNN-BiGRU	53.82	40.72	0.937	0.935	50.49	36.72	0.934	0.932
	CNN-BiLSTM + SCA	51.62	39.08	0.934	0.928	48.64	35.42	0.928	0.925
	CNN-BiGRU + SCA	48.61	37.47	0.941	0.940	45.64	34.71	0.937	0.935
Summer	CNN-BiLSTM	134.1	93.37	0.704	0.702	139.8	96.48	0.702	0.701
	CNN-BiGRU	125.3	86.71	0.719	0.714	122.6	88.34	0.714	0.711
	CNN-BiLSTM + SCA	119.8	81.62	0.725	0.722	121.6	84.71	0.722	0.721
	CNN-BiGRU + SCA	114.1	72.08	0.738	0.735	118.9	75.91	0.733	0.731
Autumn	CNN-BiLSTM	47.59	30.62	0.901	0.898	45.28	27.61	0.896	0.892
	CNN-BiGRU	44.27	28.64	0.903	0.901	40.37	24.71	0.901	0.900
	CNN-BiLSTM + SCA	43.68	25.84	0.908	0.905	41.94	22.82	0.904	0.901
	CNN-BiGRU + SCA	40.19	22.73	0.914	0.911	37.84	20.67	0.908	0.905

6. Discussion

In this study, LSTM was applied as a baseline model for streamflow prediction. The results indicated that while LSTM performed reasonably well, its accuracy was surpassed by more complex models like CNN-BiGRU and CNN-BiLSTM, particularly in peak streamflow predictions. LSTM models are widely recognized in hydrological studies for their ability to capture long-term dependencies in time-series data. Studies like Nakhaei et al. [23] have shown that LSTM models significantly outperform traditional models, particularly in capturing the non-linear and temporal dependencies of streamflow data. However, these models can be further enhanced when combined with other deep learning techniques, such as CNN, to improve feature extraction and overall prediction accuracy. BiLSTM models have been noted for their superior performance in various hydrological modeling tasks compared to unidirectional LSTM. Abdoulhalik and Ahmed [24] highlighted that BiLSTM can better capture the bidirectional dependencies in streamflow data, making it more suitable for complex hydrological forecasting. However, consistent with our study, the literature suggests that BiLSTM's performance can be further improved by integrating CNN or other deep learning architectures to enhance spatial feature extraction. GRU models are known for their ability to achieve similar performance to LSTM models but with fewer parameters and faster training times. Wegayehu et al. [26] and Vatanchi et al. [25] have demonstrated that GRU is effective for streamflow prediction, particularly in scenarios with limited data. However, similar to our findings, GRU often benefits from hybridization with CNN to improve accuracy. The use of CNN in hydrological modeling, especially when combined with RNN architectures like LSTM or GRU, has been shown to enhance model performance by effectively extracting spatial patterns from input data [27,31]. Studies such as those by Wu et al. [30] and Maiti et al. [29] support our findings, demonstrating that hybrid models like CNN-LSTM excel at handling complex, high-resolution datasets, thereby improving both accuracy and robustness in streamflow prediction. Similarly, research by Hassan and Hassan [85] and others highlights the critical role of incorporating remotely sensed data, such as SCA, into hydrological models to better represent snowmelt dynamics

and enhance streamflow predictions. The use of MODIS-derived SCA data in our study aligns with these findings and underscores its importance in improving model performance, particularly in snow-fed river basins. Our results are consistent with existing literature, which shows that hybrid models combining CNN with LSTM, BiLSTM, GRU, and BiGRU outperform traditional and standalone deep learning approaches for streamflow prediction. This advantage is particularly evident in their ability to capture complex temporal and spatial patterns, as well as peak streamflow events.

Table 10 reveals distinct seasonal patterns in model performance, highlighting the critical role of snow cover and seasonal temperature shifts in predicting streamflow in snow-dominated basins like the Upper Indus Basin (UIB). The CNN-BiGRU model with SCA data consistently achieved the best performance across all seasons, demonstrating its capability to capture complex seasonal variations. Notably, model accuracy was highest in spring, as this period represents a steady increase in snowmelt-driven flow, which the model successfully predicts. Winter accuracy, although slightly lower, remains robust due to the model's integration of SCA data, which helps capture low-flow dynamics. The summer season presented the greatest challenge due to heightened variability from glacial melt and monsoon influence, leading to higher RMSE values. Nonetheless, the CNN-BiGRU model effectively generalized across this variability, maintaining prediction reliability. This seasonal analysis underscores the added value of incorporating SCA data, particularly in snow-fed systems, and highlights the importance of model selection based on seasonal characteristics.

7. Conclusions

This study conducted a comprehensive evaluation of advanced deep learning models for streamflow prediction in the Upper Indus Basin (UIB), focusing on their ability to accurately forecast monthly and peak streamflow. The models applied included LSTM, BiLSTM, GRU, CNN, and their hybrid combinations (CNN-LSTM, CNN-BiLSTM, CNN-GRU, and CNN-BiGRU). A novel aspect of this research was the integration of MODIS-derived snow-covered area (SCA) data, which provided critical information on snowmelt dynamics, a significant contributor to streamflow in the UIB. The hybrid models, particularly CNN-BiGRU and CNN-BiLSTM, demonstrated superior performance over traditional models like LSTM and GRU. For instance, CNN-BiGRU achieved the lowest RMSE (71.6 in training and 95.7 in testing), MAE (39.62 in training and 50.7 in testing), and the highest R^2 (0.962 in training and 0.929 in testing) and NSE (0.957 in training and 0.921 in testing). This significant improvement highlights the efficacy of hybrid models in capturing complex temporal and spatial patterns in streamflow data.

The integration of SCA data from MODIS was found to enhance model accuracy substantially. For example, when SCA data were included, the CNN-BiLSTM model's RMSE improved from 83.6 to 71.6 during training and from 108.6 to 95.7 during testing. This indicates that SCA data are a crucial factor in improving the predictive capability of deep learning models in snow-dominated basins like the UIB. In peak streamflow prediction, CNN-BiGRU outperformed other models with the lowest absolute error (108.4), followed by CNN-BiLSTM (144.1). This outcome is significant as it demonstrates the model's ability to predict extreme events accurately, which is critical for flood forecasting and water resource management. The findings are consistent with the broader literature, where hybrid models integrating CNN with RNN architectures (like LSTM, BiLSTM, and GRU) are shown to outperform traditional models in hydrological forecasting tasks. This study's results reinforce the notion that combining CNN's spatial feature extraction capabilities with the temporal dependencies captured by LSTM or GRU significantly enhances model accuracy.

The study's results underscore the importance of using hybrid deep learning models for hydrological forecasting in regions like the UIB, where snow and glacier melt significantly influence streamflow. By accurately capturing both the temporal dynamics of streamflow and the spatial characteristics of snow cover, these models provide a robust framework for water resource management. The quantitative outcomes of this study sug-

gest that hybrid models could be particularly valuable for operational forecasting in similar snow-dominated basins globally. The demonstrated improvements in prediction accuracy, especially for extreme events, highlight the potential for these models to support more informed decision-making in flood risk management and water allocation. In conclusion, the integration of advanced deep learning techniques, particularly hybrid models like CNN-BiGRU and CNN-BiLSTM, with remotely sensed data like SCA offers a powerful approach for streamflow prediction. The quantitative improvements observed in this study—such as the reduction in RMSE and MAE and the increase in R^2 and NSE—demonstrate the significant potential of these models to enhance the accuracy and reliability of hydrological forecasting, ultimately contributing to better water resource management and planning in the UIB and similar regions.

The seasonal analysis findings underscore the utility of hybrid models, particularly CNN-BiGRU with SCA integration, in accurately predicting streamflow across distinct seasonal conditions. The ability of these models to handle complex temporal and spatial patterns was evident in their performance, particularly in the spring and autumn seasons. Although summer predictions were slightly less accurate due to increased flow variability, the CNN-BiGRU model demonstrated strong generalization capabilities. The seasonal breakdown highlights the importance of integrating snow cover data and supports the adoption of seasonally responsive models for hydrological forecasting in snow-dominated basins.

The accuracy of the models heavily depends on the quality and resolution of the input data. While MODIS provides valuable SCA data, the study could be limited by the availability of high-resolution, continuous datasets for other meteorological and hydrological variables. The models were trained and tested specifically for the UIB, which has unique hydrological characteristics. The results may not be directly applicable to other regions with different climatic and hydrological conditions without further calibration and validation. To improve model robustness and generalization, future studies should incorporate more diverse and higher-resolution datasets, including additional remote sensing products and in situ measurements. Expanding the temporal range of data used for training could also enhance model accuracy. Given the specific nature of the UIB, applying transfer learning techniques could allow these models to be adapted and applied to other regions with different hydrological characteristics, improving their generalizability.

Author Contributions: Conceptualization, S.H. and O.K.; Methodology, M.Z.-K. and R.M.A.; Formal analysis, S.H., A.M.S.A.-J., O.K. and R.M.A.; Data curation, W.M.; Writing—original draft, M.Z.-K., O.K. and R.M.A.; Writing—review & editing, A.M.S.A.-J. and O.K.; Visualization, W.M.; Supervision, O.K. and W.M. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study will be available upon interesting request from the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: There are no conflicts of interest in this study.

References

1. Thapa, S.; Zhao, Z.; Li, B.; Lu, L.; Fu, D.; Shi, X.; Tang, B.; Qi, H. Snowmelt-driven streamflow prediction using machine learning techniques (LSTM, NARX, GPR, and SVR). *Water* **2020**, *12*, 1734. [CrossRef]
2. Tayyab, M.; Ahmad, I.; Sun, N.; Zhou, J.; Dong, X. Application of integrated artificial neural networks based on decomposition methods to predict streamflow at Upper Indus Basin, Pakistan. *Atmosphere* **2018**, *9*, 494. [CrossRef]

3. Shah, M.I.; Khan, A.; Akbar, T.A.; Hassan, Q.K.; Khan, A.J.; Dewan, A. Predicting hydrologic responses to climate changes in highly glacierized and mountainous region Upper Indus Basin. *R. Soc. Open Sci.* **2020**, *7*, 191957. [CrossRef] [PubMed]

4. Bilal, H.; Chamhuri, S.; Bin Mokhtar, M.; Kanniah, K.D. Recent snow cover variation in the Upper Indus Basin of Gilgit Baltistan, Hindukush Karakoram Himalaya. *J. Mt. Sci.* **2019**, *16*, 296–308. [CrossRef]

5. Li, C.; Su, F.; Yang, D.; Tong, K.; Meng, F.; Kan, B. Spatiotemporal variation of snow cover over the Tibetan Plateau based on MODIS snow product, 2001–2014. *Int. J. Climatol.* **2018**, *38*, 708–728. [CrossRef]

6. Simic, A.; Fernandes, R.; Brown, R.; Romanov, P.; Park, W. Validation of VEGETATION, MODIS, and GOES+SSM/I snow-cover products over Canada based on surface snow depth observations. *Hydrol. Process.* **2004**, *18*, 1089–1104. [CrossRef]

7. Tekeli, A.E.; Akyürek, Z.; Şorman, A.A.; Şensoy, A. Using MODIS snow cover maps in modeling snowmelt runoff process in the eastern part of Turkey. *Remote Sens. Environ.* **2005**, *97*, 216–230. [CrossRef]

8. Parajka, J.; Blöschl, G. Validation of MODIS snow cover images over Austria. *Hydrol. Earth Syst. Sci.* **2006**, *10*, 679–689. [CrossRef]

9. Rittger, K.; Painter, T.H.; Dozier, J. Assessment of methods for mapping snow cover from MODIS. *Adv. Water Resour.* **2013**, *51*, 367–380. [CrossRef]

10. Hao, X.; Huang, G.; Zheng, Z.; Sun, X.; Ji, W.; Zhao, H.; Wang, J.; Li, H.; Wang, X. Development and validation of a new MODIS snow-cover-extent product over China. *Hydrol. Earth Syst. Sci.* **2022**, *26*, 1937–1952. [CrossRef]

11. Bousbaa, M.; Boudhar, A.; Kinnard, C.; Elyoussfi, H.; Karaoui, I.; Eljabiri, Y.; Bouamri, H.; Chehbouni, A. An accurate snow cover product for the Moroccan Atlas Mountains: Optimization of the MODIS NDSI index threshold and development of snow fraction estimation models. *Int. J. Appl. Earth Obs. Geoinf.* **2024**, *129*, 103851. [CrossRef]

12. Abudu, S.; Cui, C.L.; King, J.P.; Abudukadeer, K. Comparison of performance of statistical models in forecasting monthly streamflow of Kizil River, China. *Water Sci. Eng.* **2010**, *3*, 269–281.

13. Bourdin, D.R.; Fleming, S.W.; Stull, R.B. Streamflow modelling: A primer on applications, approaches and challenges. *Atmos.-Ocean* **2012**, *50*, 507–536. [CrossRef]

14. Koch, R.W. A Stochastic Streamflow Model Based on Physical Principles. *Water Resour. Res.* **1985**, *21*, 545–553. [CrossRef]

15. Sun, W.; Wang, Y.; Wang, G.; Cui, X.; Yu, J.; Zuo, D.; Xu, Z. Physically based distributed hydrological model cali-bration based on a short period of streamflow data: Case studies in four Chinese basins. *Hydrol. Earth Syst. Sci.* **2017**, *21*, 251–265. [CrossRef]

16. Ikram, R.M.A.; Goliatt, L.; Kisi, O.; Trajkovic, S.; Shahid, S. Covariance Matrix Adaptation Evolution Strategy for Improving Machine Learning Approaches in Streamflow Prediction. *Mathematics* **2022**, *10*, 2971. [CrossRef]

17. Alizamir, M.; Kisi, O.; Muhammad Adnan, R.; Kuriqi, A. Modelling reference evapotranspiration by combining neuro-fuzzy and evolutionary strategies. *Acta Geophys.* **2020**, *68*, 1113–1126. [CrossRef]

18. Adnan, R.M.; Petroselli, A.; Heddam, S.; Santos, C.A.G.; Kisi, O. Short Term Rainfall-Runoff Modelling Using Several Machine Learning Methods and a Conceptual Event-Based Model. *Stoch. Environ. Res. Risk Assess.* **2021**, *35*, 597–616. [CrossRef]

19. Zounemat-Kermani, M.; Mahdavi-Meymand, A.; Hinkelmann, R. A comprehensive survey on conventional and modern neural networks: Application to river flow forecasting. *Earth Sci. Inform.* **2021**, *14*, 893–911. [CrossRef]

20. Rahman, K.U.; Pham, Q.B.; Jadoon, K.Z.; Shahid, M.; Kushwaha, D.P.; Duan, Z.; Mohammadi, B.; Khedher, K.M.; Anh, D.T. Comparison of machine learning and process-based SWAT model in simulating streamflow in the Upper Indus Basin. *Appl. Water Sci.* **2022**, *12*, 178. [CrossRef]

21. Raaj, S.; Gupta, V.; Singh, V.; Shukla, D.P. A novel framework for peak flow estimation in the himalayan river basin by integrating SWAT model with machine learning based approach. *Earth Sci. Inform.* **2024**, *17*, 211–226. [CrossRef]

22. Mushtaq, H.; Akhtar, T.; Hashmi, M.Z.U.R.; Masood, A.; Saeed, F. Hydrologic interpretation of machine learning models for 10-daily streamflow simulation in climate sensitive upper Indus catchments. *Theor. Appl. Clim.* **2024**, *155*, 1–18.

23. Nakhaei, M.; Zanjanian, H.; Nakhaei, P.; Gheibi, M.; Moezzi, R.; Behzadian, K.; Campos, L.C. Comparative Evaluation of Deep Learning Techniques in Streamflow Monthly Prediction of the Zarrine River Basin. *Water* **2024**, *16*, 208. [CrossRef]

24. Abdoulhalik, A.; Ahmed, A.A. A Comparative Analysis of Advanced Machine Learning Techniques for River Streamflow Time-Series Forecasting. *Sustainability* **2024**, *16*, 4005. [CrossRef]

25. Vatanchi, S.M.; Etemadfard, H.; Maghrebi, M.F.; Shad, R. A comparative study on forecasting of long-term daily streamflow using ANN, ANFIS, BiLSTM and CNN-GRU-LSTM. *Water Resour. Manag.* **2023**, *37*, 4769–4785. [CrossRef]

26. Wegayehu, E.B.; Muluneh, F.B. Multivariate streamflow simulation using hybrid deep learning models. *Comput. Intell. Neurosci.* **2021**, *2021*, 5172658. [CrossRef]

27. Le, X.-H.; Kim, Y.; Van Binh, D.; Jung, S.; Nguyen, D.H.; Lee, G. Improving rainfall-runoff modeling in the Mekong river basin using bias-corrected satellite precipitation products by convolutional neural networks. *J. Hydrol.* **2024**, *630*, 130762. [CrossRef]

28. Imran, M.; Majeed, M.D.; Zaman, M.; Shahid, M.A.; Zhang, D.; Zahra, S.M.; Maqbool, Z. Artificial neural networks and regression modeling for water resources management in the upper Indus Basin. *Environ. Sci. Proc.* **2023**, *25*, 53. [CrossRef]

29. Maiti, R.; Menon, B.G.; Abraham, A. Ensemble empirical mode decomposition based deep learning models for forecasting river flow time series. *Expert Syst. Appl.* **2024**, *255*, 124550. [CrossRef]

30. Wu, J.; Wang, Z.; Hu, Y.; Tao, S.; Dong, J. Runoff Forecasting using convolutional neural networks and optimized bi-directional long short-term memory. *Water Resour. Manag.* **2023**, *37*, 937–953. [CrossRef]

31. Wang, X.; Sun, W.; Lu, F.; Zuo, R. Combining Satellite Optical and Radar Image Data for Streamflow Estimation Using a Machine Learning Method. *Remote Sens.* **2023**, *15*, 5184. [CrossRef]

32. Zhou, F.; Chen, Y.; Liu, J. Application of a new hybrid deep learning model that considers temporal and feature dependencies in rainfall–runoff simulation. *Remote Sens.* **2023**, *15*, 1395. [CrossRef]

33. Li, J.; Yuan, X. Daily streamflow forecasts based on cascade long short-term memory (LSTM) model over the Yangtze River Basin. *Water* **2023**, *15*, 1019. [CrossRef]

34. Kumar, V.; Kedam, N.; Sharma, K.V.; Mehta, D.J.; Caloiero, T. Advanced machine learning techniques to improve hydrological prediction: A comparative analysis of streamflow prediction models. *Water* **2023**, *15*, 2572. [CrossRef]

35. Wang, Y.; Liu, J.; Xu, L.; Yu, F.; Zhang, S. Streamflow Simulation with high-resolution WRF input variables based on the CNN-LSTM hybrid model and gamma test. *Water* **2023**, *15*, 1422. [CrossRef]

36. Yu, Q.; Jiang, L.; Wang, Y.; Liu, J. Enhancing streamflow simulation using hybridized machine learning models in a semi-arid basin of the Chinese loess Plateau. *J. Hydrol.* **2023**, *617*, 129115. [CrossRef]

37. Lei, H.; Li, H.; Hu, W. Enhancing the streamflow simulation of a process-based hydrological model using machine learning and multi-source data. *Ecol. Inform.* **2024**, *82*, 102755. [CrossRef]

38. Wang, Y.; Pang, G.; Wang, T.; Cong, X.; Pan, W.; Fu, X.; Wang, X.; Xu, Z. Future Reference Evapotranspiration Trends in Shandong Province, China: Based on SAO-CNN-BiGRU-Attention and CMIP6. *Agriculture* **2024**, *14*, 1556. [CrossRef]

39. Zhao, L.; Luo, T.; Jiang, X.; Zhang, B. Prediction of soil moisture using BiGRU-LSTM model with STL decomposition in Qinghai–Tibet Plateau. *PeerJ* **2023**, *11*, e15851. [CrossRef]

40. Zhang, X.; Yang, Y.; Liu, J.; Zhang, Y.; Zheng, Y. A CNN-BILSTM monthly rainfall prediction model based on SCSSA optimization. *J. Water Clim. Chang.* **2024**, *15*, 4862–4876. [CrossRef]

41. Hu, C.; Zhou, L.; Gong, Y.; Li, Y.; Deng, S. Research on Water Level Anomaly Data Alarm Based on CNN-BiLSTM-DA Model. *Water* **2023**, *15*, 1659. [CrossRef]

42. Wu, C.; Chau, K. Data-driven models for monthly streamflow time series prediction. *Eng. Appl. Artif. Intell.* **2010**, *23*, 1350–1367. [CrossRef]

43. Tang, Q.; Lettenmaier, D.P. Use of satellite snow-cover data for streamflow prediction in the Feather River Basin, California. *Int. J. Remote Sens.* **2010**, *31*, 3745–3762. [CrossRef]

44. Bennett, K.E.; Cherry, J.E.; Balk, B.; Lindsey, S. Using MODIS estimates of fractional snow cover area to improve streamflow forecasts in interior Alaska. *Hydrol. Earth Syst. Sci.* **2019**, *23*, 2439–2459. [CrossRef]

45. Ikram, R.M.A.; Hazarika, B.B.; Gupta, D.; Heddam, S.; Kisi, O. Streamflow Prediction in Mountainous Region Using New Machine Learning and Data Preprocessing Methods: A Case Study. *Neural Comput. Appl.* **2022**, 1–18. [CrossRef]

46. Adnan, R.M.; Mostafa, R.R.; Elbeltagi, A.; Yaseen, Z.M.; Shahid, S.; Kisi, O. Development of New Machine Learning Model for Streamflow Prediction: Case Studies in Pakistan. *Stoch. Environ. Res. Risk Assess.* **2022**, *36*, 999–1033. [CrossRef]

47. Ikram, R.M.A.; Ewees, A.A.; Parmar, K.S.; Yaseen, Z.M.; Shahid, S.; Kisi, O. The Viability of Extended Marine Predators Algorithm-Based Artificial Neural Networks for Streamflow Prediction. *Appl. Soft Comput.* **2022**, *131*, 109739. [CrossRef]

48. Li, J.; Pang, G.; Wang, X.; Liu, F.; Zhang, Y. Spatiotemporal Dynamics of Land Surface Albedo and Its Influencing Factors in the Qilian Mountains, Northeastern Tibetan Plateau. *Remote Sens.* **2022**, *14*, 1922. [CrossRef]

49. Mal, S.; Rani, S.; Maharana, P. Estimation of spatio-temporal variability in land surface temperature over the Ganga River Basin using MODIS data. *Geocarto Int.* **2022**, *37*, 3817–3839. [CrossRef]

50. Qin, J.; Yang, K.; Liang, S.; Zhang, H.; Ma, Y.; Guo, X.; Chen, Z. Evaluation of surface albedo from GEWEX-SRB and ISCCP-FD data against validated MODIS product over the Tibetan Plateau. *J. Geophys. Res. Atmos.* **2011**, *116*. [CrossRef]

51. Adnan, R.M.; Mostafa, R.R.; Dai, H.-L.; Heddam, S.; Masood, A.; Kisi, O. Enhancing accuracy of extreme learning machine in predicting river flow using improved reptile search algorithm. *Stoch. Environ. Res. Risk Assess.* **2023**, *37*, 3063–3083. [CrossRef]

52. Ikram, R.M.A.; Mostafa, R.R.; Chen, Z.; Islam, A.R.M.T.; Kisi, O.; Kuriqi, A.; Zounemat-Kermani, M. Advanced Hybrid Metaheuristic Machine Learning Models Application for Reference Crop Evapotranspiration Prediction. *Agronomy* **2023**, *13*, 98. [CrossRef]

53. Latif, Y.; Ma, Y.; Ma, W.; Muhammad, S.; Adnan, M.; Yaseen, M.; Fealy, R. Differentiating Snow and Glacier Melt Contribution to Runoff in the Gilgit River Basin via Degree-Day Modelling Approach. *Atmosphere* **2020**, *11*, 1023. [CrossRef]

54. Adnan, M.; Nabi, G.; Kang, S.; Zhang, G.; Adnan, R.M.; Anjum, M.N.; Iqbal, M.; Ali, A.F. Snowmelt Runoff Modelling under Projected Climate Change Patterns in the Gilgit River Basin of Northern Pakistan. *Pol. J. Environ. Stud.* **2017**, *26*, 525–542. [CrossRef]

55. Hochreiter, S. *Long Short-Term Memory*; Neural Computation MIT-Press: Cambridge, MA, USA, 1997.

56. Liu, D.R.; Hsu, Y.K.; Chen, H.Y. Air pollution prediction based on factory-aware attentional LSTM neural network. *Computing* **2020**, *103*, 75–98. [CrossRef]

57. Adnan, R.M.; Mirboluki, A.; Mehraein, M.; Malik, A.; Heddam, S.; Kisi, O. Improved prediction of monthly streamflow in a mountainous region by Metaheuristic-Enhanced deep learning and machine learning models using hydroclimatic data. *Theor. Appl. Climatol.* **2024**, *155*, 205–228. [CrossRef]

58. Wang, J.Y.; Li, J.Z.; Wang, X.X. Air quality prediction using CT-LSTM. *Neural Comput. Appl.* **2020**, *33*, 4779–4792. [CrossRef]

59. Al-Smadi, B.S. DeBERTa-BiLSTM: A multi-label classification model of Arabic medical questions using pre-trained models and deep learning. *Comput. Biol. Med.* **2024**, *170*, 107921. [CrossRef]

60. Thireou, T.; Reczko, M. Bidirectional Long Short-Term Memory Networks for Predicting the Subcellular Localization of Eukaryotic Proteins. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2007**, *4*, 441–446. [CrossRef]

61. Lu, Y.; Tang, L.; Liu, Z.; Zhou, L.; Yang, B.; Jiang, Z.; Liu, Y. Unsupervised quantitative structural damage identification method based on BiLSTM networks and probability distribution model. *J. Sound Vib.* **2024**, *590*, 118597. [CrossRef]

62. Cho, K.; Van Merriënboer, B.; Bahdanau, D.; Bengio, Y. On the properties of neural machine translation: Encod-er-decoder approaches. *arXiv* **2014**, arXiv:1409.1259. [CrossRef]

63. Mahjoub, S.; Chrifi-Alaoui, L.; Marhic, B.; Delahoche, L. Predicting Energy Consumption Using LSTM, Multi-Layer GRU and Drop-GRU Neural Networks. *Sensors* **2022**, *22*, 4062. [CrossRef] [PubMed]

64. Hamayel, M.J.; Owda, A.Y. A Novel Cryptocurrency Price Prediction Model Using GRU, LSTM and bi-LSTM Machine Learning Algorithms. *AI* **2021**, *2*, 477–496. [CrossRef]

65. Li, X.; Ma, X.; Xiao, F.; Wang, F.; Zhang, S. Application of Gated Recurrent Unit (GRU) Neural Network for Smart Batch Production Prediction. *Energies* **2020**, *13*, 6121. [CrossRef]

66. Gurumoorthy, S.; Kokku, A.K.; Falkowski-Gilski, P.; Divakarachari, P.B. Effective Air Quality Prediction Using Reinforced Swarm Optimization and Bi-Directional Gated Recurrent Unit. *Sustainability* **2023**, *15*, 11454. [CrossRef]

67. Yang, J.; Yang, F.; Zhou, Y.; Wang, D.; Li, R.; Wang, G.; Chen, W. A data-driven structural damage detection framework based on parallel convolutional neural network and bidirectional gated recurrent unit. *Inf. Sci.* **2021**, *566*, 103–117. [CrossRef]

68. Micheli, A.; Natali, M.; Pedrelli, L.; Simone, L.; Morales, M.A.; Piacenti, M.; Vozzi, F. Analysis and interpretation of ECG time series through convolutional neural networks in Brugada syndrome diagnosis. In *International Conference on Artificial Neural Networks*; Springer Nature: Cham, Switzerland, 2023; pp. 26–36.

69. Özbay, E.; Özbay, F.A.; Gharehchopogh, F.S. Visualization and classification of mushroom species with multi-feature fusion of metaheuristics-based convolutional neural network model. *Appl. Soft Comput.* **2024**, *164*, 111936. [CrossRef]

70. Fan, Y.; Ma, K.; Zhang, L.; Liu, J.; Xiong, N.; Yu, S. VeriCNN: Integrity verification of large-scale CNN training process based on zk-SNARK. *Expert Syst. Appl.* **2024**, *255*, 124531. [CrossRef]

71. Li, J.; Yan, Y.; Zhang, K.; Li, C.; Yuan, P. FPCNN: A fast privacy-preserving outsourced convolutional neural network with low-bandwidth. *Knowl.-Based Syst.* **2024**, *283*, 111181. [CrossRef]

72. Ikram, R.M.A.; Mostafa, R.R.; Chen, Z.; Parmar, K.S.; Kisi, O.; Zounemat-Kermani, M. Water Temperature Prediction Using Improved Deep Learning Methods through Reptile Search Algorithm and Weighted Mean of Vectors Optimizer. *J. Mar. Sci. Eng.* **2023**, *11*, 259. [CrossRef]

73. Emam, M.M.; Houssein, E.H.; Samee, N.A.; Alohali, M.A.; Hosney, M.E. Breast cancer diagnosis using optimized deep convolutional neural network based on transfer learning technique and improved Coati optimization algorithm. *Expert Syst. Appl.* **2024**, *255*, 124581. [CrossRef]

74. Halbouni, A.; Gunawan, T.S.; Habaebi, M.H.; Halbouni, M.; Kartiwi, M.; Ahmad, R. CNN-LSTM: Hybrid Deep Neural Network for Network Intrusion Detection System. *IEEE Access* **2022**, *10*, 99837–99849. [CrossRef]

75. Shaohu, L.; Yuandeng, W.; Rui, H. Prediction of drilling plug operation parameters based on incremental learning and CNN-LSTM. *Geoenergy Sci. Eng.* **2024**, *234*, 212631. [CrossRef]

76. Rahman, A.; Jamal, S.; Taheri, H. Remote condition monitoring of rail tracks using distributed acoustic sensing (DAS): A deep CNN-LSTM-SW based model. *Green Energy Intell. Transp.* **2024**, *3*, 100178. [CrossRef]

77. Thekkekara, J.P.; Yongchareon, S.; Liesaputra, V. An attention-based CNN-BiLSTM model for depression detection on social media text. *Expert Syst. Appl.* **2024**, *249*, 123834. [CrossRef]

78. An, Z.; Wang, F.; Wen, Y.; Hu, F.; Han, S. A real-time CNN–BiLSTM-based classifier for patient-centered AR-SSVEP active rehabilitation exoskeleton system. *Expert Syst. Appl.* **2024**, *255*, 124706. [CrossRef]

79. Tian, Y.; Wang, G.; Li, H.; Huang, Y.; Zhao, F.; Guo, Y.; Gao, J.; Lai, J. A novel deep learning method based on 2-D CNNs and GRUs for permeability prediction of tight sandstone. *Geoenergy Sci. Eng.* **2024**, *238*, 212851. [CrossRef]

80. Thanh, P.N.; Cho, M.-Y. Advanced AIoT for failure classification of industrial diesel generators based hybrid deep learning CNN-BiLSTM algorithm. *Adv. Eng. Inform.* **2024**, *62*, 102644. [CrossRef]

81. Chen, G.; Tian, H.; Xiao, T.; Xu, T.; Lei, H. Time series forecasting of oil production in Enhanced Oil Recovery system based on a novel CNN-GRU neural network. *Geoenergy Sci. Eng.* **2024**, *233*, 212528. [CrossRef]

82. Li, Q.; Zhang, X.; Ma, T.; Liu, D.; Wang, H.; Hu, W. A Multi-step ahead photovoltaic power forecasting model based on TimeGAN, Soft DTW-based K-medoids clustering, and a CNN-GRU hybrid neural network. *Energy Rep.* **2022**, *8*, 10346–10362. [CrossRef]

83. Xu, Z.; Li, Y.F.; Huang, H.Z.; Deng, Z.; Huang, Z. A novel method based on CNN-BiGRU and AM model for bearing fault diagnosis. *J. Mech. Sci. Technol.* **2024**, *38*, 3361–3369. [CrossRef]

84. Lu, Y.; Wu, X.; Liu, P.; Li, H.; Liu, W. Rice disease identification method based on improved CNN-BiGRU. *Artif. Intell. Agric.* **2023**, *9*, 100–109. [CrossRef]

85. Hassan, M.; Hassan, I. Improving ANN-based streamflow estimation models for the Upper Indus Basin using satellite-derived snow cover area. *Acta Geophys.* **2020**, *68*, 1791–1801. [CrossRef]