Water Resources Management (2026) 40:15
https://doi.org/10.1007/s11269-025-04401-5

A Comparative Evaluation of Deep Learning and Machine Learning Models for River Suspended Sediment Concentration Forecasting

Amin Gharehbaghi[1] · **Salim Heddam**[2] · **Saeid Mehdizadeh**[3] · **Sungwon Kim**[4]

Received: 1 August 2025 / Accepted: 16 October 2025

Abstract
Suspended sediment concentration (SSC) in rivers is a crucial parameter required in hydrological studies, water resources management, and many other relevant applications. This study presents a comparative assessment of deep learning (DL) and machine learning (ML) methods in river SSC prediction of two river stations on the Mississippi River, United States. To that end, two single DL models, namely recurrent neural networks (RNN) and bidirectional RNN (BiRNN) were developed. Generally, the RNN was found to outperform the BiRNN for predicting SSC. Furthermore, a convolutional neural network (CNN) was coupled on the applied DL models to create the hybrid RNN-CNN and BiRNN-CNN models. The results denoted that the BiRNN-CNN models generally performed better compared with RNN-CNN ones. Besides the four types of DL models, three forms of ML models, including adaptive boosting (AdaBoost), natural gradient boosting (NGBoost), and gradient boosting regression trees (GBRT) were also established. As a general conclusion, NGBoost and GBRT demonstrated the highest and lowest level of accuracy in river SSC forecasting. Eventually, the influence of input predictors on the outputs of models was done considering local interpretable model-agnostic explanations (LIME). Assessing the LIME outcomes for the selected samples of the test data revealed that the current daily river streamflow and one daily lagged SSC data were the most effective inputs on SSC prediction results.

Keywords River suspended sediment concentration · Forecasting · Deep learning · Machine learning · Explainability

✉ Saeid Mehdizadeh
saied.mehdizadeh@gmail.com

[1] Department of Civil Engineering, Faculty of Engineering, Hasan Kalyoncu University, Şahinbey, Gaziantep 27110, Turkey

[2] Faculty of Science, Agronomy Department, Hydraulics Division, University 20 Août 1955 Skikda, Route El Hadaik, BP 26, Skikda, Algeria

[3] Water Engineering Department, Urmia University, Urmia, Iran

[4] Department of Railroad Construction and Safety Engineering, Dongyang University, Yeongju 36040, Republic of Korea

Published online: 23 December 2025

1 Introduction

In an optimal river water management, it is essential to know the amount of sediment yield by rivers for the erosion control, bed stabilization, flood disposal, etc. (Afan et al. 2025; Bezak et al. 2025). Suspended sediment is known as one of the important components of the hydrological, geomorphological, and ecological functions of watersheds (Kumar et al. 2025). It is also one of the main factors in reducing river water quality. Precisely forecasting the suspended sediment concentration (SSC) in rivers is of great importance in several engineering domains like ecological assessment, water resources management, design and planning in operation of water structures (Fan et al. 2023; Moradinejad 2024; Gupta and Feng 2025).

Suspended sediment in rivers, including SSC can be measured directly by installing the equipment at hydrometric stations. However, these methods include numerous problems such as the relatively high cost of the measuring devices, the failure of the equipment used, the lack of skilled personnel, and finally the relatively low accuracy of measurements, specifically at low flow rates (Jarbais and Harshavardhanan 2025). Due to the mentioned problems in direct measurements, indirect methods have been usually used to predict SSC in rivers. Three major classifications of indirect methods have been proposed for estimating SSC in rivers, including empirical techniques, physically process-driven models, and artificial intelligence (AI)-based schemes (Fan et al. 2023; Nguyen et al. 2024a).

In the last few decades, researchers have used simple and empirical methods such as sediment rating curve to predict suspended sediment in rivers. However, despite their simplicity, these methods have low prediction accuracy, and therefore, the results of this method cannot be relied upon (Song et al. 2024). In contrast to the empirical models, process-driven techniques have been proposed relying on the physical processes of the studied phenomenon (e.g., SSC). Requiring a large number of data, spending a lot of costs, specifically for large rivers and watersheds, and finally calibrating the utilized process-based models for the studied areas are the possible limitations that have made their application difficult (Li and Yang 2022). Compared with the empirical and physical models, the AI paradigms comprising deep learning (DL) and machine learning (ML) have some advantages like high generalization capability, lack of the complicated mathematical relationships, their learning potential, and requiring less data (Danesh et al. 2025). Hence, they have been broadly used in hydrological parameters forecasting like SSC, which some of them are briefly presented below.

Nguyen et al. (2024b) developed four ML models for the river SSC prediction, and resulted the better accuracy of support vector machine (SVM) than the other models. Huang et al. (2021) implemented three ML and four DL methods, and concluded that the SVM and long short-term memory (LSTM) presented the best performance. Two ML models, including SVM and random forest (RF) were implemented by Dehkordi et al. (2021). Their outcomes revealed that the RF demonstrated better SSC predictions than the SVR. Joshi et al. (2024) established two ML models comprising feedforward neural network (FFNN) and cascade correlation neural network (CCNN). The results denoted the better accuracy of CCNN than FFNN. Kim et al. (2022) utilized an LSTM, and then recommended it as a reliable SSC prediction tool. A performance comparison of an LSTM with two ML models like FFNN and adaptive neuro fuzzy inference system (ANFIS) was done by Kaveh et al. (2021). The LSTM represented improved SSC forecasts compared with ML models. Dehkordi et al. (2024) developed some types of ensemble ML models, and concluded that extreme gradient boosting (XGBoost) optimized by Bayesian optimization performed the best. Two signal decomposition methods were

used by Roushangar and Alirezazadeh Sadaghiani (2025) to couple on four DL models in SSC prediction. It was revealed that hybrid models outperformed the standalone ones. The performance of an LSTM was compared with multiple regression (MR) in predicting SSC by Pham Van et al. (2023). The results indicated the superiority of LSTM over the MR. Zhang and Yang (2020) coupled the CEEMDAN with a DL model, namely gated recurrent unit (GRU), and compared its accuracy with single GRU, LSTM, and SVM. Their findings exhibited that hybrid CEEMDAN-GRU was the best-performing model for SSC forecasting. Saroughi et al. (2025) developed four hybrid models through coupling two ML models with two optimization algorithms. The results showed that the merged models illustrated better SSC forecasts then their relevant single forms. A detailed survey of previous literature clearly demonstrated that there is a research gap in comprehensive comparison of ML and DL models for river SSC forecasting. Besides, the performance of DL methods in coupled with convolutional neural networks has evaluated less in river SSC simulations.

The present study made an attempt to forecast the daily SSC time series in the Mississippi River, United States, by developing some intelligent AI-based methods. The aims of this study can be listed as follows: To (1) construct three types of ML models like adaptive boosting (AdaBoost), natural gradient boosting (NGBoost), and gradient boosting regression tree (GBRT), (2) implement four kinds of DL schemes comprising of recurrent neural networks (RNN), bidirectional RNN (BiRNN), RNN hybridized with convolutional neural networks (RNN-CNN), and BiRNN-CNN, (3) compare the efficacy of all the developed models in terms of statistically and schematically, and (4) apply local interpretable model-agnostic explanations (LIME) to find out the contributions of input predictors in the models output. To accomplish this study and therefore achieving the aforementioned goals, two river sites located on the Mississippi River were selected as the case study. The new contributions of current study relative to previous works can be listed as follows: a detailed assessment of four DL and three ML models, developing two hybrid DL frameworks comprising RNN-CNN and BiRNN-CNN, and finally applying LIME to interpret the results of established superior models, which have not been addressed so far in river SSC forecasting.

2 Materials and Methods

2.1 Case Study and Data Description

This study selected two river stations located on the Mississippi River, United States. The Mississippi River is the second largest river in the United States, and the fourth largest in the world. It begins at Lake Itasca, passes through 10 U.S. states, and finally flows into the Gulf of Mexico. The length of this river is about 3730 km. In fact, the Mississippi watershed drains almost 32 U.S. states and two U.S. territories between the Rocky and Appalachian Mountains, but its main branch is located in the United States itself.

The data used in this study consisted of the suspended sediment concentration (SSC) and river streamflow (Q) at two river sites, including USGS 07010000 Mississippi River at St. Louis (Latitude 38°37’44.4”, Longitude 90°10’47.2”) and USGS 07020500 Mississippi River at Chester (Latitude 37°54’02.7”, Longitude 89°49’48.8”). The time period of the data collected from https://waterdata.usgs.gov/nwis/ website includes the daily scale during the water years from 2003/10/01 to 2022/09/30. The data was split into the training (i.e., the

Table 1 Summary statistics of variables at the studied sites

Variables	Subset	Unit	X_{mean}	X_{max}	X_{min}	S_x	C_v	R
USGS 07010000 Station								
SSC	Training	*mg/L*	242.409	3470.000	22.900	243.538	1.005	1.000
	Validation	*mg/L*	242.714	2240.000	25.700	235.416	0.970	1.000
Q	Training	m^3/s	6924.303	26362.98	1625.387	4159.266	0.601	0.550
	Validation	m^3/s	6970.835	26362.98	1696.179	4233.068	0.607	0.564
USGS 07020500 Station								
SSC	Training	*mg/L*	254.059	1870.000	16.600	230.631	0.908	1.000
	Validation	*mg/L*	264.338	1780.000	23.600	242.117	0.916	1.000
Q	Training	m^3/s	7219.876	27014.27	1730.159	4308.682	0.597	0.578
	Validation	m^3/s	7355.697	26136.44	1809.446	4421.308	0.601	0.569

first 70% of the data) and validation (i.e., the last 30% of the data) phases. Some statistical information of the data used such as mean (X_{mean}), maximum (X_{max}), minimum (X_{min}), standard deviation (S_x), and coefficient of variation (C_v) are tabulated in Table 1. The values of R column also represent the correlations of variables with target parameter, i.e., SSC.

2.2 Models Used

2.2.1 Recurrent Neural Networks (RNN)

Recurrent neural networks (RNN) is a type of neural networks that has the characteristic of a recurrent architecture in the connections between neurons (Rumelhart et al. 1986). This architecture allows the neural networks to accumulate the case so that it can model time-variant and dynamic components. Unlike feedforward neural networks (FFNN), it can process sequence-type features utilizing its internal memory. RNN is inferred from the fact that it handles dynamic data with an infinite feature signal length, and refers to both finite impulse and infinite impulse architectures (Huang et al. 2021). Since a finite impulse based RNN is a directed acyclic graph, it can be described as a FFNN if properly solved and reconstructed. However, since an infinite impulse based RNN is a directed graph, it is impossible to explain it as a FFNN (Nguyen et al. 2024b). RNN has restricted the ability to learn long-range dependencies due to the vanishing gradient problem. This issue was solved in 1997 with the development of the long short-term memory (LSTM) architecture with gated memory (Hochreiter and Schmidhuber 1997), which became the modification of standard RNN for handling long-term dependencies. Later, gated recurrent units (GRU) was proposed as a more efficient alternative (Chung et al. 2014).

2.2.2 Bidirectional Recurrent Neural Networks (BiRNN)

The conventional RNN has restrictions as the feature information of future case cannot be reached from the current case in the description of RNN. Bidirectional recurrent neural networks (BiRNN), which was described to enhance the amount of feature information available to the networks, combines two hidden layers in opposite directions (i.e., backward and forward) to the identical label. This generative deep learning admits the final layer to obtain information from both backward (past) and forward (future) cases (Schuster and Paliwal 1997). BiRNN can be trained utilizing all available feature information in the backward and

forward of a specific time frame to overcome the limitation of conventional RNN (Dong and Qian 2022). The idea is to split the current neurons of conventional RNN into a part for the positive time direction (forward) and a part for the negative time direction (backward) (Schuster and Paliwal 1997; Tong et al. 2019). Labels from forward cases are not linked to features of backward cases, and vice versa. Notice that without the backward cases, the architecture of BiRNN simplifies to a conventional unidirectional forward RNN (Xu et al. 2021). Eliminating the forward case creates a conventional RNN with a reversed time axis.

2.2.3 Convolutional Neural Networks (CNN)

LeCun et al. (1989) developed and applied CNN for practical applications, including image classification and computational biology and so on. In addition, CNN is an effective deep learning paradigm for automatic feature extraction, and has achieved remarkable results in the field of engineering problems. Also, CNN is the best deep learning paradigm for extracting image features, including two-dimensional (2D) and three-dimensional (3D) datasets that rely on high resolution (Alizamir et al. 2025; Bahrambanan et al. 2025). CNN consists of several obvious parts, including feature layer, convolutional layer, pooling layer, fully connected layer, and the label layer, respectively. The convolutional layer carries out mathematical transformation operations on the arriving data. Basically, it can be observed the arriving information as one mathematical description, the filter as another description, and the convolution process as a mathematical technique for evaluating how the filter transforms the original information. The pooling layer can reduce the data dimensions. This can not only decrease computational demands during training phase, but also help to prevent the network from becoming specialized for training samples (Barzegar et al. 2020; Ding et al. 2021; Momeny et al. 2022). Typically, a pooling layer is utilized after a convolutional layer. A pooling layer can change the output of the network at a specific location by utilizing simple statistics for the surrounding label (Zhang et al. 2024; Yang et al. 2021).

2.2.4 Adaptive Boosting (AdaBoost)

Freund and Schapire (1997) developed AdaBoost utilizing multiple classifiers. Also, the bias of AdaBoost can be resolved by integrating multiple boosting ensemble techniques into a unified prediction problem. The sampled data divided by the previous classifier was utilized to train the continuous classifier (Liu et al. 2015). Compared to the comparable boosting ensemble techniques, AdaBoost is less likely to overfit or underfit for certain classification issues. The classifier employed in AdaBoost can be fragile. That is, a weak classifier has a large classification error, but the classification error utilizing a weak classifier is lower than that of a random classifier. For voting models, the classification efficiency of AdaBoost can be improved by the aggregate decision of all weak classifiers (Tsai and Hung 2021). AdaBoost is described as an iterative approach that modifies a weak classifier at each iteration until a bounded classification error is achieved. The sample corresponding to the training phase is given the weight. If the classifier correctly divides the sample data, the weight of sample data should decrease. Also, the probability of sample data chosen by the classifier should decrease (Busari and Lim 2021). AdaBoost can improve the accuracy of weak classifiers by modifying the weight distribution of sampled data. Therefore, each weak classifier can be linked to form a strong classifier (Weidong et al. 2022).

2.2.5 Natural Gradient Boosting (NGBoost)

The conventional boosting ensemble approaches build a predictive algorithm by formalizing the correlation between feature and label parameters (Zhou et al. 2024). NGBoost organizes a comprehensive and probabilistic method of feature and label parameters while allowing for computing prediction uncertainty through probabilistic prediction (Fang et al. 2025; Shen et al. 2022). It connects a scoring rule, a probability distribution type, and a base learner (Zhou et al. 2024). Also, NGBoost integrates natural gradients into the ensemble techniques. It builds composite predictors within the gradient boosting technique by constructing individual models that are tuned to the natural gradient (Mitrentsis and Lens 2022). This admits NGBoost to calculate the distribution parameters, which ultimately supports probability prediction (Zhou et al. 2024). In addition, NGBoost can enhance the original ensemble techniques based on gradient boosting. Since the original ensemble techniques compute multiple parameters using a single one, their performance may be worse than the gradients of other parameters (Fang et al. 2025; Shen et al. 2022).

2.2.6 Gradient Boosting Regression Trees (GBRT)

Gradient boosting tree (GBT), one of the robust and versatile ensemble techniques, can be employed for both regression (GBRT) and decision (GBDT) problems (Friedman 2001). GBRT builds a model by collecting different classes of weak decision trees in a repeated boosting process. It can make class predictions by utilizing greedy methods to generate groups of decision trees to obtain more accurate outcomes during the training phase. GBRT describes the outstanding flexibility for incorporating diverse data characteristics, managing incomplete information, and alleviating overtraining and underfitting problems. As a result, it has been widely applied to a variety of real-world engineering tasks (Dash et al. 2021; Persson et al. 2017). The overall method is to integrate these multiple trees sequentially and aggregate the predictions for all individual models to determine an ensemble outcome (Persson et al. 2017). The iterative nature allows each successive tree to target the errors specifically left behind by the previous tree, learning from outcomes and the remaining discrepancies of previous iterations to improve the predictive accuracy of ensemble (Alizamir et al. 2024). This architecture supplies GBRT with excellent integration and adaptability capabilities, allowing it to remain efficient even when faced with limited datasets due to real-world data constraints (Dash et al. 2021; Nie et al. 2021).

2.3 Local Interpretable model-agnostic Explanations (LIME)

LIME provides the calculated data based on random perturbations, and supplies explanations by modifying a scattered and weighted linear model for predictive feedback of distressed data (Daif et al. 2025; Zafar and Khan 2021). In addition, it can explain the predictive ability of a classifier based on an interpretable model, and supplies a model-independent technique for extracting localized interpretations of singular predictions (Peng et al. 2024). The process of LIME creation involves several stages as follows. Create a new dataset consisting of perturbed samples that are slightly modified from the original dataset. The original model is employed to predict the perturbed samples, weighted by their similarity to the original dataset. An interpretable model can be trained to locally approximate the prediction of origi-

nal model utilizing the weighted samples. In the final stage, the coefficients of interpretable model are employed to furnish an explanation for the original dataset (Neves et al. 2021; Wikle et al. 2023). LIME enables engineers and developers to quantify the contribution of each feature data along local sampling boundary. Also, it is suitable for all supervised algorithms, and is utilized to describe locally linear relationships by establishing a simple surrogate model for individual sample datasets (Yang et al. 2023).

2.4 Models Development

The present study developed some types of AI-based intelligent schemes for predicting the daily SSC time series at two hydrometric sites in United States, including USGS 07010000 and USGS 07020500. In this respect, two DL methods comprising of RNN and BiRNN models were firstly established. Then, two hybrid forms of DL models were developed through the hybridization of aforementioned baseline models with CNN to generate RNN-CNN and BiRNN-CNN coupled models. Finally, three ML-based techniques were proposed as a comparative study between the performance of DL and ML paradigms in river SSC forecasting. The values of hyperparameters tabulated in Table 2 were utilized to develop the models employed in this study. To construct the models, six different input scenarios were defined as listed in Table 3. Figure 1 displays the general flowchart of current study to forecast SSC.

2.5 Performance Metrics

We used four usual performance metrics comprising correlation coefficient (R), Nash-Sutcliffe efficiency (NSE), root mean square error (RMSE), and mean absolute error (MAE) to statistically evaluate the accuracy of all the implemented models for river SSC prediction. They can be represented as the following:

$$R = \frac{\sum_{i=1}^{N} \left(SSC_{o,i} - \overline{SSC_o}\right) \cdot \left(SSC_{f,i} - \overline{SSC_f}\right)}{\sqrt{\sum_{i=1}^{N} \left(SSC_{o,i} - \overline{SSC_o}\right)^2 \cdot \sum_{i=1}^{N} \left(SSC_{f,i} - \overline{SSC_f}\right)^2}} \tag{1}$$

$$NSE = 1 - \frac{\sum_{i=1}^{N} \left(SSC_{o,i} - SSC_{f,i}\right)^2}{\sum_{i=1}^{N} \left(SSC_{o,i} - \overline{SSC_o}\right)^2} \tag{2}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} \left(SSC_{o,i} - SSC_{f,i}\right)^2}{N}} \tag{3}$$

$$MAE = \frac{\left|SSC_{o,i} - SSC_{f,i}\right|}{N} \tag{4}$$

Table 2 Models hyperparameters

Models	Optimization method	Hyperparameters
RNN	Adam	Dropout = 0.01, recurrent_dropout = 0.01, SimpleRNN (30), Dense :1, activation='sigmoid', loss='mse', optimizer='adam', epochs = 150
BiRNN	Adam	Dropout = 0.01, recurrent_dropout = 0.01, Bidirectional(SimpleRNN (20)), Dense :1, activation='sigmoid', loss='mse', optimizer='adam', epochs = 100
RNN-CNN	Adam	Conv1D(filters = 128, kernel_size = 2, activation='ReLU'), MaxPooling1D(pool_size = 2), SimpleRNN(50), dropout = 0.01, recurrent_dropout = 0.01, Dense(128, activation='ReLU'), epochs = 100, optimizer='adam', loss='mse'.
BiRNN-CNN	Adam	Conv1D(filters = 64, kernel_size = 2, activation='ReLU'), MaxPooling1D(pool_size = 2), Bidirectional (SimpleRNN(20)), dropout = 0.01, recurrent_dropout = 0.01, Dense(64, activation='ReLU'), epochs = 200, optimizer='adam', loss='mse'.
AdaBoost	grid search	random_state = 0, n_estimators = 100, learning_rate = 0.01, loss='linear'
NGBoost	grid search	Dist = Normal, Score = LogScore, Base = default_tree_learner, natural_gradient = True, n_estimators = 500,learning_rate = 0.01, minibatch_frac = 1.0,col_sample = 1.0,verbose = True, verbose_eval = 100,tol = 1e-4,random_state = None,
GBRT	grid search	"n_estimators":300, "max_depth": 10, "min_samples_split": 10, "learning_rate": 0.01, "loss": "squared_error"

Table 3 Different input scenarios

Scenarios	Input variables combination	Output
Scenario1	Q_t	SSC_t
Scenario2	SSC_{t-1}	SSC_t
Scenario3	Q_t, SSC_{t-1}	SSC_t
Scenario4	Q_{t-1}, Q_t	SSC_t
Scenario5	SSC_{t-2}, SSC_{t-1}	SSC_t
Scenario6	Q_{t-1}, Q_t, SSC_{t-2}, SSC_{t-1}	SSC_t

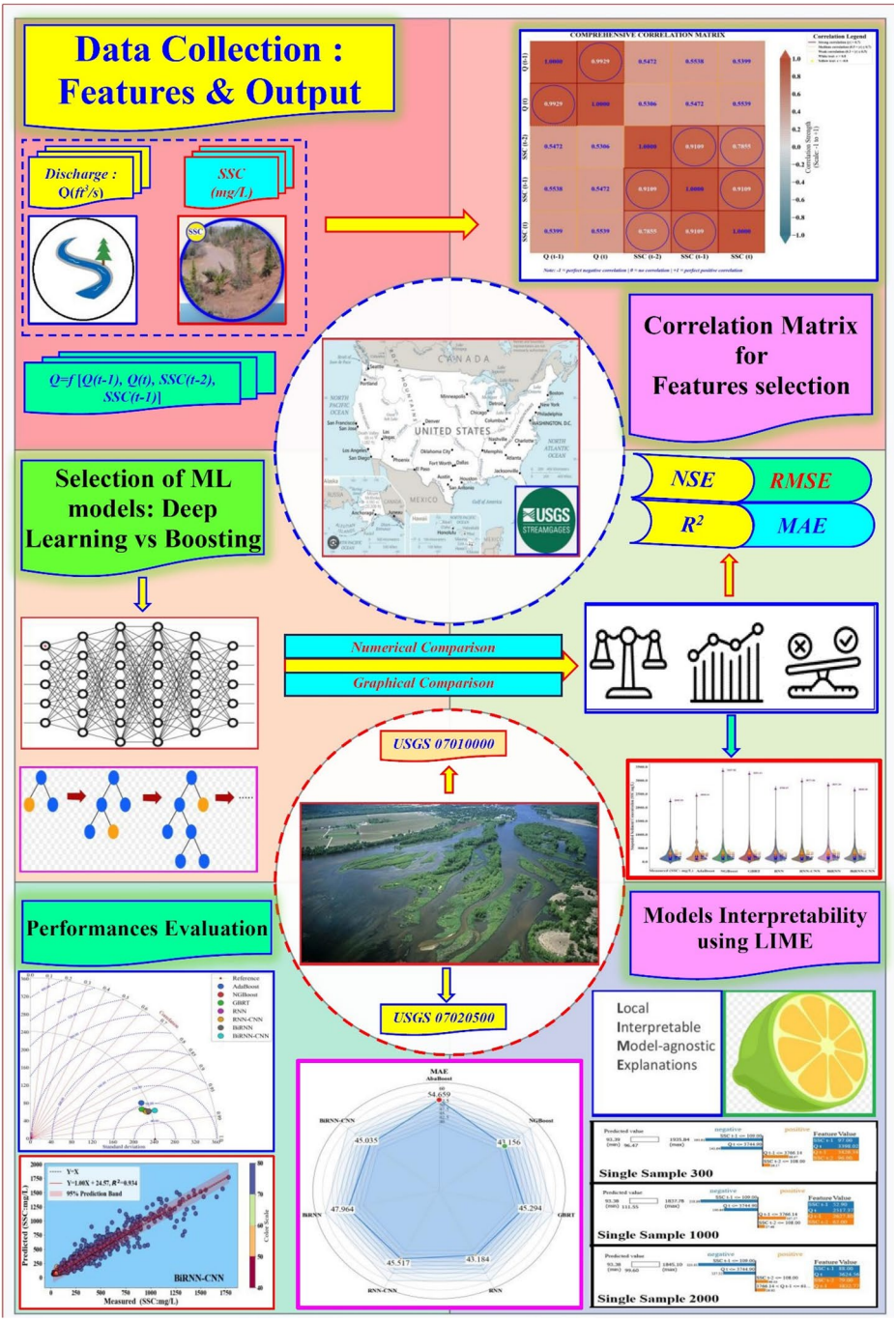

Fig. 1 The general workflow of this study

where the observed and forecasted suspended sediment concentration values are respectively shown in $SSC_{o,i}$and $SSC_{f,i}$; $\overline{SSC_o}$and $\overline{SSC_f}$respectively indicate the mean values for the observed and forecasted SSC values; and *N* represents the total observational SSC data.

3 Results and Discussion

The statistical metrics for all the simple and hybrid models during the validation period are tabulated in Table 4 for USGS 07010000 Station and Table 5 for USGS 07020500 Station. Assessing the accuracy of models developed using the Q_t input estimator in the first scenario denoted its applicability for estimating the daily SSC values with error measures in Tables 4 and 5, although the values of errors are large, indicating the relatively worst performance of models developed. The lagged SSC values with one delay, i.e., SSC_{t-1} were utilized to feed the models in the second scenario as shown in Table 3. A performance evaluation of the models using the Q_t and SSC_{t-1} inputs clearly confirmed that SSC_{t-1} predictor is much more suitable than the Q_t input for obtaining more accurate predictions of SSC. Investigating the efficiency of all the models developed through applying both the Q_t and SSC_{t-1} input predictors illustrated that it can generally lead to better SSC predictions relative to previous two input combinations defined in some cases, while the performance of some models became slightly weaker. In the fourth scenario, Q_t and Q_{t-1} inputs were simultaneously utilized as the predictors when implementing the models (Table 3). The performance comparison of all the models developed under the first and fourth input patterns indicated the generally better SSC forecasts using both the Q_t and Q_{t-1} compared to utilizing only Q_t input. Similarly, taking into account two continuous delays of SSC, i.e., SSC_{t-1} and SSC_{t-2} led to represent reliable SSC predictions relative to using solely SSC_{t-1} in all the models at both the sites; however, this conclusion is clearly visible at USGS 07020500 Station (Table 5). Finally, all the models provided the highest accuracy of SSC predictions when applying complete inputs. It can be justified considering the fact that the models at sixth scenario utilized all the available information, including Q_t, Q_{t-1}, SSC_{t-1}, and SSC_{t-2}.

Here, a performance comparison of models is done taking into account the statistical measures achieved for the models as listed in Tables 4 and 5. In general, the RNN models performed better relative to BiRNN models in both the sites for SSC forecasting. As noted, the CNN-based hybrid DL models were also developed. Comparing the accuracy of baseline RNN and BiRNN with their hybrid forms, i.e., RNN-CNN and BiRNN-CNN models demonstrated that the hybrid BiRNN-CNN models at USGS 07010000 Station surpassed the individual BiRNN models. On the other hand, the hybrid RNN-CNN models at this site performed better than their standalone forms in some cases and vice versa performed poor in some others. For USGS 07020500 Station, hybrid CNN-based models did not necessarily represent better performance than their simple models, implying that there is no guarantee to improve the accuracy of simple DL models like RNN and BiRNN by their coupling with the CNN. Among the RNN-CNN and BiRNN-CNN hybrid models, BiRNN-CNN models generally illustrated enhanced error values in comparison to RNN-CNN methods. Along with the development of DL-based techniques, three forms of ML models were also established. Taking into account the performance of three ML models, it can be concluded that NGBoost and GBRT were found to be the best and worst models at USGS 07010000 Station, respectively. Again, NGBoost was the best-performing model among the ML techniques at USGS 07020500 Station. Moreover, GBRT models at first three scenarios and AdaBoost schemes

Table 4 Results of models in forecasting SSC during validation phase: USGS 07010000 station

Models	*R*	NSE	RMSE (mg/L)	MAE (mg/L)
RNN1	0.565	0.299	197.004	106.989
RNN2	0.921	0.841	93.724	57.199
RNN3	0.919	0.828	97.740	59.970
RNN4	0.567	0.312	195.179	108.427
RNN5	0.925	0.841	93.917	60.412
RNN6	**0.938**	**0.879**	**81.877**	**43.184**
BiRNN1	0.570	0.320	194.119	116.526
BiRNN2	0.920	0.825	98.366	47.711
BiRNN3	0.921	0.838	94.645	43.974
BiRNN4	0.609	0.305	196.231	109.329
BiRNN5	0.926	0.841	93.842	53.394
BiRNN6	**0.937**	**0.873**	**84.006**	**47.964**
RNN-CNN1	0.604	0.349	189.873	105.280
RNN-CNN2	0.917	0.838	94.734	48.167
RNN-CNN3	0.913	0.820	99.975	51.743
RNN-CNN4	0.592	0.349	189.849	110.659
RNN-CNN5	0.928	0.849	91.540	49.208
RNN-CNN6	**0.933**	**0.863**	**87.125**	**45.517**
BiRNN-CNN1	0.607	0.347	190.175	104.124
BiRNN-CNN2	0.922	0.840	94.088	57.105
BiRNN-CNN3	0.922	0.845	92.700	51.727
BiRNN-CNN4	0.604	0.360	188.302	114.076
BiRNN-CNN5	0.932	0.853	90.241	49.911
BiRNN-CNN6	**0.937**	**0.876**	**83.019**	**45.035**
AdaBoost1	0.606	0.364	187.698	112.363
AdaBoost2	0.902	0.811	102.387	53.762
AdaBoost3	0.901	0.807	103.408	55.419
AdaBoost4	0.602	0.359	188.457	113.262
AdaBoost5	0.902	0.809	102.843	54.870
AdaBoost6	**0.906**	**0.818**	**100.373**	**54.659**
NGBoost1	0.590	0.346	190.355	110.878
NGBoost2	0.902	0.808	103.245	48.119
NGBoost3	0.907	0.817	100.596	48.075
NGBoost4	0.615	0.378	185.680	107.573
NGBoost5	0.921	0.846	92.328	44.650
NGBoost6	**0.927**	**0.858**	**88.668**	**43.156**
GBRT1	0.537	0.286	198.913	115.073
GBRT2	0.871	0.758	115.851	57.634
GBRT3	0.880	0.773	112.057	55.840
GBRT4	0.553	0.284	199.221	113.329
GBRT5	0.886	0.780	110.497	50.984
GBRT6	**0.906**	**0.817**	**100.708**	**45.294**

in last three scenarios indicated the worst performance at this site. An assessment of DL and ML models also approved the superiority of DL models over the ML methods at USGS 07010000 Station. This result was not achieved as a general outcome at USGS 07020500 Station, as in some scenarios, DL models were superior to ML models, and in some other scenarios, ML models were slightly better relative to DL methods.

Table 5 Results of models in forecasting SSC during validation phase: USGS 07020500 station

Models	*R*	NSE	RMSE (mg/L)	MAE (mg/L)
RNN1	0.571	0.324	198.971	119.219
RNN2	0.946	0.887	81.344	51.559
RNN3	0.946	0.894	78.737	40.903
RNN4	0.583	0.323	199.122	113.630
RNN5	0.959	0.916	70.284	42.418
RNN6	**0.966**	**0.934**	**62.402**	**33.709**
BiRNN1	0.580	0.325	198.819	129.514
BiRNN2	0.946	0.886	81.632	43.385
BiRNN3	0.946	0.889	80.489	46.548
BiRNN4	0.644	0.411	185.767	113.830
BiRNN5	0.959	0.911	72.261	40.997
BiRNN6	**0.965**	**0.920**	**68.399**	**42.791**
RNN-CNN1	0.625	0.347	195.569	112.980
RNN-CNN2	0.941	0.848	94.487	49.240
RNN-CNN3	0.944	0.860	90.513	48.378
RNN-CNN4	0.628	0.394	188.505	119.916
RNN-CNN5	0.955	0.901	76.231	41.436
RNN-CNN6	**0.964**	**0.926**	**66.026**	**39.496**
BiRNN-CNN1	0.636	0.395	188.305	119.513
BiRNN-CNN2	0.945	0.876	85.361	51.294
BiRNN-CNN3	0.943	0.864	89.200	46.577
BiRNN-CNN4	0.635	0.402	187.256	115.854
BiRNN-CNN5	0.955	0.886	81.669	52.900
BiRNN-CNN6	**0.966**	**0.920**	**68.390**	**41.895**
AdaBoost1	0.634	0.401	187.287	115.795
AdaBoost2	0.933	0.871	86.904	51.417
AdaBoost3	0.933	0.869	87.511	51.749
AdaBoost4	0.634	0.402	187.184	116.729
AdaBoost5	0.934	0.872	86.688	51.220
AdaBoost6	**0.935**	**0.875**	**85.729**	**51.101**
NGBoost1	0.638	0.406	186.597	113.056
NGBoost2	0.941	0.886	81.605	42.491
NGBoost3	0.942	0.887	81.277	42.232
NGBoost4	0.674	0.453	179.029	108.100
NGBoost5	0.956	0.914	71.104	37.486
NGBoost6	**0.961**	**0.924**	**66.818**	**35.471**
GBRT1	0.612	0.371	191.969	117.030
GBRT2	0.926	0.850	93.819	52.324
GBRT3	0.931	0.857	91.577	51.964
GBRT4	0.648	0.418	184.671	110.534
GBRT5	0.950	0.902	75.794	40.848
GBRT6	**0.955**	**0.912**	**71.717**	**37.925**

The results obtained in the present investigation have direct implications for various operational water resources management, where real-time SSC monitoring aids in improving reservoir management. The superior results from the RNN relative to BiRNN indicate that the sediment transport processes we model are forward-dependent in time. This means that predicting the current state relies more heavily on past data than on future data, mak-

ing the simpler RNN architecture more effective than the bidirectional alternative for this specific task. Additionally, the BiRNN-CNN outperforms RNN-CNN, as the CNN captures short-term sediment peaks while the BiRNN's bidirectional memory improves multi-scale dynamic representation. Further research is needed for improving the prediction accuracy and decreasing the calculated errors.

The best-performing models in each category of the applied methods in the validation stage were selected at the studied locations to comparative evaluation of them schematically. As it is apparent in Tables 4 and 5, all the models implemented through the complete input predictors were found to be superior methods, which their error metrics are bolded in Tables 4 and 5. A summary of utilized error measures, including the R, NSE, RMSE, and MAE for the best models are depicted as radar plots, which are shown in Fig. 2. For USGS 07010000 Station, the RNN considering the R, NSE, RMSE, and NGBoost focusing on the MAE metric were found to be the superior methods, whereas AdaBoost using the R and MAE, and GBRT taking into account the NSE and RMSE illustrated the worst performance. Moreover, RNN and AdaBoost models generally provided the best and worst forecasts of SSC at USGS 07020500 Station, respectively.

The accuracy of implemented models can also be assessed focusing on the measured vs. predicted SSC data scatters in the scatter diagrams as illustrated in Fig. 3. As it can be seen, ML models developed like AdaBoost, NGBoost, and GBRT represented relatively higher scatters around the perfect line (i.e., $Y = X$) and therefore lower R^2 values, while slightly lower scatters in the DL models as well as higher R^2 denote their better performance relative to ML models at both the stations. In this respect, RNN at USGS 07010000 Station, and RNN and BiRNN-CNN at USGS 07020500 were the superior models. Conversely, the AdaBoost was found to be the worst model at both the stations.

Violin plots were also prepared that are given in Fig. 4. Among the selected superior models, the violin plot of AdaBoost is relatively far from the violin plot of measured SSC data at both the locations, illustrating its poor performance in forecasting the SSC time series. However, the violin plots of remaining models are closer to the measured SSC violins, denoting their better performance compared with AdaBoost. In addition, minimum (Min) and maximum (Max) values of the predicted vs. measured SSC values can also be evaluated in the violin plots as it is clear in Fig. 4. Regarding the Min value, the predicted Min SSC values of the RNN model (i.e., 26.12 mg/L at USGS 07010000 Station and 25.48 mg/L at USGS 07020500 Station) are much closer to the actual measured SSC values (i.e., 25.70 mg/L at USGS 07010000 Station and 23.60 mg/L at USGS 07020500 Station). Additionally, the predicted Max SSC value of the AdaBoost at USGS 07010000 Station (2449.41 mg/L) is much closer to the real measured Max value (2240 mg/L), whereas the remaining models at this site cannot predict the Max value well. The AdaBoost, NGBoost, GBRT, and RNN developed at USGS 07020500 Station didn't represent reliable capability to predict Max actual SSC value (1780 mg/L); however, the Max SSC value could be well forecasted through the RNN-CNN (1756.37 mg/L), BiRNN (1764.43 mg/L), and BiRNN-CNN (1771.74 mg/L).

Besides the radar, scatter, and violin plots, the output of Taylor diagram was also extracted as shown in Fig. 5. A Taylor diagram is a visual technique utilized to schematically evaluate the models performances. The outcomes of three error measures comprising of root mean square difference, correlation coefficient, and standard deviation are merged in one plot named Taylor diagram. In this graph, the performance of each model

Fig. 2 Radar charts summarizing the obtained error metrics: (**a**) USGS 07010000 Station, (**b**) USGS 07020500 Station

is demonstrated with a specific point, being far away or close of the relevant point of any model compared to the reference point of measured data denote the worst and better accuracy of the models, respectively. As it is apparent in Fig. 5, the points of AdaBoost and GBRT models at USGS 07010000 Station, and AdaBoost at USGS 07020500 Station illustrated the highest distance with the reference point (i.e., red triangle), indicating their poor performances. In contrast, the RNN models at both the locations exhibited the best performance due to the lowest distance.

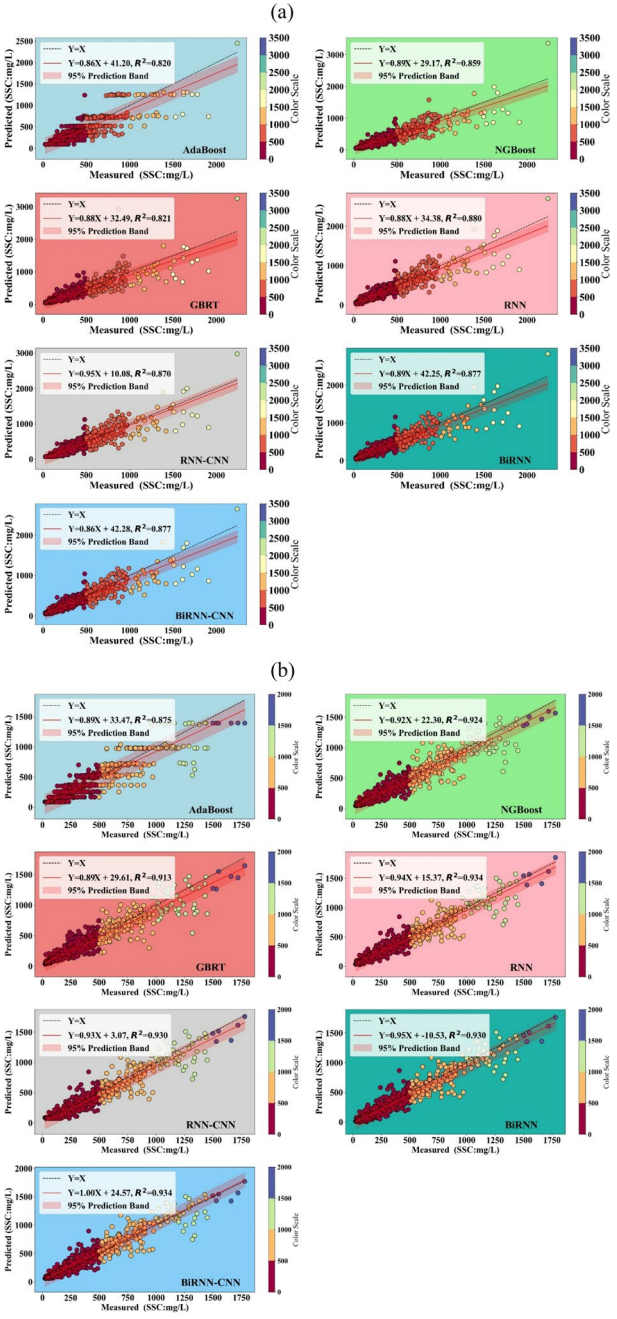

Fig. 3 Scatter plots for the superior models: (**a**) USGS 07010000 Station, (**b**) USGS 07020500 Station

At the end part of this study, the contributions of input features on the models outputs were evaluated considering the LIME outcomes as depicted in Fig. 6. The LIME plots were prepared for the RNN models among the whole developed methods as they showed the best performance at both the sites. It is worthy to note that the LIME could represent the local

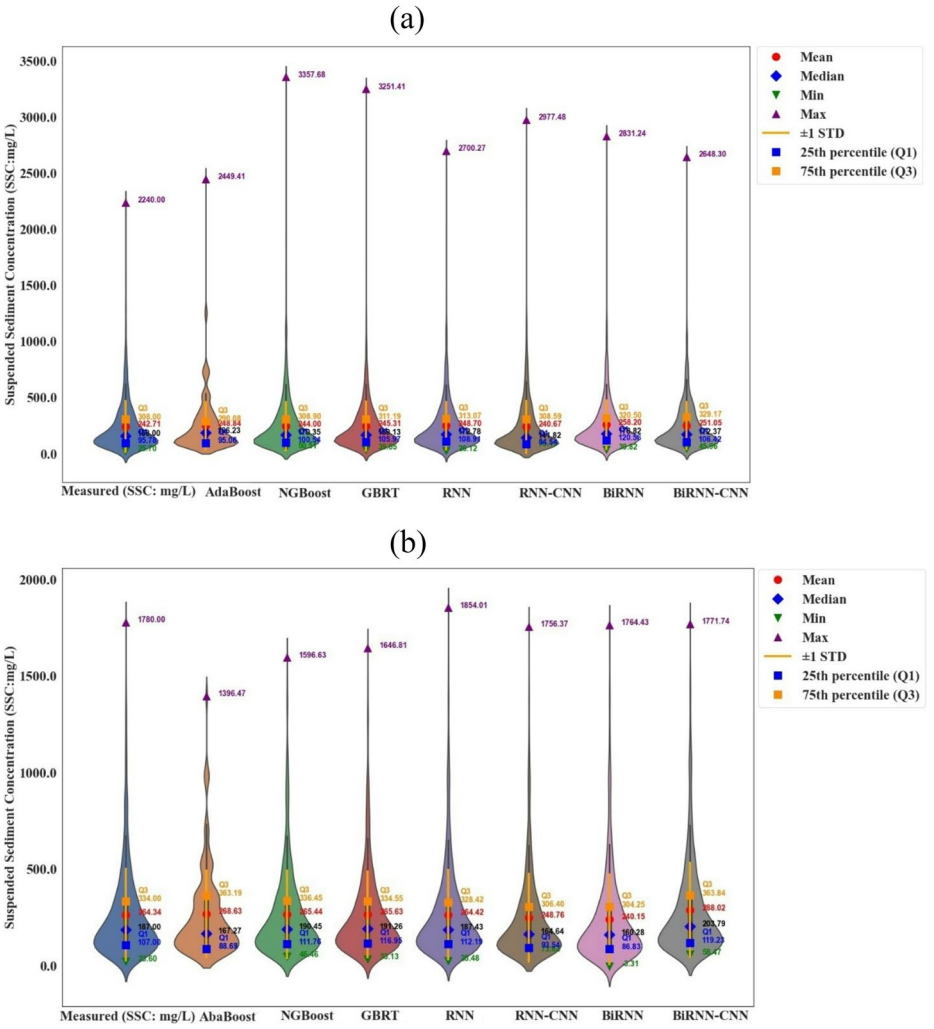

Fig. 4 Violin plots for the superior models: (**a**) USGS 07010000 Station, (**b**) USGS 07020500 Station

explainability for the specific prediction samples. For this, we selected six samples, including 1, 300, 500, 1000, 1500, and 2000 in the validation dataset. The output in LIME consists of three various sections. The left side indicates the predicted value, and min and max denote the minimum and maximum values for the forecast interval. For instance, in sample 1000, the predicted value is 79.56 mg/L with intervals of 77.32 mg/L and 2780.34 mg/L at USGS 07010000 Station, whereas they are respectively as 111.55 mg/L, 93.38 mg/L, and 1837.78 mg/L at USGS 07020500 Station. The middle section of LIME illustrates the input features negatively or positively impacting the outputs. In this context, the features highlighted with blue and orange colors lead to decreasing and increasing the predicted value, respectively. For an example, in sample 1000 of USGS 07020500 Station, $SSC_{t-1} <= 109.00$ and $Q_t <= 3744.90$ are the input predictors negatively influencing the predicted value,

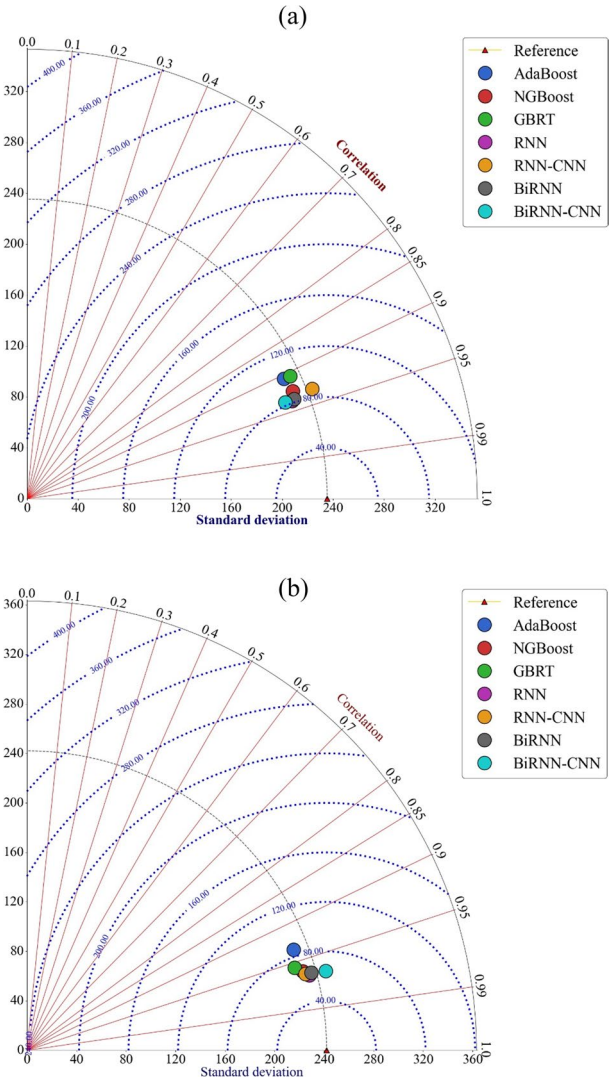

Fig. 5 Taylor diagrams for the superior models: (**a**) USGS 07010000 Station, (**b**) USGS 07020500 Station

whereas $Q_{t-1} <= 3766.14$ and $SSC_{t-2} <= 108.00$ are the inputs positively affecting the predicted SSC_t value. It is also better to mention that the input features in the middle section of LIME are ranked from the most important input to least important parameter. Investigating the LIME plots exhibited that Q_t at USGS 07010000 Station and SSC_{t-1} at USGS 07020500 Station have generally shown the highest impacts for all the samples, respectively. Furthermore, SSC_{t-2} was found to provide the least importance at both the stations and selected samples. Finally, the right side of LIME plots shows the input features and their actual values. For instance, the values of Q_t, Q_{t-1}, SSC_{t-1}, and SSC_{t-2} inputs equal to 3114.85, 3029.90, 46.80, and 47.20 at sample 2000 of USGS 07010000 Station contribute to the predicted value of SSC_t = 78.57 mg/L. As a result, the outputs of LIME could provide useful information to better discern the contributions of inputs in the models output. In this study, for predicting current daily SSC, the daily lagged time series of feature data were selected

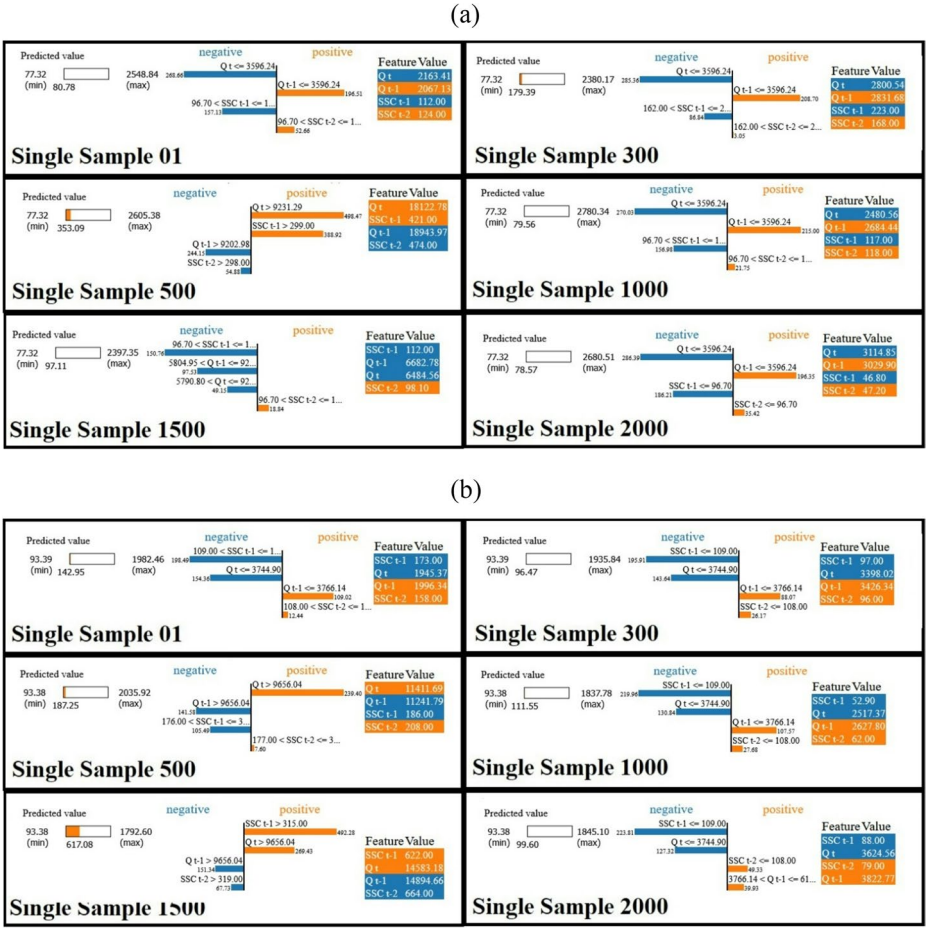

Fig. 6 LIME plots for the superior RNN models: (**a**) USGS 07010000 Station, (**b**) USGS 07020500 Station

(i.e., short-term lagged time series with high autocorrelation). It is necessary to apply LIME utilizing highly correlated lagged data (Seo et al. 2015) for feature importance analysis on impacting the positive or negative outputs. Also, the robustness of lagged time series for feature importance analysis must be confirmed by comparing the results of LIME with those of SHAP (SHapley Additive exPlanations), including the global and local interpretation (Lundberg and Lee 2017).

4 Conclusion

The present study tried to make an attempt to forecast the daily time series of SSC at two river stations (i.e., USGS 07010000 and USGS 07020500) located on the Mississippi River, United States. To achieve this aim, seven kinds of AI models, including four DL and three ML techniques were developed. The DL models included the RNN, BiRNN, RNN-CNN,

and BiRNN-CNN, whereas the ML models consisted of AdaBoost, NGBoost, and GBRT. Six input configurations were defined when developing the whole models using Q_t, Q_{t-1}, SSC_{t-1}, and SSC_{t-2}. Between the two baseline DL models, RNN illustrated better outcomes than BiRNN. Conversely, BiRNN-CNN hybrid models outperformed the hybrid RNN-CNN ones in SSC forecasting. A performance comparison of baseline and hybrid CNN-based models demonstrated that the hybrid BiRNN-CNN provided better prediction results compared with standalone BiRNN at USGS 07010000 Station, while this achievement was not a general fix outcome at RNN-CNN models at USGS 07010000 as well as all the DL models at another site. This means that it cannot be a guarantee to obtain the better prediction results via the hybrid CNN-based DL models relative to single DL ones. Among the ML models, the NGBoost and GBRT were found to present the highest and lowest accuracies, respectively. In general, the outcomes of LIME demonstrated that Q_t at USGS 07010000 and SSC_{t-1} at USGS 07020500 were found to represent the highest impacts on the models forecasts, respectively. Furthermore, the least-influencing input feature at both the stations was SSC_{t-2}. It is a necessity for river engineers to be aware of the rivers SSC values for better sediment management. In this context, the implemented models in this study could be beneficial tools for precisely predicting the rivers SSC.

This study focused on developing three ML and four DL models in SSC forecasting. It is suggested that other various types of ML and DL methods could be utilized in future and the outcomes compared with this research. Diverse types of ML and DL techniques are also recommended to be hybridized with signal decomposition methods and optimization algorithms to propose hybrid predictive SSC tools. Focusing on DL frameworks, it is suggested that diverse baseline DL models like LSTM, BiLSTM, transformers, as well as their hybrids with CNN and RNN could be implemented when predicting the rivers SSC values. This work applied the lagged SSC and river streamflow as the possible input predictors as they were available at the studied locations. It is therefore recommended that other river information like river water quality data and climatic data such as temperature and rainfall could be utilized as the new inputs to further improve the performance of predictive SSC models.

Author contributions Amin Gharehbaghi: Conceptualization, Formal analysis, Investigation, Validation, Writing – original draft. Salim Heddam: Software, Formal analysis, Validation, Writing – original draft, Writing – review & editing. Saeid Mehdizadeh: Supervision, Data curation, Investigation, Writing – original draft, Writing – review & editing. Sungwon Kim: Methodology, Writing – original draft, Writing – review & editing.

Funding Not applicable.

Data Availability Data will be available upon reasonable request.

Declarations

Ethics approval Not Applicable.

Consent to publish All authors agree to publish.

Consent to participate Not Applicable.

Competing interests The authors declare no competing interests.

References

Afan HA, Wan Mohtar WHM, Aksoy M, Ahmed AN, Khaleel F, Khan MMH, El-Shafie A (2025) A multi-functional genetic algorithm-neural network model for predicting suspended sediment loads. Water Resour Manag 39(5):2033–2048

Alizamir M, Gholampour A, Kim S, Keshtegar B, Jung W-t (2024) Designing a reliable machine learning system for accurately estimating the ultimate condition of FRP-confined concrete. Sci Rep 14(1):20466

Alizamir M, Moradveisi K, Ahmed KO, Bahrami J, Kim S, Heddam S (2025) An efficient data fusion model based on Bayesian model averaging for robust water quality prediction using deep learning strategies. Expert Syst Appl 261:125499

Bahrambanan F, Alizamir M, Moradveisi K, Heddam S, Kim S, Kim S et al (2025) The development of an efficient artificial intelligence-based classification approach for colorectal cancer response to radiochemotherapy: deep learning vs. machine learning. Sci Rep 15(1):62

Barzegar R, Aalami MT, Adamowski J (2020) Short-term water quality variable prediction using a hybrid CNN–LSTM deep learning model. Stoch Environ Res Risk Assess 34(2):415–433

Bezak N, Lebar K, Bai Y, Rusjan S (2025) Using machine learning to predict suspended sediment transport under climate change. Water Resour Manag 39(7):3311–3326

Busari GA, Lim DH (2021) Crude oil price prediction: a comparison between AdaBoost-LSTM and AdaBoost-GRU for improving forecasting performance. Comput Chem Eng 155:107513

Chung J, Gulcehre C, Cho K, Bengio Y (2014) Empirical evaluation of gated recurrent neural networks on sequence modeling. ArXiv Preprint arXiv:1412.3555

Daif N, Di Nunno F, Granata F, Difi S, Kisi O, Heddam S et al (2025) Forecasting maximal and minimal air temperatures using explainable machine learning: Shapley additive explanation versus local interpretable model-agnostic explanations. Stoch Environ Res Risk Assess 39(6):2551–2581

Danesh M, Gharehbaghi A, Mehdizadeh S, Danesh A (2025) A comparative assessment of machine learning and deep learning models for the daily river streamflow forecasting. Water Resour Manag 39(4):1911–1930

Dash SK, Roccotelli M, Khansama RR, Fanti MP, Mangini AM (2021) Long term household electricity demand forecasting based on RNN-GBRT model and a novel energy theft detection method. Appl Sci 11(18):8612

Dehkordi AT, Ghasemi H, Zoej MJV (2021) Machine learning-based estimation of suspended sediment concentration along Missouri River using remote sensing imageries in Google Earth Engine. Paper presented at the 2021 7th International Conference on Signal Processing and Intelligent Systems (ICSPIS)

Dehkordi AT, Zoej MJV, Jafari M, Ghasemi H, Mehran A (2024) Leveraging remote sensing data and machine learning models to estimate suspended sediment concentration (SSC), a vital water quality parameter to assess soil erosion effects. Advanced Tools for Studying Soil Erosion Processes. Elsevier, pp 97–114

Ding H, Yang L, Cheng Z, Yang Z (2021) A remaining useful life prediction method for bearing based on deep neural networks. Measurement 172:108878

Dong X, Qian L (2022) Semi-supervised bidirectional RNN for misinformation detection. Machine Learning with Applications 10:100428

Fan J, Liu X, Li W (2023) Daily suspended sediment concentration forecast in the upper reach of Yellow River using a comprehensive integrated deep learning model. J Hydrol 623:129732

Fang X, Zhang J, Yu X, Zhang S, Kong D, Wang X et al (2025) Accurate estimation of suspended sediment concentration integrated remote sensing information and a novel stacking machine learning model. Stoch Environ Res Risk Assess 39(4):1517–1535

Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. J Comput Syst Sci 55(1):119–139

Friedman JH (2001) Greedy function approximation: a gradient boosting machine. Ann Stat 1189–1232

Gupta A, Feng D (2025) Regional scale simulations of daily suspended sediment concentration at gauged and ungauged rivers using deep learning. J Hydrol 660:133111

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780

Huang C-C, Chang M-J, Lin G-F, Wu M-C, Wang P-H (2021) Real-time forecasting of suspended sediment concentrations in reservoirs by the optimal integration of multiple machine learning techniques. J Hydrol Reg Stud 34:100804

Jarbais G, Harshavardhanan P (2025) Comparative analysis of machine learning models for daily suspended sediment concentration prediction in environmental monitoring. Sādhanā 50(2):63

Joshi B, Singh VK, Vishwakarma DK, Ghorbani MA, Kim S, Gupta S et al (2024) A comparative survey between cascade correlation neural network (CCNN) and feedforward neural network (FFNN) machine learning models for forecasting suspended sediment concentration. Sci Rep 14(1):10638

Kaveh K, Kaveh H, Bui MD, Rutschmann P (2021) Long short-term memory for predicting daily suspended sediment concentration. Eng Comput 37:2013–2027
Kim HS, He M, Sandhu P (2022) Suspended sediment concentration estimation in the Sacramento-San Joaquin Delta of California using long short-term memory networks. Hydrol Process 36(10):e14694
Kumar S, Deshpande V, Agarwal M (2025) Standalone and hybrid machine learning approaches to predict sediment load in an alluvial channel. Eng Appl Artif Intell 150:110578
LeCun Y, Boser B, Denker J, Henderson D, Howard R, Hubbard W, Jackel L (1989) Handwritten digit recognition with a back-propagation network. Adv Neural Inf Proc Syst 2
Li S, Yang J (2022) Modelling of suspended sediment load by bayesian optimized machine learning methods with seasonal adjustment. Eng Appl Comput Fluid Mech 16(1):1883–1901
Liu H, Tian H-q, Li Y-f, Zhang L (2015) Comparison of four adaboost algorithm based artificial neural networks in wind speed predictions. Energy Convers Manage 92:67–81
Lundberg SM, Lee SI (2017) A unified approach to interpreting model predictions. Adv Neural Inf Process Syst 30
Mitrentsis G, Lens H (2022) An interpretable probabilistic model for short-term solar power forecasting using natural gradient boosting. Appl Energy 309:118473
Momeny M, Neshat AA, Gholizadeh A, Jafarnezhad A, Rahmanzadeh E, Marhamati M et al (2022) Greedy Autoaugment for classification of mycobacterium tuberculosis image via generalized deep CNN using mixed pooling based on minimum square rough entropy. Comput Biol Med 141:105175
Moradinejad A (2024) Suspended load modeling of river using soft computing techniques. Water Resour Manag 38(6):1965–1986
Neves I, Folgado D, Santos S, Barandas M, Campagner A, Ronzio L et al (2021) Interpretable heartbeat classification using local model-agnostic explanations on ECGs. Comput Biol Med 133:104393
Nguyen SQ, Nguyen LC, Ngo-Duc T, Ouillon S (2024b) Applying a machine learning-based method for the prediction of suspended sediment concentration in the red river basin. Model Earth Syst Environ 10(2):2675–2692
Nguyen TTH, Vu DQ, Doan NP, Chi HTK, Li P, Van Binh D et al (2024a) Reconstructing suspended sediment concentrations in the Mekong River Basin via semi-supervised-based deep neural networks. Sci Total Environ 955:176758
Nie P, Roccotelli M, Fanti MP, Ming Z, Li Z (2021) Prediction of home energy consumption based on gradient boosting regression tree. Energy Rep 7:1246–1255
Peng L, Lv SX, Wang L (2024) Explainable machine learning techniques based on attention gate recurrent unit and local interpretable model-agnostic explanations for multivariate wind speed forecasting. J Forecast 43(6):2064–2087
Persson C, Bacher P, Shiga T, Madsen H (2017) Multi-site solar power forecasting using gradient boosted regression trees. Sol Energy 150:423–436
Pham Van C, Le H, Van Chin L (2023) Estimation of daily suspended sediment concentration in the Ca River Basin using a sediment rating curve, multiple regression, and long short-term memory model. J Water Clim Change 14(12):4356–4375
Roushangar K, Alirezazadeh Sadaghiani A (2025) Innovative deep learning and signal decomposition approaches for enhanced Spatial and Temporal suspended sediment concentration prediction. Environ Sci Pollut Res 1–26
Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323(6088):533–536
Saroughi M, Katipoğlu OM, Kartal V, Simsek O, Kilinc HC, Pande CB (2025) Developing sediment concentration prediction in the Euphrates River catchment, Türkiye, with a honey Badger and Coati optimization-based hybrid algorithm. Environ Monit Assess 197(7):1–30
Schuster M, Paliwal KK (1997) Bidirectional recurrent neural networks. IEEE Trans Signal Process 45(11):2673–2681
Seo Y, Kim S, Singh VP (2015) Multistep-ahead flood forecasting using wavelet and data-driven methods. KSCE J Civ Eng 19(2):401–417
Shen K, Qin H, Zhou J, Liu G (2022) Runoff probability prediction model based on natural gradient boosting with tree-structured Parzen estimator optimization. Water 14(4):545
Song Y, Chaemchuen P, Rahmani F, Zhi W, Li L, Liu X et al (2024) Deep learning insights into suspended sediment concentrations across the conterminous United States: strengths and limitations. J Hydrol 639:131573
Tong W, Li L, Zhou X, Hamilton A, Zhang K (2019) Deep learning PM2.5 concentrations with bidirectional LSTM RNN. Air Qual Atmos Health 12(4):411–423
Tsai J-K, Hung C-H (2021) Improving AdaBoost classifier to predict enterprise performance after COVID-19. Mathematics 9(18):2215

Weidong L, Suhayb MK, Thangavelu L, Marhoon HA, Pustokhina I, Alqsair UF et al (2022) Implementation of adaboost and genetic algorithm machine learning models in prediction of adsorption capacity of nanocomposite materials. J Mol Liq 350:118527

Wikle CK, Datta A, Hari BV, Boone EL, Sahoo I, Kavila I et al (2023) An illustration of model agnostic explainability methods applied to environmental data. Environmetrics 34(1):e2772

Xu Y, Yan X, Wu Y, Hu Y, Liang W, Zhang J (2021) Hierarchical bidirectional RNN for safety-enhanced B5G heterogeneous networks. IEEE Trans Netw Sci Eng 8(4):2946–2957

Yang M, Xu C, Bai Y, Ma M, Su X (2023) Investigating black-box model for wind power forecasting using local interpretable model-agnostic explanations algorithm. CSEE J Power Energy Syst 11(1):227–242

Yang Y, Xiong Q, Wu C, Zou Q, Yu Y, Yi H, Gao M (2021) A study on water quality prediction by a hybrid CNN-LSTM model with attention mechanism. Environ Sci Pollut Res 28(39):55129–55139

Zafar MR, Khan N (2021) Deterministic local interpretable model-agnostic explanations for stable explainability. Mach Learn Knowl Extr 3(3):525–541

Zhang M, Zhang Z, Wang X, Liao Z, Wang L (2024) The use of attention-enhanced CNN-LSTM models for multi-indicator and time-series predictions of surface water quality. Water Resour Manage 38(15):6103–6119

Zhang X, Yang Y (2020) Suspended sediment concentration forecast based on CEEMDAN-GRU model. Water Supply 20(5):1787–1798

Zhou Z, Cao J, Shi X, Zhang W, Huang W (2024) Probabilistic rutting model using NGBoost and SHAP: incorporating other performance indicators. Constr Build Mater 438:137052