Journal of Water Process Engineering 79 (2025) 109013

Contents lists available at ScienceDirect

Journal of Water Process Engineering

journal homepage: www.elsevier.com/locate/jwpe

Development of explainable hybrid quantum-inspired recurrent neural networks for predicting groundwater quality: A case study at West Azerbaijan, Iran

Milad Sharafi [a], Amin Gharehbaghi [b], Saeid Mehdizadeh [a,*]

[a] Department of Water Engineering, Urmia University, Urmia, Iran
[b] Department of Civil Engineering, Faculty of Engineering, Hasan Kalyoncu University, 27110, Şahinbey, Gaziantep, Türkiye

ARTICLE INFO

Editor: Soroush Abolfathi

Keywords:
Total dissolved solids prediction
West Azerbaijan Province
Quantum inspired recurrent neural network
Hybrid models
SHapley additive explanation

ABSTRACT

Effective groundwater quality monitoring is essential for ensuring sustainable water resource management. Total dissolved solids (TDS) is a key indicator of groundwater quality. This study presents advanced hybrid deep learning frameworks for forecasting TDS levels at West Azerbaijan, Iran. Six diverse input combinations of pH, Mg, total alkalinity, HCO_3, Ca, and total hardness were defined. A quantum-inspired recurrent neural network (QRNN) was first developed as baseline model. Two hybrid models, QRNN-CNN and GBO-QRNN-CNN, were then introduced to enhance predictive performance by integrating convolutional neural network (CNN) and gradient-based optimizer (GBO). Comparative evaluations demonstrated that both hybrid models outperformed the standalone QRNN, with GBO-QRNN-CNN achieving the highest accuracy. Root mean square error of TDS prediction via the best model (GBO-QRNN-CNN-6) was reduced by 48.36 % compared with baseline QRNN-6. Moreover, long short-term memory (LSTM) was implemented, denoting its lower accuracy than QRNN-based models. Additionally, SHapley Additive exPlanation (SHAP) was employed to assess the influence of input variables, revealing that total hardness and calcium had the highest impacts on TDS predictions. The QRNN frameworks proposed in this study taking into account the outcomes of SHAP, offer powerful TDS predictive tools for data-driven groundwater quality assessment, supporting more informed decision-making in water resource management.

1. Introduction

Groundwater is among one of the main natural sources of available water utilized for several applications such as agriculture, irrigation, drinking, and industry [1–3]. By determining the quality of groundwater, an estimate of the health of these groundwater resources can be made, and, in accordance with its properties, its type of use can be specified [4]. Recently, several factors comprising of population growth, urbanization, climate change, and improper management of water resources have led to deteriorate and therefore reduce the quality of groundwater resources [5–7].

A diverse group of variables can be used to monitor and consequently manage the quality of groundwater resources, including pH [8], total hardness [9], sulfate [10], electrical conductivity, EC [11], total dissolved solids, TDS [12]. Of them, the amount of TDS could provide helpful information about the groundwater salinity, as one of the primary risks for groundwater resources, specifically in arid regions [12,13]. Forecasting groundwater TDS with a dependable degree of precision is required in related applications for hydrological and agricultural researchers, to protect fresh groundwater, as well as protection and sustainability of groundwater resources [12,14].

Direct and indirect techniques could be employed to specify the groundwater quality like TDS. Despite their high accuracy, direct laboratory methods are expensive to use, time-consuming, and require extensive calculations. Alternatively, other types of indirect methods, including numerical and deterministic systems could be applied in groundwater quality forecasting [12,15]. However, some problems consisting the aquifer's heterogeneity, spatiotemporal diversity of groundwater quality variables, and complicated hydrochemical processes have caused substantial challenges at using these models [16]. Due to the listed drawbacks, the researchers have recently been encouraged to use alternative methods, including artificial intelligence

* Corresponding author.
E-mail address: s.mehdizadeh@urmia.ac.ir (S. Mehdizadeh).

https://doi.org/10.1016/j.jwpe.2025.109013
Received 22 August 2025; Received in revised form 13 October 2025; Accepted 26 October 2025
Available online 28 October 2025

(AI) techniques. Compared to the direct prediction systems, the AI methods are affordable and time saving [17,18]. Handling complex and non-linear processes in any datasets like hydro-climatological data could be a challenging issue when developing machine learning (ML)-based models, whereas they could be well extracted by deep learning (DL) techniques through automatic feature extraction in time series data and discovery of their hidden patterns [19–21].

Literature review clearly showed that groundwater quality modeling using AI has been addressed by the literature. In this respect, various AI frameworks like DL and ML schemes have been extensively utilized for predicting the different groundwater quality variables [22–26]. For instance, Yassin et al. [14] forecasted the groundwater salinity on the basis of TDS data via the development of four ML models, and found out that Gaussian process regression (GPR) outperformed the other models. Ewusi et al. [15] modeled the TDS values of water supply systems including the groundwater, surface water, and drinking water through applying some ML methods, and concluded that the GPR and back-propagation neural network (BPNN) were the best-performing models for water quality indices forecasting. Six different optimizers were used by Banadkooki et al. [27] to couple on three ML models for prediction of groundwater TDS. The results illustrated that the hybrid models performed better than their relevant simple forms. El Bilali et al. [28] predicted six groundwater quality variables, including the TDS, salinity, etc. via the establishment of four ML models, and confirmed the better potential of Adaboost and random forest (RF) compared to the support vector regression (SVR) and artificial neural networks (ANN). Aryafar et al. [29] recommended genetic programming (GP) for groundwater TDS forecasting, and reported that the GP surpassed both the ANN and adaptive neuro-fuzzy inference system (ANFIS), and therefore could be utilized as a promising forecasting tool. Pan et al. [30] developed a hybrid of principal component regression (PCR) and dual-step multiple linear regression (MLR), as well as a BPNN, for modeling the groundwater TDS. Their findings revealed that the hybrid of PCR and dual-step MLR illustrated better TDS values than the BPNN. Elzain et al. [12] implemented three versions of tree-based ML models for predicting groundwater TDS levels, and concluded that the Bagging regression (BR) was found to be the best model. Abba et al. [31] coupled an ANFIS on three meta-heuristic algorithms for modeling the TDS of a coastal aquifer, and stated that the developed hybrid models demonstrated better TDS estimates than the simple ANFIS. Two groundwater quality variables such as TDS and pH were predicted by Mahmoudi et al. [32] applying three ML techniques and their merged forms in bagging (BG) and boosting (BT) structures. Their outcomes demonstrated that the CHAID-BT (i.e., Chi-square automatic interaction detection- boosting) was the most precise method when forecasting the TDS. Gulati et al. [33] estimated TDS of groundwater through developing three SVMs, three ANNs, and an ANFIS. The findings demonstrated that the ANN optimized by Bayesian regularization represented the highest accuracy. Farooq et al. [34] applied MLR and ANNs for predicting groundwater TDS levels, and reported the superiority of ANN over conventional MLR. Jamshidzadeh e al. [11] implemented two standalone GPR and LSTM, as well as four hybrid models for groundwater TDS forecasting. Their findings revealed that the CNN-LSTM-GPR was found to illustrate better TDS forecasts relative to other methods.

As outlined, forecasting the groundwater quality variables could be done taking into account the various variables such as pH, EC, nitrate, sulfate, TDS, etc. In this context, less attention has been paid to TDS during the development of intelligent systems. Hence, the current study aimed at predicting the groundwater TDS levels at some urban water supply wells located in West Azerbaijan Province, Iran, through developing four types of DL models. Firstly, a Long Short-Term Memory (LSTM) was developed. Then, a standalone Quantum-Inspired Recurrent Neural Network (QRNN) was implemented. It was also tried to improve the performance of QRNN by merging it with Convolutional Neural Network (CNN) and Gradient-Based Optimizer (GBO) to propose two types of hybrid QRNN-based model, including QRNN-CNN and GBO-QRNN-CNN. Six diverse input configurations were considered utilizing other groundwater quality variables. Investigating the model's accuracies was conducted through applying some error measures and visual comparative diagrams. SHapley Additive exPlanation (SHAP) technique was finally applied to evaluate how the input predictors could affect the output results of the developed modes. It is worthwhile to say that DL frameworks have been used less when forecasting the TDS levels of groundwater in preceding works, while this study proposed a standalone LSTM, QRNN and its hybrid versions with CNN and GBO to propose novel QRNN-CNN and GBO-QRNN-CNN hybrid schemes. Additionally, a detailed literature review revealed that there was a knowledge gap in the development of quantum-inspired DL models for predicting groundwater quality variables such as TDS; however, this study attempted to overcome this gap by proposing baseline QRNN, as well as novel QRNN-CNN and GBO-QRNN-CNN hybrid frameworks. Furthermore, the application of SHAP in analyzing the impacts of inputs on the output results of models is another innovation of this research, which has not been used so far in previous literature for predicting groundwater TDS levels.

2. Materials and methods

2.1. Study region and data used description

This study utilized a dataset comprising 312 samples collected from groundwater wells supplying urban water resources in West Azerbaijan Province, Iran, between 2020 and 2024. Before modeling, the dataset was carefully preprocessed to ensure quality and consistency. The dataset was free of missing and outlier values. To ensure that features with different scales did not unduly influence the learning process, all input variables were normalized using *Z*-scores. Each sample includes measurements of six physicochemical parameters: potential of hydrogen (pH), magnesium (Mg), total alkalinity, bicarbonate (HCO_3), calcium (Ca), and total hardness. The target variable is total dissolved solids (TDS).

The dataset was obtained from the West Azerbaijan Province Water and Wastewater Company. To ensure data quality and consistency, preprocessing steps such as standardization were performed. In the absence of an external dataset from a different region or time period, model stability and generalization were assessed through repeated random sub-sampling. Specifically, the modeling process was conducted over six independent runs, each using different random splits of the dataset into training (67 %, 209 samples) and testing (33 %, 103 samples) subsets. Performance metrics were recorded for each run, and consistent results across repetitions provided confidence in the model's robustness. Although k-fold cross-validation was not explicitly applied, this repeated evaluation strategy allowed each sample to contribute to multiple train-test scenarios, approximating the benefits of cross-validation and mitigating the risk of overfitting. Additionally, for hyperparameter optimization using the Gradient-Based Optimizer (GBO), the dataset was split into training and testing sets as described above. The test data remained completely unseen during model development and was used solely for final evaluation. Reported metrics reflect the aggregated results from all runs, offering a comprehensive assessment of model performance. The statistical criteria of the parameters used, including mean (Mean), minimum (Min), maximum (Max), standard deviation (Std), and coefficient of variation (CV), are given in Table 1.

2.2. Models

2.2.1. Long short-term memory (LSTM) neural network

The Long Short-Term Memory (LSTM) network is a complex recurrent neural network (RNN) specially designed to learn long-term dependencies and temporal sequences better than standard RNNs [35]. In contrast to the standard RNNs, which are plagued by vanishing

Table 1
Statistical criteria for utilized parameters.

Parameter	Unit	Statistical criterion				
		Mean	Min	Max	Std	CV
pH	mg/l	7.42	6.53	8.05	0.27	0.04
Mg	mg/l	26.99	7.80	75.60	8.81	0.33
Total alkalinity	mg/l as $CaCO_3$	282.48	122.40	524.00	64.84	0.23
HCO_3	mg/l	343.63	149.30	639.30	78.61	0.23
Ca	mg/l	86.36	11.04	140.80	23.16	0.27
Total hardness	mg/l as $CaCO_3$	327.79	136.00	500.00	76.81	0.23
TDS	mg/l	452.52	0.00	922.00	114.49	0.25

gradients during training, the LSTM networks possess a memory cell architecture that retains information over long time intervals [36]. The characteristic architecture renders LSTM very suitable for forecasting problems. At the center of each LSTM cell is a three-gated memory cell: input gate, forget gate, and output gate [36]. The gates control the flow of the information, and the network determines what values will be retained, what needs to be forgotten, and what will be output. The below are the operations performed in an LSTM unit:

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

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

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

$$C_t = f_t^{*} C_{t-1} + i_t^{*} \widetilde{C}_t \tag{4}$$

$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t^{*} \tanh(C_t) \tag{6}$$

where, $\sigma$is the sigmoid activation function and x_t is the input vector at time t. The hidden state, and the state of the cell at the previous time are represented as h_{t-1} and C_{t-1}, also, i_t, f_t, and o_t denote respectively input, forget, and output gates, and $\widetilde{C}_t$ is the cell state. *Tanh* is the hyperbolic tangent activation function, and b, W are respectively biases and weights.

LSTM models are more capable of dealing with nonlinear relationships and noisy data compared to conventional statistical approaches, leading to higher prediction accuracy. LSTM networks, nevertheless, have drawbacks like high computation expenses, sensitivity to parameter optimization, and requirements for large datasets in order to function optimally. In spite of such constraints, their applicability in the prediction of TDS is in their strength and ability to characterize intricate temporal dynamics, rendering them a useful tool in environmental monitoring [37], suspended sediment load concentration [38], and water quality prediction [39].

2.2.2. Quantum-inspired recurrent neural network (QRNN)

Quantum-Inspired Recurrent Neural Networks (QRNNs) are an advanced form of recurrent architectures that integrate principles derived from quantum mechanics, such as superposition, unitary transformations, and phase interference, into classical neural network models. These mechanisms are designed to enhance memory retention and learning stability, particularly when modeling sequential or time-dependent data. In contrast to conventional RNNs, which utilize real-valued hidden states and weight matrices, QRNNs represent hidden states using complex-valued vectors and evolve them through unitary transformation matrices. This structure ensures that information is preserved across long time steps, addressing the vanishing or exploding gradient problems commonly associated with traditional RNNs [40].

One key innovation in QRNNs is the representation of hidden states using quantum probability amplitudes rather than classical deterministic values. The transition between states follows a unitary transformation [41], preserving information across long-term dependencies more efficiently than traditional RNNs. Mathematically, the update of the hidden state in a QRNN can be formulated as:

$$h_t = U(\theta).h_{t-1} + V(\theta).x_t \tag{7}$$

where, h_t represents the hidden state at time t, x_t is the input at time t, $U(\theta)$and $V(\theta)$ are unitary transformation matrices parameterized by θ, ensuring reversible and stable state evolution. The unitary nature of $U(\theta)$ prevents information loss, overcoming key limitations of classical RNN architectures.

Another key distinction lies in the nonlinear activation function. Classical RNNs typically employ activation functions like tanh or sigmoid, which are defined over the real domain [42]. However, because QRNN operates in the complex domain, it utilizes nonlinearities designed for complex-valued inputs. A prominent example, modReLU, is given in Eq. (8) as:

$$\mathrm{modRe}LU(z) = \mathrm{Re}LU(|z| + b).\left(z/|z|\right) \tag{8}$$

where, $z \in \mathbb{C}$ is the complex activation, $|z|$is its magnitude, and b is a learnable bias term. This formulation allows the *model* to encode both magnitude and phase information, enhancing its ability to model temporal dependencies in hydro chemical systems.

In this study, QRNN was implemented using sequences derived from physicochemical water quality data across multiple years (2020–2024), and it demonstrated improved predictive accuracy, reduced training instability, and better generalization compared to classical RNN architectures. The QRNN model, inspired by the principles of quantum computing, introduces a probabilistic and interference-based mechanism into the recurrent structure, enabling more expressive modeling of sequential uncertainty and long-range dependencies. Unlike standard RNNs, QRNNs leverage quantum state amplitudes and phase rotations, which allows the model to learn richer representations of sequential patterns with fewer parameters and potentially better generalization. These advantages make QRNN a compelling choice for forecasting in environmental and water resource applications. In summary, the difference between QRNN and conventional RNNs can be summarized in four cases: (1) use of unitary transformations (norm preservation), (2) operation in complex number space, (3) use of quantum-inspired activations (modReLU), and (4) increased capacity to model long-range dependencies without fading gradients. Additionally, the use of complex-valued activation functions and phase-based nonlinearity enables more flexible representation of patterns and interactions among variables [43].

2.2.3. Convolutional neural network (CNN)

A Convolutional Neural Network (CNN) is a deep learning architecture designed specifically for processing structured grid-like data, such as images [44] and time-series signals [45]. CNNs have revolutionized computer vision by effectively capturing spatial hierarchies through convolutional operations [46]. A typical CNN consists of multiple layers that sequentially transform raw input data into high-level representations [47]:

1. Convolutional Layer: the convolutional layer applies a set of trainable filters (kernels) to extract local features from the input data. The operation is defined as:

$$h_{ij}^{(l)} = \sum_m \sum_n W_{mn}^{(l)} x_{(i+m)(j+n)}^{(l-1)} + b^{(l)} \tag{9}$$

where, $h_{ij}^{(l)}$is the output at position (i,j) in layer l, $W_{mn}^{(l)}$represents the filter weights, $x_{(i+m)(j+n)}^{(l-1)}$is the receptive field from the previous layer, and $b^{(l)}$is

the bias term. This operation preserves spatial relationships and allows CNNs to detect patterns such as edges, textures, and shapes.

2. Activation Function: non-linearity is introduced using activation functions like ReLU (Rectified Linear Unit), which accelerates convergence and prevents vanishing gradients:

$$f(x) = \max(0, x) \tag{10}$$

3. Pooling Layer: pooling layers reduce spatial dimensions while retaining important features. A common operation is max pooling and enhances computational efficiency and reduces overfitting, defined as:

$$h_{ij}^{(l)} = \max\left(x_{(i+m)(j+n)}^{(l-1)}\right) \tag{11}$$

4. Fully Connected Layer (FC): flattened feature maps are passed to fully connected layers, where each neuron is connected to all previous activations. The final output is computed using:

$$y = \sigma(Wh + b) \tag{12}$$

where, the term y represents the output of the fully connected (dense) layer, W and b are learnable parameters, and $\sigma$is the activation function.

In this research, a CNN was incorporated into QRNN for learning higher-order feature relationships of groundwater data before the iterative processing layer. Encapsulating CNNs before iterative layers better accounts for local relationships among input features, reduces the need for manual or engineering selection, and improves overall temporal forecast accuracy. CNN and QRNN's synergy combine the pattern extraction ability of CNN and the temporal memory of QRNN, and provides an efficient hybrid for TDS forecasting from multivariate data.

2.2.4. Gradient-based optimization (GBO)

Gradient-Based Optimization (GBO) is a modern metaheuristic algorithm developed to address complex and high-dimensional optimization tasks, particularly those where traditional gradient-based methods fall short due to non-differentiable or multimodal objective functions [48]. This study is driven to use GBO for two reasons. 1) The hyperparameter tuning of their deep learning is fundamentally a non-convex, multimodal optimization problem. This can become computationally expensive when applying methods like grid search or Bayesian optimization. 2) Standard metaheuristics such as genetic algorithm or particle swarm optimization may either experience premature convergence or take too many iterations to reach competitive solutions. GBO avoids the problems of costly computation, difficult convergence, and unnecessary iterations with its dual-phase search strategy and adaptive trade-off between exploration and exploitation.

In the present study, GBO was employed to fine-tune three critical hyperparameters of the proposed hybrid QRNN-CNN model: learning rate, number of GRU units, and dropout rate. The algorithm iteratively updates a population of candidate solutions through a two-phase search process. In the main search phase, the solution update rule is driven by an estimated gradient-like term and stochastic noise, defined as:

$$X_i^{t+1} = X_i^t + \alpha . G\left(X_i^t\right) + \beta . R\left(X_i^t\right) \tag{13}$$

where, X_i^t denotes the position of the i-th candidate at iteration t, $G\left(X_i^t\right)$is the gradient approximation, $R\left(X_i^t\right)$is a random exploration term, and α, $\beta$are adaptive control coefficients that balance exploitation and exploration. To prevent stagnation in local optima, GBO introduces a Local Escaping Operator (LEO), which perturbs promising solutions to promote diversity in the population. The LEO equation is given by:

$$X_i^{t+1} = X_i^t + \gamma . \left(X_{best} - X_i^t\right) + \delta . rand() \tag{14}$$

where X_{best} is the best solution found so far, $\gamma$and $\delta$are user-defined coefficients, and $rand()$is a uniformly distributed random number in [0,1]. This operator introduces directional bias toward the best solution while maintaining stochastic variability, helping the algorithm escape from local minima.

The integration of GBO into the model development pipeline significantly enhanced model calibration by automating the hyperparameter tuning process, resulting in improved predictive accuracy and reduced overfitting. Furthermore, the method offers a balance between convergence speed and global search capability, making it efficient for applications with limited computational resources [49].

2.3. SHapley additive exPlanations (SHAP)

SHapley Additive exPlanations (SHAP) is a game-theoretic approach used to interpret complex machine learning models by assigning each feature an importance value based on its contribution to the model's predictions. By decomposing a model's output into additive attributions, SHAP provides an intuitive understanding of how features influence predictions, allowing researchers to assess model behavior with transparency [50]. Unlike traditional feature importance methods, SHAP accounts for all possible feature interactions by computing their marginal contributions across different coalitions of features, making it a robust and theoretically sound interpretability framework [51].

One of the primary strengths of SHAP is its ability to provide both local and global interpretability, offering insights at the individual prediction level while also capturing broader trends across the dataset [51]. This dual capability makes SHAP particularly valuable in domains requiring high interpretability, such as healthcare, finance, and environmental science, where understanding model decisions is critical [52,53]. Additionally, SHAP supports various model types, including tree-based algorithms, deep learning networks, and ensemble methods, through optimized approximation techniques that improve computational efficiency.

2.4. Models' development

In this study, an intelligent two-stage hybrid model was developed to predict total dissolved solids (TDS), integrating a convolutional neural network (CNN), a quantum-inspired recurrent neural network (QRNN), and the gradient-based optimization (GBO) algorithm. This combination enhances the model's ability to process time-series data, capture spatial-temporal patterns, and improve prediction accuracy through adaptive hyperparameter tuning.

The model was created using TensorFlow 2.13 and Keras for neural networks, Scikit-learn for data pre-processing and evaluation, and NumPy and Pandas for numerical operations and data management. Before modeling, the dataset underwent several pre-processing techniques to ensure data quality and stability. Prior records that possessed inconsistent values were excluded from analysis, and only complete validated samples were included in the analysis. In the interest of limiting noise, the dataset was cross-referenced with laboratory quality-control reports, and standardized Z-score normalization was applied so that each feature with different scales contributed equally during training to the overall calculation. These techniques guaranteed that the final dataset was consistent and trustworthy, allowing for robust training and convergence of the model to occur. Visualization was performed using Matplotlib and Seaborn. All experiments were conducted on a standard workstation equipped with an Intel Core i7–11,700 CPU (2.50 GHz), 32 GB RAM, and an NVIDIA RTX 3060 GPU (12 GB) running Windows 10. Instead of employing k-fold cross-validation, a train-test split strategy was utilized to evaluate model performance. This decision was made due to the temporal and spatial nature of the dataset,

which required maintaining the chronological and regional integrity of the data to avoid data leakage and unrealistic model performance estimates. Additionally, the dataset size and domain-specific characteristics limited the feasibility of creating multiple folds that would be representative and independent enough for cross-validation. Despite the lack of k-fold validation, rigorous evaluation on an independent test set demonstrated stable and reliable model performance, minimizing concerns regarding overfitting. The total time required for the GBO optimization phase was approximately 15–20 min, which was acceptable given the trade-off between global search accuracy and runtime. Also, average training time for the proposed GBO-QRNN-CNN model was approximately 40–60 min per run, with memory usage below 12 GB, making it feasible for deployment by water management agencies with limited computational resources. The model's implementation with TensorFlow also provides flexibility for future adaptation to cloud-based or CPU-only environments if necessary.

During the data preparation phase, the dataset was sourced from a reliable database and underwent initial processing to separate input and output variables. To optimize the model's training process, the dataset was split into training and testing subsets, and all input features were normalized within a specified range using standardization techniques. This step not only improved the learning speed but also mitigated the impact of outliers, contributing to the model's overall stability.

The proposed model consists of two primary components operating in tandem. The first component is a convolutional neural network, which acts as a spatial feature extractor. Through one-dimensional convolutional layer, CNN captures essential spatial patterns in the input data. Activation functions and dropout layer are applied to prevent overfitting, and the extracted features are then flattened into a one-dimensional vector before being passed to the next stage. The second component is a quantum recurrent neural network, which incorporates memory units based on quantum mechanics principles to model temporal dependencies in the time-series data. QRNN exhibits a strong ability to detect hidden patterns in sequential data, effectively processing the spatial features extracted by CNN while accounting for their temporal relationships. To avoid model complexity and overfitting on the limited dataset, a single 1D convolutional layer was employed in the hybrid architecture. Preliminary experiments with deeper CNNs did not result in significant improvements and introduced a higher risk of overfitting.

While numerous optimization methods, such as Particle Swarm Optimization (PSO), Marine Predators Algorithm (MPA), etc. have been successful in training models and optimizing parameters, the use of CNNs in this research is for a deeply different and complementary purpose. The motivation for using CNN is that groundwater quality variables, such as pH, Mg, Ca, and hardness, often exhibit localized interactions that may not be accounted for by a linear model. The convolutional layer learns these spatially structured interactions automatically, using richer representations to pass to the recurrent QRNN layer for improved predictive capabilities. CNNs are used not as an optimizer but as a feature extractor module in the hybrid QRNN-CNN model and contribute significantly to making the model learn spatial patterns and complex interactions between the input variables. Groundwater quality data, of which the TDS physicochemical parameters are often used in the prediction, often consist of structured relationships, latent interdependencies, and local patterns among features. Such localized patterns can be captured effectively by CNNs through convolutional operations, which are themselves adaptive filters. By the use of 1D convolutions applied over the input feature space, CNN layer enhances the capability of the model to learn informative features automatically, and not necessarily hand-engineered or optimizable with readily available optimization algorithms. In addition, the combination of CNN and QRNN allows for the model to learn temporal relationships (via QRNN), which are both important in environmental and hydrologic modeling, where variables have the tendency to both interact in space and time. Such a synergy offers a broader learning context than optimization alone, which tends to be focused on adjusting the model parameters but does not assist directly toward more informative feature representations learning.

To achieve optimal performance, the model's hyperparameters, including learning rate and the number of recurrent units, were fine-tuned using the gradient-based optimization algorithm. In this process, an initial set of hyperparameter values was generated, and each configuration was evaluated based on prediction error metrics. The optimization algorithm iteratively refined these values to identify the optimal configuration, ensuring improved model performance. Once the model was trained, its predictive capability was assessed on the test dataset using evaluation metrics such as root mean square error and correlation coefficient. In the proposed GBO-QRNN-CNN hybrid model, the GBO is employed not for training the network weights, but specifically for optimizing key hyperparameters that significantly influence the model's learning capacity and generalization performance. The motivation behind using GBO for hyperparameter tuning stems from the fact that manual selection or grid search methods are often inefficient and may fail to find optimal configurations in complex, non-convex landscapes. GBO can explore the search space more effectively by balancing exploitation and exploration. The optimization searched within predefined ranges: learning rates varied between 0.0001 and 0.01, and the number of QRNN units ranged from 10 to 100. The GBO was run for 10 generations with 5 particles per generation, iteratively evaluating hyperparameter sets based on mean squared error on a validation subset. The final optimized parameters for the GBO-QRNN-CNN model included a learning rate of 0.0008, a batch size of 12, 64 CNN filters with a kernel size of 3 and ReLU activation, a dropout rate of 0.25, and 150 QRNN units with Tanh activation. The number of training epochs was adaptive and determined through the optimization process, while the Adam optimizer was used with optimized learning rates. These finalized hyperparameters are summarized in Table 2. This systematic approach ensured robust hyperparameter tuning beyond manual try-and-error.

Table 2
Utilized hyperparameters to increase the accuracy of models.

Hyperparameter	QRNN	QRNN-CNN	GBO-QRNN-CNN	LSTM
Learning Rate	0.001	0.001	0.0008 (optimized by GBO)	0.001
Batch Size	16	16	12 (optimized by GBO)	16
Number of CNN Filters	–	64	64	–
CNN Kernel Size	–	3	3	–
CNN Activation Function	–	ReLU	ReLU	–
Dropout Rate	0.2	0.2	0.25 (optimized by GBO)	–
QRNN Units	100	128	150 (optimized by GBO)	–
QRNN Activation Function	Tanh	Tanh	Tanh	–
Number of Training Epochs	50	50	Adaptive (early stopping)	50
Optimizer Used	Adam	Adam	Adam (optimized by GBO)	Adam
GBO Population Size	–	–	30	–
GBO Max Iterations	–	–	50	–
GBO Coefficients	–	–	$\alpha = 0.5$, $\beta = 0.3$, $\gamma = 0.4$, $\delta = 0.2$	–
Escaping Probability (P_e)	–	–	0.5	–
LEO Scaling Coefficient (α)	–	–	1.5	–
LSTM Units	–	–	–	50 (2 layers)
LSTM Activation Function	–	–	–	ReLU

According to Table 2, the Gradient-Based Optimizer enhanced with a Local Escaping Operator was employed to optimize the hyperparameters of the proposed QRNN-CNN model. Specifically, GBO/LEO was used to determine the most suitable values for the learning rate and the number of GRU units, which are crucial in striking a balance between model convergence and generalization. It is important to emphasize that GBO/LEO was applied solely as a pre-training hyperparameter optimization tool; the weights and biases of the neural network were optimized using the Adam optimizer during the actual training phase. This hybrid strategy allows the model to benefit from the global search capabilities of GBO/LEO while leveraging the fast convergence properties of Adam for weight updates. The LEO improves the exploitation-exploration balance of GBO by preventing premature convergence to local optima. In this work, the additional parameters introduced by LEO were set based on recommendations from the original GBO/LEO formulation. The escaping probability was set to 0.5, and the scaling coefficient was set to 1.5. These values were kept constant to reduce tuning complexity and computational load while maintaining consistency with prior applications of the algorithm. The optimization process used a population size of 4 and 2 iterations, resulting in a manageable search space and feasible runtime.

The dataset was initially divided into two major subsets: training (67 %) and testing (33 %), using stratified random sampling to preserve the overall distribution. The testing set was held out and remained entirely untouched throughout the optimization process to ensure unbiased model evaluation. During the hyperparameter optimization phase using models, the training set was further internally split into training and validation subsets using a random 80/20 split. This internal validation set was used to compute the fitness of each candidate solution, based on the mean squared error. This separation ensured that the optimization was performed on data unseen by the final model and prevented any form of test data leakage. After the optimal hyperparameters were identified, the model was retrained using the entire training set (67 %), and its performance was then evaluated on the unseen test set (33 %), for which metrics were reported. Thus, the test error reported in the study corresponds to a model configuration selected via validation error, not test error, adhering to standard machine learning protocol.

The mean squared error on the validation data was used as the fitness function to evaluate and compare the performance of each particle in the population. Each candidate solution was evaluated by training the model for only 2 epochs during the search phase to accelerate the process. After identifying the optimal configuration, the model was retrained using the full training set for 30 epochs, and performance was assessed on the held-out test set to ensure unbiased evaluation. This approach successfully reduced manual trial-and-error in model tuning and contributed to improved generalization performance of the final model. In summary, the general flowchart of the models developed are given in Fig. 1.

2.5. Evaluation criteria

In the present study, five error evaluation metrics including coefficient of correlation (R), root mean square error (RMSE), normalized RMSE (NRMSE), Nash-Sutcliffe efficiency (NSE), and Willmott's index (WI) were used in assessing the performance of simple LSTM, QRNN, and hybrid QRNN forms, i.e., QRNN-CNN and GBO-QRNN-CNN. Their mathematical formulas can be written as follows:

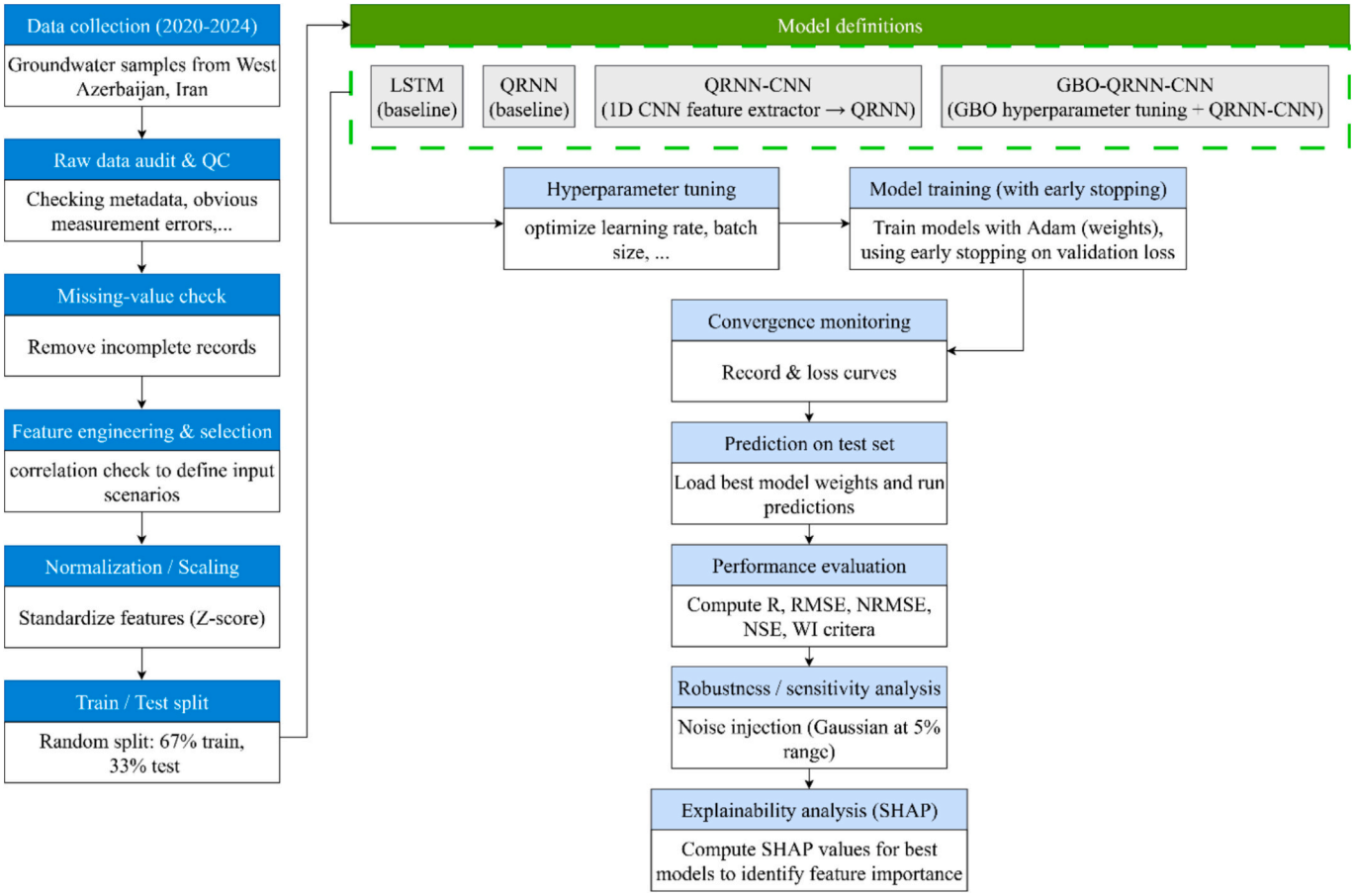

Fig. 1. Models implementation flowchart.

$$R = \frac{\sum_{i=1}^{N}\left(TDS_{o,i} - \overline{TDS_o}\right)\cdot\left(TDS_{p,i} - \overline{TDS_p}\right)}{\sqrt{\sum_{i=1}^{N}\left(TDS_{o,i} - \overline{TDS_o}\right)^2 \cdot \sum_{i=1}^{N}\left(TDS_{p,i} - \overline{TDS_p}\right)^2}} \quad (15)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}\left(TDS_{o,i} - TDS_{p,i}\right)^2}{N}} \quad (16)$$

$$NRMSE = \frac{RMSE}{TDS_{o,\max} - TDS_{o,\min}} \quad (17)$$

$$NSE = 1 - \frac{\sum_{i=1}^{N}\left(TDS_{o,i} - TDS_{p,i}\right)^2}{\sum_{i=1}^{N}\left(TDS_{o,i} - \overline{TDS_o}\right)^2} \quad (18)$$

$$WI = 1 - \left[\frac{\sum_{i=1}^{N}\left(TDS_{o,i} - TDS_{p,i}\right)^2}{\sum_{i=1}^{N}\left(\left|TDS_{o,i} - \overline{TDS_o}\right| + \left|TDS_{p,i} - \overline{TDS_o}\right|\right)^2}\right] \quad (19)$$

where, $TDS_{o,i}$and $TDS_{p,i}$illustrate the ith observed and predicted TDS values, respectively; $\overline{TDS_o}$and $\overline{TDS_p}$indicate the mean values for the observed and predicted TDS values, respectively; $TDS_{o,\max}$and $TDS_{o,\min}$denote the maximum and minimum values for the observational TDS values; and N is the total number of observational data. It is clear that lower values of the RMSE and NRMSE, and higher values of the R, NSE, and WI demonstrate the better accuracy of the relevant model in predicting the TDS values.

3. Results and discussion

The groundwater TDS levels measured in some urban water supply wells located in West Azerbaijan Province, Iran, were predicted in the current research. In this context, six various input configurations were taken into consideration by applying six groundwater variables, including pH, Mg, total alkalinity, HCO_3, Ca, total hardness as listed below: (i) scenario1: pH; (ii) scenario2: pH, Mg; (iii) scenario3: pH, Mg, total alkalinity; (iv) scenario4: pH, Mg, total alkalinity, HCO_3; (v) scenario5: pH, Mg, total alkalinity, HCO_3, Ca; (vi) scenario6: pH, Mg, total alkalinity, HCO_3, Ca, total hardness.

The modeling scenarios in this study were structured based on the Pearson correlation coefficient between each input variable and the target variable, TDS. Specifically, the variables were introduced in ascending order of their individual linear correlation with TDS, starting with pH, which exhibited the lowest correlation, and ending with total hardness, which had the highest correlation. This strategy was designed to examine the incremental benefit of adding more strongly correlated features to the predictive model and to assess how prediction accuracy evolves as input relevance increases. First, an LSTM model was implemented with the statistical metrics listed in Table 3 during the test phase. It is worthy to mention that the LSTM and QRNN-based models numbered with one (LSTM-1, …, GBO-QRNN-CNN-1) and six (LSTM-6, …, GBO-QRNN-CNN-6) are developed using the inputs of first and sixth scenarios, respectively. A performance evaluation of the six developed LSTM models denoted their capability for forecasting TDS levels via the utilized input predictors, with the LSTM-1 and LSTM-6 were found to represent the lowest and highest levels of precision, respectively.

Next, another DL framework, namely QRNN was developed utilizing other groundwater quality data. The results of developed QRNN models in terms of statistical metrics under different input patterns during the testing stage are presented in Table 3. As it is apparent, pH could be used as an independent predictor in QRNN-1 model to predict TDS levels, although its performance is not satisfactory due to the relatively higher values of RMSE and NRMSE, as well as lower values for the R, NSE, and WI metrics. Therefore, more input combinations were considered in QRNN-2 to QRNN-6 models to improve the forecasting performance of TDS values. A performance comparison of the developed QRNN models clearly illustrated that the accuracy of QRNN models could be generally enhanced through increasing the input estimators taking into account the values of diverse error measures used to evaluate the models' accuracy.

Table 3
The values of statistical metrics achieved during the testing stage.

Model	Evaluation criteria				
	R	RMSE (mg/l)	NRMSE	NSE	WI
LSTM-1	0.35	103.38	0.15	0.07	0.23
QRNN-1	0.54	93.41	0.13	0.24	0.52
QRNN-CNN-1	0.57	88.20	0.12	0.32	0.68
GBO-QRNN-CNN-1	0.61	84.77	0.12	0.37	0.72
LSTM-2	0.70	100.38	0.14	0.12	0.25
QRNN-2	0.71	77.12	0.11	0.48	0.76
QRNN-CNN-2	0.77	68.53	0.10	0.59	0.86
GBO-QRNN-CNN-2	0.79	65.80	0.09	0.62	0.88
LSTM-3	0.68	91.42	0.13	0.27	0.50
QRNN-3	0.77	79.68	0.11	0.45	0.72
QRNN-CNN-3	0.77	67.90	0.10	0.60	0.86
GBO-QRNN-CNN-3	0.81	64.41	0.09	0.64	0.89
LSTM-4	0.69	84.90	0.12	0.37	0.63
QRNN-4	0.84	71.06	0.10	0.56	0.79
QRNN-CNN-4	0.86	55.65	0.08	0.73	0.91
GBO-QRNN-CNN-4	0.90	47.39	0.07	0.80	0.94
LSTM-5	0.73	78.13	0.11	0.47	0.74
QRNN-5	0.90	47.79	0.07	0.80	0.94
QRNN-CNN-5	0.92	42.78	0.06	0.84	0.96
GBO-QRNN-CNN-5	0.93	40.47	0.06	0.86	0.96
LSTM-6	0.78	71.53	0.10	0.55	0.79
QRNN-6	0.94	49.19	0.07	0.79	0.92
QRNN-CNN-6	0.96	31.89	0.05	0.91	0.98
GBO-QRNN-CNN-6	0.97	25.40	0.04	0.94	0.99

The theoretical motivation for utilizing QRNN and its hybrid models lies in their efficiency at capturing short- and long-term dependencies in time-series data. Unlike LSTM networks, QRNN employs convolutional operations for local feature extraction and also uses recurrent pooling mechanisms to preserve sequence information for faster training and better generalization. The QRNN structure is particularly suited to groundwater TDS prediction, where hydrochemical parameters are often periodically autocorrelated and usually display some sort of complex dependency that is not simple linear. The comparative results shown in Table 3 provide some empirical evidence supporting this assumption, as QRNN-based models outperformed the LSTM benchmarks in all tests, as they present a better capacity to represent the complex spatio-temporal representations of groundwater quality variables.

As already noted, the chief aim of this study was to propose improved forms of QRNN models. To this end, a hybrid DL scheme was firstly established via merging the QRNN and CNN to create the QRNN-CNN coupled model. The same input combinations applied for the LSTM and QRNN were used during the establishment of QRNN-CNN models. Taking into account the values of error evaluation criteria for the proposed merged method in Table 3, it can be concluded that the general results of proposed QRNN-CNN hybrid technique were the same as the outcomes of standalone QRNN model. They can be listed as follows: poor performance of the first scenario (QRNN-CNN-1) using only pH predictor, improved performance of the hybrid model with increasing the number of inputs, as well as the best accuracy of the hybrid model in the scenario with full-input (QRNN-CNN-6). The hybrid QRNN-CNN model also demonstrated improved performance compared with the single QRNN.

Besides the development of simple QRNN and hybrid QRNN-CNN models, a novel two-stage merged model named GBO-QRNN-CNN was also proposed in this study by coupling the QRNN with GBO and CNN.

The values of error metrics achieved for the mentioned coupled model are shown in Table 3. The general outcomes obtained for the QRNN and QRNN-CNN models were also concluded for the novel proposed GBO-QRNN-CNN merged framework. In addition, assessing the performance of developed two-stage hybrid method with simple QRNN and coupled QRNN-CNN methods clearly demonstrates that the GBO-QRNN-CNN was the best-performing model for forecasting the groundwater TDS levels because of the enhanced error measures. This improvement can be attributed to CNN's ability to extract spatial features, QRNN's efficiency in modeling sequential dependencies, and GBO's role in fine-tuning model parameters. The values of error metrics for the best groundwater TDS predictions were observed at GBO-QRNN-CNN-6 model as $R = 0.97$, RMSE = 25.40 mg/l, NRMSE = 0.04, NSE = 0.94, WI = 0.99. The values of R, RMSE, NRMSE, NSE, and WI metrics in the GBO-QRNN-CNN-6 relative to baseline QRNN-6 were improved by 3.19 %, 48.36 %, 42.86 %, 18.99 %, and 7.61 %, respectively.

While the level of performance enhancement from the application of GBO may seem modest in absolute numerical terms, it is a consistent enhancement across all evaluation metrics, contributing to the overall robustness of the proposed framework. The main advantage of GBO is its optimization capabilities through the gradient-based updating strategy of GBO, and the local escaping operator, in adaptively balancing the global exploration and the local exploitation. This mechanism has facilitated the model to have smoother convergence and is less likely to get stuck in local minima before making it to the global minima. This contributes to more stability and reproducibility over multiple runs. Thus, even with relatively small or modest numerical gains, GBO contributes to the stability of convergence, optimization efficiencies, and generalizability of the hybrid GBO-QRNN-CNN model over the non-optimized models.

To assess the statistical significance of the improvements observed in the hybrid models (QRNN-CNN and GBO-QRNN-CNN) compared to the baseline QRNN and traditional LSTM, paired *t*-tests were performed on the two performance measures, NSE and WI, on the sixth scenario. Pairwise comparisons were performed between the four models (LSTM, QRNN, QRNN-CNN, and GBO-QRNN-CNN). The results, which are presented in Fig. 2, show that both QRNN-CNN and GBO-QRNN-CNN models outperformed the QRNN and LSTM models significantly in most cases (p-value <0.05). These findings statistically confirm the added value of integrating CNN layers and optimization strategies such as GBO in the predictive modeling framework.

Fig. 3 shows sensitivity tests under different data conditions. Two challenging data situations of noisy data and reduced training set size were used in order to test sensitivity to ascertain the robustness of the employed models. For noise injection, Gaussian noise was added to the input features (mean = 0, σ = 5 % of data range) and the performance of the model was evaluated in all cases using NSE and WI indices. For sparse data, to simulate data scarcity, 50 % and 70 % of the original training data were used to train the models. Robustness and sensitivity analysis outcomes show that the proposed hybrid models, especially GBO-QRNN-CNN, can maintain acceptable predictive performance in unfavorable conditions of noisy inputs and smaller training data (70 % and 50 %). As expected, the performance of all models degrades, but GBO-QRNN-CNN consistently performs the best and suffers the minimum accuracy loss. In comparison, LSTM suffers higher degradation since it has poor robustness. The outcomes prove the stability and preparedness of the hybrid model for real-life hydro chemical prediction under compromised data quality and availability. In addition to the stability under noisy and reduced data, the convergence behavior of the models was carefully examined. The training and validation loss curves showed smooth and uniform declines in each period without any divergence or unstable oscillation, confirming the stable optimization dynamics. The use of a combined GBO approach to tune the hyperparameters and the use of Adam as the weight update regime improved the convergence speed and prevented premature trapping in local optima. Early stopping was used to prevent overfitting, and the final model configuration was also made based on the convergence of the decision. By repeating the modeling process on six independent random training-test splits, the results showed low variance among the metrics, indicating the reliability and repeatability displayed in the convergence process. The overall results showed that the combined approaches used not only performed better in terms of accuracy, but were also reliable in terms of convergence and any stability in the optimization possible during the training period.

In addition to evaluating the accuracy of simple and hybrid models from a statistical perspective, some graphical representations were also provided to further investigate the improvements in performance of the proposed hybrid models compared to the simple QRNN model. In this regard, the scatter, violin, and Taylor plots were prepared for all the simple and hybrid forms of QRNN under the different input combinations. Fig. 4 illustrates the scatter plots for the observed vs. predicted TDS values by the standalone and merged models. The blue dots indicate the scatter of each data point, the dotted blue line indicates the intercept line, and the red line indicates the perfect X = Y line. As it is clear, the scatter of datapoints in both the hybrid QRNN models is lesser than the simple QRNN, indicating the better accuracy of the hybrid models. However, the two-stage GBO-QRNN-CNN models exhibit the lowest scatters. This conclusion could also be confirmed considering the higher

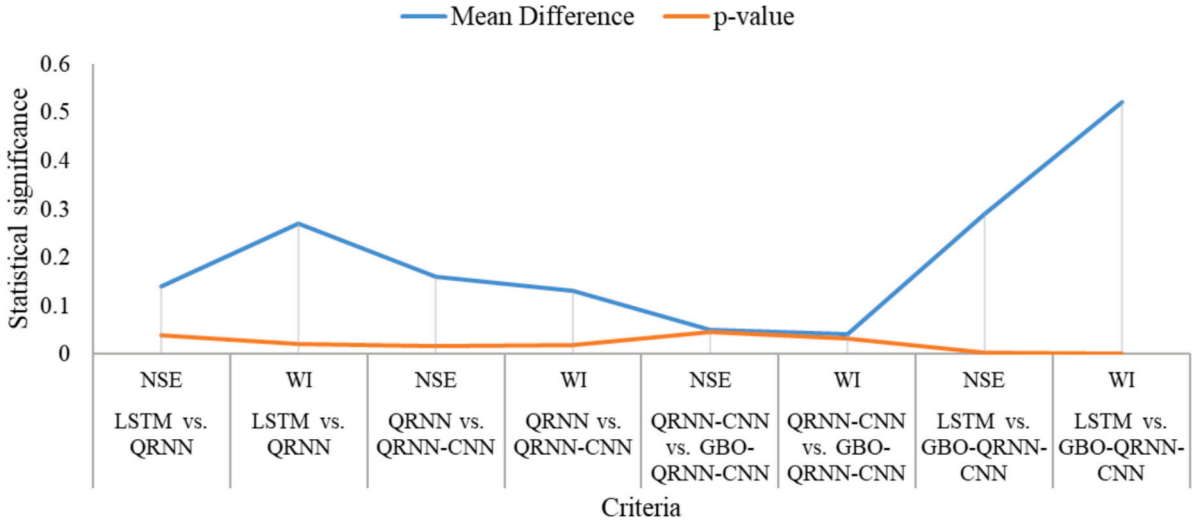

Fig. 2. Statistical significance of model comparison.

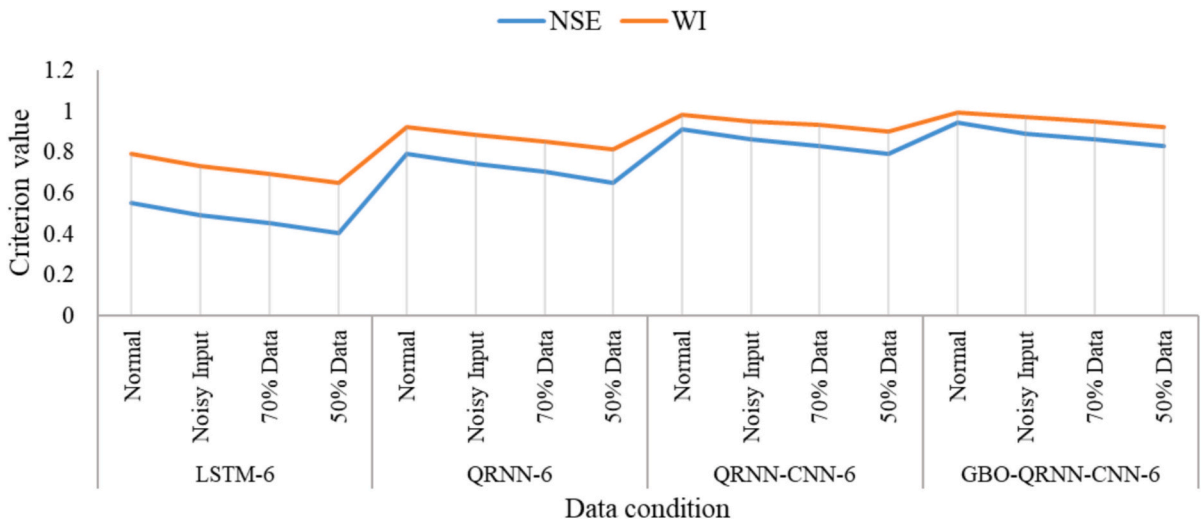

Fig. 3. Sensitivity tests under challenging data conditions.

values of R^2 in the hybrid models compared with the relevant single QRNN ones. On the other hand, the LSTM models represented the highest scatters, denoting its poor performance. Additionally, the scatter of points in all the hybrid and simple models decreases with increasing the number of inputs, indicating a significant impact of more inputs on the performance of the developed models.

Besides the scatter plots, violin plots were also prepared that are shown in Fig. 5. A performance evaluation of the violin plots for the different input patterns clearly indicates that the simple LSTM and QRNN, and hybrid QRNN-based models with fewer inputs cannot predict the data distribution well, whereas models with more inputs could better represent violin plots closer to the corresponding violin plots of the observational values, indicating their reliable ability to predict TDS levels more accurately. In this context, the violin plot of the hybrid GBO-QRNN-CNN-6 with full-input predictors provides the closest similarity to the violin plot of observed TDS data.

Fig. 6 also demonstrates the Taylor diagrams for the whole simple and coupled forms of QRNN developed under the diverse input configurations. The outcomes of two statistical metrics including the correlation coefficient and standard deviation are merged in a Taylor diagram to investigate the model's performances graphically. In a Taylor diagram, each point demonstrates the results of a model, and the proximity or distance of the corresponding point of each model compared to the observational data point indicates the performance of any model in predicting the observational data. As shown in Fig. 6, the standalone LSTM and QRNN models, as well as all the models developed using fewer inputs demonstrate more distances compared to the reference point, which indicated unfavorable performance of respective models. Conversely, the points related to the models, especially the hybrid models, established with more input predictors are closer to the reference point.

Fig. 7 presents the SHAP plots to illustrate the impact of various features on predicting TDS in different models. It is worthwhile to mention that SHAP plots are prepared for the simple LSTM-6, QRNN-6, and hybrid QRNN-CNN-6 and GBO-QRNN-CNN-6 models as they showed generally the best performance among the whole models implemented. Positive SHAP values indicate an increase in TDS, while negative values suggest a decrease. The color of the points represents the feature values (blue: low value, red: high value). The findings from the SHAP plots for the standalone LSTM and QRNN models indicate that total hardness and Ca have the most significant influence on TDS, with higher values of these features leading to an increase in TDS. Remaining four variables also affect TDS but to a lesser extent, particularly in the QRNN. It is observed that low values of total hardness and Ca generally result in lower TDS, whereas high values of these features have a positive effect. Additionally, pH and total alkalinity in QRNN exhibit a complex influence, as both high and low values can have varying effects on TDS.

For the hybrid QRNN-CNN model, total hardness and Ca had a greater impact on TDS compared to the standalone model. Notably, pH and total alkalinity exhibited a more significant effect on TDS in comparison to the QRNN model. However, Mg showed a high impact on TDS at SHAP = 20, with a noticeable clustering of points at this value. Moreover, in this model, HCO_3 demonstrated ambiguous behavior relative to other parameters. In the hybrid GBO-QRNN-CNN model, the influence of Mg on TDS was increased, while the other parameters showed no substantial changes compared to the QRNN-CNN model. These results highlight the importance of monitoring water hardness and Ca in groundwater quality management, as controlling these variables can contribute to improving groundwater resource quality. On the other side, both the mentioned variables/inputs in whole the four standalone and hybrid QRNN models demonstrated the positive impacts, i.e., an increased value for the total hardness and Ca lead to the increased TDS value. This means that decreasing the total hardness and Ca of groundwater will lead to reduce the TDS levels. Hence, decision-makers should implement policies to reduce the total hardness and Ca of groundwater. There are several methods to reduce total hardness and calcium in groundwater. Some of these methods include: using ion exchange filters, utilizing sodium carbonate, applying magnetic and electronic devices, and increasing groundwater temperature. The choice of an appropriate technique depends on the hardness and calcium levels, budget, and specific needs of the user.

A thorough comparison was done between the LSTM model and QRNN-based models, i.e., QRNN, QRNN-CNN, and GBO-QRNN-CNN, on the basis of architectural complexity, optimization processes, and training parameters (Table 2). The LSTM model applied a two-layer structure consisting of 50 LSTM units in each layer and ReLU activation, Adam optimizer (learning rate = 0.001), batch size = 16, and 50 epochs. While the LSTM architecture is simple and relatively easy to optimize, it does not have some features aimed at enhancing learning capacity, such as dropout layer, convolutional processing, or metaheuristic-based parameter optimization.

In contrast, the QRNN-based models were structurally more advanced and adaptable. The baseline QRNN model used 100 Tanh-activated recurrent units with dropout (0.2), while QRNN-CNN used convolutional feature extraction (64 filters, kernel size 3, ReLU activation) along with 128 QRNN units. The most advanced model, GBO-QRNN-CNN, used architectural optimization efficiency with the

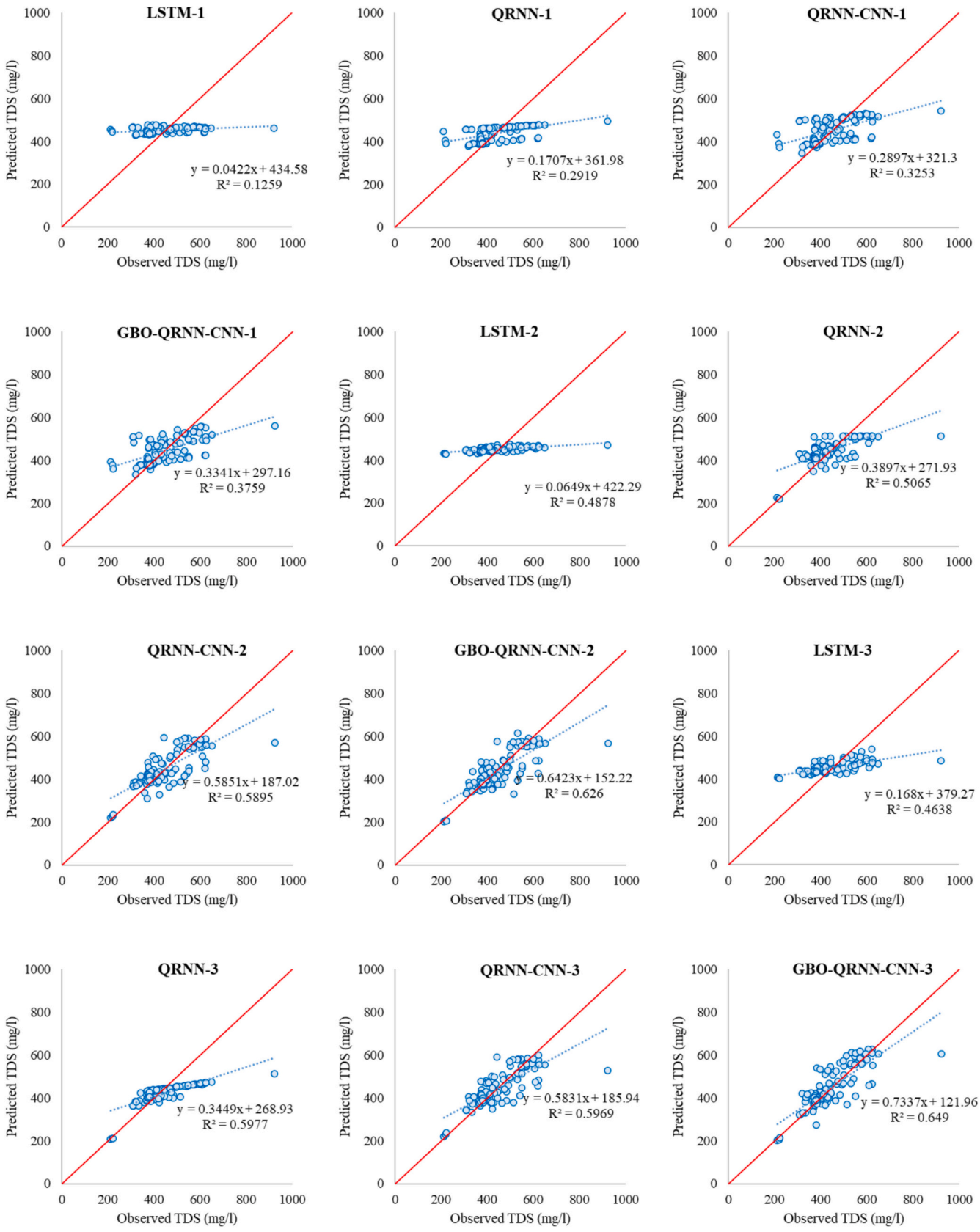

Fig. 4. Scatter plots of the standalone and hybrid QRNN models.

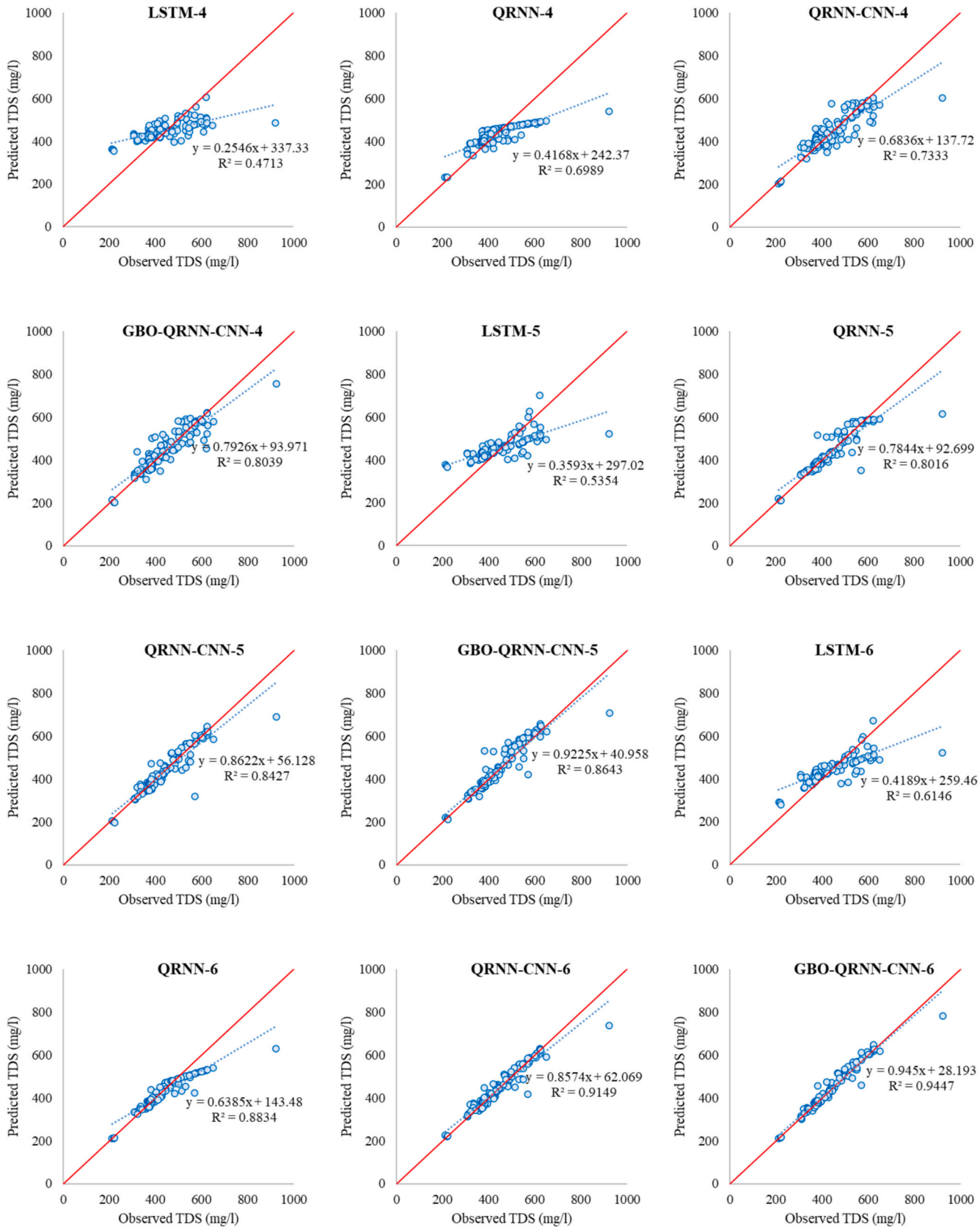

Fig. 4. (*continued*).

Gradient-Based Optimizer (GBO) adjusting learning rate (0.0008), batch size (12), dropout rate (0.25), and number of QRNN units (150). This adaptive approach had a population size of 4 and did 5 iterations with coefficients ($\alpha = 0.5$, $\beta = 0.3$, $\gamma = 0.4$, $\delta = 0.2$) to produce a more customized and perhaps more generalizable model. Although LSTM worked decently well, the QRNN-based models were more adaptable

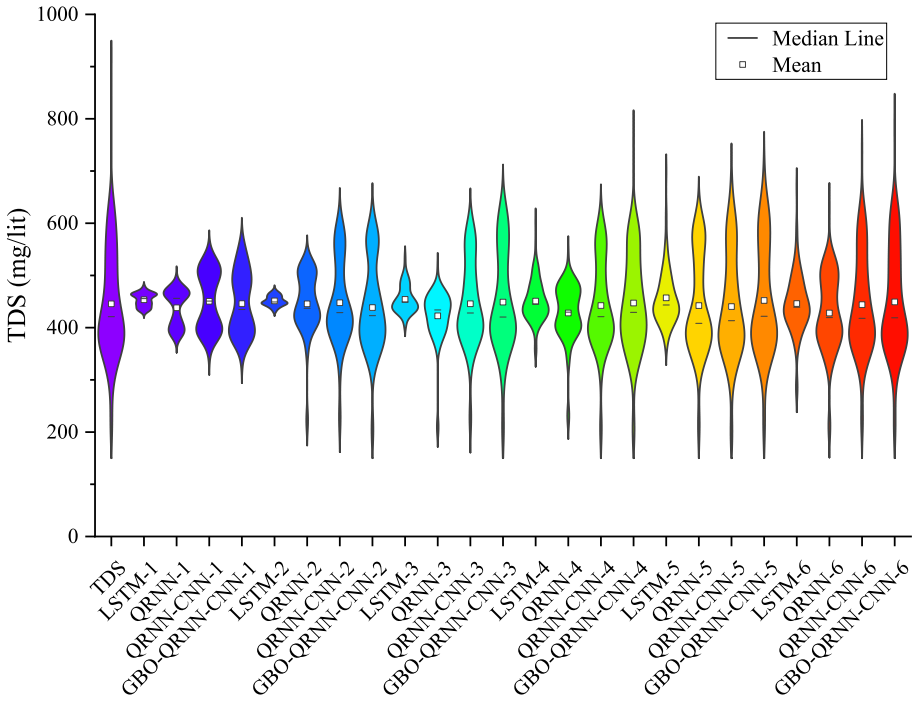

Fig. 5. Violin plots for the standalone and hybrid QRNN models.

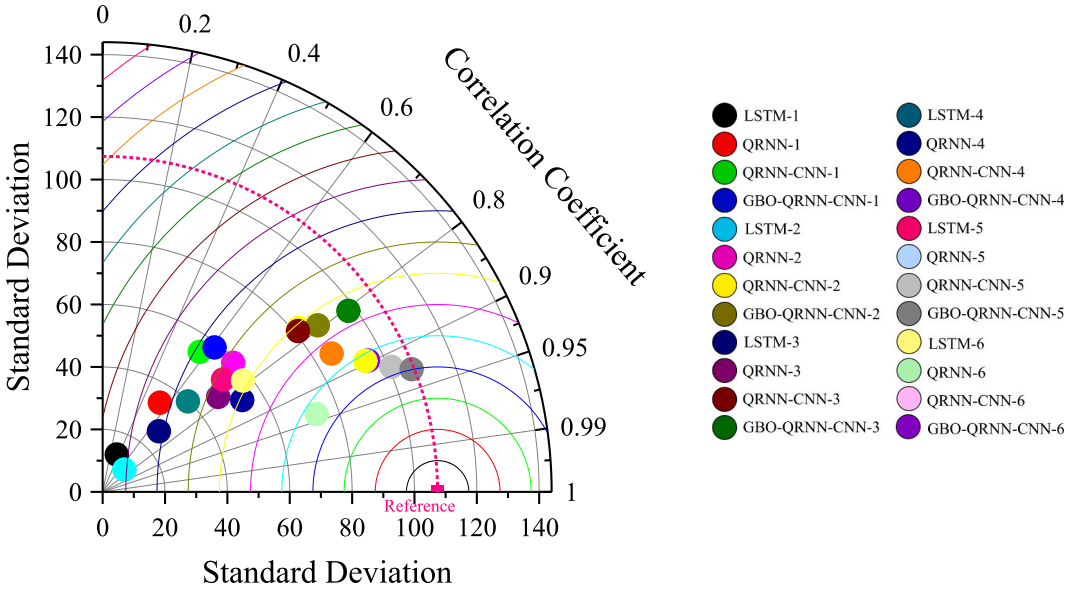

Fig. 6. Taylor diagram for the standalone and hybrid QRNN models.

and scalable. The use of CNNs facilitated improved spatial pattern learning, and with GBO, model convergence and tuning were significantly improved. Interestingly, the test RMSEs of the QRNN models were competitive and, for the majority of cases, lower than LSTM with increased architectural complexity. More significantly, QRNN-based models, especially the GBO-QRNN-CNN, converged better with early stopping and required less human tuning due to automatic optimization. Briefly, even though the LSTM model is straightforward, the QRNN-based models, particularly those that include CNN and GBO, have better learning, more robust performance across varying conditions, and more efficient optimization. All these render QRNN a more practical and scalable solution compared to LSTM, particularly for applications that require stronger accuracy, adaptability, and automatic model optimization.

As concluded, the merged QRNN-CNN model illustrated better TDS forecasts than the standalone QRNN for all the considered input patterns. The potential of CNN in improving the performance of various types of DL models has been confirmed in literature by scholars in forecasting different hydro-climatological and environmental parameters. For example, Danesh et al. [19] developed a hybrid CNN-LSTM for river streamflow forecasting, and compared its performance with single LSTM. They found out that the proposed CNN-LSTM hybrid framework showed better outcomes compared with simple LSTM. Hybrid DL-based models including CNN-LSTM and CNN-BiLSTM (Bi directional LSTM) were proposed by Hu et al. [54] for urban water demand forecasting, and confirmed their dependable performance. Geetha et al. [55]

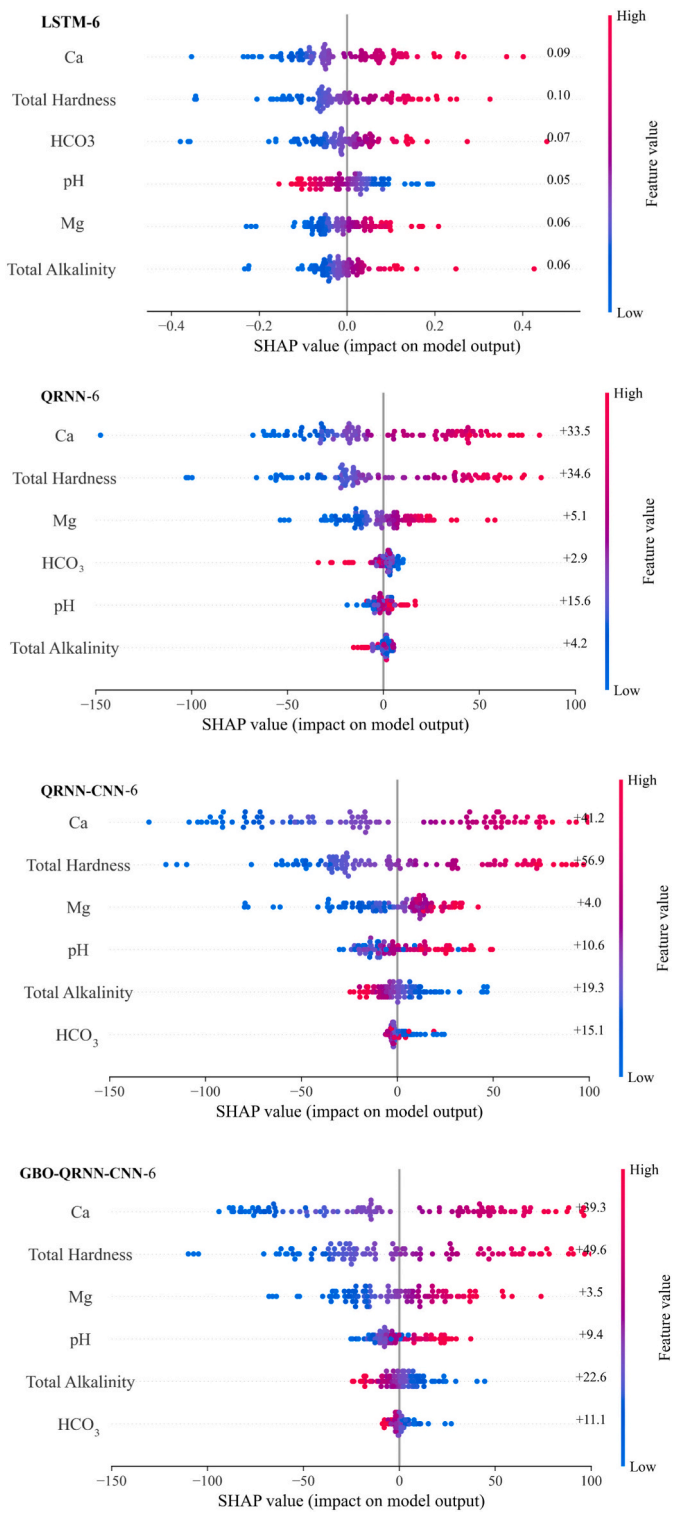

Fig. 7. SHAP plots for the best-performing standalone and hybrid models.

established the hybrid CNN-BiLSTM model for predicting river water quality, and concluded that the hybrid model improved the performance of simple model. Haq et al. [56] proposed two hybrid DL-based models by coupling the CNN with LSTM and gated recurrent unit (GRU) for water quality prediction. It was revealed that the hybrid CNN-LSTM and CNN-GRU models were found to represent better results than their standalone versions. Similarly, Davoudi et al. [57], as well as Uluocak and Bilgili [58] developed CNN-LSTM and CNN-GRU coupled models for forecasting nitrate in surface waters and air temperature, respectively, and stated that the proposed hybrid models outperformed the standalone LSTM and GRU. Ullah et al. [59] developed a CNN-BiGRU hybrid model for short-term load prediction, and reported its dependable potential.

In addition, the key finding of the present study was the superior performance of a two-stage hybrid model of GBO-QRNN-CNN creating by simultaneous hybridizing the GBO and CNN on QRNN. Similar two-stage hybrid DL models have been proposed in previous researches. Wang et al. [60] proposed a two-stage DL model named BWO-CNN-GRU by combining Beluga whale optimizer (BWO) and CNN with the GRU for predicting dissolved oxygen concentration. The results illustrated that the recommended model performed better than the GRU and CNN-GRU. Yang et al. [61] merged the CNN, GRU, and a grey whale optimization (GWO) to develop the hybrid GWO-CNN-GRU model for prediction of soil moisture content, and concluded that the proposed model performed the best. A two-stage hybrid DL model was proposed by Mei et al. [62] in forecasting the turbidity and permanganate index of wastewater through coupling the CNN, BiGRU, and an optimizer named improved sparrow search algorithm (ISSA), and confirmed its reliable potential. Li et al. [63] hybridized the CNN, BiGRU, and Bayesian optimizer (BO) to establish the BO-CNN-BiGRU hybrid technique in predicting sea level height, and compared its performance with the BiGRU and CNN-BiGRU. They reported that the proposed two-stage BO-CNN-BiGRU hybrid method showed the best performance. An intelligent irrigation system was established by Mishra et al. [64] through combining the CNN, BiGRU, and genetic algorithm (GA). The results denoted the superior capability of proposed GA-CNN-BiGRU than the BiGRU and CNN-BiGRU models. Furthermore, some various kinds of two-stage hybrid deep learning models have been also presented in diverse engineering domains like PSO (particle swarm optimization)-CNN-Bi-LSTM [65], PSO-CNN-LSTM, GWO-CNN-LSTM, and COA (Coati optimization algorithm)-CNN-LSTM [66], WOA (whale optimization algorithm)-CNN-attention LSTM, GA (genetic algorithm)-CNN-attention LSTM, and PSO-CNN-attention LSTM [67], DBO (dung beetle optimizer)-multi-head attention (MH)-CNN-GRU, DBO-MH-CNN-LSTM, IDBO (improved DBO)-MH-CNN-GRU, and IDBO-MH-CNN-LSTM [68]. As a result, improved hybrid forms of DL models developed by their coupling with the CNN and optimization algorithms could be used as the promising alternatives to standalone DL methods when predicting the various surface and groundwater quality variables, as well as hydro-climatological parameters.

As outlined, the two-stage GBO-QRNN-CNN-6 hybrid model was found to surpass the other simple and hybrid models developed in the current study (R = 0.97, RMSE = 25.40 mg/l, NRMSE = 0.04, NSE = 0.94, WI = 0.99). In fact, this model is fed with six independent input predictors, including pH, Mg, total alkalinity, HCO_3, Ca, and total hardness. The efficiency of the aforementioned superior model for predicting groundwater TDS levels could be compared with the accuracy of best-performing models proposed in literature. In this regard, it was focused on the values of commonly-used error metrics like RMSE, R, and NSE during the test/validation phase. Yassin et al. [14] reported the values of R and NSE equal to 0.999 and 0.998 for a standalone GPR model with eight inputs comprising total hardness, Na^+, Mg^{2+}, Ca^{2+}, F^-, Cl^-, Fe, and EC. Banadkooki et al. [27] forecasted groundwater TDS levels through developing some types of intelligent models by applying Mg, Ca, and HCO_3 inputs. The authors achieved the RMSE = 3.112 mg/l and NSE = 0.93 for the best method, including the ANFIS-moth flame optimization (MFO) hybrid model. Elzain et al. [12] stated that the groundwater TDS levels could be efficiently forecasted via the best model of bagging regression (BA) developed in level 2, i.e., by extracting the information from Extra Trees Regression (ETR) and CatBoost Regression (CBR) methods established via three inputs of Cl, K, and Sr. The values of R = 0.997 and NSE = 0.996 were reported by the authors for the mentioned superior model. El Bilali et al. [28] concluded that AdaBoost applying three inputs of EC, temperature, and pH was a superior method for an accurate predicting the groundwater TDS levels

with error measures of R = 0.99 and RMSE = 182.0 mg/l. Ewusi et al. [15] found out that the GPR (R = 0.9899, NSE = 0.959, and RMSE = 8.983 mg/l) fed with nine inputs, including As, Cd, Hg, Cu, CN, pH, Turbidity, EC, and total suspend solids (TSS) illustrated the best TDS levels forecasts compared with other methods developed. Jamshidzadeh et al. [11] predicted the groundwater TDS levels of coastal aquifers using HCO_3, Na, Ca, and Mg inputs. They achieved the best NSE equal to 0.95 for CNN-LSTM-GPR hybrid model. As a general conclusion, the performance of any of the models developed to forecast the groundwater quality variables like TDS can depend on a number of factors, namely the type of intelligent models implemented, the independent variables and their numbers used as the input predictors when development of the models, study region, and therefore, the nature of used groundwater quality data and so on.

A precise groundwater TDS forecast is required to better monitor the groundwater quality, as well as for decision makers and planners. So, all the models proposed in this study, especially the hybrid QRNNs, could be utilized as the robust intelligent systems for precisely predicting groundwater quality like TDS levels. As the outcomes revealed, the two-stage GBO-QRNN-CNN hybrid model was found to represent superior TDS predictions. Therefore, it is necessary to pay more attention to simultaneous coupling of the CNN and the optimization algorithms such as GBO used in this study on DL models to improve the performance of baseline DL frameworks. Furthermore, the present study proposed a quantum-inspired RNN (QRNN) and its hybrid forms, including QRNN-CNN and GBO-QRNN-CNN, while previous studies have less addressed this issue.

4. Conclusions, limitations, and future directions

In this study, the groundwater TDS levels at urban water supply wells located in West Azerbaijan Province, Iran, were predicted through developing four intelligent systems. For this, two DL-based techniques named LSTM and QRNN were initially established. Two enhanced forms of the QRNN were then proposed to improve the TDS forecasts. So, QRNN-CNN and GBO-QRNN-CNN hybrid models were established by merging the standalone QRNN with CNN and GBO. Six various input combinations were defined using six groundwater quality variables. A survey of preceding studies in literature clearly illustrated that there was a knowledge gap in implementing quantum-inspired DL-based models to predict the groundwater quality like TDS, while this study proposed a baseline QRNN and its two novel hybrid variants named QRNN-CNN and GBO-QRNN-CNN. The outcomes denoted that two hybrid versions of QRNN illustrated better accuracies compared with simple QRNN; however, the GBO-QRNN-CNN was the best-performing method. A performance evaluation of the simple and hybrid models also demonstrated that their performances were generally improved by increasing the number of input estimators, the full-input scenario showed the best accuracy. Analyzing the relative importance of input predictors on model's outputs utilizing the SHAP demonstrated that total hardness and Ca represented the most importance. Overall, this study highlighted the effectiveness of integrating deep neural networks with advanced optimization techniques for forecasting complex environmental phenomena like groundwater TDS levels.

Although the proposed models, specifically the QRNN-based hybrid methods, demonstrated dependable accuracies on the data from West Azerbaijan, their generalization to other climatic regions (such as arid, semi-arid or humid zones) requires further investigation using regionally diverse datasets. While the models demonstrated good predictive potential for the years 2020–2024, the relatively short record may not include long-term trends or decadal-scale changes in groundwater quality in response to climate change, land use change, or other anthropogenic factors. A longer record would stand a better chance of capturing such processes and improving the temporal stability of the model. Furthermore, the analysis here only looked at a subset of readily available water-quality measurements (pH, Mg, Ca, total hardness, total alkalinity, and HCO_3). Leaving out possibly important variables such as temperature, precipitation, electrical conductivity (EC), and land use measurements may limit the overall predictive potential and generalizability of the models. Addition of such variables in future, with longer series, could lead to even greater model accuracy, allows for better inference on the determinants of TDS variability, and increase the transferability of the model across regions and climatic zones. The present research employed deterministic deep learning methods, but various techniques were utilized to indirectly assess the uncertainty and robustness of the models. These techniques included repeating train–test splits, dropout, and early stopping as means of controlling overfitting, and sensitivity analyses with noisy and reduced training data. Consistent results across these studies indicate stability in predicting behavior for model convergence as reliable. However, it will be beneficial for future studies to more overtly incorporate uncertainty quantification approaches such as Monte Carlo dropout, Bayesian deep learning, or modeling with ensembles to directly capture the range of prediction confidence and further quantify the uncertainty associated with groundwater quality forecasting.

This study applied only two DL models named LSTM and QRNN, whereas diverse types of DL frameworks including GRU, Bi-LSTM, Bi-GRU could be used in future for forecasting the values of groundwater TDS. In following, the aforementioned DL methods could be hybridized with the CNN, and the obtained outcomes could be compared with the results of current study. Additionally, this research focused only on an optimizer, i.e., GBO, while future research could utilize different optimization algorithms, including marine predators' algorithm, spiral dynamic search algorithm, moth flame optimization algorithms, and many more algorithms when developing hybrid versions of DL schemes. Besides the hybrid models proposed in this study, various signal decomposition techniques could also be employed to create the coupled models to predict groundwater TDS. It is also recommended that the proposed methodologies could be used in future for predicting diverse groundwater and surface water quality variables. The outcomes of SHAP clearly illustrated that it is a beneficial tool to discern the possible interactions among a set of input and output data. It is therefore suggested that it could be used further in future when forecasting the different environmental variables like groundwater and surface water quality. As literature review revealed, the quantum-inspired DL models have been less noticed for predicting groundwater quality variables. Hence, it is highly recommended that the quantum-inspired versions of DL frameworks should be considered further in future works.

CRediT authorship contribution statement

Milad Sharafi: Writing – original draft, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Amin Gharehbaghi:** Writing – review & editing, Writing – original draft, Investigation, Formal analysis. **Saeid Mehdizadeh:** Writing – review & editing, Writing – original draft, Validation, Supervision, Investigation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Dataset and the code can be made available upon reasonable request.

References

[1] H. Raheja, A. Goel, M. Pal, Prediction of groundwater quality indices using machine learning algorithms, Water Practice & Technology 17 (2022) 336–351.

[2] R. Haggerty, J. Sun, H. Yu, Y. Li, Application of machine learning in groundwater quality modeling-a comprehensive review, Water Res. 233 (2023) 119745.
[3] A.M. Jibrin, M. Al-Suwaiyan, A. Aldrees, S. Dan'azumi, J. Usman, S.I. Abba, M. A. Yassin, M. Scholz, S.S. Sammen, Machine learning predictive insight of water pollution and groundwater quality in the Eastern Province of Saudi Arabia, Sci. Rep. 14 (2024) 20031.
[4] J.A. Torres-Martinez, J. Mahlknecht, M. Kumar, F.J. Loge, D. Kaown, Advancing groundwater quality predictions: machine learning challenges and solutions, Sci. Total Environ. 949 (2024) 174973.
[5] Y. Razandi, H.R. Pourghasemi, N.S. Neisani, O. Rahmati, Application of analytical hierarchy process, frequency ratio, and certainty factor models for groundwater potential mapping using GIS, Earth Sci. Inf. 8 (2015) 867–883.
[6] G. Busico, N. Kazakis, E. Cuoco, N. Colombani, D. Tedesco, K. Voudouris, M. Mastrocicco, A novel hybrid method of specific vulnerability to anthropogenic pollution using multivariate statistical and regression analyses, Water Res. 171 (2020) 115386.
[7] S. Singha, S. Pasupuleti, S.S. Singha, R. Singh, S. Kumar, Prediction of groundwater quality using efficient machine learning technique, Chemosphere 276 (2021) 130265.
[8] L.A. DeSimone, J.P. Pope, K.M. Ransom, Machine-learning models to map pH and redox conditions in groundwater in a layered aquifer system, Northern Atlantic Coastal Plain, eastern USA, Journal of Hydrology: Regional Studies 30 (2020) 100697.
[9] A. Mosavi, F.S. Hosseini, B. Choubin, M. Abdolshahnejad, H. Gharechaee, A. Lahijanzadeh, A.A. Dineva, Susceptibility prediction of groundwater hardness using ensemble machine learning models, Water 12 (2020) 2770.
[10] Y. Tian, Q. Liu, Y. Ji, Q. Dang, Y. Sun, X. He, Y. Liu, J. Su, Prediction of sulfate concentrations in groundwater in areas with complex hydrogeological conditions based on machine learning, Sci. Total Environ. 923 (2024) 171312.
[11] Z. Jamshidzadeh, S.D. Latif, M. Ehteram, Z. Sheikh Khozani, A.N. Ahmed, M. Sherif, A. El-Shafie, An advanced hybrid deep learning model for predicting total dissolved solids and electrical conductivity (EC) in coastal aquifers, Environ. Sci. Eur. 36 (2024) 20.
[12] H.E. Elzain, O. Abdalla, H.A. Ahmed, A. Kacimov, A. Al-Maktoumi, K. Al-Higgi, M. Abdallah, M.A. Yassin, V. Senapathi, An innovative approach for predicting groundwater TDS using optimized ensemble machine learning algorithms at two levels of modeling strategy, J. Environ. Manag. 351 (2024) 119896.
[13] S. Abba, M. Benaafi, A. Usman, I.H. Aljundi, Sandstone groundwater salinization modelling using physicochemical variables in southern Saudi Arabia: application of novel data intelligent algorithms, Ain Shams Eng. J. 14 (2023) 101894.
[14] M.A. Yassin, A. Usman, S. Abba, D.U. Ozsahin, I.H. Aljundi, Intelligent learning algorithms integrated with feature engineering for sustainable groundwater salinization modelling: Eastern Province of Saudi Arabia, Results in Engineering 20 (2023) 101434.
[15] A. Ewusi, I. Ahenkorah, D. Aikins, Modelling of total dissolved solids in water supply systems using regression and supervised machine learning approaches, Appl Water Sci 11 (2021) 1–16.
[16] D.T. Bui, K. Khosravi, M. Karimi, G. Busico, Z.S. Khozani, H. Nguyen, M. Mastrocicco, D. Tedesco, E. Cuoco, N. Kazakis, Enhancing nitrate and strontium concentration prediction in groundwater by using new data mining algorithm, Sci. Total Environ. 715 (2020) 136836.
[17] F. Ding, W. Zhang, S. Cao, S. Hao, L. Chen, X. Xie, W. Li, M. Jiang, Optimization of water quality index models using machine learning approaches, Water Res. 243 (2023) 120337.
[18] M.G. Uddin, A. Rahman, S. Nash, M.T.M. Diganta, A.M. Sajib, M. Moniruzzaman, A.I. Olbert, Marine waters assessment using improved water quality model incorporating machine learning approaches, J. Environ. Manag. 344 (2023) 118368.
[19] M. Danesh, A. Gharehbaghi, S. Mehdizadeh, A. Danesh, A comparative assessment of machine learning and deep learning models for the daily river streamflow forecasting, Water Resour. Manag. 39 (2025) 1911–1930.
[20] W.-c. Wang, F.-r. Ye, Y.-y. Wang, M. Gu, A singular spectrum analysis-enhanced BiTCN-selfattention model for runoff prediction, Earth Sci. Inf. 18 (2025) 31.
[21] W.-c. Wang, W.-c. Tian, X.-x. Hu, Y.-h. Hong, F.-x. Chai, D.-m. Xu, Dttr: Encoding and decoding monthly runoff prediction model based on deep temporal attention convolution and multimodal fusion, J. Hydrol. 643 (2024) 131996.
[22] J. Ramakrishnan, R. John, D. Mavaluru, R.S. Ravali, K. Srinivasan, Transforming groundwater sustainability, management and development through deep learning, Groundw. Sustain. Dev. 27 (2024) 101366.
[23] M. Paul, A. Kumar, H. Kumar, V.A. Mary, S. Jancy, J. Grace, Groundwater quality prediction using deep learning, in: in: 2024 International Conference on Expert Clouds and Applications (ICOECA), IEEE, 2024, pp. 917–923.
[24] K. Selvakumarasamy, R. Mahaveerakannan, D.R. Kumar, R. Balamanigandan, T. R. Kumar, Predicting groundwater quality with advanced deep learning models: Improving environmental management by addressing pollution issues and obtaining more accurate forecasts, in: in: 2024 8th International Conference on Electronics, Communication and Aerospace Technology (ICECA), IEEE, 2024, pp. 1301–1305.
[25] M. Kasiselvanathan, A. Suresh, M. Sinduja, K. Prajna, Prediction of ground water quality in western regions of Tamilnadu using LSTM network, Groundw. Sustain. Dev. 25 (2024) 101156.
[26] A. Demir Yetiş, N. İlhan, H. Kara, Integrating deep learning and regression models for accurate prediction of groundwater fluoride contamination in old city in Bitlis province, eastern Anatolia region, Türkiye, Environ. Sci. Pollut. Res. 31 (2024) 47201–47219.
[27] F.B. Banadkooki, M. Ehteram, F. Panahi, S.S. Sammen, F.B. Othman, A. El-Shafie, Estimation of total dissolved solids (TDS) using new hybrid machine learning models, J. Hydrol. 587 (2020) 124989.
[28] A. El Bilali, A. Taleb, Y. Brouziyne, Groundwater quality forecasting using machine learning algorithms for irrigation purposes, Agric. Water Manag. 245 (2021) 106625.
[29] A. Aryafar, V. Khosravi, H. Zarepourfard, R. Rooki, Evolving genetic programming and other AI-based models for estimating groundwater quality parameters of the Khezri plain, Eastern Iran, Environmental earth sciences 78 (2019) 69.
[30] C. Pan, K.T.W. Ng, B. Fallah, A. Richter, Evaluation of the bias and precision of regression techniques and machine learning approaches in total dissolved solids modeling of an urban aquifer, Environ. Sci. Pollut. Res. 26 (2019) 1821–1833.
[31] S. Abba, M. Benaafi, A. Usman, D.U. Ozsahin, B. Tawabini, I.H. Aljundi, Mapping of groundwater salinization and modelling using meta-heuristic algorithms for the coastal aquifer of eastern Saudi Arabia, Sci. Total Environ. 858 (2023) 159697.
[32] M. Mahmoudi, A. Mahdavi-Meymand, A. AlDallal, M. Zounemat-Kermani, Improving groundwater quality predictions in semi-arid regions using ensemble learning models, Environ. Sci. Pollut. Res. 32 (2025) 1985–2006.
[33] S. Gulati, A. Bansal, A. Pal, Estimating Total dissolved solids in groundwater using machine learning models, Nat. Resour. Res. 34 (2025) 1623–1644.
[34] M.U. Farooq, A.M. Zafar, W. Raheem, M.I. Jalees, A. Aly Hassan, Assessment of algorithm performance on predicting total dissolved solids using artificial neural network and multiple linear regression for the groundwater data, Water 14 (2022) 2002.
[35] A. Graves, Long short-term memory, Supervised sequence labelling with recurrent neural networks, 2012, pp. 37–45.
[36] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (1997) 1735–1780.
[37] K. Greff, R.K. Srivastava, J. Koutník, B.R. Steunebrink, J. Schmidhuber, LSTM: a search space odyssey, IEEE transactions on neural networks and learning systems 28 (2016) 2222–2232.
[38] S. Shadkani, Y. Hemmatzadeh, A. Pak, S. Abolfathi, Prediction of suspended sediment concentration in fluvial flows using novel hybrid deep learning model, Int. J. Sed. Res. 40 (4) (2025) 573–587.
[39] K. Khosravi, A.A. Farooque, M. Karbasi, M. Ali, S. Heddam, A. Faghfouri, S. Abolfathi, Enhanced water quality prediction model using advanced hybridized resampling alternating tree-based and deep learning algorithms, Environ. Sci. Pollut. Res. (2025) 1–20.
[40] S. Tomar, R. Tripathi, S. Kumar. Comprehensive survey of QML: from data analysis to algorithmic advancements, 2025. https://doi.org/10.48550/arXiv.2501.09528.
[41] Y. Li, Z. Wang, R. Han, S. Shi, J. Li, R. Shang, H. Zheng, G. Zhong, Y. Gu, Quantum recurrent neural networks for sequential learning, Neural Netw. 166 (2023) 148–161.
[42] H.K. Moghaddam, E. Abtahizadeh, S. Abolfathi, Sustainable water allocation under climate change: deep learning approaches to predict drinking water shortages, J. Environ. Manag. 385 (2025) 125600.
[43] I.-C. Chen, H. Singh, V. Anukruti, B. Quanz, K. Yogaraj, A survey of classical and quantum sequence models, in: in: 2024 16th International Conference on COMmunication Systems & NETworkS (COMSNETS), IEEE, 2024, pp. 1006–1011.
[44] S. Borzooei, L. Scabini, G. Miranda, S. Daneshgar, L. Deblieck, O. Bruno, P. De Langhe, B. De Baets, I. Nopens, E. Torfs, Evaluation of activated sludge settling characteristics from microscopy images with deep convolutional neural networks and transfer learning, J Water Process Eng 64 (2024) 105692.
[45] E.R. Coutinho, J.G. Madeira, D.G. Borges, M.V. Springer, E.M. de Oliveira, A. L. Coutinho, Multi-step forecasting of meteorological time series using CNN-LSTM with decomposition methods, Water Resour. Manag. (2025) 1–26.
[46] N. Ketkar, J. Moolayil, Convolutional neural networks, in: Deep learning with Python: learn best practices of deep learning models with PyTorch, Springer, 2021, pp. 197–242.
[47] H.H. Aghdam, E.J. Heravi, Guide to convolutional neural networks 10, Springer, New York, NY, 2017, p. 51.
[48] I. Ahmadianfar, O. Bozorg-Haddad, X. Chu, Gradient-based optimizer: a new metaheuristic optimization algorithm, Inf. Sci. 540 (2020) 131–159.
[49] Y. Jiang, Q. Luo, Y. Wei, L. Abualigah, Y. Zhou, An efficient binary gradient-based optimizer for feature selection, Math. Biosci. Eng. 18 (2021) 3813–3854.
[50] E. Albini, J. Long, D. Dervovic, D. Magazzeni, Counterfactual Shapley Additive Explanations, in: Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency, 2022, pp. 1054–1070.
[51] S.M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, Adv. Neural Inf. Proces. Syst. 30 (2017).
[52] C. Yang, X. Guan, Q. Xu, W. Xing, X. Chen, J. Chen, P. Jia, How can SHAP (SHapley additive exPlanations) interpretations improve deep learning based urban cellular automata model? Comput. Environ. Urban. Syst. 111 (2024) 102133.
[53] M. Sharafi, S. Shadkani, A. Pak, A. Motadayen, S. Samadianfard, E. Aras, V. Nourani, Developing long short-term memory combined with numerical first order differential optimization and clockwork recurrent neural network to predict suspended sediment load, Earth Sci. Inf. 18 (2025) 5.
[54] P. Hu, J. Tong, J. Wang, Y. Yang, L. de Oliveira Turci, A Hybrid Model Based on CNN and Bi-LSTM for Urban Water Demand Prediction, in: IEEE Congress on Evolutionary Computation (CEC) 2019, 2019, pp. 1088–1094. IEEE.
[55] T. Geetha, C. Chellaswamy, E. Raja, K. Venkatachalam, Deep learning for river water quality monitoring: a CNN-BiLSTM approach along the Kaveri River, Sustainable Water Resources Management 10 (2024) 125.
[56] K.R.A. Haq, V. Harigovindan, Water quality prediction for smart aquaculture using hybrid deep learning models, IEEE Access 10 (2022) 60078–60098.

[57] S. Davoudi, K. Roushangar, Innovative approaches to surface water quality management: advancing nitrate (NO3) forecasting with hybrid CNN-LSTM and CNN-GRU techniques, Model. Earth Syst. Environ. 11 (2025) 80.
[58] I. Uluocak, M. Bilgili, Daily air temperature forecasting using LSTM-CNN and GRU-CNN models, Acta Geophys. 72 (2024) 2107–2126.
[59] K. Ullah, W. Akram, A. Hassan, S.A.S. Bokhari, S. Abid, H. Yousaf, A. Farooq, Hybrid CNN–BiGRU model with attention mechanism for enhanced short-term load forecasting, Energy Rep. 14 (2025) 2570–2577.
[60] T. Wang, L. Ding, D. Zhang, J. Chen, A hybrid model combined deep neural network and Beluga whale optimizer for China urban dissolved oxygen concentration forecasting, Water 16 (2024) 2966.
[61] Y. Yang, H. Li, M. Sun, X. Liu, L. Cao, A study on hyperspectral soil moisture content prediction by incorporating a hybrid neural network into stacking ensemble learning, Agronomy 14 (2024) 2054.
[62] P. Mei, M. Li, Q. Zhang, J. Peng, Y. Bao, S. Zhang, Optimization model of process parameters for waterworks based on CNN-ISSA-BiGRU, J. Hydrol. 633 (2024) 130933.
[63] X. Li, S. Zhou, F. Wang, A CNN-BiGRU Sea level height prediction model combined with bayesian optimization algorithm, Ocean Eng. 315 (2025) 119849.
[64] S. Mishra, R. Manikandan, N. Mishra, G. Niphadkar, M. Shobana, M.A. Khan, IoT based smart irrigation monitoring and controlling system using GA-CNN-BiGRU model, in: Hybrid and Advanced Technologies, CRC Press, 2025, pp. 7–12.
[65] P. Saini, B. Nagpal, PSO-CNN-bi-LSTM: a hybrid optimization-enabled deep learning model for smart farming, Environ. Model. Assess. 29 (2024) 517–534.
[66] M. Abou Houran, S.M.S. Bukhari, M.H. Zafar, M. Mansoor, W. Chen, COA-CNN-LSTM: coati optimization algorithm-based hybrid deep learning model for PV/wind power forecasting in smart grid applications, Appl. Energy 349 (2023) 121638.
[67] X. Cui, J. Zhu, L. Jia, J. Wang, Y. Wu, A novel heat load prediction model of district heating system based on hybrid whale optimization algorithm (WOA) and CNN-LSTM with attention mechanism, Energy 312 (2024) 133536.
[68] J. Cai, B. Sun, H. Wang, Y. Zheng, S. Zhou, H. Li, Y. Huang, P. Zong, Application of the improved dung beetle optimizer, muti-head attention and hybrid deep learning algorithms to groundwater depth prediction in the Ningxia area, China, Atmospheric and Oceanic Science Letters 18 (2025) 100497.