

Article

Not peer-reviewed version

Water Quality Identification: Integrating IoT Sensors and Deep Learning for Near-Real-Time Water Quality Assessment

[Christina Tsolaki](#) , [George Kokkonis](#) , [Stavros Valsamidis](#) , [Sotirios Kontogiannis](#) *

Posted Date: 15 April 2026

doi: 10.20944/preprints202604.1037.v1

Keywords: smart cities; water quality; IoT sensors; machine learning; real-time monitoring; environmental monitoring; deep learning; low-cost sensors






Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Water Quality Identification: Integrating IoT Sensors and Deep Learning for Near-Real-Time Water Quality Assessment

Christina Tsolaki ¹, George Kokkonis ² , Stavros Valsamidis ³  and Sotirios Kontogiannis ^{1,*} 

¹ Laboratory Team of Distributed Microcomputer Systems, Department of Mathematics, University of Ioannina, 45110 Ioannina, Greece

² Department of Information and Electronic Engineering, International Hellenic University, 57001 Thessaloniki, Greece

³ Dept. of Accounting and Finance, Democritus University of Thrace, Ag. Loukas, 65404 Kavala, Greece

* Correspondence: skontog@uoi.gr

Abstract

The increasing demand for sustainable and affordable smart-city infrastructure has intensified the need for low-cost, near-real-time water-quality monitoring systems. In this study, we propose Water-QI, a low-cost Internet of Things (IoT)-based environmental monitoring platform that combines budget-friendly sensors with deep learning for Water Quality Index (WQI) assessment and forecasting. The sensing platform measures five key physicochemical parameters, namely temperature, total dissolved solids (TDS), pH, turbidity, and electrical conductivity, enabling continuous multi-parameter monitoring in urban water environments. To model temporal variations in water quality under both cloud-based and edge-oriented deployment scenarios, we evaluate multiple Gated Recurrent Unit (GRU) architectures with different widths and depths. Experiments are conducted at two temporal resolutions, hourly and minute-level, in order to examine the trade-off between predictive accuracy and computational cost. In the hourly scenario, the single-layer GRU with 64 units achieved the best overall balance, reaching a validation RMSE of 0.0281 and a test R^2 of 0.9820, while deeper stacked GRU models degraded performance substantially. In the minute-resolution scenario, shallow wider GRU models produced the best results, with the single-layer GRU with 512 units attaining the lowest validation RMSE (0.025548) and the 256-unit variant achieving nearly identical accuracy with much lower inference cost. The results show that increasing model width can yield marginal improvements at high temporal granularity, whereas excessive recurrent depth consistently harms convergence and generalization. Overall, the findings indicate that shallow GRU architectures provide the most practical solution for accurate, low-cost, and scalable near-real-time water-quality forecasting. In particular, the 64-unit GRU is the most suitable choice for hourly low-complexity operation, while the 256-unit GRU offers the best speed-accuracy trade-off for minute-level edge inference on resource-constrained devices.

Keywords: smart cities; water quality; IoT sensors; machine learning; real-time monitoring; environmental monitoring; deep learning; low-cost sensors

1. Introduction

Modern cities experience mounting pressure as population density rises, resulting in significant strain on natural resources and infrastructure. Among the most pressing concerns is water pollution, which requires urgent resolution. Addressing this issue necessitates real-time water quality measurement rather than exclusive reliance on laboratory analysis. However, most existing water quality assessment methods remain prohibitively expensive and are not yet implemented for real-time monitoring.

Recent advancements in the Internet of Things (IoT) and cloud technologies, driven by the fourth industrial revolution (Industry 4.0), have enabled continuous, near-real-time monitoring and

processing of environmental data. This research examines the role of low-cost sensors, combined with emerging technologies such as machine learning and artificial intelligence, in facilitating real-time water quality measurement across urban environments. Such capabilities empower cities to implement timely preventive measures, thereby enhancing urban quality of life.

Recent developments in sensor technology have significantly reduced the cost barriers to environmental monitoring. The study [1] presents a low-cost IoT-based water quality monitoring system that combines Arduino microcontrollers and cloud-connected sensors to measure real-time parameters, including pH, turbidity, temperature, and total dissolved solids (TDS). This system achieves 91% accuracy levels, 20% higher than traditional models—through a feed-forward artificial neural network optimized using a hybrid Genetic Algorithm–Particle Swarm Optimization (GA-PSO) approach. Its affordability and automation capabilities make it highly applicable across various domains, including drinking water safety, agriculture, aquaculture, industrial water monitoring, and smart city infrastructure.

In support of the movement toward accessible monitoring solutions, [2] highlights low-cost machine learning and IoT-based technologies for real-time water quality assessment. The research identifies several cost-effective strategies, including solar disinfection (SODIS), ceramic and bamboo charcoal filters, treadle and rope pumps, low-cost drip irrigation systems, underground storage tanks, and nature-based solutions such as microalgae filtration. Furthermore, carbon nanotube-based chemical sensors and community-based water management practices enhance the accessibility and sustainability of water quality management.

Low-cost sensor technologies typically measure essential water quality parameters, offering valuable insights without requiring complex equipment. According to [3], variables such as pH, temperature, and electrical conductivity serve as fundamental inputs for machine learning models that forecast groundwater quality for irrigation. By reducing the need for extensive laboratory investigations, this approach enables near-real-time, cost-effective prediction of critical irrigation indicators, including total dissolved solids (TDS), electrical conductivity, turbidity, and potential salinity.

Similarly, [4] demonstrates that a limited set of characteristics, including temperature, turbidity, pH, and total dissolved solids, can yield accurate water quality predictions. The study conducted in Pakistan's Rawal watershed shows that supervised machine learning methods can serve as cost-effective alternatives to conventional laboratory tests while maintaining sufficient predictive accuracy.

Despite their advantages, low-cost sensor systems encounter several challenges. The authors of [5] observe that although traditional laboratory methods are more precise, they are also costly and time-consuming. In contrast, low-cost systems provide real-time monitoring, remote data transmission, and instant alerts, but require periodic calibration to maintain accuracy. The study identifies key barriers to implementation, including insufficient data management, limited model explainability, and low reproducibility.

To address these limitations, [6] proposes integrating Long Short-Term Memory (LSTM) networks with denoising techniques such as wavelet transform and moving average, alongside random forest-based feature selection to eliminate noise and collinear variables. This methodology mitigates data inconsistency and sensor limitations through advanced preprocessing and model optimization. Consequently, incorporating low-cost water quality monitoring systems into smart city frameworks provides significant benefits for urban management and sustainability. As noted by [1], affordable and automated monitoring systems are highly applicable to drinking water safety, agriculture, aquaculture, industrial water monitoring, and broader smart city infrastructure.

The authors of [7] position machine learning as a transformative tool for urban water management, enabling rapid responses to flooding, contamination, and system failures while reducing infrastructure costs. The research underscores the value of low-cost surrogate models for cities with budget constraints and recommends that future studies focus on enhancing model transferability across diverse urban contexts and adapting to evolving infrastructure conditions. Practical implementation is illustrated in [8] through the WaterS system, which utilizes Sigfox for IoT connectivity. This open-source

approach supports scalability and collaborative development, addressing challenges such as increased packet error rates in dense deployments by evaluating protocols and optimizing communication.

This paper introduces a distributed platform, Water Quality Identification (Water-QI), designed for periodic, hourly, or near-real-time minute-level monitoring of water quality attributes at the source. The platform leverages low-cost sensors and a high spatial density of GPS-based IoT nodes to monitor qualitative drinking water attributes in urban environments. Additionally, it utilizes existing city Wi-Fi infrastructure, incorporates predictive models either on-device or in the cloud, and employs a device-level correlation function for immediate calculation of the Water Quality Index (WQI). The integration of Deep Learning GRU models for measurement prediction and WQI calculations further enhances the platform's suitability for edge-level computations.

The structure of this paper is as follows: Section 2 presents a review of related work, emphasizing the differentiation and potential of machine learning and deep learning methods for predicting and classifying water quality attributes. Section 3 details the proposed approach, Section 4 discusses experimental results obtained using deep learning models, and Section 5 summarizes these findings. Finally, Section 6 provides the conclusion.

2. Related Work

This section reviews the existing literature on machine learning and deep learning approaches for water quality monitoring, categorizing them into traditional methods and advanced deep learning architectures, and then compares their performance. The performance metrics evaluated across these studies are consistent with those summarized in the review by [9]. This survey shows that regression tasks commonly use R^2 , RMSE, and MSE, whereas classification tasks rely on accuracy, precision, and recall. The review corroborates the findings of individual studies, noting that models like Machine Learning XGBoost and Deep Learning LSTMs often achieve high predictive performance due to their ability to handle complex, non-linear relationships in water quality data.

2.1. Machine Learning Models for Water Quality Assessment

In recent years, machine learning methods have been extensively applied to the prediction and classification of water quality parameters. To assess model performance, researchers commonly utilize established metrics. For regression tasks, the primary indicators include the coefficient of determination (R^2), Mean Absolute Error (MAE), and Root Mean Square Error (RMSE). For classification tasks, standard metrics comprise accuracy, precision, recall, and F1-score. Each metric provides a distinct perspective on model performance and prediction reliability.

The study by [10] evaluates multiple machine learning models, including the adaptive neuro-fuzzy inference system (ANFIS), feed-forward neural networks (FFNN), and K-nearest neighbors (KNN), for predicting the water quality index (WQI) and classifying water quality (Water Quality Classification, WQC). The findings demonstrate that the ANFIS model achieved a regression coefficient of 96.17% for WQI prediction, while the FFNN attained 100% accuracy for WQC classification.

The comparative analysis presented in [11] examines residual chlorine prediction in drinking water using various machine learning approaches. Among the individual models evaluated, Deep Learning models exhibited strong performance results. The study also investigates multi-model ensembles (MMEs), revealing that optimal model combinations can explain up to 74% of the variance in observed chlorine levels, thereby offering cost-effective alternatives to complex physical models. Further research by [12] reports that Multiple Linear Regression (MLR) achieved high regression performance ($R^2 = 0.9992$, RMSE = 0.338) in predicting drinking water quality in seawater desalination plants. SHAP analysis identified total bacterial count, TDS, sodium, chloride, and residual chlorine as critical features, supporting an interpretable machine learning-based approach for sustainable plant operations.

Similarly, Khan et al. [13] implemented a Water Quality Prediction and Classification (WQPC) system utilizing an SVM classifier. The model was primarily evaluated using accuracy, calculated from

the confusion matrix components: True Positives, True Negatives, False Positives, and False Negatives. The proposed system reported a high prediction accuracy ranging from 92% to 95%.

Garcia et al. [14] evaluated the performance of Decision Tree, Random Forest, and K-Nearest Neighbors (KNN) algorithms for classifying groundwater quality based on Total Dissolved Solids (TDS). The models were assessed using accuracy, precision, recall, F1-score, and cross-validation. Both Decision Tree and Random Forest achieved superficial perfect scores (100%) across all metrics, which was likely due to overfitting. In contrast, KNN achieved an accuracy of 92.9%, with an average cross-validation score of 93.7% and a standard deviation of 3.74%. The authors concluded that tree-based models provided superior performance and greater stability for water quality classification tasks.

Patil et al. [15] compared several machine learning algorithms, including SVM, Random Forest, KNN, and Logistic Regression, to predict water potability. Among these, the SVM classifier achieved the highest accuracy of 64%, with precision values of 0.71 for non-potable and 0.56 for potable water. The study focused on standard classification metrics rather than regression-based errors such as MSE or RMSE, highlighting the importance of class-wise performance and model robustness in handling imbalanced datasets instead.

Ding et al. [16] improved Water Quality Index (WQI) modeling using CatBoost, SVM, Logistic Regression (LR), XGBoost, and LightGBM. The models were assessed through standard uncertainty (SU) and sensitivity analysis using multiple linear regression (MLR), reporting both R^2 and RMSE values. The CatBoost-based model demonstrated the lowest uncertainty (0.559–0.903), while the SVM and LR models showed higher uncertainty levels (0.576–1.034). Additionally, the study examined different aggregation functions (SGM, WQM, SWM) and discussed how they influenced model sensitivity and overall reliability.

The study by Iyer et al. [17] explored the use of machine learning algorithms, including SVM, Random Forest, and Decision Tree, for predicting water quality. Model performance was evaluated using accuracy, precision, recall, and F1-score. The authors found that Random Forest achieved the highest accuracy of 68%, outperforming the other models. However, the study did not report additional error metrics such as MSE or RMSE.

In the work of Prasad and Ranjit [18], various machine learning algorithms, including Logistic Regression, SVM, Naive Bayes, KNN, Decision Tree, Random Forest, AdaBoost, and XGBoost, were applied for water quality classification. Accuracy was the main evaluation metric, with both Decision Tree and XGBoost achieving 99% accuracy. Other metrics, such as precision, recall, and F1-score, were also included, though regression-based errors, such as MSE and RMSE, were not reported.

Walczak and Walczak [19] conducted a comparative analysis of four machine learning algorithms such as Neural Networks, Random Forest, KNN, and multivariate Linear Regression for predicting the Water Quality Index (WQI) with a reduced set of input variables. Model performance was assessed using Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the coefficient of determination (R^2). Both the Neural Network and Linear Regression models attained an R^2 value of 0.999 when trained on the complete dataset; however, RMSE values varied depending on the number of predictors included.

Karthick et al. [20] evaluated 14 machine learning algorithms for water quality classification, employing advanced preprocessing techniques such as the Yeo-Johnson transformation, principal component analysis (PCA), and Synthetic Minority Over-sampling Technique (SMOTE). Model performance was measured using accuracy, precision, recall, F1-score, and ROC AUC. The XGBoost model achieved the highest accuracy at 96.31% without SMOTE, while LightGBM also demonstrated strong performance.

Shams et al. [21] assessed several machine learning models for predicting both the Water Quality Index (WQI) and Water Quality Classification (WQC). The authors applied grid search for hyperparameter optimization and evaluated regression models using Mean Absolute Error (MAE), Median Absolute Error (MedAE), Mean Squared Error (MSE), and the coefficient of determination (R^2). For classification tasks, accuracy, recall, precision, and F1-score are employed as evaluation metrics. The

Multi-Layer Perceptron (MLP) model achieved the highest regression performance, with an R^2 of 99.8% and an MSE of 2.8×10^{-5} . For classification, the Gradient Boosting (GB) model attained an accuracy of 99.5%.

Prabu et al. [22] focused on anomaly detection in water treatment plants using a modified Quality Index (QI) and an encoder-decoder architecture. They evaluated their model using accuracy, precision, recall, Critical Success Index (CSI). Their proposed model achieved 89.18% accuracy, 85.54% precision, 94.02% recall, demonstrating strong performance in real-time anomaly detection and water quality monitoring.

Similarly, the authors at [23] demonstrate that their proposed WDT-ANFIS model, which combines Wavelet De-noising Techniques with Adaptive Neuro-Fuzzy Inference Systems, significantly improved prediction accuracy for water quality parameters, achieving R^2 values of at least 0.9 for most parameters while outperforming other approaches like MLP-ANN and RBF-ANN.

The authors of [24] proposed two hybrid models: CEEMDAN-XGBoost and CEEMDAN-RF for short-term water quality prediction, combining decision tree algorithms with the CEEMDAN denoising method. Using hourly data from the Tualatin River, these models demonstrated superior accuracy and stability compared with benchmarks such as LSTM and SVM, with CEEMDAN-RF excelling at predicting temperature and dissolved oxygen, while CEEMDAN-XGBoost performed better for pH and turbidity prediction. The study by [25] addresses the common challenge of class imbalance in water-quality datasets by using an adaptive synthetic sampling (ADASYN) approach to generate synthetic instances above the threshold. Their evaluation of four machine learning models using k-nearest neighbors, boosting decision trees (BDT), support vector machines (SVM), and multi-layer perceptron neural networks (MLP-ANN) showed that BDT and MLP-ANN achieved the highest accuracy and sensitivity (over 90% and 75% respectively).

For water quality classification, the Random Forest Classifier consistently achieved the highest accuracy, reaching 98.2%, with precision, recall, and F1-score all equal to 0.98, as reported in [26]. K-Nearest Neighbors (KNN) and Support Vector Machine (SVM) models also demonstrated strong performance, achieving accuracy levels of 97%–98%. These findings underscore the robustness of ensemble and kernel-based methods for multi-class environmental prediction tasks. Furthermore, [26] reports that models such as the Extra Tree Regressor (ETR) and Random Forest Regressor (RFR) achieve R^2 values close to 0.99, with RMSE values between 1.55 and 1.71 and MAE below 0.75. Ridge Regression, Artificial Neural Networks (ANN), and ANFIS models also maintained competitive performance, with R^2 values above 0.95.

The authors of [27] applied a Linear model (LTSP-Linear) for water quality prediction at the Huangyang Reservoir, focusing on dissolved oxygen (DO), pH, and turbidity. Model evaluation using MSE and MAE indicated reductions of 8.55% in MSE and 10.51% in MAE compared to deep learning models such as LSTM and Informer, as well as the statistical ARIMA. For short-term DO prediction (96-step), the Linear model achieved an MSE of 0.05479 and an MAE of 0.15481, demonstrating high efficiency and performance despite its simple architecture.

The comprehensive review in [28] highlights the effectiveness of models such as Extreme Gradient Boosting (XGBoost) and Long Short-Term Memory (LSTM) networks for water quality prediction. The analysis demonstrates that XGBoost achieves up to 99.99% accuracy, while LSTM attains an R^2 of 0.9999 when predicting key parameters, including chlorophyll-a, turbidity, salinity, and dissolved oxygen.

In the context of water-quality prediction, the authors introduce a new performance metric, the *Water Quality Score (WQS)*, to facilitate the evaluation of models proposed in the literature. The score is computed according to Equation (1):

$$WQS = \alpha \cdot (1 - RMSE) + (1 - \alpha) \cdot R^2 \quad (1)$$

where $RMSE$ represents the normalized root mean square error of the model predictions, R^2 is the coefficient of determination, and $\alpha \in [0.5, 1)$ is a weighting factor. Setting $\alpha = 0.5$ assigns equal

importance to the error and goodness-of-fit terms, while $\alpha = 1$ eliminates the contribution of R^2 entirely. The authors adopt $\alpha = 0.8$, emphasizing predictive accuracy over explained variance. This choice is motivated by the nature of water-quality datasets, in which measurements are often noisy and acquired at relatively long probing intervals, commonly greater than 24 hours, making variance-based metrics less informative. If WQS values are above 1, then it is considered as an underfitted result or the attributes that contribute to RMSE value have not been normalized.

An exploratory analysis of the proposed Water Quality Score (WQS) shows that its behavior is strongly governed by the normalized RMSE term, since for $\alpha = 0.8$ predictive error contributes 80% of the final score, while R^2 contributes only 20%. Under the intended use of the metric, RMSE is expected to be normalized and therefore typically remain in the interval $[0, 1]$. In that case, $1 - RMSE$ is nonnegative, and because R^2 is also usually bounded by $[0, 1]$, WQS is expected to lie approximately within $[0, 1]$, with higher values indicating better overall predictive performance. A value close to 1 reflects both low prediction error and strong explanatory power, whereas moderate values indicate a trade-off between acceptable fit and non-negligible error. Negative WQS values arise when the normalized RMSE exceeds 1, making the term $(1 - RMSE)$ negative and causing the error contribution to outweigh the positive contribution of R^2 . Such cases suggest either very poor predictive accuracy or that RMSE was not properly normalized. Conversely, WQS values greater than 1 should not normally occur if RMSE is correctly normalized and $R^2 \leq 1$; therefore, values above 1 strongly indicate a scaling inconsistency, a calculation error, or the use of a non-normalized RMSE.

The literature review data summarized in Table 1 highlights a significant shift in environmental monitoring, moving away from simple classification toward complex, time-aware forecasting. At the heart of this effort is the concept of water potability, which is essentially a measure of whether water is safe for long-term human consumption without health risks. This is not determined by a single "clean" look, but by a delicate balance of physical and chemical parameters, such as pH, mineral concentration via TDS and EC, and the presence of suspended particles measured as turbidity.

When analyzing high-performance scores reported in the literature, a degree of scientific skepticism is necessary. While seeing a coefficient of determination (R^2) of 0.9999 might seem like a superficial achievement, in the messy world of real-world environmental sensors, such high values are often a red flag. Therefore, near-perfect values can be superficial, frequently indicating that a model has overfitted to a very specific, noise-free dataset or that the experimental conditions were too controlled to be realistic. In a true urban environment, sensor drift, biofouling, and sudden weather changes introduce unpredictability that makes absolute perfection nearly impossible. Therefore, in our research, we prioritize models that demonstrate high reliability in the 94–96% range, as they tend to be more robust and better at generalizing to the actual complexity of city water networks. By focusing on realistic loss metrics like RMSE and curve fit metrics like the R^2 we also ensure that our proposition system remains easily re-evaluated by others. In the following Section 2.2, a literature review of Deep Learning models is presented.

Table 1. Performance Metrics of Traditional Machine Learning and Shallow Neural Network Architectures.

Regression Tasks					
R^2 value	RMSE Value	Score(WQS)	Model Architecture	Superficial	Study
0.9239	0.0540	0.9416	ANFIS (5 hidden layers NN/Sugeno-Fuzzy)	No	[10](Table 4)
0.9992	0.3377	0.7299	MLR (Linear regression-20 input parameters)	Yes, underfitting	[12](Table 6)
0.998	0.00529	0.9954	MLP (Numerous hidden layers-unspecified)	Yes, overfitting	[21](Table 9)
0.99	1.55	-0.242	Extra Trees Regressor	Yes-High RMSE	[26](Table 4)
0.000	0.028	0.78	Linear Regression Model (LRM)	Small Dataset-Superficial fit of $R^2 = 1$, set to zero	[19](Table 4)
-*	0.241	0.607	LTSF-Linear	Simple architecture	[27](Table 2)
0.94	0.15	0.868	WDT-ANFIS	No	[23](Table 5 / Fig 12)
0.736	0.0054	0.94288	Multi-model Ensembles-preferably DL than ML models	No	[11](Table 1)
-*	0.096	0.72	CEEMDAN RF / Data denoising	No (Hybrid-Ensemble)	[24](Table 3)
0.722	0.0843	0.87696	CatBoost (Uncertainty-based modeling)	Focus on SU	[16](Sec 4.2.3/Figure 8)
Classification Tasks					
	Metric	Metric value	Model Architecture	Superficial	Study
	Accuracy	0.982	Random Forest Classifier	Yes-Small Dataset	[26](Table 5)
	Accuracy	0.963	XGBoost (without SMOTE)	No	[20](Table 5)
	Accuracy	1	Decision Tree and Random Forest	Yes- Overfitting	[14](Table 3)
	Accuracy	0.64	Support Vector Machine (SVM)	Yes (Imbalanced dataset)	[15](Sec. Results)
	Accuracy	0.8506	Random Forest	No	[18](Table 7)
	Accuracy	0.995	Gradient Boosting (GB)	No	[21](Table 6)
	Accuracy	0.69	Support Vector Machine (SVM)	Yes (Poor minority class prediction)	[17](Table 1 / Sec. Results & Discussion)
	Accuracy	0.8918	Encoder-Decoder (Anomaly detection)	No	[22](Table 9)
	Accuracy	0.92	MLP-ANN	No	[25](Sec 4.4)
	Accuracy	1	Decision Tree & Random Forest	Yes (Multicollinearity & Data Leakage / Overfitting)	[14](Tables 3 & 4)

* Values with no calculated R^2 are considered as $R^2 \rightarrow 0$.

2.2. Deep Learning Models for Water Quality Assessment

Deep learning has emerged as a powerful tool for predicting water quality, enabling models to capture complex temporal dependencies and nonlinear relationships in multivariate time-series data. Techniques such as Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and hybrid architectures have demonstrated superior performance in forecasting critical water-quality parameters, including dissolved oxygen (DO), chlorophyll a, and turbidity. These models leverage historical sensor data to provide accurate, real-time predictions, facilitating proactive water-quality management and early-warning systems for aquaculture and environmental monitoring.

Recent advances in deep learning have enabled significant improvements in the accuracy of water quality predictions. The authors of [29] present an integrated Artificial Ecosystem Optimization with Deep Learning Enabled Water Quality Prediction and Classification (AEODL-WQPC) model that delivers rapid, cost-effective, and highly accurate water quality analysis. The approach employs an Optimal Stacked Bidirectional Gated Recurrent Unit (OSBiGRU) for water quality index prediction and an Improved Elman Neural Network (IENN) for classification. This model achieved superior

performance with a mean squared error (MSE) of 0.0021, root mean squared error (RMSE) of 0.0458, and regression accuracy approaching 96%.

Wang et al. [30] proposed a comprehensive weighting method combining entropy weighting and Pearson correlation for feature selection in water quality prediction. They compared models including SVM, MLP, RF, XGBoost, and LSTM using metrics such as MSE, RMSE, Nash–Sutcliffe Efficiency (NSE), and R^2 . The LSTM model outperformed the other models, especially in predicting Dissolved Oxygen (DO), achieving an R^2 of 0.882, an MSE of 3.361, and an RMSE of 1.827. The study highlights the importance of feature selection and temporal modeling in improving prediction accuracy.

Similarly, the authors of [31] evaluated Automated Deep Learning (AutoDL) for water quality assessment and compared it with conventional deep learning models, including ANN, RNN, LSTM, and CNN. Their findings show that LSTM achieved 92% accuracy for binary classification and 94% for multiclass classification, while CNN achieved 95% for binary classification and 91% for multiclass classification. Although conventional DL models slightly outperformed AutoDL, the latter significantly reduced manual effort by automating model selection and tuning.

The study at [32] examined the use of Artificial Neural Networks (ANN) to predict 6 Water Quality parameters of the Langat River, Malaysia. The reported results were incredibly promising, achieving R^2 values of 0.88–0.99 during the testing phase and very low RMSE values (as low as 7.7×10^{-23}). The ANN with 10 hidden neurons and the Levenberg-Marquardt algorithm, in particular, yielded the most stable results and the lowest errors in predicting Phosphate and TSS.

In the context of river water quality, the authors of [33], developed an LSTM model enhanced with an attention mechanism (AT-LSTM) to predict dissolved oxygen in the Burnett River, Australia. The model was evaluated using Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and the coefficient of determination (R^2), achieving values of 0.094, 0.130, and 0.953, respectively. The inclusion of the attention mechanism significantly improved the model's ability to focus on relevant temporal features, resulting in 23.9% and 27.7% reductions in RMSE and MAE, respectively, compared to the standard LSTM.

For aquaculture applications, Gandhi et al. [34] applied both LSTM and GRU models to predict key water-quality parameters, including salinity, pH, DO, and temperature. Through extensive hyperparameter optimization, the models were assessed using MAE, MSE, RMSE. On the ADAK dataset, the GRU model achieved an MAE of 0.0272 and an RMSE of 0.036 for DO prediction, demonstrating a favorable balance between computational efficiency and forecasting accuracy.

Addressing the challenge of incomplete datasets due to spurious measurements, the authors of [35] proposed a Kalman Filter-based LSTM encoder-decoder model (KF-LSTM) incorporating an attention mechanism. This approach effectively reconstructed missing values and captured long-term dependencies, achieving an MAE of 0.30, an MSE of 0.16, an RMSE of 0.40, and an R^2 of 0.94 on the test set. The model achieved 10% reductions in MAE and 21.2% in MSE relative to a traditional LSTM, highlighting its robustness in real-world monitoring scenarios with frequent data gaps.

To improve forecasting accuracy, the authors of [36] introduced a hybrid model combining Ensemble Empirical Mode Decomposition (EEMD), Multivariate Linear Regression (MLR), and LSTM (EEMD-MLR-LSTM) to predict phytoplankton levels. Evaluated using MAE, MSE, and RMSE, the model achieved an MAE of 0.0375, MSE of 0.0024, and RMSE of 0.0489 for a 6-hour prediction horizon. The integration of EEMD improved feature extraction from non-stationary signals, yielding superior performance compared to standalone models and underscoring the value of hybrid approaches in complex aquatic ecosystems.

Hyperparameter tuning and the use of alternative network architectures have proven critical for enhancing prediction accuracy. The authors of [37] proposed a methodology that combines an LSTM model with the Grasshopper Optimization Algorithm (GOA) for automatic hyperparameter tuning. This technique, evaluated on a dataset of 1600 samples, achieved an average accuracy of 92.22%, precision of 92.30%, and an F1-score of 92.19% on the test set, outperforming conventional

methods such as SVM and Decision Trees. This underscores the importance of automated optimization for fully leveraging the potential of deep learning models.

The comprehensive review in [38] consolidates evidence on the efficacy of various Artificial Neural Network (ANN) architectures for this task. Their analysis shows that Recurrent Neural Network (RNN)-based models, particularly LSTM, achieve the highest performance, with measurement accuracies ranging from 96% to 98%. The review systematically compares models such as Feedforward Neural Networks (FFNN), Multi-Layer Perceptron (MLP), and Radial Basis Function Neural Networks (RBF), establishing a taxonomy that guides the selection of the most suitable architecture based on data characteristics and prediction goals, thereby providing a valuable roadmap for future research.

Addressing the challenge of data pre-processing, which is crucial for model performance, [39] emphasized the use of Z-score normalization and advanced neural networks, such as the Nonlinear Autoregressive Neural Network (NARNET) and LSTM. Their research, which also employed machine learning models such as the Support Vector Machine (SVM) for water quality classification, demonstrated that NARNET achieved a slightly higher regression coefficient (R^2 value of 96.17%) than LSTM (R^2 value of 94.21%) for predicting the Water Quality Index (WQI). This work highlights that the choice between different advanced neural network architectures can be nuanced and depends on the specific prediction task, whether regression (WQI) or classification (WQC).

Authors at [40] introduced a hybrid LSTM model combined with Gray Wolf Optimization (GWO) and Fish Swarm Optimization (FSO) for predicting DO, COD, and $\text{NH}_3\text{-N}$ in the Thamirabarani River. The model was assessed using RMSE, MAE, and R^2 . The LSTM-GWO-FSO model achieved an RMSE of 0.083, MAE of 0.055, and R^2 of 0.94 for DO, outperforming traditional ANN, BPNN, and RNN models and demonstrating the value of metaheuristic optimization in enhancing deep learning predictions for water quality.

For nationwide tap water quality monitoring in South Korea, the authors of [41] developed and compared deep learning models, including LSTM, GRU, and SCINet, against an ARIMA baseline. Performance was measured using MAE, and RMSE. The optimized SCINet model achieved the best results, with an average MAE of 0.0003, RMSE of 0.0006, illustrating the potential of advanced deep learning architectures for high-accuracy, real-time water quality forecasting in large-scale supply systems.

In a comprehensive methodology for water quality prediction using machine learning, the authors of [42], outlined the use of the XGBoost algorithm, emphasizing the importance of data preprocessing, feature selection, and model evaluation. The proposed framework uses regression metrics such as MAE, MSE, RMSE, and R^2 , as well as classification metrics including accuracy, precision, recall, and F1-score, providing a structured approach for developing robust, accurate predictive models tailored for environmental monitoring and decision-making. Table 2, presents the DL model results achieved in the recent literature.

Table 2. Performance Metrics of Deep Learning Architectures in Water Quality Monitoring.

Regression Tasks					
R^2 value	RMSE Value	Score(WQS)	Model Architecture	Superficial	Study
0.9421	0.3206	0.732	LSTM (Z-score normalization)	No	[39](Table 6)
0.9617	0.3678	0.6982	NARNET (Time-series)	No	[39](Table 6)
0.953	0.130	0.8866	AT-LSTM (Attention Mechanism)	No	[33](Table 4)
0.94	0.40	0.668	KF-LSTM (Kalman Filter)	No	[35](Table 3)
-*	0.008	0.7936	SCINet (1D CNN-NN hybrid model)	No	[41] (Table 7 mean values)
0.908	0.036	0.9528	GRU (Hyperparameter Optimized)	No	[34](Figure 4)
0.957	0.0489	0.9523	EEMD-MLR-LSTM (Hybrid)	No	[36](Table 3)
0.94	0.083	0.9216	LSTM-GWO-FSO (Metaheuristic)	No	[40](Table 1)
0.882	1.827	-0.4852	LSTM (Temporal modeling)	Yes-High RMSE	[30](Table 5)
0.97	0.019	0.9782	NN-10 hidden layers	No	[32](Table 1 mean values)
0.985	0.0378	0.9668	LSTM (Standard)	No	[43](Table 3)
Classification Tasks					
Metric	Metric value	Model Architecture	Superficial	Study	
Accuracy	0.96	OSBiGRU (Hybrid Optimization)	No	[29](Table 4)	
Accuracy	0.951	CNN (Convolutional)	No	[31](Table 3)	
Accuracy	0.926	LSTM (Binary Classification)	No	[31](Table 3)	
Accuracy	0.9222	LSTM-GOA (Grasshopper Opt)	No	[37](Table 2)	

* Values with no calculated R^2 are considered as $R^2 \rightarrow 0$.

According to the literature review, using DL models for regression tasks, SCINet [41] has demonstrated strong performance in complex time-series forecasting tasks due to its hierarchical architecture, which captures multi-scale temporal patterns and long-range dependencies more effectively than conventional sequential models in many cases. Nevertheless, LSTM remains a robust and widely adopted baseline, particularly for simpler tasks or smaller datasets; therefore, SCINet should be considered a potentially superior alternative in specific forecasting settings rather than a universally better model [44,45].

Based on the best-performing regression results reported in Table 2, the highest Water Quality Score (WQS) is achieved by a deep NN model with 10 hidden layers [32], followed by the standard LSTM [43], GRU [34], and hybrid LSTM [36]. The top score of the deep feedforward NNs [32], similar to stranded-NN [46] suggests that, in some water-quality prediction settings, carefully optimized dense architectures can outperform more complex sequential models when the underlying relationships are strongly nonlinear but do not require sophisticated temporal decomposition, however their increased depths may sufficiently degrade and narrow their performance to specific temporal intervals [46]. Nevertheless, the strong performance of the standard LSTM and the close-to-LSTM performance of the GRU, as a lighter edge-computational candidate, further confirm their effectiveness as a reliable regression baseline for water-quality forecasting, particularly in capturing temporal dependencies.

From the regression models studies in Table 2, GRU is especially relevant for sensor-based water-quality forecasting because it offers a simpler gated recurrent architecture than LSTM, with fewer parameters and lower computational complexity, while often retaining comparable predictive accuracy in sequence modeling tasks. This makes GRU attractive for fast inference and resource-constrained edge deployments. Prior comparative work has shown that GRU can be comparable to LSTM and may even converge faster under equal parameter budgets, while hydrology and water-related forecasting studies likewise note that GRU is less complicated and uses fewer parameters than LSTM, making it a practical choice for regression on environmental time series [47].

For classification tasks, the importance of CNN models should be emphasized, particularly one-dimensional convolutional models, which can learn local temporal and cross-channel patterns directly from raw or minimally processed multivariate sensor sequences, avoiding heavy feature engineering while remaining highly effective for classification and anomaly-related decision tasks in water monitoring systems [48]. Hybrid GRU models may outperform CNNs, as shown in [29], by offering a much larger set of hyperparameters and requiring more optimization effort.

2.3. ML-DL Comparative Analysis

The choice between Machine Learning (ML) and Deep Learning (DL) is no longer a theoretical debate but a strategic decision based on the results established in Tables 1 and 2 accordingly. Our analysis reveals a clear performance hierarchy: for high-precision regression and forecasting, Deep Learning is the definitive leader, while for robust classification of static data, traditional Machine Learning remains highly competitive. The evaluation of Machine Learning (ML) and Deep Learning (DL) architectures for water quality monitoring reveals a significant disparity between raw statistical performance and actual model reliability. While traditional models often report near-perfect metrics, a deeper inspection suggests that these results frequently stem from architectural limitations or data-specific artifacts rather than genuine predictive power.

The performance of traditional ML models in regression tasks is characterized by extreme variance. On the one hand, sophisticated fuzzy-logic integrations like the ANFIS model [10] demonstrate robust predictive capabilities, with an R^2 of 0.9239 and a high Water Quality Score (WQS) of 0.9416, without signs of overfitting. Similarly, multi-model ensembles [11] achieve a remarkable WQS of 0.94288, suggesting that aggregating weaker learners can mitigate the non-linearity of hydrological data.

However, a critical skepticism must be applied to models reporting perfect or near-perfect scores. For instance, the MLP architecture in [21] reports an R^2 of 0.998 and an RMSE of 0.00529, yet this is flagged as a clear case of overfitting. Such superficial performance is often an artifact of small datasets or insufficient rigorous cross-validation. This is further evidenced by the Linear Regression Model

in [19], where an initial R^2 of 1 was identified as a superficial fit and subsequently set to 0 for the analysis. In classification tasks, the trend continues; while Gradient Boosting achieves a high accuracy of 0.995 (see [21]), other models like the Support Vector Machine (SVM) struggle significantly, dropping to accuracies of 0.64 (see [15]) and 0.69 (see [17]). These failures are typically attributed to imbalanced datasets and to shallow models' inability to capture minority-class features in complex water-quality matrices.

In contrast to traditional methods, Deep Learning (DL) models exhibit more consistent and honest performance metrics. The dominance of Recurrent Neural Networks (RNNs), specifically Long Short-Term Memory (LSTM) networks, is evident across the literature. Standard LSTMs and their optimized variants, such as the hyperparameter-optimized GRU [34] ($WQS = 0.9528$) and the temporal modeling in [43] ($R^2 = 0.985$), consistently achieve high accuracy without being categorized as overfitting or underfitting.

The integration of attention mechanisms and hybrid optimization further stabilizes these models. For example, the AT-LSTM [33] utilizes an attention mechanism to weigh critical temporal features, achieving an R^2 of 0.953. Hybrid approaches that combine data denoising with DL, such as the EEMD-MLR-LSTM [36], show that pre-processing raw sensor data is vital for achieving high scores ($WQS = 0.9523$). In classification, DL models like the OSBiGRU [29] and CNN-based architectures [31] maintain high accuracies (above 0.95), demonstrating superior robustness against the noise and complexity inherent in environmental time-series data compared to their ML counterparts.

A cross-comparison between the two paradigms reveals that while traditional ML can occasionally match DL in terms of R^2 or Accuracy, it is far more prone to superficiality. The results from [14], which show an accuracy of 1.0 for Decision Trees and Random Forests, are prime examples of models that likely suffer from data leakage or multicollinearity. These models fail to generalize to real-world scenarios where water quality parameters are subject to stochastic environmental shifts. Deep Learning models appear to overcome this superficiality trap by leveraging their deep hierarchical structures to learn more abstract representations of the data. While a traditional Random Forest might show signs of overfitting on small datasets [26], DL models like the 10-layer Neural Network [32] achieve a high R^2 of 0.97 through genuine pattern recognition.

Furthermore, the RMSE values in DL are generally more realistic; whereas some ML models report suspiciously low errors, the DL cohort maintains a balance between high R^2 and meaningful RMSE loss (e.g., [39]), indicating a better calibration to the physical reality of the water body. Ultimately, the transition from shallow to deep architectures represents a shift from fitting the noise to modeling the system, making DL the preferred choice for reliable, long-term water quality monitoring. Ultimately, our research bridges these two worlds. By selecting a GRU-based architecture, we leverage the superior regression capabilities of Deep Learning reported in the literature while maintaining a lean computational profile that rivals the efficiency of traditional ML. This shift toward time-aware modeling is what makes a real-time smart city infrastructure truly proactive rather than just descriptive.

To summarize, traditional machine learning algorithms deliver reliable performance on static datasets, while Deep Learning is the definitive choice for the complex, high-stakes requirements of real-time urban water monitoring. The fundamental advantage of Deep Learning lies in its ability to treat water quality not as a series of isolated measurements, but as a living, sequential narrative. In this context, architectures such as the Long Short-Term Memory (LSTM) and the Gated Recurrent Unit (GRU) are the undisputed gold standard. These recurrent models move beyond simple pattern matching to achieve high-fidelity forecasting, capturing the intricate, non-linear dependencies that traditional models typically overlook, as evidenced by the work of [36] and [34], who demonstrated that recurrent architectures significantly outperform traditional regressors in complex aquatic ecosystems.

The superiority of the Deep Learning paradigm is further solidified by its capacity to scale with the massive volumes of data generated by modern smart city IoT grids. Rather than reaching a performance plateau, these models continue to improve as data density increases. Studies such as those by [37] and [40] illustrate that when deep learning is combined with advanced metaheuristic

optimization, it pushes the boundaries of predictive accuracy far beyond the reach of classical statistical methods. In our framework, the choice of a GRU-based Deep Learning architecture is a deliberate move toward a more AI driven and proactive environmental infrastructure, one that prioritizes the temporal depth and predictive momentum needed to ensure public safety in a dynamic, often unpredictable urban landscape.

3. Materials and Methods

This section introduces a distributed drinking water monitoring system called the Water Quality Identification IoT system (Water Quality Identification, Water-QI). The subsequent subsections detail the end-to-end high-level system architecture, the IoT device, implemented communication methods and application protocols, and the proposed deep learning models for localized Water Quality Index predictions. These models are designed for extensibility and edge predictability. Additionally, the evaluation metrics, dataset, proposed models, and training hyperparameters are described.

3.1. Proposed System Architecture

The proposed Water-QI platform is a cost-effective Internet of Things (IoT) system developed for real-time monitoring, visualization, and prediction of water quality, with a focus on the Water Quality Index (WQI). The system architecture integrates a field IoT telemetry device, cloud-based data transmission, a web-based data management and visualization environment, and a mobile application. This configuration enables continuous monitoring of water conditions, reducing dependence on periodic laboratory analysis. The platform automatically collects measurements from the IoT sensing node, transmits data to the cloud via existing Wi-Fi infrastructure, and displays both raw measurements and the calculated WQI through intuitive user interfaces. Beyond real-time monitoring, the system offers historical data inspection, statistical analysis, alert management, and configurable parameter weighting for WQI calculation.

At the cloud level, the platform utilizes the open-source ThingsBoard AS [49] to manage device communication, data visualization, and remote supervision. Data storage is performed using the Cassandra NoSQL database provided by ThingsBoard [50]. The communication workflow links the end node to the cloud through telemetry services, while the application server hosts the predictive component. Specifically, a deep-learning algorithm based on a variable-depth and cells of gated recurrent units as the recurrent neural network (GRU-RNN) infrastructure model, that operates on a cloud Virtual server that operates on top of a container similar to the thingsAI paradigm [51]. To estimate and forecast WQI trends from incoming sensor data streams. This edge-to-cloud architecture enables the system to monitor current water conditions as a weighted cumulative index, facilitating early warning and proactive decision-making in smart city and environmental monitoring contexts. Figure 1 presents the proposed Water-QI system architecture.

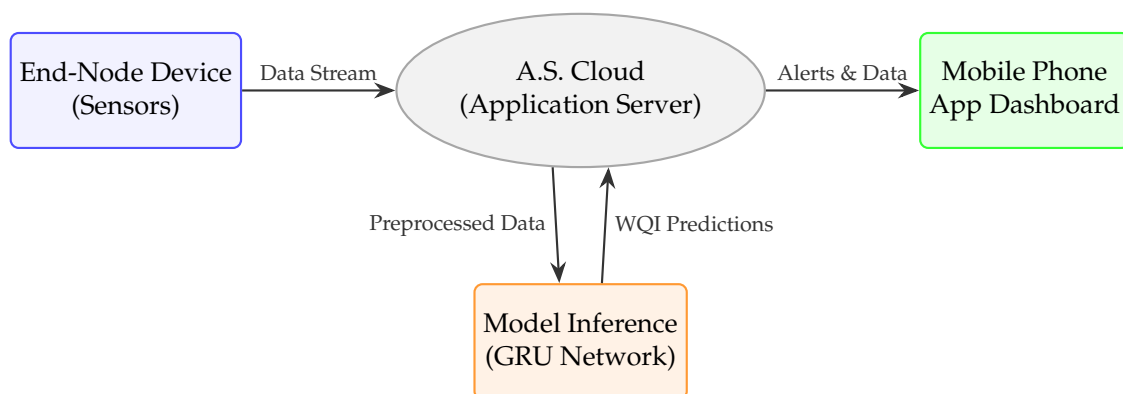


Figure 1. Proposed system architecture the Water-QI system.

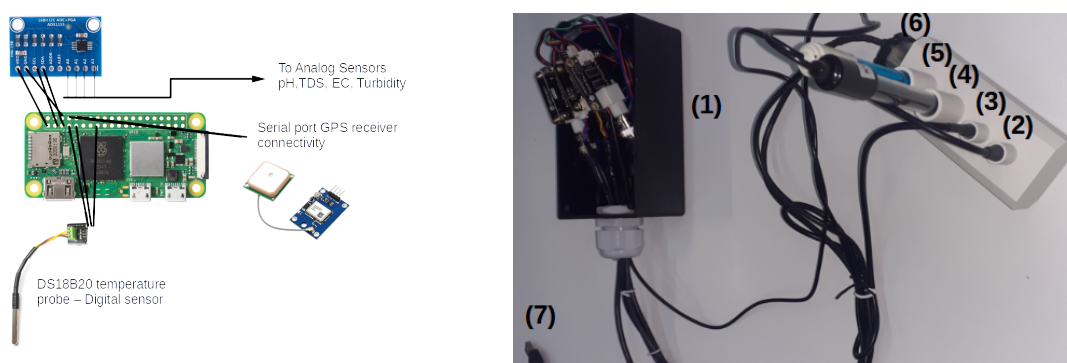
The Water-QI system also includes a mobile monitoring application developed in Flutter/Dart, designed to provide real-time supervision of the Water-QI IoT device via a cross-platform Android and iOS interface. In the uploaded project description, the mobile application is presented as a companion to the open-source ThingsBoard application server, which is responsible for telemetry collection, device supervision, alert exchange, and parameter configuration [52]. Within this Water-QI architecture, the mobile application allows users to inspect live sensor measurements, review water-quality history, and monitor the operational state of the field device through a portable interface, while the ThingsBoard backend manages data storage, dashboards, and server-side services.

Different protocols are utilized for the collection of data per IoT end node device of the Water-QI: 1) the MQTT beacon protocol, 2) the HTTP telemetry protocol, and 3) the HTTP request-back control protocol. The MQTT beacon protocol is a real-time protocol for sending beacons from an IoT device to the ThingsBoard A.S. broker. The beacon packer includes AES-128-encrypted information about the IoT device UUID, the device sensory measurement period T_m , the data transmission period to the A.S. T_p , the AS command update period for the device control protocol T_c , and the beacon location expressed in latitude and longitude coordinates. The HTTP over SSL telemetry protocol is using the method POST to submit a JSON encoded string of measurements to the Water-QI AS. Finally, the control protocol is a HTTP over SSL request-response protocol initiated periodically from the end node with the purpose to receive any updated information of probing intervals (periods) WQI weight parameters and lat,lng coordinated in the map, if the device does not include a GPS receiver to automatical location updates. The following section 3.2 provides additional information regarding the IoT device's sensors, measurements, and protocols, including functionality and interoperability.

3.2. End-Node IoT Device

A primary objective in the design of the Water-QI IoT end node device with edge capabilities adhered was to demonstrate that high-fidelity environmental monitoring can be achieved using budget-friendly, off-the-shelf components. The sensor suite was carefully curated to balance extreme affordability with the data reliability required for deep learning applications. For water temperature monitoring, we selected the DS18B20 digital stainless probe. This sensor provides a highly stable One-Wire digital output at a fraction of the cost of industrial-grade thermocouples or thermometers, making it an ideal candidate for large-scale, distributed urban deployments.

To maintain the Water-QI device IoT implementation with edge capabilities using a low-power ARM multi-core processor, while ensuring multi-parametric low-cost analysis, we integrated a series of analog sensors attached to the RPi zero 2W board via an I2C ADC board (ADS1115) as illustrated in Figure 2a. The actual implemented prototype includes the DFR0300 for Electrical Conductivity (EC) sensor (see Figure 2b.(4)), the SEN0244 for Total Dissolved Solids (TDS) sensor (see Figure 2(b).(1)), the Groove V1.0 sensor meter, for Turbidity measurements (see Figure 2(b).(6)), the SEN0161-V2 sensor for pH assessment (see Figure 2b.(5)) and the DS18B20 temperature sensor (see Figure 2b.(2)). The device is powered on using a 5V USB type-A connector (see Figure 2b.(7)) and uploads measurements to the cloud AS using Wi-Fi connectivity provided by the RPi Wi-Fi transponder. These probes were specifically chosen because they offer a cost-effective entry point for smart city infrastructure without sacrificing the precision needed to calculate an accurate Water Quality Index (WQI), since we are mainly focusing on measurement deviations rather than the actual accurate values. Even if monthly calibration is needed, by opting for these accessible analog modules over expensive laboratory-grade equipment and focusing on real-time acquisition of measurement changes, we ensure that the proposed system remains financially viable for municipalities with limited budgets, facilitating the transition toward pervasive and sustainable water management. Furthermore, the device's capability to include a GPS receiver (NEO 6M GPS module) connected to the RPi's serial port, if selected or statically assigned localization GPS coordinates, makes the Water-QI system's distributed approach fundamental for monitoring water-quality deviations at city-district levels. Figure 2a shows the actual device and its interface with the analog sensors mentioned, while Figure 2b.(1), illustrates the actual PoC implementation that was put to the test, without the use of a GPS receiver as shown in Figure 2(a).



(a) Connectivity diagram of the proposed sensing node based on the Raspberry Pi Zero 2 W. (b) Actual proof-of-concept (PoC) implementation of the sensing node.

Figure 2. IoT water-quality sensing node architecture and physical prototype. The left subfigure illustrates the connectivity diagram where the Raspberry Pi Zero 2 W communicates with the ADS1115 analog-to-digital converter through the I²C interface. The right subfigure shows the actual proof-of-concept implementation of the sensing node.

The probing Water-QI IoT node is built around the Raspberry Pi Zero 2W microprocessor, a compact single-board computer featuring a quad-core 64-bit ARM Cortex-A53 CPU at 1 GHz, 512 MB LPDDR2 RAM, integrated 2.4 GHz 802.11 b/g/n Wi-Fi, Bluetooth 4.2, mini-HDMI, micro-USB OTG, CSI camera connector, and a 40-pin GPIO header. The RPi zero 2W interfaces with an ADS1115 analog-to-digital converter over the I²C bus to acquire the outputs of the analog water-quality probes. The ADS1115 is connected to the Raspberry Pi through GPIO2 (SDA) and GPIO3 (SCL), while its four analog 16-bit input channels are assigned as follows: AIN0 to the DFRobot SEN0161-V2 pH sensor, AIN1 to the Grove Turbidity Sensor Meter V1.0, AIN2 to the DFRobot SEN0244 TDS sensor, and AIN3 to the DFRobot DFR0300 electrical conductivity sensor. The pH conditioning board operates at 3.3–5.5 V with an analog output of 0–3.0 V, the TDS board operates at 3.3–5.5 V with an analog output of 0–2.3 V, and the EC board operates at 3.0–5.0 V with an analog output of 0–3.4 V. The Grove turbidity sensor supports 3.3 V/5 V operation and provides both analog and digital output; in the proposed setup, it is configured in analog mode and connected directly to AIN1. In addition, water temperature is measured using a DS18B20 digital sensor connected to GPIO4 via the Raspberry Pi 1-Wire interface, with a 4.7 k Ω pull-up resistor between the data line and 3.3 V. All sensors share a common ground, and the DS18B20 temperature reading can also be used for compensation in conductivity and TDS-related calculations. Finally, the GPS receiver with an IPX uFL antenna included is connected via the GPIO 13-14 UART serial port of the RPi Zero 2W MPU.

The National Sanitation Foundation Water Quality Index (NSF-WQI) was proposed by Brown et al. [53], as a refinement of the earlier index-based water-quality assessment concept introduced by Horton [54]. Horton's contribution is generally recognized as the first formal WQI framework, designed to compress multiple physicochemical observations into a single interpretable score for surface-water assessment. Brown and colleagues extended this idea into the NSF-WQI by adopting a multiplicative model of weighting parameters and rating procedure, which made the index easier to apply and helped establish it as one of the most widely used WQI formulations for rivers and other surface waters. Like the Horton model, the NSF-WQI preserves the four basic components that characterize most classical water-quality indices: (i) parameter selection, namely the choice of the physical, chemical, and biological variables to be included; (ii) transformation of raw measurements into sub-indices, so that heterogeneous variables with different units can be mapped onto a common quality scale, (iii) parameter weighting, through which more influential variables receive greater importance in the final score, and (iv) aggregation of the weighted sub-indices into a single composite WQI value. These four elements remain the conceptual backbone of many later WQI variants [55,56].

The NSF-WQI has since been widely applied to evaluate surface-water quality across diverse environmental and management settings, including rivers affected by urban, agricultural, and industrial pressures. For example, Abrahao et al. [57] applied index-based analysis to assess a stream receiving industrial effluents, illustrating the practical use of WQI methods in pollution-impact studies. More broadly, the popularity of the NSF-WQI stems from its ability to reduce complex monitoring datasets into a concise, communicable measure of overall water status while retaining the essential logic of Horton's original formulation. The historical development of water quality indices, from Horton's original formulation to the NSF-WQI and later variants, has been extensively reviewed in [58].

For the real-time edge-device implementation, the weighting strategy was derived by adapting nominal literature-based WQI coefficients to the reduced parameter set available in the proposed sensing platform. Specifically, NSF-WQI-type formulations assign expert-defined raw weights to several physicochemical variables, including pH ($w_{pH}^{raw} = 0.12$), temperature ($w_{temp}^{raw} = 0.10$), turbidity ($w_{Tb}^{raw} = 0.08$), and total solids ($w_{TDS}^{raw} = 0.08$) (see [59], Table 2). These coefficients, however, do not constitute a complete weighting scheme for the present five-parameter system, since they originate from a broader multi-parameter index and sum to only 0.38 across the overlapping variables. Moreover, electrical conductivity is not explicitly included in the original NSF-WQI formulation and is therefore introduced here as an application-specific extension with raw coefficient $w_{EC}^{raw} = 0.08$. To obtain a valid edge-computable WQI, all raw coefficients are normalized based on Equation (2).

$$\hat{w}_i = \frac{w_i}{\sum_{j=1}^5 w_j} \quad (2)$$

where \hat{w}_i denotes the normalized weight of the i -th measured parameter, w_i is the corresponding raw weight before normalization, $i \in 1, \dots, 5$ indexes the five sensory attribute variables of the proposed Water-QI platform, and j is the summation index used to accumulate the raw weights of all five parameters in the denominator. Thus, the final weights satisfy $\sum_i \hat{w}_i = 1$, or equivalently 100%. In this way, the final percentages are not directly copied from the bibliography, but are obtained through proportional renormalization of literature-inspired coefficients over the subset of parameters actually measured at the IoT device level.

According to the Horton model, which is one of the earliest and most influential weighted-arithmetic WQI formulations, five WQI classes are commonly used: very good (91–100), good (71–90), poor (51–70), bad (31–50), and very bad (0–30) [54,55]. Furthermore, there is also the canonical NSF-WQI, which evolved from Horton-type formulations, that does not explicitly include electrical conductivity (EC) and uses total solids rather than total dissolved solids (TDS) among its standard variables [55,59]. Therefore, while the final WQI interpretation in this study follows an established five-class Horton-type scale for practical comparison, the individual sub-index equations for turbidity, pH, temperature, TDS, and EC are min-max tailored in the proposed Water-QI platform and measure attributive weights expressed as a quality score, where minimal values are better.

In depth, using the raw literature-inspired coefficients $w_{pH} = 0.12$, $w_{temp} = 0.10$, $w_{Tb} = 0.08$, $w_{TDS} = 0.08$, and the application-specific extension $w_{EC} = 0.08$, and based on Equation (2), the total raw weight becomes $\sum_{i=1}^5 w_i = 0.46$. The final normalized weights are then obtained as $\hat{w}_i = \frac{w_i}{0.46}$, which yields $\hat{w}_{pH} = 0.2609$, $\hat{w}_{temp} = 0.2174$, $\hat{w}_{Tb} = 0.1739$, $\hat{w}_{TDS} = 0.1739$, and $\hat{w}_{EC} = 0.1739$. The final weighting scheme for the Water-QI system becomes 26.09% for pH (set to 25%), 21.74% for temperature (set to 15% to denote the minimal significance of temperature over the other parameters, since it is rather constant for underground water pipelines and city installations), and 17.39% for turbidity, TDS, and EC, respectively (set to 20% to denote the importance over temperature), summing exactly to 100%. Table 3 summarizes the WQI classes as well as the mathematical formulation for the selected parameters for the WQI index calculation performed by the Water-QI IoT device. Table 3 presents the Horton/NSF-WQI attributes classification with respect to the Horton classification and the Water-QI score based mainly on min-max normalizations, the per measure normalization process, and the final WQI index value acting as a classification index value that is inversely proportional to Horton

classification values. Furthermore, the NSF-WQI is disregarded, and the TDS metric for Total Solids is used, along with Temperature and EC values, each with its min-max limitations, in accordance with the NSF-WQI classification.

Table 3. WQI interpretation classes and parameter sub-index formulas used in the proposed Water-QI edge-device implementation.

Category	WQI classification score in this paper	Interpretation / Formula
Excellent	0–30	Water quality is considered very good.
Good	31–50	Water quality is acceptable with minor concerns.
Poor	51–70	Water quality shows noticeable degradation.
Bad	71–90	Water quality is unsuitable without treatment.
Very bad	91–100	Water quality is severely degraded.
NSF-WQI attribute indices (value 1.0 is better)		
Turbidity	0–5 NTU	$Q_{Tb} = 100 \cdot \max\left(0, \min\left(1, \frac{5 - Tb}{5}\right)\right)$
pH	6.5–8.5	$Q_{pH} = 100 \cdot \max\left(0, 1 - \frac{ pH - 7.0 }{1.5}\right)$
Temp	0–40 °C	$Q_T = 100 \cdot \max\left(0, \min\left(1, \frac{40 - T}{40}\right)\right)$
TDS	0–500 mg/L	$Q_{TDS} = 100 \cdot \max\left(0, \min\left(1, \frac{500 - TDS}{500}\right)\right)$
EC	0–2000 μ S/cm	$Q_{EC} = 100 \cdot \max\left(0, \min\left(1, \frac{2000 - EC}{2000}\right)\right)$
Min–max normalized implementation used in this work (value 0.0 is better)		
Turbidity	0–5 NTU	$Tb^{\text{norm}} = \frac{Tb - Tb_{\min}}{Tb_{\max} - Tb_{\min}}$
pH	6.5–8.5	$pH^{\text{norm}} = \frac{ pH - 7.5 }{1.5}$
Temp	0–40 °C	$T^{\text{norm}} = \frac{T - T_{\min}}{T_{\max} - T_{\min}}$
TDS	0–500 mg/L	$TDS^{\text{norm}} = \frac{TDS - TDS_{\min}}{TDS_{\max} - TDS_{\min}}$
EC	0–2000 μ S/cm	$EC^{\text{norm}} = \frac{EC - EC_{\min}}{EC_{\max} - EC_{\min}}$
WQI index	$WQI = 100 \cdot \frac{1.5 T^{\text{norm}} + 2.0 TDS^{\text{norm}} + 2.0 EC^{\text{norm}} + 2.5 pH^{\text{norm}} + 2.0 Tb^{\text{norm}}}{10}$	

To ensure the Water-QI system reliability, specific operational thresholds were defined in accordance with WHO and Environmental Protection Agency guidelines [60,61]. In the proposed Water-QI implementation, the five monitored variables are combined through an application-specific weighted score rather than a canonical Horton or NSF-WQI formulation. With respect to drinking-water suitability, turbidity should ideally remain below 1 NTU and, in practice, not exceed 5 NTU. The pH value is commonly considered acceptable in the range 6.5–8.5, and total dissolved solids (TDS) are typically limited to 500 mg/L. By contrast, neither electrical conductivity (EC) nor temperature have a single universal WHO/EPA health-based drinking-water limit in the same sense, so in the present work they should be interpreted as operational surrogate variables whose influence is set by the custom weighting scheme ($w_{pH} = 2.5$, $w_T = 1.5$, and $w_{Tb} = w_{TDS} = w_{EC} = 2.0$). Consequently, the resulting WQI index score is best described as a custom 0–100 water-quality score derived from min–max

normalized measurements. In terms of class interpretation, the adopted bands as mentioned in Table 3 are closest in direction to the NSF-WQI classification limits, where higher values denote better quality.

A critical design choice in our architecture is the deployment of two separate physical sensors for EC and TDS. Although these parameters are theoretically correlated, where TDS (mg/L) is estimated as $k \times EC$ ($\mu\text{S}/\text{cm}$), with a typical conversion factor of $k \approx 0.98$, a single-sensor approach would introduce a static dependency that fails in complex environments. By utilizing distinct sensing elements, we overcome the limitations of pre-determined linear estimation. This redundancy allows the system to capture specific ionic fluctuations that a simple mathematical conversion might miss. For instance, one sensor may detect a spike in a specific mineral salt that alters the water's conductive profile differently than its total dissolved solids. This dual-sensing strategy prevents blind spots in the detection logic, ensuring that if one sensor reaches its sensitivity limit or encounters a specific type of ionic interference, the other remains as a fail-safe to maintain the integrity of the Water Quality Index (WQI) calculation. According to regulations, TDS values above 500.0 ppm are considered medium/fair and set as very high for drinking water. Moreover, TDS values above 1200ppm are considered unacceptable. In accordance, electroconductivity (EC) is considered unacceptable for drinking water if a value of 2000.0 $\mu\text{S}/\text{cm}$ and above is detected (see Table 3).

Temperature measurements for the Water-QI node are performed using a DS18B20 sensor. This is because thermal variations significantly affect ion mobility. Maintaining water temperature between 5°C and 15°C is considered ideal for palatability and the prevention of microbial regrowth, which becomes a significant risk at temperatures exceeding 25°C or with temperature variations of 10°C (penalty value of 100). The following section 3.3 describes the metrics used in the authors' experimentation.

3.3. Metrics Used

To evaluate the performance of our prediction models, we utilize standard regression metrics widely adopted in the literature for water quality forecasting. Specifically, our evaluation is based on the Root Mean Square Error (RMSE) and the Coefficient of Determination (R^2):

Root Mean Square Error (RMSE): Indicates the standard deviation of prediction errors. It is highly sensitive to large errors and provides interpretability in the same unit as the scaled target variable. It is defined according to Equation (3).

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (3)$$

where n denotes the total number of observations, y_i is the actual value of the target variable for the i -th observation, and \hat{y}_i is the corresponding predicted value produced by the model.

Coefficient of Determination (R^2): Measures the proportion of the variance in the dependent variable (WQI) that is predictable from the independent variables. A score closer to 1 indicates a perfect fit. It is defined according to Equation (4).

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (4)$$

where y_i is the actual value of the target variable for the i -th observation, \hat{y}_i is the predicted value for the i -th observation, \bar{y} is the mean of the observed target values, and n is the total number of observations.

3.4. Proposed Deep Learning Models for WQI Prediction

Following the extensive comparative analysis of existing literature, we designed a targeted experimental framework. While complex hybrid models are popular, they often require significant computational power, which contradicts the philosophy of low-cost IoT smart city deployments.

Instead, our approach focuses exclusively on Gated Recurrent Units (GRU). GRUs offer a streamlined alternative to LSTMs, requiring fewer computational resources and memory while maintaining excellent retention of temporal dependencies in time-series data.

To thoroughly evaluate the trade-off between predictive accuracy, temporal granularity, and computational cost, we developed and trained three distinct GRU architectures:

The Standard Hourly Model (Lightweight, periodic): Designed with practical, low-cost IoT deployment in mind. It consists of 2 GRU layers with 64 units each, processing the 24-step hourly input. This model is engineered to be computationally inexpensive, capable of running on edge devices, while capturing the general daily trend of the Water Quality Index (WQI).

The High-Capacity Minute Model (Heavyweight, near-real-time): Built to capture maximum detail, this model processes the full 1440-step input. It utilizes 2 GRU layers but significantly increases the network's capacity to 256 units per layer. It serves as our benchmark for maximum achievable accuracy, albeit at a higher computational cost.

The Deep GRU Model (Stacked alternative - Depthwise, near-real-time): To test the limits of network depth and investigate potential diminishing returns, we constructed an unusually deep 10-layer GRU architecture (64 units per layer), processing the 1440-step input. This experimental model acts as a stress test to determine if extremely deep recurrent networks justify their massive training times in the context of environmental monitoring.

To ensure training stability and prevent the models from simply memorizing the training data (overfitting), we applied rigorous regularization techniques across all three architectures. Batch Normalization was applied after every GRU layer to stabilize the learning process, followed immediately by a 20% Dropout rate to promote generalization. The final features are passed through a fully connected (Dense) layer and reshaped to output the exact forecasted sequence (either the next 24 hours or the next 1440 minutes) for the five key water parameters (Temperature, TDS, EC, pH, and Turbidity).

The models in the minute-resolution temporal scale have been classified according to their GRU items depth as:

Small models: Models of a single layer depth and their corresponding multi-layer deep derivatives of 64–128 GRU items per layer

Medium models: Models of at least 256–512 GRU items per layer and their corresponding stacked layer counterparts.

Large models: Models of at least 512 GRU items per layer, with significant representatives, the 1024 and 2048 GRU item models per layer.

3.5. Data Collection and Preprocessing Steps

Effective data preprocessing and feature engineering are crucial for maximizing model performance, especially when working with environmental data that often contains noise and inconsistencies [62], demonstrating the importance of feature selection in their application of a Gated Recurrent Unit (GRU) neural network for measurements predictions. Similarly, [6] addresses the non-stationarity and jitter inherent in environmental data through multi-step forecasting strategies and various train-test splits. In accordance, the research by [4] further emphasizes the value of preprocessing techniques—including normalization and feature selection in reducing computational overhead while enhancing predictive accuracy.

Following these directions, the authors use the OpenData available dataset of EYATH, for the city of Thessaloniki, Greece [63]. The dataset contains monthly measurements from 49 selected areas in and around the city of Thessaloniki from 2021–2025. This structured monthly-collected dataset has been temporally fuzzy-interpolated to provide minute-level measurements of Temperature, pH, EC, TDS, and turbidity. Therefore, the collected temporal sensory measurement data are partitioned into two distinct temporal resolutions to train their models accordingly:

1. High-Resolution (Minute-by-Minute): The input sequence consists of 1440 time steps, representing every single minute of a 24-hour period. This allows the model to observe micro-fluctuations and transient spikes in water quality.
2. Standard Resolution (Hourly intervals): The input sequence is condensed into 24 time steps, representing hourly averages. This significantly reduces the data's dimensionality, filtering out potential sensor noise.

The following sections 4, 5 present the authors' experimental results and discussion.

4. Experimental Scenarios

To evaluate the effectiveness of our proposed GRU architectures, we structured our experiments around the data temporal resolutions described previously. The training data annotation process involved formatting the sequential datasets to predict the designated forecast horizon (24 steps for hourly, 1440 steps for minute-level).

4.1. Model Training and Hyperparameters

Table 4 summarizes the hyperparameters for the training models' scenarios in both low- and high-resolution cases. The optimization setup uses the Adam optimizer with a learning rate of 0.001, RMSE, MAE as monitoring metrics, and R^2 at the testing dataset, batch size 16, and a maximum of 100 training epochs.

The GRU forecasting models are trained as a multivariate sequence-to-sequence predictor using five water-quality variables: Temperature, TDS, EC, pH, and Turbidity. The main architectural hyperparameters are one or multiple stacked recurrent layers ($L=1-10$), the layer width of GRU units, followed by a batch normalization layer and a dropout layer with a dropout rate of 0.2, after the recurrent block. Finally, a dense projection-flatten layer that maps the hidden representation to $24 \times 5 = 120$ neurons for the hourly resolution case and $1440 \times 5 = 7200$ neurons for the minute resolution case. Then, the output values layer follows, with the same number of neurons indicating the temporal prediction length (hourly or minute-graded).

From the periodic-low temporal resolution case, where the hourly measurement dataset is used, the periodic temporal coverage input depth is `SEQ_LEN=24`, meaning that each training sample contains 24 past hourly observations, corresponding to one full day of historical context. Similarly, the prediction horizon is set to `PRED_LEN=24`, so the network forecasts the next 24 hours on an hour-by-hour basis. Hence, the model learns a one-day to one-day mapping, using 24 past hours to predict the next 24 future hours of measurements.

From the near-real-time, high-temporal-resolution perspective, the model has a very large temporal depth. The input depth is `SEQ_LEN=1440`, meaning each training sample contains 1440 past time steps. Since the data are sampled at a minute resolution, this corresponds to one full day of historical context. The time window is also 1440 steps, because `PRED_LEN=1440`, so the network predicts the next 24 hours, minute-by-minute. Therefore, the model learns a one-day-to-one-day mapping: 1440 past minutes are used to forecast 1440 future minutes. In addition, the temporal sampling stride is 60, so neighboring training windows overlap heavily while advancing by one hour.

The preprocessing and training hyperparameters also play an important role. Before sequence generation, the high-temporal-resolution raw sensor data are smoothed with a 30-sample rolling window to reduce short-term noise. The dataset is then split chronologically, with 10% reserved for testing and 10% of the remaining training portion used for validation, preserving temporal order by setting shuffle to False. Training is further regulated by `ReduceLROnPlateau` with a factor of 0.5 and a patience of 2, and by early stopping with a patience of 8 and best-weight restoration. The following Sections 4.2 and 4.3 summarize the experimental results.

Table 4. Training hyperparameters of the GRU forecasting models

Hyperparameter	Value	Description
Historical depth window (SEQ_LEN)	1440 (minute) / 24 (hourly)	Number of past observations used as input. This corresponds to 1440 minutes (24 h) for minute-resolution data, or 24 hourly samples (24 h) for hourly-resolution data.
Prediction horizon (PRED_LEN)	1440 (minute) / 24 (hourly)	Number of future observations predicted by the model. This corresponds to forecasting the next 1440 minutes (24 h) for minute data, or the next 24 hourly steps (24 h) for hourly data.
Number of input features	5	Multivariate input composed of Temp, TDS, EC, pH, and Turbidity.
Number of GRU layers (L)	1	The recurrent architecture uses a single GRU layer.
GRU units/layer (U)	64 128 256 512 1024 2048	The number of GRUs/layer.
Batch normalization	Yes	Applied after the GRU layer to stabilize learning.
Dropout rate	0.2	Dropout applied after batch normalization for regularization.
Optimizer	Adam	Optimization algorithm used for training.
Learning rate	0.001	Initial learning rate of the Adam optimizer.
Epochs	100	Maximum number of training epochs.
Batch size	16	Number of samples per gradient update.
Dense output size	1440 \times 5 for minute resolution, 24 \times 5 for hour resolution	Final fully connected layer producing all future values before reshaping to (1440, 5), (24, 5).
Optimizer	Adam	Optimization algorithm used for training.

4.2. Scenario I: Low Temporal Resolution Data Experimentation

The authors trained three distinct GRU configurations: the standard small-scale model with 64 GRU units, the heavy model with 256 GRU units, and the deep, large model with multiple GRU layers, each containing 64 GRU units. All models have also been examined with different stacked layer configurations (2, 4, and 10) on an hourly-averaged dataset over 100 epochs. The hourly resolution is highly representative of typical smart city IoT deployments, particularly when data transmission and power consumption must be carefully balanced. The learning curves, which illustrate both training and validation RMSE, reveal significant insights into how network complexity affects environmental time-series forecasting.

As shown in Figure 3, the small GRU model with 64 units (standard GRU) achieved the best results for its size and the hourly dataset, converging quickly and yielding a validation RMSE of approximately 0.028. with a R^2 above 0.98. This result suggests that a lightweight recurrent architecture is sufficient to capture the dominant temporal patterns in the hourly water-quality data. Table 5 summarizes the validation RMSE and test R^2 values for all hourly GRU configurations.

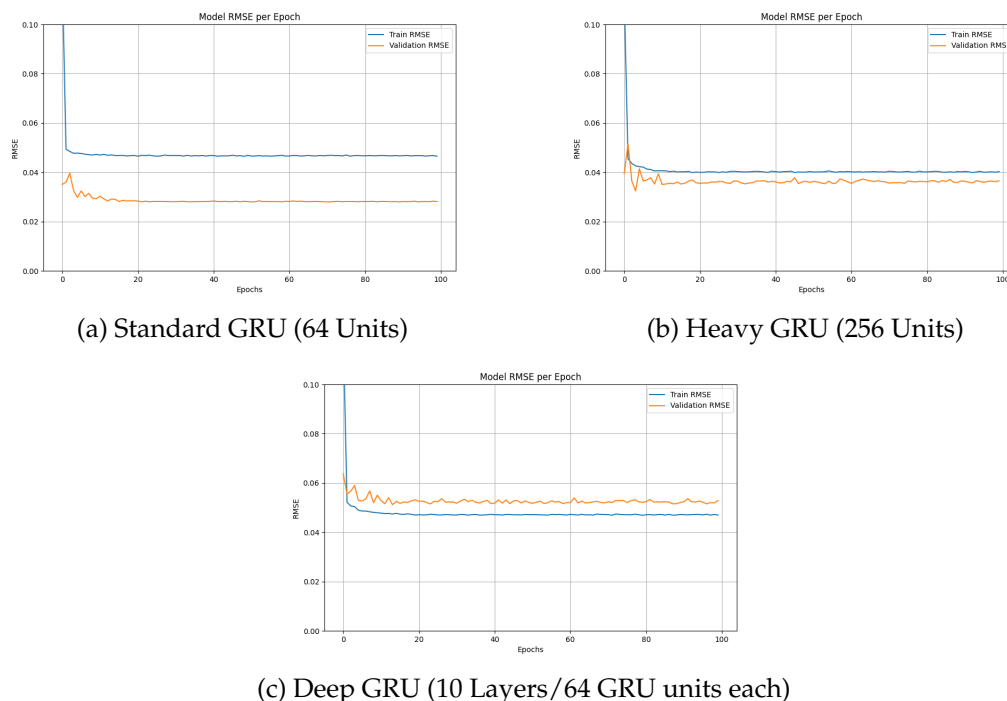


Figure 3. Training and Validation RMSE across 100 epochs for the three evaluated GRU architectures on hourly resolution data.

Table 5. Detailed performance evaluation for all GRU architectures using the hourly resolution dataset.

Model Architecture (Hourly)	Validation RMSE	Test R^2
Standard GRU (1 Layer - 64 units)	0.0281	0.9820
Heavy GRU (256 units)	0.0365	0.9796
Deep GRU (2 Layers - 64 units)	0.0389	0.9756
Deep GRU (4 Layers - 64 units)	0.0405	0.9541
Deep GRU (10 Layers - 64 units)	0.0529	0.9246

Interestingly, drastically increasing the network's size in the medium GRU model scale of 256 GRU units (heavy GRU) yielded worse results than the standard GRU model, raising the validation RMSE to roughly 0.0084 (0.84% - still less than 1% worse). While technically superior to the standard GRU, this poor accuracy can be explained by the lower resolution of the training dataset, with fewer short- and long-term characteristics that a less-dense GRU can easily capture. Increasing the number of units from 64 to 256 did not improve performance. On the contrary, the heavy GRU achieved a higher validation RMSE and a slightly lower test R^2 . This indicates that the additional model capacity did not translate into better generalization for the hourly dataset.

The most revealing finding came from the deep, large stackable GRU model. Despite its 10 layers of depth, the model struggled with diminishing returns and inherent instability, ultimately plateauing at a significantly higher validation RMSE of 0.053. This provides empirical evidence that

blindly adding depth to recurrent neural networks for standard environmental forecasting can be counterproductive, leading to optimization hurdles without improving generalization.

Beyond the error metrics, we evaluated the models' practical utility by simulating a 24-hour-ahead forecasting scenario. The predictions were converted back into their real-world values to calculate the final Water Quality Index (WQI), as illustrated in Figure 4.

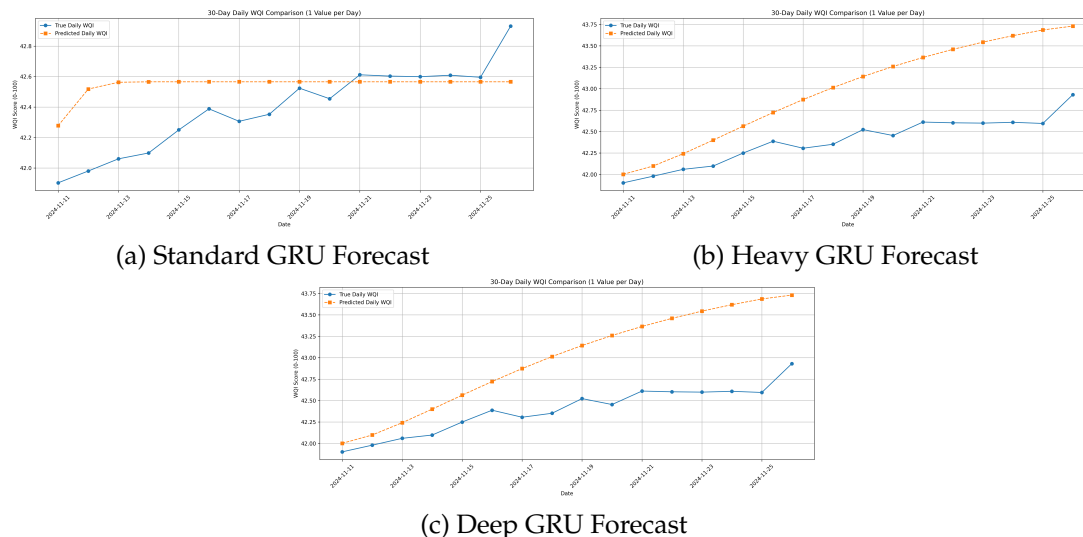


Figure 4. Next 24-hour WQI prediction using the hourly-resolution models.

Observing Figure 4, all three hourly-resolution models consistently classified the forecasted water quality within the *good* zone according to the characterization adopted in this work (WQI=31–50). In contrast to the previous interpretation, the predicted values do not fall in a very poor regime. Instead, they remain in a relatively narrow interval of approximately 42–46. The standard 64-unit GRU in Figure 4(a) provides the closest agreement with the true daily WQI series. Its predictions remain nearly flat around 42.3–42.6 and closely follow the observed mild upward trend. This behavior is physically reasonable. Daily averaged water-quality measurements usually exhibit substantial inertia and do not change abruptly unless there are major contamination events. By comparison, the heavier GRU model in Figure 4(b) shows a systematic positive drift. The predicted WQI rises from about 42.0 to 43.7, whereas the true series remains much more stable. The deep GRU model in Figure 4(c) amplifies this effect even further. It produces a stronger monotonic overestimation, reaching approximately 45.6 by the end of the forecasting horizon. Therefore, all three models preserve the same category-level interpretation: good water quality throughout the 30-day horizon. However, the standard GRU clearly offers the best practical trade-off between forecast stability, category consistency, and numerical fidelity to the observed daily WQI trajectory. This makes it the most suitable candidate for deployment on resource-constrained edge or end-node Water-QI devices. In such cases, reliable category-level monitoring and low computational overhead are more important than unnecessarily complex architectures. The following subsection 4.3 examines the three representative model categories using a minute resolution temporal dataset.

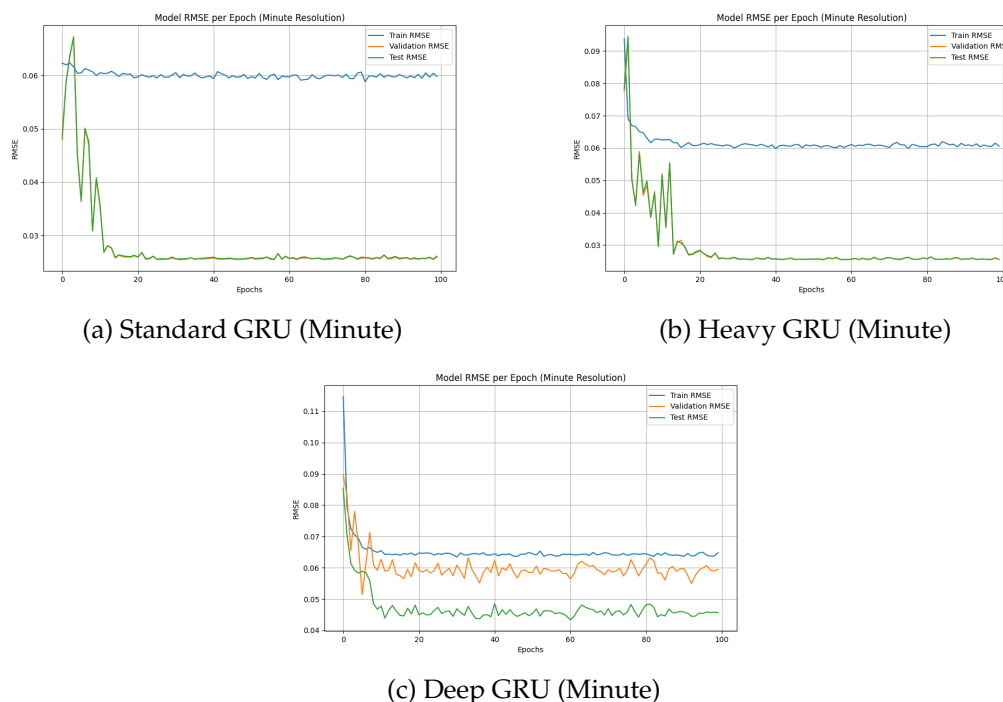
4.3. Scenario II: High Temporal Resolution Data Experimentation

The hourly models proved highly efficient for general trend monitoring; relying solely on averaged data might, in theory, obscure critical, short-lived anomalies. To investigate whether high-frequency sampling offers a strategic advantage, we trained the exact same three GRU architectures using minute-by-minute data (a massive sequence length of 1440 steps per sample). The most immediate observation from this experiment was the staggering computational toll. Transitioning from an hourly (24 steps) to a minute (1440 steps) resolution exponentially increased the processing load. Table 6 summarizes the results.

Table 6. Detailed performance evaluation for all GRU architectures using the minute resolution dataset.

Model Architecture	Layers	Validation RMSE	Test R^2
GRU (64 units)	1 (Standard GRU)	0.025981	0.984846
	2	0.027072	0.983401
	4	0.035431	0.974196
	10 (Deep GRU)	0.078124	0.849364
GRU (128 units)	1	0.025697	0.985331
	4	0.031230	0.976415
GRU (256 units)	1	0.025552	0.985445
	4	0.028994	0.937943
GRU (512 units)	1 (Heavy GRU)	0.025548	0.985448
	2	0.027008	0.976421
GRU (1024 units)	1	0.025608	0.985260
GRU (2048 units)	1	0.026411	0.985454

Looking at the RMSE error and R^2 , the heavy GRU model of 256 or 512 units (similar losses according to Table 6), specifically the GRU 512, achieved the lowest overall validation RMSE of approximately 0.025548. The standard GRU model of 64 units closely followed with an RMSE around 0.027. Just like in the hourly experiment, the deep GRU model struggled significantly, stabilizing at a much higher RMSE loss around 0.07, reaffirming that excessive depth hinders learning in this context. Furthermore, deeper models (GRU 1024, GRU 2048) performed similarly or slightly worse than the GRU-512 model. This indicates that, for the provided dataset, extending the GRU units beyond 512 does not yield better performance (less than 1% improvement in RMSE). Figure 5, presents the representative models (standard, heavy, deep), RMSE train, validation and evaluation curves over training epochs.

**Figure 5.** Training and Validation RMSE for the minute-resolution models across 100 epochs.

The minute-resolution experiment shows that shallow GRU architectures remain the most effective even under very high temporal granularity. As seen in Figure 5, both the standard GRU (1 layer, 64

units) and the wider shallow variants converge rapidly within the first few epochs and stabilize at very low error levels. The best overall validation RMSE was achieved by the heavy GRU (1 layer, 512 units) with 0.025548, followed almost identically by the 1-layer 256-unit model with 0.025552. Compared with the standard 1-layer 64 GRU units model (0.025981), these correspond to small RMSE reductions of 1.67% and 1.65%, respectively, nevertheless above 1%, indicating that increasing width still provides a marginal benefit.

In contrast, the deep GRU (10 layers, 64 units) performed substantially worse, yielding a validation RMSE of 0.078124, which is 200.70% higher than the standard model and 205.79% higher than the heavy model. A similar pattern is observed in the test R^2 values: the heavy and 256-unit shallow models provide small improvements over the standard architecture, whereas the deep model drops sharply to 0.849364. Overall, these results confirm that for minute-resolution sequences, widening a shallow GRU still offers minor gains, while excessive depth severely impairs convergence and generalization. The following section 5 provides a summary of the experimentation and explores the use of the examined best-case models and their performance, offering edge inference capabilities to the end node Water-QI device.

4.4. Scenario III: Edge Computation Performance of Minute Resolution Models

Using an ESP32 microcontroller as the central processing unit for on-device GRU inference, our preliminary experimentation showed that only relatively small recurrent models, approximately in the range of 10–32 GRU cells together with their associated parameters, can be loaded within the memory limits of a dual-core 32-bit ESP32 platform with 4–8 MB RAM. Under these constraints, the device can support only hourly-scale inference, typically with a temporal input window of 12–24 past hours to produce a forecast horizon of 10–24 future hours for a single measurement variable. Consequently, ESP32-class microcontrollers are considered insufficient for multivariate predictive inference with minute-resolution data and subsequent WQI estimation at the edge.

For this reason, the Raspberry Pi Zero 2W platform was selected for the proposed Water-QI edge prototype. This device provides a 64-bit quad-core ARM processor and 512 MB of RAM, enabling the deployment of more demanding minute-resolution GRU models. To examine a lower-bound embedded execution scenario, the experiments were conducted on this hardware under a 32-bit Raspberry Pi OS configuration, using a custom build of TensorFlow 2.4.0 [64] with Python 3.7. Table 7 summarizes the measured memory footprint and inference time for the examined GRU architectures.

Table 7. Inference performance of the examined GRU architectures on a quad-core 32-bit edge device for a 24-hour forecasting horizon. using the minute-level setup that predicts 1440×5 minute-resolution samples. Memory values correspond to the approximate FP32 footprint of the loaded model, while inference times are rough ARM CPU-only estimates.

Model	Loaded Model Memory (MB)	Minute resolution 24h (1440-point) Inference (s)
GRU-64	15.23	0.831
GRU-256	25.74	3.872
GRU-512	35.13	13.215
GRU-1024	61.13	49.647
GRU-2048	113.61	199.51
Stacked GRU (10 \times 64)	80.08	6.370

Comparing the inference-time measurements of Table 7 with the minute-resolution validation errors reported in Table 6, a clear speed–accuracy trade-off emerges for the single-layer models. The best numerical validation RMSE is achieved by the GRU-512 model (0.025548), but the GRU-256 model is only 0.0157% worse in RMSE 0.025552, while completing inference 70.70% faster (3.872 s versus 13.215 s). Likewise, the GRU-64 model is 93.71% faster than GRU-512, at the cost of only a 1.69% increase in RMSE. In contrast, increasing the model size beyond 512 units does not yield a meaningful

accuracy benefit: GRU-1024 is 275.69% slower than GRU-512, while its RMSE is 0.235% worse; GRU-2048 is 1409.72% slower, and its RMSE is 3.38% worse. Therefore, from an edge-computing perspective, the GRU-256 configuration provides the most favorable practical balance between predictive accuracy and execution speed, followed by GRU-512, which fine-grains accuracy while deliberately increasing speed, within the marginal context of minute-level inference. A similarly strong conclusion is obtained for deep stacked models. The 10-layer GRU with 64 units per layer requires 6.370 s for a 24-hour minute-resolution forecast, which is 666.55% slower than the single-layer GRU-64 model (0.831 s), while its validation RMSE increases from 0.025981 to 0.078124, corresponding to a 200.70% error increase. Hence, deeper stacking is disadvantageous not only in predictive quality but also in edge-execution efficiency. Moreover, for near-real-time, minute-scale deployment, a full 1440-point forecast should complete within 60 s to sustain timely rolling updates. Under this criterion, models whose inference time exceeds 60 s cannot provide near-real-time minute-level operation; therefore, the GRU-2048 model 199.51 s inference time is unsuitable for practical minute-scale edge inference, while GRU-1024 49.647 s is pretty close to the operational limit.

5. Discussion of the Results

To provide a clear comparative evaluation of the reported experiments, Table 8 summarizes the validation RMSE and test R^2 values achieved by representative GRU architectures across both temporal-resolution scenarios, Table 8 compares validation RMSE and test R^2 for GRU architectures in both temporal-resolution scenarios.

Table 8. Performance metrics of representative GRU architectures for WQI prediction.

Scenario (Resolution)	Model Architecture	Validation RMSE	Test R^2
Scenario I (Hourly)	Standard GRU (64 units)	0.0281	0.9820
	Heavy GRU (256 units)	0.0365	0.9796
	Deep GRU (10 Layers - 64 units)	0.0529	0.9246
Scenario II (Minute)	Standard GRU (64 units)	0.0259	0.9840
	Heavy GRU (256 units)	0.0255	0.985448
	Very heavy GRU (2048 units)	0.02641	0.985454
	Deep GRU (10 Layers - 64 units)	0.0781	0.8490

For the minute-resolution scenario, increasing model capacity from 64 to 256 GRU units yields a small but measurable improvement in validation accuracy: the validation RMSE decreases from 0.0259 to 0.0255 (a 1.54

Further increasing the number of units to 2048 does not improve RMSE. While the 2048-unit model attains the numerically highest test R^2 (0.985454), its advantage over the 256-unit model is negligible, and its validation RMSE is approximately 3.57% worse. This saturation effect suggests that larger single-layer GRU models offer no meaningful practical gains for this dataset.

The deep stacked GRU model performs substantially worse than the shallow minute-resolution models, with a validation RMSE of 0.0781 and test R^2 of 0.8490. These findings reinforce the conclusion that increasing layer depth is not beneficial under the examined conditions, and that shallow GRU architectures generalize more effectively than deeper stacked variants.

Minute-resolution shallow models achieve slightly better validation RMSE than their hourly counterparts. For example, the standard 64-unit GRU improves from 0.0281 in the hourly scenario to 0.0259 in the minute scenario, an RMSE reduction of approximately 7.83%. Likewise, the 256-unit GRU improves from 0.0365 to 0.0255, reducing RMSE by approximately 30.14%. In both settings, shallow architectures outperform deeper stacked variants.

A direct comparison between the best hourly and minute-resolution models further highlights the benefit of finer temporal granularity. The best hourly model, namely the standard GRU with 64 units, achieves a validation RMSE of 0.0281 and a test R^2 of 0.9820. In contrast, the best minute-resolution

model, namely the single-layer GRU with 256 units, achieves a lower validation RMSE of 0.0255 and a higher test R^2 of 0.985448. This corresponds to an absolute RMSE reduction of 0.0026, or approximately 9.25%, together with an absolute increase of 0.003448 in test R^2 . These results suggest that the minute-resolution setting offers a modest but consistent predictive advantage over the best-performing hourly configuration.

Among the minute-resolution models reported in Table 8, the single-layer 256-unit GRU provides the best trade-off between predictive accuracy and model complexity. Although the 2048-unit model yields a marginally higher test R^2 , its validation RMSE is worse, and its practical advantage is negligible. Therefore, the final results support the use of a shallow single-layer GRU architecture and indicate that performance saturates beyond the moderate-capacity regime, while deeper stacking consistently degrades prediction accuracy.

In the minute-resolution data scenario, the experiments in section 4.3 show a clear, consistent effect of network layer depth when the number of GRU cells is small. In the 64-cell configurations, increasing the number of layers from 1 to 2, 4, and 10 leads to a progressive deterioration in predictive accuracy, as evidenced by the increase in test RMSE from 0.026 to 0.027, 0.034, and 0.082, respectively, together with the corresponding decrease in R^2 from 0.984 to 0.983, 0.974, and 0.84. Therefore, deeper stacking is not beneficial for this dataset and instead introduces substantial performance degradation.

For medium-sized (heavy) single-layer models, the experimental results indicate a gradual improvement in predictive accuracy as the number of GRU cells increases from 128 to 256 and 512, while the test R^2 values remain very high in all three cases. However, these gains are extremely small, especially between the 256-cell and 512-cell models, where the relative RMSE improvement using the 512-cell model's minute dataset is only about 1%. This suggests that increasing the number of cells beyond 256 yields only marginal benefit in this performance region. The best trade-off for this dataset is achieved by a single-layer GRU model with 512 cells, or equivalently by models in the same saturation region, since their predictive differences are minimal. Although the 2048-cell model yields the lowest numerical test RMSE, its advantage over the 512-cell model is too small to justify the four-fold increase in GRU cells. Therefore, the final results support the use of a shallow single-layer architecture and indicate that performance improvement follows a saturation pattern with diminishing returns, while deeper stacking consistently degrades prediction accuracy under the examined experimental conditions.

6. Conclusions

In this study, we investigated the integration of low-cost IoT sensing with GRU-based deep learning models for near-real-time and periodic water-quality assessment in smart-city environments. The proposed Water-QI platform combines affordable hardware, cloud-supported telemetry, and predictive analytics to estimate water-quality behavior using five measured parameters: temperature, TDS, EC, pH, and turbidity. The results confirm that reliable forecasting can be achieved without resorting to excessively complex architectures, which is especially important for practical deployment in budget-constrained urban infrastructures.

The experimental evaluation across hourly and minute-resolution scenarios showed that shallow GRU models consistently outperform deeper stacked alternatives. In the hourly case, the single-layer 64-unit GRU achieved the best overall performance, with a validation RMSE of 0.0281 and a test R^2 of 0.9820, making it the most suitable solution for low-cost and computationally efficient periodic monitoring. In the minute-resolution case, wider, shallower models provided slightly better predictive accuracy, with the 512-unit GRU achieving the lowest validation RMSE and the 256-unit GRU delivering nearly identical performance at substantially faster inference. These findings indicate that increasing model width yields small gains at very fine temporal granularity, whereas increasing recurrent depth leads to clear degradation in both convergence behavior and generalization.

From a practical edge-computing perspective, the results highlight a clear trade-off between predictive performance and execution cost. Although the 512-unit model achieved the best numerical

validation accuracy, the 256-unit model emerged as the most balanced configuration for minute-level forecasting on embedded ARM-based hardware. In contrast, very large or deeply stacked GRU models introduced substantial computational overhead without providing meaningful predictive benefit. Therefore, the experimental evidence supports deploying shallow GRU architectures as the most effective design choice for scalable and resource-aware real-time water-quality monitoring systems.

This study has several limitations that we should acknowledge. First, we derived the dataset from monthly open data records and temporally interpolated them to produce hourly and minute-level sequences; although this preprocessing enabled controlled forecasting experiments, the resulting high-resolution series do not fully replicate the behavior of truly continuous field measurements. Second, we focused our experiments on a reduced set of five physicochemical parameters in a single geographical context, which may limit the direct generalizability of the findings to other water networks or hydro-environmental conditions. Third, the proposed forecasting framework primarily models normal temporal evolution and does not explicitly address rare contamination incidents, abrupt anomalies, or sensor failures. Finally, although we evaluated edge inference on representative embedded hardware, we have limited long-term field validation under real operating conditions, including sensor drift, calibration degradation, communication instability, and environmental interference.

Future research will focus on extending the proposed Water-QI framework toward real multi-node spatial-temporal deployments across broader urban water networks. A first priority is the collection of genuine high-frequency sensor data from distributed IoT nodes to validate the models under fully realistic operating conditions and reduce reliance on interpolated sequences. In addition, future work will investigate hybrid and graph-based learning approaches for jointly modeling temporal evolution and spatial dependencies among sensing locations. Further directions include incorporating anomaly-detection mechanisms for sudden contamination events, uncertainty-aware prediction, adaptive calibration and drift compensation strategies, and online or federated learning schemes that enable models to continuously improve while maintaining low communication overhead. These extensions will strengthen the robustness, transferability, and operational value of the Water-QI platform for smart-city water management.

Finally, this work demonstrates that low-cost IoT sensing, combined with carefully selected shallow GRU models, can provide accurate, computationally feasible water-quality forecasting. The study shows that practical predictive performance is achieved not by maximizing architectural complexity, but by balancing temporal resolution, model capacity, and deployment constraints. In this sense, the proposed Water-QI framework offers a realistic pathway toward scalable, intelligent, and proactive water-quality monitoring in smart-city environments.

Author Contributions: Conceptualization, S.K.; methodology, S.K., G.K.; software, S.K. and C.T.; validation, C.T. and S.V.; formal analysis, S.K. and G.K.; investigation, C.T.; resources, S.K. and C.T.; data curation, S.K. and C.T.; writing—original draft preparation, C.T.; review and editing, S.K., S.V. and G.K.; visualization, C.T.; supervision, S.K.; project administration, S.K. and C.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ADC	Analog-to-Digital Converter
ADASYN	Adaptive Synthetic Sampling
ANN	Artificial Neural Network
ARIMA	Autoregressive Integrated Moving Average
ARM	Advanced RISC Machine
AS	Application Server
AUC	Area Under the Curve
CNN	Convolutional Neural Network
CPU	Central Processing Unit
DL	Deep Learning
GPIO	General-Purpose Input/Output
GRU	Gated Recurrent Unit
HTTP	Hypertext Transfer Protocol
I2C	Inter-Integrated Circuit
IoT	Internet of Things
JSON	JavaScript Object Notation
LSTM	Long Short-Term Memory
ML	Machine Learning
MLP	Multi-Layer Perceptron
MQTT	Message Queuing Telemetry Transport
NN	Neural Network
NSF_WQI	National Sanitation Foundation Water Quality Index
RBF	Radial Basis Function
ROC	Receiver Operating Characteristic
SCINet	Sample Convolution and Interaction Network
SMOTE	Synthetic Minority Over-sampling Technique
SSL	Secure Sockets Layer
WQS	Water Quality System
XGBOOST	eXtreme Gradient Boosting

References

1. Bamini, A.; Jengan, C.; Agarwal, S.; Kim, H.; Stephan, P.; Stephan, T. IoT-Based Automatic Water Quality Monitoring System with Optimized Neural Network. *KSII Transactions on Internet and Information Systems* **2024**, *18*, 46–63. <https://doi.org/10.3837/tiis.2024.01.004>.
2. Kyriatsakas, G. Exploring Machine Learning Applications for Improving Drinking Water Quality. Ph.d. dissertation, The University of Sheffield, Cambridge, MA, USA, 2021. Available at: <https://etheses.whiterose.ac.uk/id/eprint/30179/>, [Online; accessed Mar 2024].
3. El Bilali, A.; Taleb, A.; Brouziyne, Y. Groundwater Quality Forecasting Using Machine Learning Algorithms for Irrigation Purposes. *Agricultural Water Management* **2021**, *245*, 106625. <https://doi.org/10.1016/j.agwat.2020.106625>.
4. Ahmed, U.; Mumtaz, R.; Anwar, H.; Shah, A.A.; Irfan, R.; García-Nieto, J. Efficient Water Quality Prediction Using Supervised Machine Learning. *Water* **2019**, *11*, 2210. <https://doi.org/10.3390/w11112210>.
5. Lowe, M.; Qin, R.; Mao, X. A Review on Machine Learning, Artificial Intelligence, and Smart Technology in Water Treatment and Monitoring. *Water* **2022**, *14*, 1384. <https://doi.org/10.3390/w14091384>.
6. Nong, X.; He, Y.; Chen, L.; Wei, J. Machine Learning-Based Evolution of Water Quality Prediction Model: An Integrated Robust Framework for Comparative Application on Periodic Return and Jitter Data. *Environmental Pollution* **2025**, *369*, 125834. <https://doi.org/10.1016/j.envpol.2025.125834>.
7. Garzón, A.; Kapelan, Z.; Langeveld, J.; Taormina, R. Machine Learning-Based Surrogate Modeling for Urban Water Networks: Review and Future Research Directions. *Water Resources Research* **2022**, *58*, e2021WR031808. <https://doi.org/10.1029/2021WR031808>.

8. Boccadoro, P.; Daniele, V.; Di Gennaro, P.; Lofù, D.; Tedeschi, P. Water Quality Prediction on a Sigfox-compliant IoT Device: The Road Ahead of WaterS. *Ad Hoc Networks* **2022**, *126*, 102749. <https://doi.org/10.1016/j.adhoc.2021.102749>.
9. Zhu, M.; Wang, J.; Yang, X.; Zhang, Y.; Zhang, L.; Ren, H.; Wu, B.; Ye, L. A Review of the Application of Machine Learning in Water Quality Evaluation. *Eco-Environment & Health* **2022**, *1*, 107–116. <https://doi.org/10.1016/j.eehl.2022.06.001>.
10. Hmoud Al-Adhaileh, M.; Waselallah Alsaade, F. Modelling and Prediction of Water Quality by Using Artificial Intelligence. *Sustainability* **2021**, *13*, 4259. <https://doi.org/10.3390/su13084259>.
11. Onyutha, C. Multiple Statistical Model Ensemble Predictions of Residual Chlorine in Drinking Water: Applications of Various Deep Learning and Machine Learning Algorithms. *Journal of Environmental and Public Health* **2022**, *2022*, 7104752. <https://doi.org/10.1155/2022/7104752>.
12. Sharaan, M.; Elshemy, M.M.; Fujii, M.; Ibrahim, M.G.; Nada, A.M. Water Quality Prediction and Classification for Drinking Water from Seawater Desalination Plants Using Machine Learning Algorithms, 2024. <https://doi.org/https://doi.org/10.2139/ssrn.4999808>.
13. Khan, P.F.; Zaheen, S.Z.; Sunder, D.P.S.; Shirisha, M.K.; Kotoju, D.R.; Ayyappa, R.M.K. Water Quality Prediction and Classification Using Machine Learning. *International Journal of Research Publication and Reviews* **2025**, *6*, 8425–8435. Available at: <https://ijrpr.com/uploads/V6ISSUE5/IJRPR45788.pdf>.
14. Garcia, J.; Heo, J.; Kim, C. Machine Learning Algorithms for Water Quality Management Using Total Dissolved Solids (TDS) Data Analysis. *Water* **2024**, *16*, 2639. <https://doi.org/10.3390/w16182639>.
15. Patil, S.V.; Wankhade, N.R.; Bagal, S.B.; Patel, M.T. Water Quality Analysis and Prediction Using Machine Learning. *Journal of Information Systems Engineering and Management* **2025**, *10*, 1069–1073. <https://doi.org/10.52783/jisem.v10i39s.7429>.
16. Ding, F.; Hao, S.; Zhang, W.; Jiang, M.; Chen, L.; Yuan, H.; Wang, N.; Li, W.; Xie, X. Using Multiple Machine Learning Algorithms to Optimize the Water Quality Index Model and Their Applicability. *Ecological Indicators* **2025**, *172*, 113299. <https://doi.org/10.1016/j.ecolind.2025.113299>.
17. Iyer, S.; Kaushik, S.; Nandal, P. Water Quality Prediction Using Machine Learning. *MR International Journal of Engineering and Technology* **2023**, *10*, 60–62. <https://doi.org/10.58864/mrijet.2023.10.1.8>.
18. Padmaja, P.; Sai, C.S.D.; Teja, V.K.; Ragav, A.P.; Babji, P. Water Quality Prediction Using Machine Learning Algorithms. *Journal of Emerging Technologies and Innovative Research* **2023**, *10*, c711–c721. Available at: <https://www.jetir.org/papers/JETIR2304287.pdf>.
19. Walczak, N.; Walczak, Z. Assessing the Feasibility of Using Machine Learning Algorithms to Determine Reservoir Water Quality Based on a Reduced Set of Predictors. *Ecological Indicators* **2025**, *175*, 113556. <https://doi.org/10.1016/j.ecolind.2025.113556>.
20. Karthick, K.; Krishnan, S.; Manikandan, R. Water Quality Prediction: A Data-Driven Approach Exploiting Advanced Machine Learning Algorithms with Data Augmentation. *Journal of Water and Climate Change* **2024**, *15*, 431–452. <https://doi.org/10.2166/wcc.2023.540>.
21. Shams, M.Y.; Elshewey, A.M.; El-kenawy, E.S.M.; Ibrahim, A.; Talaat, F.M.; Tarek, Z. Water quality prediction using machine learning models based on grid search method. *Multimedia Tools and Applications* **2024**, *83*, 35307–35334. <https://doi.org/10.1007/s11042-023-16737-4>.
22. Prabu, P.; Alluhaidan, A.S.; Aziz, R.; Basheer, S. Comparative analysis of machine learning models for detecting water quality anomalies in treatment plants. *Scientific Reports* **2025**, *15*, 30453. <https://doi.org/10.1038/s41598-025-15517-4>.
23. Najah Ahmed, A.; Binti Othman, F.; Abdulmohsin Afan, H.; Khaleel Ibrahim, R.; Ming Fai, C.; Shabbir Hosain, M.; Ehteram, M.; Elshafie, A. Machine Learning Methods for Better Water Quality Prediction. *Journal of Hydrology* **2019**, *578*, 124084. <https://doi.org/10.1016/j.jhydrol.2019.124084>.
24. Lu, H.; Ma, X. Hybrid Decision Tree-Based Machine Learning Models for Short-Term Water Quality Prediction. *Chemosphere* **2020**, *249*, 126169. <https://doi.org/10.1016/j.chemosphere.2020.126169>.
25. Xu, T.; Coco, G.; Neale, M. A Predictive Model of Recreational Water Quality Based on Adaptive Synthetic Sampling Algorithms and Machine Learning. *Water Research* **2020**, *177*, 115788. <https://doi.org/10.1016/j.watres.2020.115788>.
26. Lokman, A.; Ismail, W.Z.W.; Aziz, N.A.A. A Review of Water Quality Forecasting and Classification Using Machine Learning Models and Statistical Analysis. *Water* **2025**, *17*, 2243. <https://doi.org/10.3390/w17152243>.
27. Chen, J.; Wei, X.; Liu, Y.; Zhao, C.; Liu, Z.; Bao, Z. Deep Learning for Water Quality Prediction—A Case Study of the Huangyang Reservoir. *Applied Sciences* **2024**, *14*, 8755. <https://doi.org/10.3390/app14198755>.

28. Yan, X.; Zhang, T.; Du, W.; Meng, Q.; Xu, X.; Zhao, X. A Comprehensive Review of Machine Learning for Water Quality Prediction over the Past Five Years. *Journal of Marine Science and Engineering* **2024**, *12*, 159. <https://doi.org/10.3390/jmse12010159>.
29. Islam, N.; Irshad, K. Artificial Ecosystem Optimization with Deep Learning Enabled Water Quality Prediction and Classification Model. *Chemosphere* **2022**, *309*, 136615. <https://doi.org/10.1016/j.chemosphere.2022.136615>.
30. Wang, X.; Li, Y.; Qiao, Q.; Tavares, A.; Liang, Y. Water Quality Prediction Based on Machine Learning and Comprehensive Weighting Methods. *Entropy* **2023**, *25*, 1186. <https://doi.org/10.3390/e25081186>.
31. Prasad, D.V.V.; Venkataramana, L.Y.; Kumar, P.S.; Prasannamedha, G.; Harshana, S.; Srividya, S.J.; Harrinei, K.; Indraganti, S. Analysis and Prediction of Water Quality Using Deep Learning and Auto Deep Learning Techniques. *Science of The Total Environment* **2022**, *821*, 153311. <https://doi.org/10.1016/j.scitotenv.2022.153311>.
32. Rizal, N.N.M.; Hayder, G.; Yusof, K.A. Water Quality Predictive Analytics Using an Artificial Neural Network with a Graphical User Interface. *Water* **2022**, *14*, 1221. <https://doi.org/10.3390/w14081221>.
33. Chen, H.; Yang, J.; Fu, X.; Zheng, Q.; Song, X.; Fu, Z.; Wang, J.; Liang, Y.; Yin, H.; Liu, Z.; et al. Water Quality Prediction Based on LSTM and Attention Mechanism: A Case Study of the Burnett River, Australia. *Sustainability* **2022**, *14*, 13231. <https://doi.org/10.3390/su142013231>.
34. Rahul Gandh, D.; Rasheed Abdul Haq, K.P.; Harigovindan, V.P.; Bhide, A. LSTM and GRU based Accurate Water Quality Prediction for Smart Aquaculture. *Journal of Physics: Conference Series* **2023**, *2466*, 012027. Publisher: IOP Publishing, <https://doi.org/10.1088/1742-6596/2466/1/012027>.
35. Cai, H.; Zhang, C.; Xu, J.; Wang, F.; Xiao, L.; Huang, S.; Zhang, Y. Water Quality Prediction Based on the KF-LSTM Encoder-Decoder Network: A Case Study with Missing Data Collection. *Water* **2023**, *15*, 2542. <https://doi.org/10.3390/w15142542>.
36. Eze, E.; Kirby, S.; Attridge, J.; Ajmal, T. Aquaculture 4.0: Hybrid Neural Network Multivariate Water Quality Parameters Forecasting Model. *Scientific Reports* **2023**, *13*, 16129. <https://doi.org/10.1038/s41598-023-41602-7>.
37. Sathya Preiya, V.M.; Subramanian, P.; Soniya, M.; Pugalenth, R.; M, S.P.V. Water Quality Index Prediction and Classification Using Hyperparameter Tuned Deep Learning Approach. *Global NEST Journal* **2024**, *26*, 1–8. <https://doi.org/10.30955/gnj.005821>.
38. Jaffar, A.; Thamrin, N.M.; Ali, M.S.A.M.; Misnan, M.F.; Yassin, A.I.M. Water Quality Prediction Using LSTM-RNN: A Review. *Journal of Sustainability Science and Management* **2022**, *17*, 204–225. <https://doi.org/10.46754/jssm.2022.07.015>.
39. Aldhyani, T.H.H.; Al-Yaari, M.; Alkahtani, H.; Maashi, M. Water Quality Prediction Using Artificial Intelligence Algorithms. *Applied Bionics and Biomechanics* **2020**, *2020*, 6659314. <https://doi.org/10.1155/2020/6659314>.
40. Perumal, B.; Rajarethinam, N.; Velusamy, A.D.; Sundramurthy, V.P. Water Quality Prediction Based on Hybrid Deep Learning Algorithm. *Advances in Civil Engineering* **2023**, *2023*, 6644681. <https://doi.org/10.1155/2023/6644681>.
41. Im, Y.; Song, G.; Lee, J.; Cho, M. Deep Learning Methods for Predicting Tap-Water Quality Time Series in South Korea. *Water* **2022**, *14*, 3766. <https://doi.org/10.3390/w14223766>.
42. Nagalakshmi, P.; Kumar, P.G. Water Quality Prediction Using Machine Learning Technique. *International Journal of Scientific Research in Engineering and Management (IJSREM)* **2024**, *8*, 1–9. <https://doi.org/10.55041/IJSREM36721>.
43. Elmotawakkili, A.; Enneya, N.; Bhagat, S.K.; Ouda, M.M.; Kumar, V. Advanced Machine Learning Models for Robust Prediction of Water Quality Index and Classification. *Journal of Hydroinformatics* **2025**, *27*, 299–319. <https://doi.org/10.2166/hydro.2025.290>.
44. Liu, M.; Zeng, A.; Chen, M.; Xu, Z.; Lai, Q.; Ma, L.; Xu, Q. SCINet: Time Series Modeling and Forecasting with Sample Convolution and Interaction, 2022, [arXiv:cs.LG/2106.09305].
45. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Computation* **1997**, *9*, 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>.
46. Kontogiannis, S.; Gkamas, T.; Pikridas, C. Deep Learning Stranded Neural Network Model for the Detection of Sensory Triggered Events. *Algorithms* **2023**, *16*. <https://doi.org/10.3390/a16040202>.
47. Gao, S.; Huang, Y.; Zhang, S.; Han, J.; Wang, G.; Zhang, M.; Lin, Q. Short-term runoff prediction with GRU and LSTM networks without requiring time step optimization during sample generation. *Journal of Hydrology* **2020**, *589*, 125188. <https://doi.org/10.1016/j.jhydrol.2020.125188>.

48. Tornyeviadzi, H.M.; Seidu, R. Leakage detection in water distribution networks via 1D CNN deep autoencoder for multivariate SCADA data. *Engineering Applications of Artificial Intelligence* **2023**, *122*, 106062. <https://doi.org/https://doi.org/10.1016/j.engappai.2023.106062>.
49. ThingsBoard. ThingsBoard Open-source IoT Platform. <https://thingsboard.io/>, 2019. [Online; Accessed 10 November 2021].
50. Apache Foundation. Cassandra, Open Source NoSQL Database. <https://cassandra.apache.org/>, 2015. [Online; Accessed 1st August 2021].
51. Kontogiannis, S.; Koundouras, S.; Pikridas, C. Proposed Fuzzy-Stranded-Neural Network Model That Utilizes IoT Plant-Level Sensory Monitoring and Distributed Services for the Early Detection of Downy Mildew in Viticulture. *Computers* **2024**, *13*. <https://doi.org/10.3390/computers13030063>.
52. ThingsBoard. ThingsBoard Mobile Application. https://github.com/thingsboard/flutter_thingsboard_app, 2024. [Online; Accessed 20 September 2025].
53. Brown, R.M.; McClelland, N.I.; Deininger, R.A.; Tozer, R.G. A Water Quality Index—Crashing the Psychological Barrier. *Water and Sewage Works* **1970**, *117*, 339–343. https://doi.org/10.1007/978-1-4684-1698-5_15.
54. Horton, R.K. An Index Number System for Rating Water Quality. *Journal of the Water Pollution Control Federation* **1965**, *37*, 300–306.
55. Uddin, M.G.; Nash, S.; Olbert, A.I. A review of water quality index models and their use for assessing surface water quality. *Ecological Indicators* **2021**, *122*, 107218. <https://doi.org/10.1016/j.ecolind.2020.107218>.
56. Patel, D.D.; Mehta, D.J.; Azamathulla, H.M.; Shaikh, M.M.; Jha, S.; Rathnayake, U. Application of the Weighted Arithmetic Water Quality Index in Assessing Groundwater Quality: A Case Study of the South Gujarat Region. *Water* **2023**, *15*, 3512. <https://doi.org/10.3390/w15193512>.
57. Abrahão, R.; Carvalho, M.; da Silva Jr., W.R.; Machado, T.T.V.; Gadelha, C.L.M.; Hernandez, M.I.M. Use of Index Analysis to Evaluate the Water Quality of a Stream Receiving Industrial Effluents. *Water SA* **2007**, *33*, 459–466. <https://doi.org/10.4314/wsa.v33i4.52940>.
58. Lumb, A.; Sharma, T.C.; Bibeault, J.F. A Review of Genesis and Evolution of Water Quality Index (WQI) and Some Future Directions. *Water Quality, Exposure and Health* **2011**, *3*, 11–24. <https://doi.org/10.1007/s12403-011-0040-0>.
59. Garcia, C.A.B.; Silva, I.S.; Mendonça, M.C.S.; Garcia, H.L. Evaluation of Water Quality Indices: Use, Evolution and Future Perspectives. In *Advances in Environmental Monitoring and Assessment*; Sarvajayakesavalu, S., Ed.; IntechOpen: London, 2018; chapter 2. <https://doi.org/10.5772/intechopen.79408>.
60. World Health Organization. Guidelines for Drinking-water Quality: Fourth Edition Incorporating the First and Second Addenda, 2022. Available at: <https://www.who.int/publications/i/item/9789240045064>, [Online; accessed 15 November 2025].
61. United States Environmental Protection Agency. Drinking Water Regulations and Contaminants, 2025. Available at: <https://www.epa.gov/ground-water-and-drinking-water/national-primary-drinking-water-regulations>, [Online; accessed 10 December 2025].
62. Jiang, Y.; Li, C.; Sun, L.; Guo, D.; Zhang, Y.; Wang, W. A Deep Learning Algorithm for Multi-Source Data Fusion to Predict Water Quality of Urban Sewer Networks. *Journal of Cleaner Production* **2021**, *318*, 128533. <https://doi.org/10.1016/j.jclepro.2021.128533>.
63. EYATH S.A.. Water Measurements in the area of Thessaloniki, Greece. Public page linking to area-level water quality measurements and historical data., 2026. Available at: <https://theses.whiterose.ac.uk/id/eprint/30179/>, [Online; accessed 12 January 2026].
64. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: a system for large-scale machine learning. In Proceedings of the Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation, USA, 2016; OSDI'16, pp. 265–283. <https://doi.org/10.5555/3026877.3026899>.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.