P_2O: AI-Driven Framework for Managing and Securing Wastewater Treatment Plants

Ajay Kulkarni[1]; Mehmet Yardimci[2]; Md Nazmul Kabir Sikder[3]; and Feras A. Batarseh[4]

Abstract: Wastewater treatment plants (WWTPs) are critical infrastructures responsible for processing wastewater before discharging effluent to rivers and other potential uses. WWTPs use large, connected deep tunnels for storing sanitary and wet-weather flows for treatment. However, wastewater in those systems cannot exceed safe tunnel levels in order to prevent overflows of untreated wastewater into the environment. Further, WWTPs are among the 16 national lifeline infrastructure sectors in which the utilization of sensor technology has increased, making the sectors vulnerable to all forms of cyber threats. Considering these challenges, the work presented in this manuscript uncovers the role of AI at WWTPs by focusing on two problems: tunnel water-level prediction and detection of security threats. This is done by proposing an AI framework: P_2O (prediction, protection, and optimization). The prediction module forecasts the tunnel water level using deep-learning models based on the current wastewater flow in the tunnel and other inputs from the sensors and gauges. The protection module focuses on classifying the intentionality of an anomaly, i.e., whether an attack is adversarial in nature or merely an outlier, using recurrent neural network models. Last, the optimization module aims to provide actionable recommendations to pump operators using a genetic algorithm. The experimental results of P_2O indicate that the prediction module can predict the tunnel water level with 85% accuracy, and the protection module can detect about 97% of intentional attacks on WWTPs. AI models within P_2O are evaluated; the experimental results are presented and discussed. **DOI: 10.1061/JOEEDU.EEENG-7266.** © *2023 American Society of Civil Engineers.*

Practical Applications: This manuscript presents P_2O, which is a novel AI framework that can predict about 85% of wastewater overflow incidences and about 95% of intentional cyberattacks on a WWTP, as indicated in the experiments. The deployment of P_2O at a WWTP is essential, especially considering the adverse effects of overflowing wastewater on the environment (i.e., rivers and other water bodies). Moreover, cyberattacks on WWTPs can be subtle, making them challenging to detect; on average, most of them are noticed within one week to one month after the attack. This makes national infrastructure vulnerable to external and internal threats, influencing the well-being of water bodies and overall national security. P_2O provides a real-time monitoring interface and can recommend optimal actions in different scenarios (i.e., outliers) for pump operators and process engineers at WWTPs.

Introduction

Wastewater treatment plants (WWTP) process the wastewater collected from cities, households, factories, and more, before discharging it (effluent) for reuse (such as reclaimed water or for agriculture) or to a river or another water body (Corominas et al. 2018). Water plants are complex systems that utilize advanced network devices to improve operational problems. It is common in WWTPs (especially in populated areas) to use large connected tunnels for storing sanitary and wet-weather flows for treatment (Owolabi et al. 2022). At WWTPs, the decisions on pumping the stored wastewater from tunnels need to be made in a short time because the wastewater cannot exceed the tunnel's safe levels, which can cause the overflow of the untreated water (Corominas et al. 2018; Schütze et al. 2002) or an overuse of chemicals. This reason makes calculation time a critical issue (Schütze et al. 2002). The U.S. Environmental Protection Agency (EPA) reports between 23,000 and 75,000 incidences (between 11,400,000 and 37,900,000 L of wastewater annually) of overflowing untreated wastewater into the environment (Robison 1991). This overflowed wastewater harms the soil, air, and rivers (Owolabi et al. 2022). It also leads to public health issues, such as gastrointestinal outbreaks (Sojobi and Zayed 2022). It has been noted that wastewater treatment consumes about 12.6% of the total energy by public utilities (Sanders and Webber 2012), which makes up about 30% of the total operation and maintenance costs in a WWTP (Robison 1991) and makes it essential to have a solution that predicts the overflow of the wastewater while minimizing the potential overflow risks and greatly helping critical decision-making processes (such as pumping and adding chemicals). This can be achieved using artificial intelligence (AI) for different downstream tasks to provide sophisticated decision support at WWTPs.

Most modern WWTPs utilize cyberphysical mechanical actuators, electrical sensors, and internet components communicating via a computer network to supervise and configure the treatment process. Large WWTPs have hundreds of sensors, actuators, and complex connections between electrical devices to the control and protection switching gear (CPSG), making it vulnerable to cyberattacks (Adepu and Mathur 2018). These cyberattacks are launched with "minimum perturbation," which deems them challenging to detect (Adepu and Mathur 2016b). We present two examples of

[1] Postdoctoral Associate, Commonwealth Cyber Initiative (CCI), Virginia Tech, Arlington, VA 22203. ORCID: https://orcid.org/0000-0002-3620-2670

[2] Ph.D. Candidate, Dept. of Computer Science, Virginia Tech, Blacksburg, VA 24061.

[3] Ph.D. Candidate, Bradley Dept. of Electrical and Computer Engineering (ECE), Virginia Tech, Arlington, VA 22203.

[4] Associate Professor, Dept. of Biological Systems Engineering (BSE), Virginia Tech, Blacksburg, VA 24060 (corresponding author). ORCID: https://orcid.org/0000-0002-6062-2747. Email: batarseh@vt.edu

Note. This manuscript was submitted on November 29, 2022; approved on March 21, 2023; published online on June 17, 2023. Discussion period open until November 17, 2023; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Environmental Engineering***, © ASCE, ISSN 0733-9372.

such cyberattacks on sensors, as plotted in Fig. 1. These examples are based on the SWaT data set (Goh et al. 2016), and Goh et al. (2016) present more details on these attacks. In Fig. 1(a), the two vertical lines indicates that attackers manipulated level transmitter sensor 101's (LIT10) readings from the system in a marked timeframe. An experienced operator might detect this anomaly only if they monitor system values frequently. However, in some cases, even under the supervision of an expert, an attack may not be detected. Another example of an attack is shown in Fig. 1(b), in which the attacker mimics the patterns of the system while manipulating the data, which is a complex attack instance for an expert to detect. These are examples of single-stage single-point attacks, focusing on precisely one point in a CPSG. However, attackers can also launch multistage multipoint attacks, which can occur at multiple stages of the process from multiple attack points, making detecting the attacks even more difficult.

This makes monitoring and interpretation of the data crucial for operational decision-making while ensuring safety, security, and efficiency at a water facility (Tuptuk et al. 2021). These reasons constitute a need for a solution that forecasts the wastewater level, detects potential cybersecurity threats, and utilizes these insights to optimize the processes in WWTPs (Radanliev et al. 2021). Considering these aspects to assist treatment operators at WWTPs, this paper presents an AI solution: prediction; protection; and optimization (P_2O). Thus, the main contribution of this paper is to provide a three-way solution using AI as a visual tool to provide actionable insights into pump operators. P_2O is developed by answering the following two research questions (RQs).

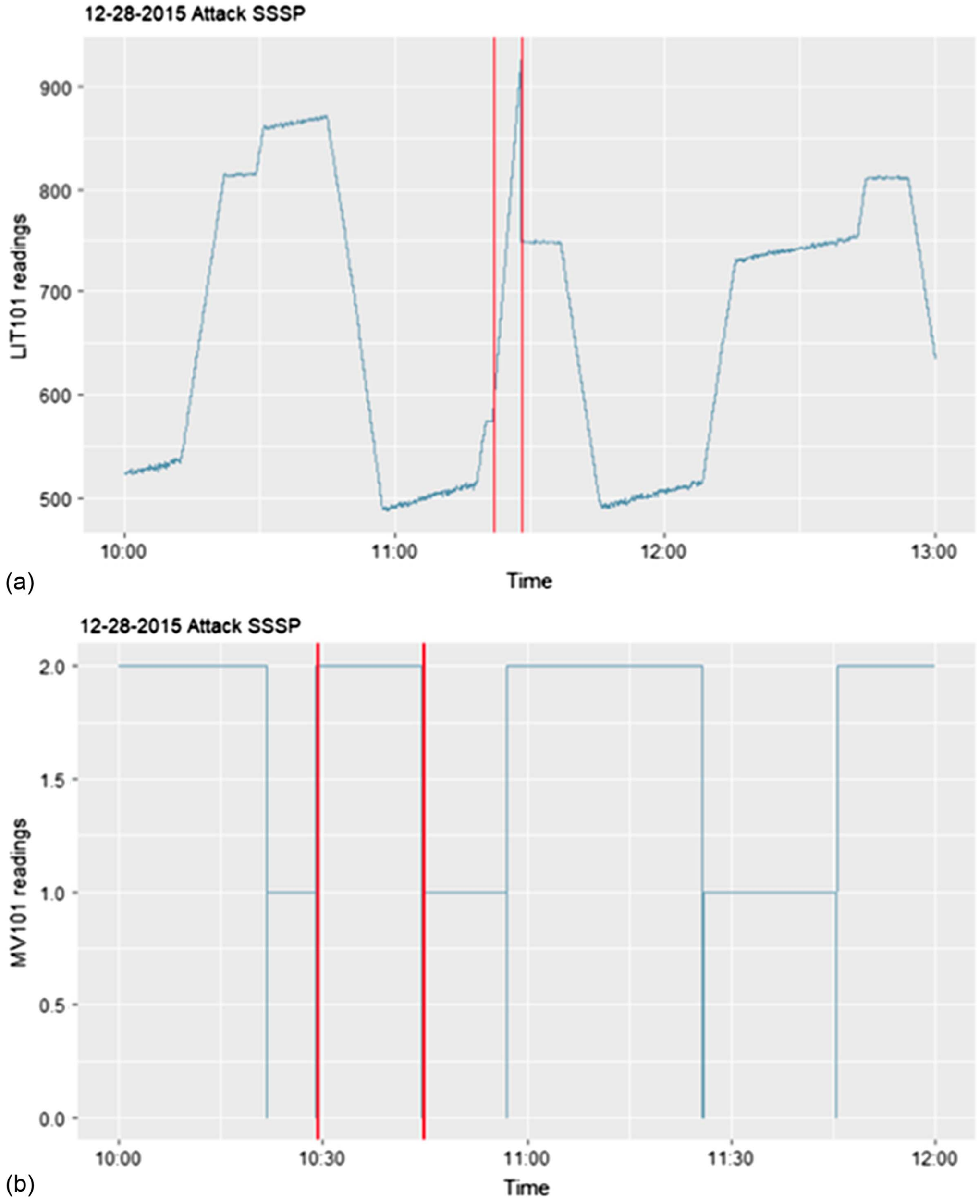


Fig. 1. (a) Attack on the LIT101 sensor (vertical lines starting at 10:29), which manipulates the data collected from the sensor. (b) Attack on the MV101 sensor, which began at 10:30 and ended at 10:45.

RQ1: Which AI approach provides accurate predictions in P_2O for predicting tunnel water levels?

RQ1 compares ML and DL performance in predicting tunnel water levels. It is based on Hypothesis 1 (H_1): $\mathrm{Acc}(\mathrm{DL}(wl_p)) > \mathrm{Acc}(\mathrm{ML}(wl_p))$. This states that the accuracy (Acc) of predicting the wastewater level (wl_p) using a DL model is higher than that in an ML model.

RQ2: How can DL models detect cyberattacks at WWTPs?

RQ2 focuses on the protection module of P_2O. It is based on comparing two recurrent neural network (RNN) models using classification metrics. RQ2 is answered based on Hypothesis 2 (H_2): $\mathrm{Acc}(\mathrm{LSTM}(A_C)) > \mathrm{Acc}(\mathrm{GRU}(A_C))$. This states that the overall accuracy (Acc) of detecting cyberattacks (A_c) of the long short-term memory (LSTM) model is higher than the gated recurrent units (GRU) model.

The structure of this paper is as follows: The next section presents recent cyberattack examples on WWTPs by highlighting the need to use DL models for detecting attacks. The "Methodology" section presents P_2O and explains the methods used in the prediction, protection, and optimization modules. "Results" section presents experimental results by highlighting the performance and comparisons of DL and ML models. The "Discussions" section presents the relevance of results to decision-making and utilization of P_2O. Lastly, the "Conclusions" section provides future work and other concluding remarks.

Literature Review

Cyberphysical systems (CPS) are "Intelligently networked systems with embedded sensors, processors, and actuators that are designed to sense and interact with the physical world (including human users), and support real-time, guaranteed performance in safety-critical applications" (Wang et al. 2015; Forest 2006). WWTPs are CPSs, and safeguarding them is a national priority (Flynn 2020). Adepu and Mathur (2016a) noted that cyberattacks have increasingly targeted water treatment systems in recent years, and they are ranked third in the Kaspersky ICS CERT vulnerability report (Alanazi et al. 2022). Illyes et al. (2018) provide two reasons for this pattern: (1) due to the expansion of the Internet of Things (IoT); and (2) the proliferation of AI in the decision-making processes. Further, Hassanzadeh et al. (2020) presented 15 disclosed, documented, and malicious cybersecurity incidents in the water sector, among which two recent incidents are the Florida Water Supply (FWS) hack in 2021 (Miller et al. 2021) and the Riviera Beach Water Utility (RBWU) attack in 2019. In the FWS hack, the hacker gained remote access to the programmable logic controller (PLC) unit that controls sodium hydroxide levels in the water. The hacker increased the amount of sodium hydroxide content in the water by 110-fold; fortunately, the attack was mitigated before the toxic levels of chemicals were diffused into the distribution network. In the RBWU incident, ransomware, a common type of cyberattack, was launched, which paralyzed the computer systems controlling pumping stations, water-quality testing, and payment operations. The government authorities paid 65 bitcoins (approximately $600,000) to the attacker in a few days; after two weeks, however, water pump stations and water quality testing systems were partially available. Further, on January 15, 2021, an intrusion happened at the water treatment plant that served parts of the San Francisco Bay area (Collier 2021). The hacker had the username and password of an employee's TeamViewer account. The hacker tried to poison the drinking water by deleting the programs that treat the drinking water. It took one day to discover this hack; the authorities acted by changing the password and reinstalling the programs. In these examples, the systems were breached; yet, authorities could notice the intrusions only after investigating traffic and data flow. These incidents highlighted the vulnerability of these infrastructures and high relevance to public safety. Considering these details, Jian-Hua Li (Li 2018) made a case for developing and using DL-based AI models for malware classification and intrusion detection. Further, Hindy et al. (2019) used the modbus penetration testing framework (SMOD) data set to improve security information and event management of water infrastructures. In their study, the authors used six ML models for scenario classification and compared them based on classification accuracy. The authors noted that the *k*-nearest neighbors indicated 94% accuracy in detecting anomalies. In another study, Albahar et al. (2020) used the SMOD data set to detect malicious acts from nonmalicious ones based on neural networks. The authors compared different models by analyzing the confusion matrix generated from the results. The authors reported greater than 60% accuracy in detecting malicious activities and about 44% accuracy in detecting operational scenarios. Moradbeikie et al. (2020) conducted experiments to improve safety via fast and accurate hazard detection. For these experiments, the authors categorized data into six classes: normal data; transient failure; permanent failure; random attack;stealthy attack; and false alarm. The authors then compared the performance of different ML models for attack detection. The authors further used precision, recall, F-measure, false positive rate, and accuracy; they reported about 97% accuracy on hazard detection and noted that it could reduce about 60% of the time in the system recovery reconfiguration. Sahu et al. (2021) proposed a fusion engine that can improve detection accuracy by fusing features to detect cyberattacks in power systems at CPSs. This study utilized F1 score, precision, and recall for evaluating intrusion detection and classification. The authors reported that the fusion engine could improve performance by an average of 15% to 20% (based on F1 scores). Faramondi et al. (2021) used ML techniques for detecting and categorizing threats in CPS using a water distribution testbed. The authors compared four ML techniques based on accuracy, recall, precision, and F1 score. Based on these metrics, the authors reported the highest accuracy (99%) for the random forest (RF) model. Last, a study conducted by Perrone et al. (2021) for threat recognition in critical CPS compared five ML models based on accuracy, precision, recall, specificity, F-measure, and G-mean. The authors reported that the RF model showed the best accuracy (90.2%) for threat recognition compared with other models. Considering these similar studies, our work primarily focuses on DL-based models for detecting and classifying malicious activities (while comparing that to other ML models), DL models proved superior to ML and more scalabale than existing state-of-the-art works. To achieve this, two DL models are developed and compared based on accuracy, precision, recall, and F1 score to select the best model for P_2O's protection module.

Methodology

The P_2O solution consists of three major AI-driven modules, i.e., prediction, protection, and optimization, as shown in Fig. 2. Two data sets have been used to develop these modules, and their details are provided in the subsection "Datasets." The details of the methodology used for the prediction module are presented in the subsection "Prediction Module." The protection module focuses on classifying the intentionality of anomalies. To demonstrate its application in a WWTP, SMOD, is utilized, and the details on the methodology are provided in the subsection "Protection Module." Details of the optimization module are presented in the subsection "Optimization Module," which aims to provide actionable

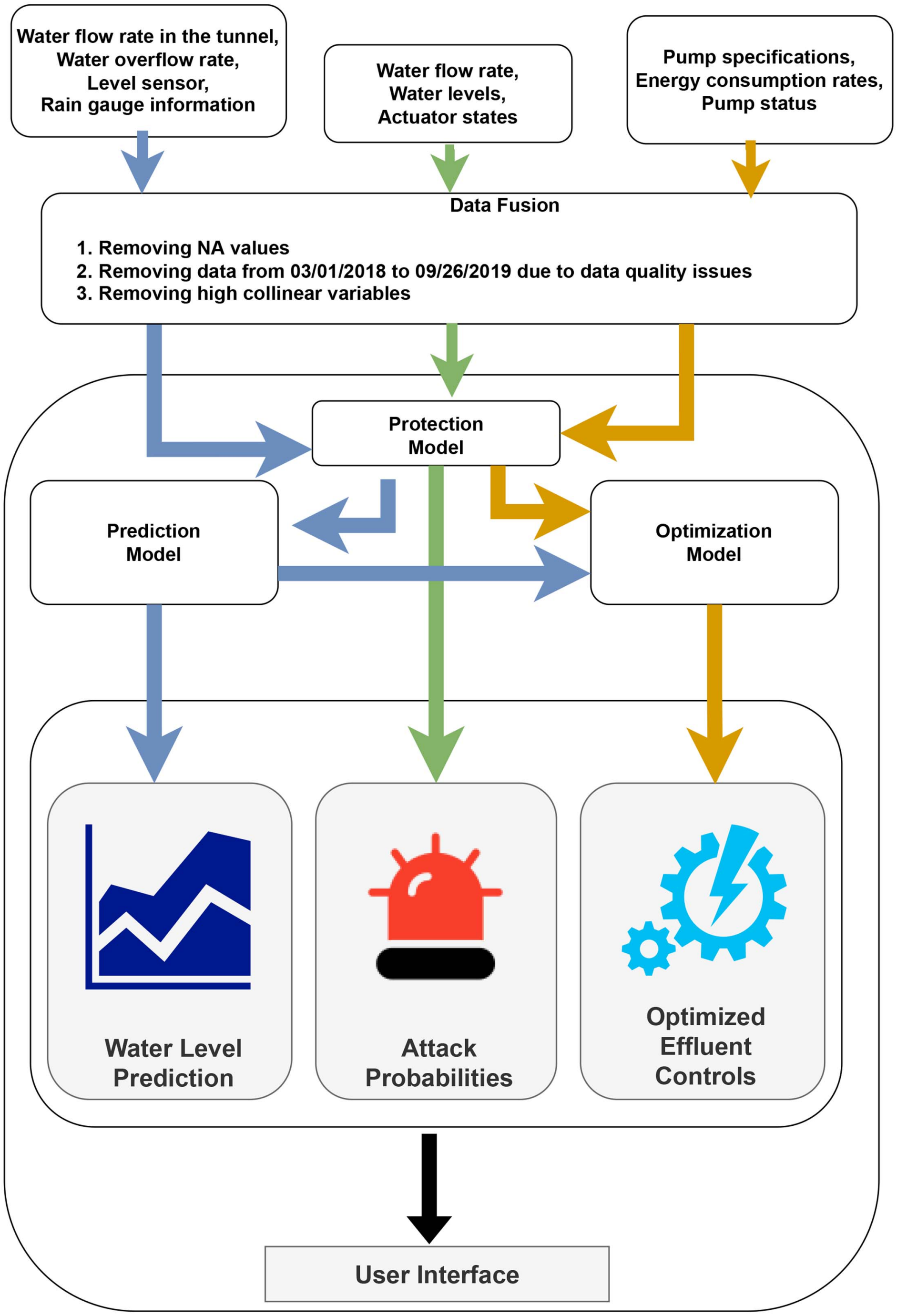


Fig. 2. P_2O consists of three modules: prediction; protection; and optimization.

recommendations, such as when to start pumping water from the tunnel, how many pumps to operate, and with what capacity these pumps should be run (Batarseh et al. 2022).

Data Sets

The data set for water-level prediction and optimization is obtained from a WWTP (name not revealed due to nondisclosure reasons), while for the protection module, the SMOD data set is utilized. The SMOD data set (Laso et al. 2017) is obtained from the open-source water testbeds because there are no publicly available real data sets with sufficient complexity of a modern water infrastructure (Goh et al. 2016; Laso et al. 2017). The SMOD data set is useful to build AI for designing and evaluating defense mechanisms for water infrastructures (Laso et al. 2017), making it a suitable choice for this work. The details of the data sets are presented in the next section.

Data Set for Prediction and Optimization Modules

The data for the tunnel water-level prediction is collected from one of the largest advanced WWTP in the world (which cannot be shared due to a nondisclosure condition from the WWTP). This plant treats about 1,135.62 million liters of wastewater daily, but at a peak flow, it can treat up to 3.79 billion liters daily. To fulfill the tunnel water-level prediction objective, data from March 1, 2018, to March 26, 2022, with 5 min intervals, are collected. The collected data includes six spreadsheets, with each providing details on (1) major flows coming in the tunnel; (2) overflow from the

Table 1. Fifteen attack situations and associated labels were used for classification using the SMOD data set

Situation	Affected component	Operational scenario	Label
Normal	None	Normal	Normal
Plastic bag	Ultrasound sensor	Accident/sabotage	Intentional attack
Blocked Measure 1	Ultrasound sensor	Breakdown/sabotage	Unknown
Blocked Measure 2	Ultrasound sensor	Breakdown/sabotage	Unknown
Floating objects in main tank (two objects)	Ultrasound sensor	Accident/sabotage	Intentional attack
Floating objects in main tank (seven objects)	Ultrasound sensor	Accident/sabotage	Intentional attack
Humidity	Ultrasound sensor	Breakdown	Outlier event
Discrete Sensor 1 failure	Discrete Sensor 1	Breakdown	Outlier event
Discrete Sensor 2 failure	Discrete Sensor 2	Breakdown	Outlier event
Denial of service attack	Network	Cyberattack	Intentional attack
Spoofing	Network	Cyberattack	Intentional attack
Wrong connection	Network	Breakdown/sabotage	Unknown
Person hitting the tanks (low intensity)	Whole subsystem	Sabotage	Intentional attack
Person hitting the tanks (medium intensity)	Whole subsystem	Sabotage	Intentional attack
Person hitting the tanks (high intensity)	Whole subsystem	Sabotage	Intentional attack

Source: Data from Laso et al. (2017).

tunnel system to the river; (3) level sensors in the tunnel; (4) rainfall; (5) flow meters associated with pumps used for dewatering the tunnel; and (6) the other critical main plant flows. Each file has 243 columns and 428,244 rows, i.e., the overall dimension of the data used in the analysis is 1,458 columns and 2,569,464 rows. Most of the columns (about 95%) in each file have NA values that indicate a need to preprocess the data. Further, the data on pump utilization, overflow from the tunnel to the river, and water mass have been used in the optimization module.

Data Set for Protection Module

The SMOD (Laso et al. 2017) data set has been utilized for the intention classification and attack situation detection at a WWTP. The authors (Laso et al. 2017) of the SMOD data set provide a temporal series representing details on 15 situations, affected components, and five operational scenarios: normal; anomalies; breakdown; sabotages; and cyberattacks. They also note that SMOD is useful in surveillance and security applications for CPSs, especially in training the algorithms that assess data alteration and service degradation. The SMOD data set is generated using an ultrasound depth sensor, four discrete sensors, two pumps, and two tanks for storing water. The data were collected at every 0.1 s, but the execution time for each operational scenario was different. For intention classification, the details are presented in Table 1.

Prediction Module

This subsection focuses on ML and ANN-based DL models for tunnel water-level prediction. A schematic diagram of the methodology used for this module is shown in Fig. 3. As shown in the figure, this module has five components: data preprocessing; exploratory data analysis (EDA); model development; hyperparameter tuning; and model evaluation and selection. Details on these components are as follows.

Data Preprocessing for Wastewater-Level Prediction

The data used for this study include 243 columns in each file; thus, the first task was to understand the not available (NA) sensor readings in the data. The reason for NAs is due to the format of the data produced by the reporting tool while fetching the data. Thus, NAs were removed from the data, and the output is identified. After preprocessing, selecting output, and combining the data into a dataframe, there were 42 columns in the data. This combined dataframe also had NAs in the first 60,301 rows, which were removed. Finally, at the end of the preprocessing phase, the data consisted of 42 columns and 367,943 rows.

In the next step, two different versions of the preprocessed data were created, based on principal component analysis (PCA) and sampling, to understand the effects of data-processing techniques and maintain AI assurance. Abdi and Williams (2010) provided evidence that PCA is a widely used technique that provides a

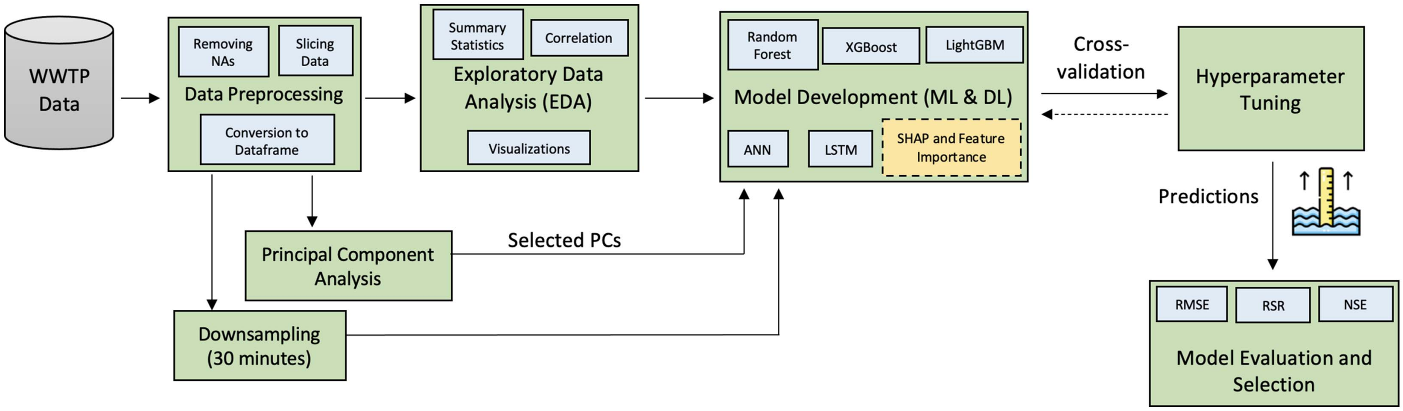


Fig. 3. Schematic diagram of the methodology used for tunnel water-level prediction.

set of uncorrelated variables from a set of correlated variables. Thus, considering collinearity in the data, the PCA technique for preprocessing was selected. The second data set was produced based on downsampling and was performed by selecting the observations based on intervals of 30 mins. This way, two versions of the data were produced based on the raw data at the end of the preprocessing phase.

Summary Statistics and Graphical Representations

EDA helps to find or understand patterns in the data; it is a fundamental step after data collection (Komorowski et al. 2016). To achieve this, four techniques, i.e., summary statistics, visualization, correlations, and variance inflation factor (VIF), have been used for the tunnel water prediction level. First, the summary statistics, such as minimum, maximum, median, interquartile range, mean, and standard deviation, are calculated for all the sensors. Next, time-series plots of different sensors are produced to understand the visual patterns in the data. Next, Pearson's correlation coefficient (Ratner 2009) is used to uncover the strength of a linear relationship between every two sensors. Finally, a multicollinearity check in the data is performed using VIF analysis. The literature (James et al. 2013) indicated that a VIF value higher than or equal to 5 indicates multicollinearity, which affects the regression models negatively (Alin 2010). This recommendation was followed while performing a multicollinearity check.

Model Selection and Development for Prediction Module

Ardabili et al. (2020) pointed out the rise of DL and ML applications for data-driven decision-making at WWTPs. A survey conducted by Corominas et al. (2018) revealed that artificial neural network (ANN) and PCA are the top-two widely used methods in the literature. Furthermore, the authors noted that ANN, PCA, and regression, along with four other techniques, reached the plateau of productivity; these methods have a large number of citations and are well-established in the research field (Corominas et al. 2018). Generally, ML models are simple, computationally efficient, and can perform better when the amount of data is small (Thompson et al. 2020). On the other hand, DL models are complex, computationally expensive, and less explainable but provide accurate results on large data sets (Thompson et al. 2020). This makes it challenging to select a correct model that provides accurate predictions while maintaining complexity. Based on these reasons and considering a scenario of high-stake decision-making, a comparison among ML-based models, ANN, and RNN-based DL models for water-level prediction is performed.

Explainability can help improve scientific understanding and create trust by showing the importance of different variables to the decisions (Batarseh et al. 2021; Doshi-Velez and Kim 2017; Gilpin et al. 2018; Adadi and Berrada 2018). According to Kabir Sikder et al. (2022), SHapley Additive exPlanations (SHAP) (Lundberg and Lee 2017), a game theory-based black box explainer, has recently gained traction in DL models' deployments. It also has been revealed that the tree-based ML models and LSTM with SHAP provide the best approach in explaining the model predictions for multivariate time-series data (Zanzotto 2019). Accordingly, considering the multivariate aspect of time-series forecasting, three ensemble tree-based ML models and two DL models are selected to predict the tunnel water level. Three selected ML models are RF, eXtreme gradient boosting (XGBoost), and light gradient boosting machine (LightGBM), while two selected DL models are feed-forward ANN (FF-ANN) and LSTM. The equations (Bruce et al. 2020) used in modeling RF and boosted trees (XGboost and LightGBM) are presented in Eqs. (1) and (2)

$$\hat{F}_{rf}^{B} = \frac{1}{B}\sum_{b=1}^{B} T(x;\theta_b) \tag{1}$$

where B = a number of trees; $T(x;\theta_b)$ = a forest of b trees where θ_b characterizes the bth RF tree in terms of split variables, cutpoints, and terminal-node values

$$\hat{F} = \alpha_1\hat{f}_1 + \alpha_2\hat{f}_2 + \cdots + \alpha_M\hat{f}_M \tag{2}$$

where $\alpha_1, \alpha_2, \ldots, \alpha_M$ = the weights to train $\hat{f}_1, \hat{f}_2, \ldots, \hat{f}_M$ models that minimize the error e_m to provide the ensemble model $\hat{F}$.

The equations used in developing FF-ANN (James et al. 2013) and LSTM (Yu et al. 2019) are presented in Eqs. (3) and (4)

$$F(X) = \beta_0 + \sum_{k=1}^{k} \beta_k g\left(w_{k0} + \sum_{j=1}^{p} w_{kj}X_j\right) \tag{3}$$

where k = hidden layers; β = bias values; function $g(..)$ = the activation functions for p variables; and w = the weight values

$$y^t = g_2(W_3 g_1(W_1 a^{t-1} + W_2 a^t + b_a) + b_y) \tag{4}$$

where t = a time-step; W_1, W_2, W_3 = the coefficients; and b_a, b_y = the bias values temporally shared with two activation functions, i.e., g_1 and g_2, which produce an output y^t for that time-step.

Hyperparameter Tuning of Tunnel Water-Level Prediction Models

This process started by splitting the preprocessed data into training (70%) and testing (30%) sets. The training data set was used further for hyperparameter tuning, i.e., an optimization process. Kochenderfer and Wheeler (2019) provided a general philosophy for hyperparameter tuning, i.e., "determine which hyperparameters to tune and their search space, adjust them from coarse to fine, evaluate the performance of the model with different parameter sets, and determine the optimal combination." Yu and Zhu (2020) noted different search algorithms, i.e., grid search, random search, Bayesian optimization and its variants, and tree parzen estimators, which can help to determine the optimal combination of hyperparameters. For P_2O, a grid search and random search algorithms are used for this purpose. A grid search algorithm is an exhaustive approach that provides the most accurate prediction with the optimal combination (Yu and Zhu 2020). It is a widely used algorithm because of its mathematical simplicity (Bergstra and Bengio 2012); it is also used for tuning tree-based models. The random search algorithm is a randomized search technique that improves grid search (Bergstra and Bengio 2012). As noted by Yu and Zhu (2020), random search is more effective than grid search in most cases, and flexible resource allocation and parallelization are the main advantages of this method (Feurer and Hutter 2019). Thus, for these reasons and considering the computational complexity of DL models, a random search technique is selected for tuning hyperparameters of DL models.

For tree-based models, a grid search is performed using threefold cross-validation (CV) on the training set (Asadollahfardi et al. 2022), consisting of 70% of the original data (Rahnama et al. 2020). Usually, the cross-sectional CV methods split the data randomly using random seeds in training and validation sets. The data obtained from these methods do not mimic the temporal uncertainty, creating gaps in the time series (Willemain 2013). Further, these methods may also lead to information leakage in the model, which affects the model's performance on unseen data (Willemain 2013). Therefore, the CV process should be based on a temporal partition. A figure (Fig. S1) explaining this concept is provided in

Table 2. Hyperparameter details for RF, XGBoost, and LightGBM

Parameter	Description	Values
	RF model hyperparameters	
*n*_estimator	Number of trees in the forest	100, 200, 300, 400, 500
max_depth	Maximum depth of the tree	5, 10, 20, 30, 40,50
max_features	Number of features to consider when looking for the best split	auto, sqrt
min_samples_split	Minimum number of samples required to split an internal node	2, 3, 5, 7, 9
min_samples_leaf	Minimum number of samples required to be at a leaf node	1, 3, 5
	XGBoost model hyperparameters	
eta	Learning rate or step size shrinkage	0.05, 0.1,0.2,0.3
*n*_estimator	Number of boosting rounds	10, 20, 50, 100, 200, 300
max_depth	Maximum depth of a tree	2, 4, 6, 7, 8, 10
colsample_bytree	Subsample ratio of columns when constructing each tree	0.5, 0.7, 0.9, 1
alpha	L1 regularization	0, 0.5, 1
	LightGBM model hyperparameters	
learning_rate	Learning rate	0.05, 0.1, 0.2, 0.3
num_leaves	Complexity of tree model	50, 60, 70, 80, 100
num_iterations	Number of iterations	10, 20, 50, 100, 200, 300
max_depth	Maximum depth of a tree	2, 4, 6, 8, 10
bagging_fraction	Subsample ratio of columns when constructing each tree	0.5, 0.7, 0.9, 1
lambda_l1	L1 regularization	0, 0.5, 1

the Supplemental Materials. The temporal partition can be performed using a time-series split in which the observations are split along with sequences (Hyndman and Athanasopoulos 2018). After performing the CV using a time-series split, the training data are divided into training and validation sets. Next, the training set is used for building RF, XGBoost, and LightGBM models, while the validation set is used to assess each model's performance in the hyperparameter tuning process. The details on the selection of hyperparameters are provided in Table 2, which are based on guidance provided by Kuhn and Johnson (2013).

The hyperparameter tuning for FF-ANN and LSTM is performed using a random search algorithm as suggested by Asadollahfardi et al. (2018) based on a time-series CV. The hyperparameter tuning for DL models can be categorized into two groups: (i) model training; and (ii) model design (Yu and Zhu 2020). The most critical hyperparameters for model training are batch size and learning rate because they determine convergence speed; for the model design, the number of neurons and number of hidden layers are important (Yu and Zhu 2020; Charu 2018). In this study, the hyperparameters from both groups are tuned to find the best suitable configuration. This task is performed using the last 20% of the training data as a validation set to assess the model's performance in predicting tunnel water level. Details on the hyperparameters of FF-ANN and LSTM are provided in the Supplemental Materials.

Model Evaluation and Selection Metrics

The selection of evaluation metrics is based on the performance measures and evaluation criteria for hydrologic and water quality models provided by Moriais et al. (2007). The AI models are evaluated based on three metrics: root mean squared error (RMSE); RMSE-observation standard deviation ratio (RSR); and Nash–Sutcliffe efficient (NSE) (Park et al. 2022). The formulas for these metrics are provided as follows:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - p_i)^2}{n}} \tag{5}$$

$$\mathrm{RSR} = \frac{\sqrt{\sum_{i=1}^{n}(y_i - p_i)^2}}{\sqrt{\sum_{i=1}^{n}(y_i - \bar{p})^2}} \tag{6}$$

$$\mathrm{NSE} = 1 - \frac{\sum_{i=1}^{n}(y_i - p_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{7}$$

In Eqs. (5)–(7), y_i = the observed value; p_i = the predicted value; n = total data points in the data set; $\bar{p}$ = the average of predicted values; and $\bar{y}$ = the average of observed values. RMSE is the square root of the average squared error (Chai and Draxler 2014), which ranges from 0 to ∞. RSR ranges from 0 to 1, where the smaller the RSR the better the model, and a value <0.7 is considered satisfactory (Moriasi et al. 2007; Bennett et al. 2013). The value of NSE ranges from $-\infty$ to 1, and the model is considered a better fit if it reaches 1 (Moriasi et al. 2007).

Protection Module

The detection and classification of malicious activities are performed using the SMOD data set. For this study, the data are collected through PLC using three registers: Register 2 (stores the binary state of the four discrete sensors); Register 3 (records the binary state of Pumps 1 and 2); and Register 4 (contains the value of the ultrasound depth sensor). These data are then used for classifying the attack intentions. A schematic diagram of the methodology used for detecting and classifying malicious activities using SMOD is shown in Fig. 4. The details on data preprocessing, model development, and evaluations are presented next.

Data Preparation for Protection Module

Data preprocessing is performed in four steps: (1) data conversion; (2) data normalization; (3) datatype conversion; and (4) oversampling. In the first step, the sensor values from Register 2 are extracted and converted into a binary format, resulting in seven new columns. In the second step, data normalization is performed using min-max normalization to ensure that all the sensor values have the same scale. Next, the normalized data are converted into a timestamp, so each data point forms a time-series window for the model to learn. After analyzing the data, it was found that there is an imbalance in the data, i.e., unequal class distribution (Kulkarni et al. 2020). Considering this data quality issue, an oversampling technique, i.e., a data-level method, is applied to increase the minority class instances to match them with the majority class instances.

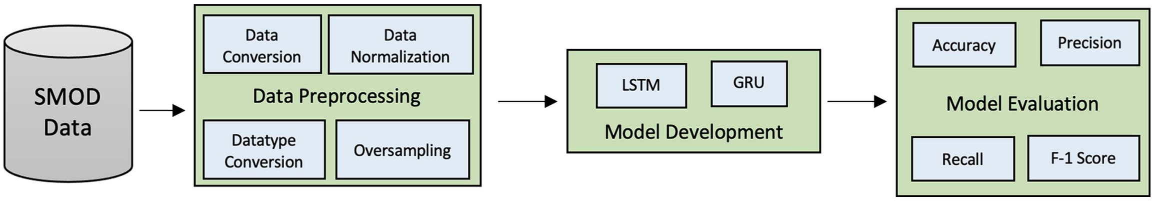


Fig. 4. Schematic diagram of the methodology used for detecting and classifying malicious activities using SMOD.

The oversampling is performed using the synthetic minority oversampling technique (SMOTE) (Chawla et al. 2002), a popular oversampling method in data mining literature (Fernández et al. 2018). After applying SMOTE, each minority class was oversampled to 23,287 instances to match the majority class. Details about unequal class distribution in the data set are provided in the Supplemental Materials (Fig S2).

Model Development for Classifying Cyberphysical Attacks

Model development is started by dividing 80% of the prepossessed data as a training set and the remaining 20% as a test set. Next, two DL models, i.e., GRU and LSTM, have been selected with a softmax activation function to detect and classify malicious activities. The GRU model is also an RNN-based model suitable for predicting sequences and time-series-based data (LeCun et al. 2015). It has also been noted that GRU is similar to LSTM but computationally inexpensive (Hettiarachchi and Ranasinghe 2019); both, however, are popular to use for time-series (Lim and Zohren 2021). Considering these reasons, GRU and LSTM models are compared. In this study, the hyperparameters from both groups, i.e., model training (learning rate) and model design parameters (number of hidden layers and neurons), are tuned to find the best suitable configuration. This task is performed using the last 20% of the training data as a validation set to assess the model's performance. For the hyperparameter tuning, the weights and biases (W&B) (Biewald 2020) platform is used, which automates the hyperparameter optimization using the grid-search method. Details on the range of hyperparameters used for tuning are provided in the Supplemental Materials.

Model Evaluation Metrics

Model evaluations are performed using a confusion-matrix-based approach. A confusion matrix compares the difference between the observed and predicted values (Kulkarni et al. 2020). It also provides details on true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), which offer more insights into the classifier. Further, using the values from the confusion matrix, the accuracy, precision, recall, and F_1 score are calculated. Accuracy indicates the overall accuracy of the classifier, while precision shows the accuracy of the model in predicting positive instances, and recall denotes the strength of a model predicting positive outcomes (Bruce et al. 2020). The F_1 score is a weighted harmonic mean between precision and recall, which provides the trade-off between correctness and coverage. The formulas for accuracy, precision, recall, and F_1 score are provided as follows:

$$\text{Accuracy} = \frac{\text{TN} + \text{TP}}{\text{TN} + \text{FP} + \text{FN} + \text{TP}} \tag{8}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{9}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{10}$$

$$F_1\text{score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{11}$$

This way, GRU and LSTM are compared based on these classification evaluation metrics.

Optimization Module

The optimization module in P_2O aims to minimize the amount of wastewater sent to the wet-weather treatment process using an optimization technique, as shown in Fig. 5. It can be achieved by controlling the influent flow to the tunnel and the wet-weather treatment plant by optimizing the actions and using industrial pumps in the WWTP. Currently, the total depth of a tunnel at the WWTP is 137 ft, which is 120 ft below sea level. The WWTP uses five large industrial pumps that can move water between 189 and 315Ml/d (million liters per day), and one lesser capacity pump can move water between 11 and 37 Ml/d. These pumps directly control the amount of wastewater treated by chemical means. Thus, this module applies an optimization technique for finding the optimal actions and their expected effect on the level of wastewater in the tunnel. The optimization module uses inputs from sensors, actuators, pump states, and forecasted water levels. It provides actionable recommendations such as when to start pumping water to the tunnel, how many pumps to operate, and what capacity these pumps should be run for the operators. This is achieved using a genetic algorithm (GA), i.e., a classical metaheuristic algorithm inspired by natural selection (Hingston et al. 2008). Dokeroglu et al. (2019) noted that GA is a widely used optimization algorithm compared with other methods; due to its popularity, GA is selected for this task. To perform optimization, the GA follows selection, crossover, and mutation (Katoch et al. 2021). Based on Zhong et al. (2005), the tournament selection operator is suitable for the problems in which an individual's fitness is essential. Thus, tournament selection is suitable considering individual pumps' importance. Next, the half-uniform crossover operator is selected because it provides fast convergence to local minima and enables a fast search of solution space (Picek et al. 2012). Finally, the multiple-bit flip operator is used for the mutation because it provides a high variance between generations (Chicano et al. 2015), which is especially important considering the water flow variable. The schematic diagram of the optimization module of P_2O is shown in Fig. 5. It can be seen that the GA takes input from the AI model (tunnel water-level prediction) and WWTP data (pump capacity, pump threshold, current tunnel water level, and water mass). The data from the WWTP are provided as input to the GA model via interpolation. The interpolation unit is responsible for estimating a function value (Lunardi 2018) $f(x_i)$ for n inputs, i.e., $x_1, x_2, \ldots, x_n$, and to make the water level into water mass conversion; measurement samples from the tunnel from different time steps have been used. This is performed

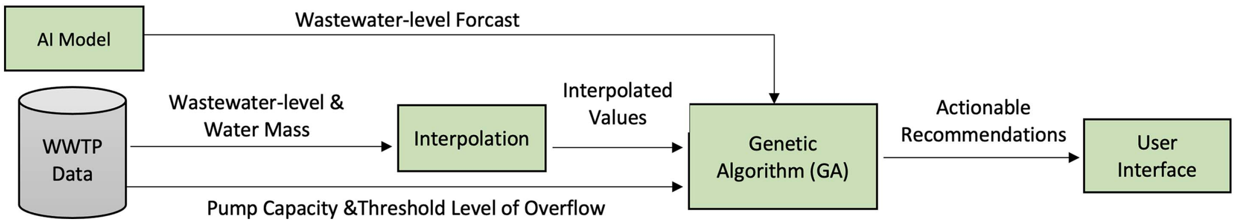


Fig. 5. Schematic diagram of the methodology used to perform optimization for providing actionable recommendations to pump operators.

in Python using polynomial interpolation by dividing 50% of the data for training and the remaining 50% for measuring an empirical error for the models. Based on the approximation error, Newton's divided differences method (Das and Chakrabarty 2016) provided the lowest approximation error and was selected in the interpolation unit. Based on this data, the GA creates the chromosomes of each pump P for n pumps such that the pump's start time (S) [00:00, 23:55] → [0,287] encoded with nine bits, run time (R) [00:00, 23:55] → $[0, 48]$, and operation capacity (C) [0%,100%] → [0,100] combined into a binary string, as shown in Eq. (12). For example, the encoding for the P_2 can be shown as $P_1{:}\{S_1 = 12{:}00, R_1 = 00{:}35, C_1 = 50\}, P_2{:}\{S_2 = 13{:}05, R_2 = 00{:}15, C_2 = 25\}$, which provides $010010000{,}000111, 0110010|010011101{,}000011, 0011010$

$$S1, R1, C1|S2, R2, C2| \dots |Sn, Rn, Cn \tag{12}$$

Considering this information, the formulation for the optimization problem is provided in Eq. (13). In this formulation, the objective is to minimize $\sum_i^n C_i R_i$, with the constraints $0 \geq C_i \geq 100$ where $C_i \in \mathbb{Z}$, $0 \geq S_i \geq 287$ where $S_i \in \mathbb{Z}$, and $0 \geq R_i \geq 48$ where $R_i \in \mathbb{Z}$.

$$I(wl_t) - I(d) \leq \frac{1}{12} \sum_i^n \sum_{j=S_i}^{\min((S_i+R_i),t)} C_i, \quad \forall\ t \in (0, 48] \tag{13}$$

where the tunnel water-level prediction from the prediction model at a time (t) = wl_t; the danger level threshold = d; and the water level to mass conversion interpolation function = $I(l)$.

Results

This section presents the results of three modules, i.e., prediction, protection, and optimization, implemented in P_2O. The prediction module uses preprocessed data for the experiments, with 42 columns, i.e., sensors, and 367,943 rows. The results of the prediction module are presented next, focusing on comparisons and selecting an AI model for wastewater prediction. After that, the results for the protection module are presented; they show a performance comparison of LSTM and GRU for intention classification and attack situation detection. Finally, the results of the optimization module are discussed, providing details on the simulated scenarios based on GA for actionable recommendations to pump operators.

Results: Prediction Module

This section presents the prediction module's results by providing details on summary statistics, visualizations, hyperparameter tuning, and model performance.

Summary Statistics and Visual Inspection

The summary statistics, i.e., minimum, maximum, median, interquartile range, mean, and standard deviation, are calculated for all the sensors. For the tunnel water-level depth sensor (output), the observation ranges from −121.21 (0 is sea level; negative values indicate below sea level) to 15.76, while the mean and standard deviation values are −114.125 and 10.413, respectively. Further, sensor observations are also visualized to check the patterns, as shown in Fig. 6. Based on the visual inspection, it is easy to identify that most values are negative (367,058) while very few are positive (851). Information from one of the WWTP's process engineers notes that the overflow from the tunnel occurs when the wastewater level observation reaches 3. Thus, based on the EDA, it can be seen that, in the last four years (2018–2022), there have been 94 incidences at the WWTP when the wastewater overflowed from the tunnel.

Next, Pearson's correlation coefficients are calculated to investigate the relationship between the wastewater level depth sensor and other sensors. The coefficients indicated that the outflow sensor

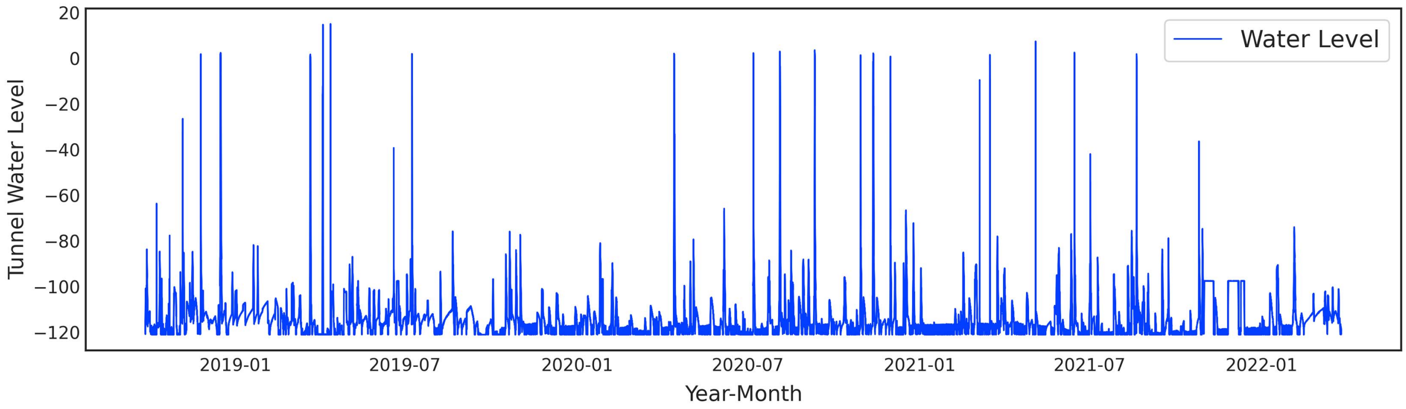


Fig. 6. Wastewaterlevel sensor observations from 2018 to 2022.

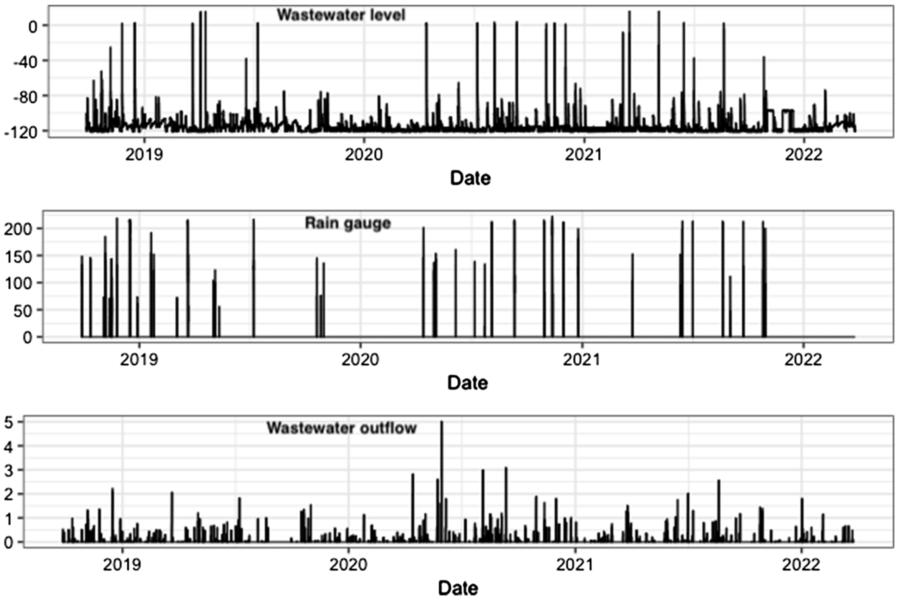


Fig. 7. Visual patterns of wastewater level, rain gauge, and wastewater outflow.

showed the highest (0.57) and rain gauges showed the second highest (0.56) significant positive correlations with the wastewater level depth sensor. Overall, most of the variables (24) showed weak positive correlations (between 0 and 3), while some (15) variables indicated a moderate positive correlation (between 0.3 and 0.7). The summary statistics show that the minimum and maximum values for the rain gauge and wastewater outflow sensor were 0, 5.01, and 0, 221.95, respectively. The time-series plots of these three variables, i.e., wastewater level, rain gauge, and wastewater outflow, are shown in Fig. 7. Based on the correlations, the VIF analysis indicated the multicollinearity between all the rain gauges and some sensors measuring the other critical main plant flows. Thus, three sensors from the rain gauge and four sensors from the other critical main plant flows are removed to eliminate multicollinearity. After performing VIF analysis, the final version of the data included 35 sensors (output sensor and time axis), and the same seven sensors were also removed from the data derived from downsampling.

Further, PCA is performed on the preprocessed data with 41 sensors (without the dependent variable), and a Scree plot is used to visualize the explained variance ratio for every principal component (PC). It was observed that the first PC contributes the most (about 20%); then, there is a gradual increase in the explained variance after PC 10. Thus, a threshold of 70% of cumulative explained variance was set after visual inspection, and 15 PCs were selected for the model development.

Hyperparameter Tuning for ML and DL Models

The hyperparameter selection for RF, XGBoost, and LightGBM is performed for three versions of the preprocessed data. The details of the selected hypermeters for each version of the data are provided in Table 3. This hyperparameter tuning procedure resulted in 10,800, 17,280, and 86,400 model fits to find the optimal combinations for the RF, XGBoost, and LightGBM models, respectively. For the FF-ANN, a random search algorithm was executed for five trials, and five different model configurations were tested in each trial. Thus, a total of 100 FF-ANN models are developed and evaluated based on mean absolute error (MAE) to find the optimal configuration of hyperparameters. During the training process, the batch size was set to 500; epochs were set to 200, and 20% of the data from the training set were used as a validation set. Further, the loss was set to MAE during training phase. The optimal hyperparameters obtained from the experiments are provided in Table 4. A random search algorithm was executed for tuning hyperparameters in the development of a multivariate LSTM model. The LSTM model is trained for 500 epochs with a batch size of eight. Further, a cubic loss function is used as an objective function for minimization in this process. The results indicated that the LSTM model with a configuration of 512 neurons with one hidden layer and a learning rate of 0.001 performed the best. The architecture of LSTM used for predicting wastewater level is shown in Fig. 8.

Table 3. Tuned hyperparameters for RF, XGBoost, and LightGBM

Hyperparameter	All data	Downsampled data	PCA processed data
	RF model		
*n*_estimator	200	300	100
max_depth	10	10	5
max_features	sqrt	sqrt	auto
min_samples_split	7	2	3
min_samples_leaf	1	1	5
	XGBoost model		
eta	0.05	0.2	0.3
*n*_estimator	100	50	20
max_depth	4	2	2
colsample_bytree	0.5	0.5	1
alpha	1	1	0.5
	LightGBM model		
learning_rate	0.1	0.1	0.2
num_leaves	50	50	100
num_iterations	200	300	10
max_depth	2	2	8
bagging_fraction	0.5	0.5	0.5
lambda_l1	0.5	0.5	1

Table 4. Details on the optimal hyperparameters obtained using the random search algorithm for the FF-ANN model

Data	No. of hidden layers	No. of neurons	Activation functions	Learning rate	Execution time
All (96,801)	2	480 160	linear tanh	0.0001	4:23:31
Downsampled (68,417)	4	32 160 320 32	relu tanh tanh relu	0.001	3:56:30
PCA (56,033)	4	352 128 32 32	relu linear relu relu	0.0001	7:10:23

Model Comparison Based on RMSE, RSR, NSE, and R^2

The RF, LightGBM, XGBoost, and FF-ANN results are presented in Table 5. It can be observed that the RF model performs better with the downsampled data compared with other versions. The same pattern can also be observed for the LightGBM model, but XGBoost and FF-ANN models perform better with all the columns. For the LSTM models, three input sequences (12, 24, 30) and four output sequences (2, 4, 6, 8) are evaluated. The results for these configurations are shown in Fig. 9. The important results noted from the experiments are as follows:

- After comparing four models, the least RMSE (7.515) and RSR (0.771) values are noted for the LightGBM model with downsampled data.
- The LightGBM model with downsampled data indicates the highest NSE (0.413) compared with other models.
- For the test set, for 30 h input sequence, the NSE values are negative for all the output sequence hours.
- The LSTM model with a 12 h input sequence and 2 h output sequence indicates the lowest RMSE (0.036), RSR (0.276), and highest NSE (0.723) values.
- The 24 h input sequence and 2 h output sequence indicate the lowest RMSE (0.036), RSR (0.260), and highest NSE (0.739) values for this configuration.
- Overall, it can be noted that the LSTM model with a 24 h input sequence and a 2 h output sequence manifests the best performance.

In the WWTP, the sea surface level (equal to 0) is used as a reference to measure the wastewater level. The stored wastewater is collected in the underground tunnels below the sea surface level (less than 0, which makes it negative). Considering this, a threshold is selected to provide soft warning predictions to check the model's performance. For a soft warning, a threshold of –50 m (50 m below sea level; total tunnel depth is 120 m below sea level) is selected for the potential effluent overflow. Based on this threshold, the selected LSTM model correctly predicted 85% incidence of overflow in the test data set. The results for the overflow threshold are visualized and shown in Fig. 10.

Intention Classification and Attack Situation Detection

The results from the hyperparameter tuning have indicated that the LSTM model with two hidden layers, with 600 neurons and a learning rate of 0.001, provided the best configuration. This selected model is trained for 1,000 epochs with a batch size of 200 to perform two experiments. In the first experiment, intention classification is performed using the SMOD data set with four labels (i.e., normal, intentional attack, outlier event, and unknown) using two RNN models: GRU and LSTM. The SMOD data set is imbalanced data, and oversampling is performed to tackle this data quality problem. Due to this reason, the DL models are developed for both versions of the data set, i.e., oversampled and original. For model comparisons, accuracy, precision, recall, and F_1 score are calculated and presented using a bar plot, as shown in Fig. 11(a). Further, confusion matrices are calculated to compare the performance of LSTM and GRU on oversampled data, which are presented in Tables 6 and 7, respectively. The important results noted from the experiments are as follows:

- The result indicates higher values of evaluation metrics for LSTM and GRU when the data are oversampled.
- The results also show that the LSTM model performs better than the GRU model on oversampled data based on all the evaluation metrics.
- Based on the confusion matrix, the GRU model shows the maximum accuracy while predicting normal operations (99.55%) and the least accuracy (17.52%) for outlier events.
- For the LSTM model, the maximum accuracy can be noted for predicting normal operations (98.97%) and the least for outlier events (33.48%).

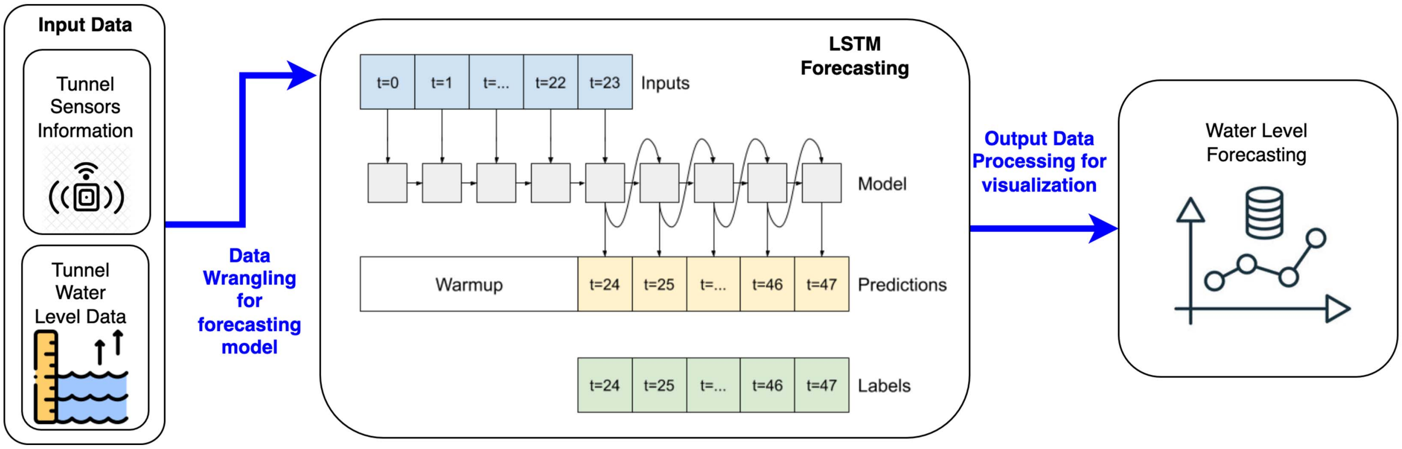


Fig. 8. Architecture of LSTM used for wastewater-level predictions.

Table 5. Comparison of RF, LightGBM, XGBoost, and FF-ANN using RMSE, RSR, NSE, and R^2

Model	Data	RMSE	RSR	NSE	R^2
Random forest	All	7.628	0.784	0.385	0.41
	Downsampled	7.577	0.774	0.395	0.43
	PCA	7.857	0.807	0.347	0.36
LightGBM	All	7.548	0.775	0.398	0.42
	Downsampled	7.515	0.771	0.405	0.42
	PCA	7.824	0.771	0.405	0.01
XGBoost	All	7.450	0.765	0.413	0.40
	Downsampled	7.615	0.781	0.389	0.39
	PCA	7.984	0.820	0.326	0.37
FF-ANN	All	8.195	0.842	0.290	0.33
	Downsampled	8.228	0.844	0.287	0.29

- For outlier events, the GRU model classifies most labels as unknown (43.94%), and a similar pattern can be observed for the LSTM model, which also classifies most labels as unknown (31.13%).
- Overall, the GRU model can classify about 97% of the intentional attacks, while LSTM can classify about 95% correctly.

The second experiment is performed to classify 15 attack situations using LSTM and GRU models. This experiment also uses oversampled and normal versions of the data for model development. The results for both models are calculated and shown using a bar plot in Fig. 11(b). Further, confusion matrices are calculated to compare the performance of LSTM and GRU models on oversampled data and are presented in the Supplemental Materials (Figs. S3 and S4). The important results noted from the experiments are as follows:

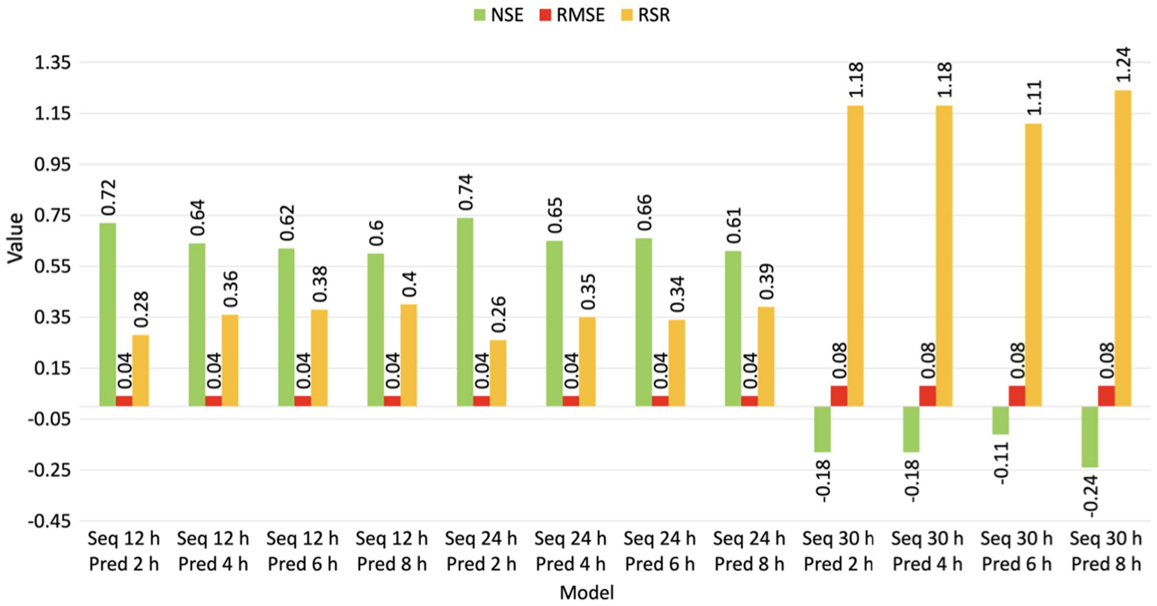


Fig. 9. LSTM model with a 24 h input sequence and a 2 h output sequence shows the best performance on the test data set.

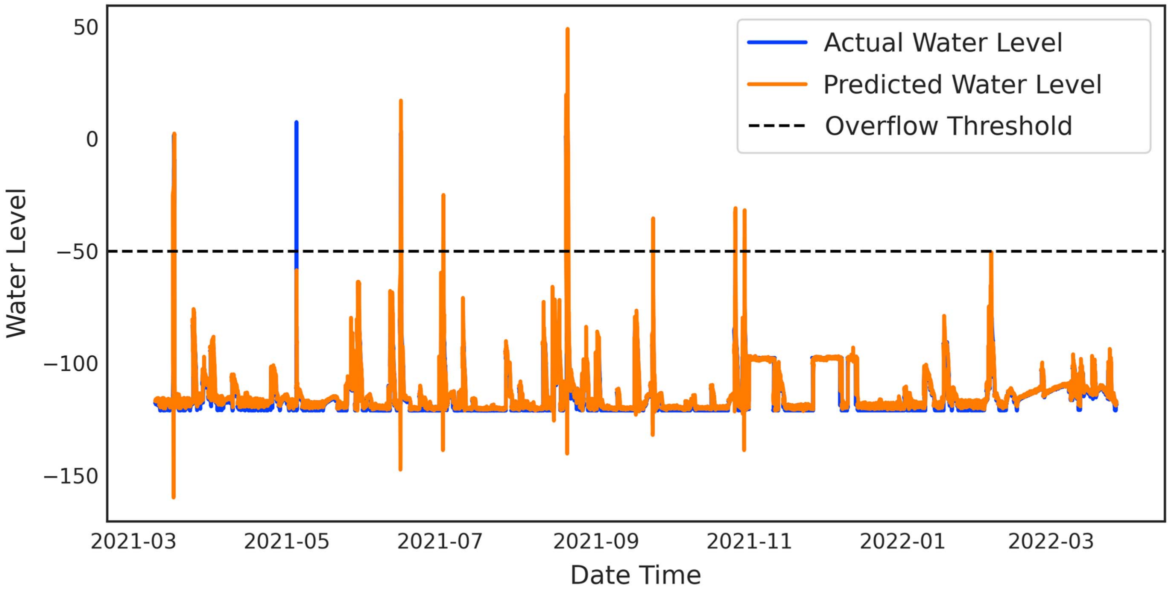


Fig. 10. LSTM model (24 h input sequence and 2 h output) prediction on test data set with –50 m (85% accuracy at 50 m below sea level) as the peak threshold.

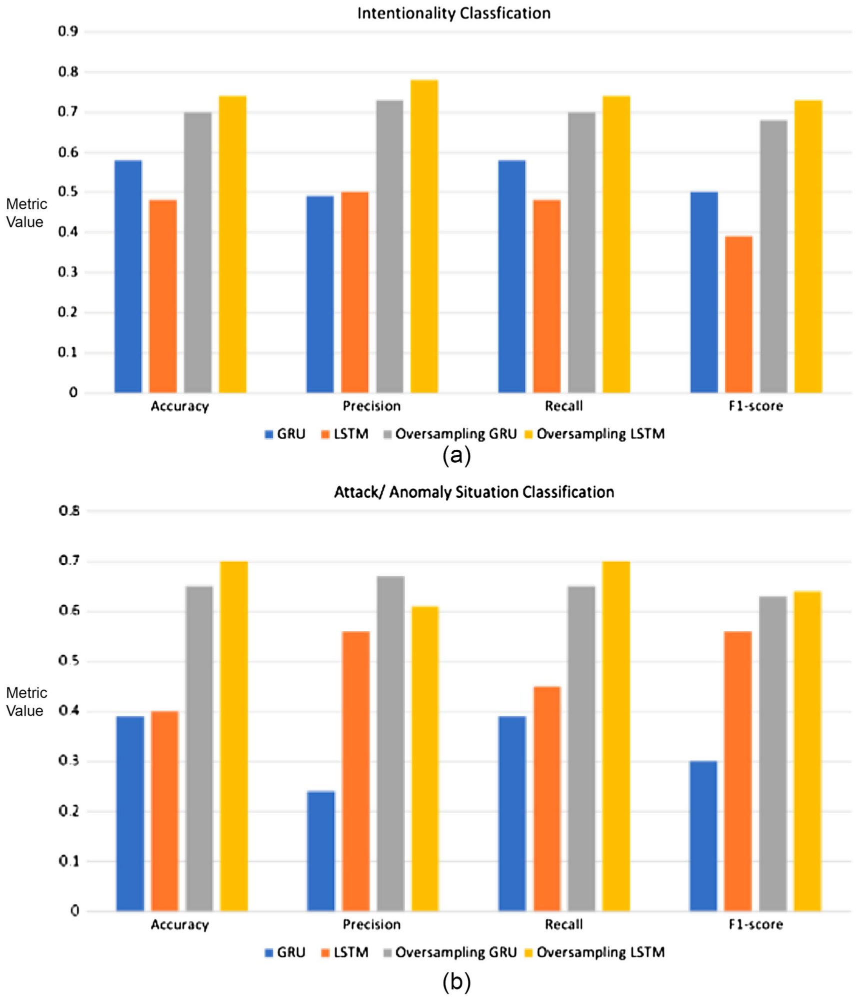


Fig. 11. (a) LSTM model performs better than the GRU model for the intention classification. (b) LSTM model performs better than the GRU model for the attack situation classification.

Table 6. Confusion matrix for the LSTM model indicates the maximum accuracy for predicting normal operations (100%) and the least for outlier events (13.44%)

	Predicted			
Actual	Normal (%)	Intentional attack (%)	Outlier event (%)	Unknown (%)
Normal	98.97	0	1.30	0
Intentional attack	0.12	94.89	4.17	0.82
Outlier event	15.46	19.33	33.48	31.13
Unknown	23.29	1.39	3.78	71.54

Table 7. Confusion matrix for the GRU model indicates the maximum accuracy while predicting normal operations (99.57%) and the least accuracy (2.59%) for outlier events

	Predicted			
Actual	Normal (%)	Intentional attack (%)	Outlier event (%)	Unknown (%)
Normal	99.57	0	0.30	0.15
Intentional attack	0.21	96.74	0.53	2.52
Outlier event	14.49	24.05	17.52	43.94
Unknown	23.44	2.45	5.03	69.07

- The bar plot in Fig. 11(b) indicates higher values of all evaluation metrics for LSTM and GRU models for the oversampled version of the data, which matches the pattern also observed in the previous experiment.
- The results show that the LSTM model performs better than the GRU model based on accuracy, recall, and F1-score metrics, while the GRU model performs better considering the precision metric.
- The confusion matrix for the GRU model denotes low accuracy for most attack situations (11 out of 15), while the highest accuracy is noted for the Blocked Measure 1 situation.
- For the LSTM model, the confusion matrix denotes only four attack situations with low accuracy, and the highest accuracy is noted for six attack situations: Blocked Measure 1; Blocked Measure 2; denial of service attack; Hits 1; Hits 2; and Hits 3.
- The GRU and LSTM show similar performance in classifying the normal operation scenarios.
- Overall, the LSTM model also performs better classifying spoofing (89.03%) and Poly 7 (86.05%) scenarios than the GRU model.

Results: Optimization Module

The predictions produced by the LSTM model with a 24 h input and a 2 h output sequences are used as inputs in the optimization module. The optimization module generated the optimal pump operation times based on the input predictions, pump specifications, and the safety threshold. Also, the prediction inputs were perturbed by 5% for testing the optimization module. Next, the GA model was executed 100 times, resulting in less than 3% deviation from the optimum after 50 generations with an initial solution population of 50. After 100 generations, the deviation from the optimal value becomes less than 1%. The simulation showed that following the actions suggested by the GA model reduced the amount of influent directed to the wet-weather treatment plant by 23% as opposed to the status quo. Additionally, it prevented any overflow incident under extreme wet weather conditions for five years of data over 100 iterations with 5% perturbation each.

Discussion

AI for Wastewater-Level Prediction

The experiments for the prediction module focus on developing and comparing ML and DL models to select the most accurate model for predicting tunnel water levels at WWTPs. The results indicate that the selected ML models (RF, LightGBM, and XGBoost) could not perform well in the prediction task. The literature has indicated that a model with an RSR value <0.7 is considered satisfactory; if the NSE reaches 1, then the model is considered a better fit (Moriasi et al. 2007; Bennett et al. 2013). Considering these criteria, ML models and FF-ANN fail to consider satisfactory or good models for predicting tunnel water levels. The unsatisfactory performance is due to the lack of predictive power in predicting the peaks, i.e., overflow of the wastewater from the tunnel. The data set indicates 262 incidences of water overflow (223 in the training set and 39 in the test set) due to which these models could not learn these patterns from the data. Further, the time-series aspect of the data was also not included during the development of these models, which may also have affected the prediction ability of the models. For the LSTM models, all the configurations show RSR values <0.7, and most also show higher NSE values than tree-based and FF-ANN models. In the LSTM models with a 30 h input sequence, the NSE values are negative, indicating they are the worst fit for the prediction task. If only the NSE values are considered, overall, the LSTM models with a 30 h input sequence indicate the worst performance. Considering this, the LSTM model with a 24 h input sequence and a 2 h output sequence is the best model and can capture 85% incidence of overflow in the test data set. This answers RQ1 and proves H_1, which states that the accuracy of predicting the wastewater level using a DL model is higher than an ML model. These experiments also highlight the importance of hyperparameter tuning, especially while using DL models. The top-five variables that affect the selected LSTM model's predictions are tunnel water level (30%), water flow sensor (18%), total flow (12%), Pump 5 (11%), and treatment flow (10%). These insights and predictions can help a WWTP devise action plans to promote the desired operational outcomes.

AI for Detecting Security Threats at WWTP

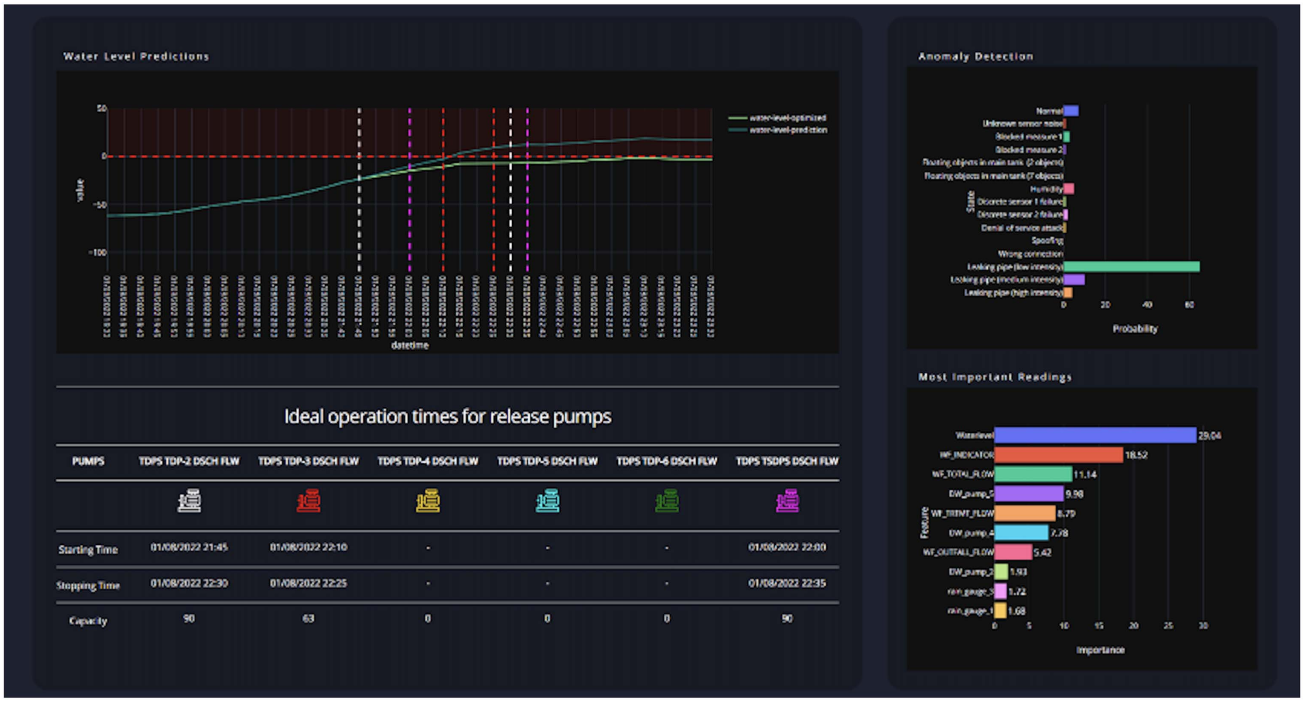
RQ2 is answered by exploring the capability of AI in detecting security threats at a WWTP. The literature (Hassanzadeh et al. 2020) has already exposed how frequently attacks occur at WWTPs, primarily when an attacker attacks the sensors and manipulates the data (Tuptuk et al. 2021). Thus, to explore the capability of AI for detecting security threats, two experiments, i.e., intention classification and attack situation detection, are performed using the SMOD data set. The results for both experiments are mentioned in the previous section. The intention classification experiment helps to classify whether the occurred situation is normal, an outlier event, an intentional attack, or something else, i.e., unknown. Overall, the results of this experiment indicate that the LSTM model performs better than the GRU model, considering higher accuracy, recall, and F-1 score. Considering these values, one downside of the LSTM model is the misclassification of intentional attacks as outlier events. The LSTM model misclassifies about 4% of intentional attacks as outlier events, while the GRU model misclassifies about 0.5% of intentional attacks as outlier events. Overall, the GRU model is more accurate (96.74%) when classifying intentional attacks indicating the most accurate for detecting cyber-related adversaries.

The second experiment's results highlight the DL models' capability to detect attack types in WWTPs. It can be inferred that the LSTM model performs more accurately than the GRU model. The LSTM model can detect six attack scenarios correctly, i.e., with 100% accuracy, and detect normal operations with about 99% accuracy. For misclassification, the GRU model misclassified two attacks, i.e., high blocked (0.02%) and Hits 1 (0.04%), as normal operations, but the LSTM misclassifies three attacks, i.e., high blocked (0.02%), Hits 1 (0.04%), and spoofing (0.13%), as normal operations. This highlights an important point: the GRU model misclassifies lesser attacks as normal operations, but overall accuracy is lesser than the LSTM model. Considering this, for both experiments, the LSTM model indicated higher accuracy than the GRU model, proving H_2.

The EPA establishes policies and thresholds for WWTP's effluent discharge, which, if not followed, these plants will get penalized under the Clean Water Act (Robison 1991). Let's take an example of hospital wastewater, which has a higher concentration of pharmaceuticals (antibiotics and heavy metals) and pathogen counts, which increases the dangerous substance concentration in related urban effluent (Khan et al. 2020, 2021). The processes in a WWTP, such as aeration, chemical coagulation, activated sludge process, trickling filters, and rotations biological contactors, affect the degree of microbe destruction (Luo et al. 2014). Thus, if an attacker compromises one of these processes, it may affect the quality of the wastewater treatment and, therefore, the environment (which causes EPA penalties, too). WWTPs must detect intentional attacks more accurately because the cost of an undetected intentional attack is higher and more disastrous than misclassifying a normal operation as an attack (i.e., false alarm). In this case, the pump operators and process engineers should be on high alert and then take substantial action to halt some pats of the system. The GRU model presented, however, can help with detecting such incidents and minimizing such related misrepresentations.

GA for Actionable Recommendation

One of the problems faced in WWTPs is to decide how much of the influent will be treated in a wet-weather treatment plant rather than the complete chemical treatment during extreme weather events. The wet-weather treatment plant uses many chemicals to treat wastewater (Reardon 2005). This process is much faster than the complete treatment process, but it is also more costly, labor intensive, and produces a suboptimal effluent quality (Throneburg et al. 2014). Using a wet-weather treatment process is still necessary to handle the large amounts of wastewater incoming to the facility during these disaster scenarios without any overflows (Peters and Zitomer 2021). The results of the optimization module indicate the effectiveness of the proposed model for optimal operation. This is partially due to the real-time updates of the proposed



Ideal operation times for release pumps

PUMPS	TDPS TDP-2 DSCH FLW	TDPS TDP-3 DSCH FLW	TDPS TDP-4 DSCH FLW	TDPS TDP-5 DSCH FLW	TDPS TDP-6 DSCH FLW	TDPS TSDPS DSCH FLW
Starting Time	01/08/2022 21:45	01/08/2022 22:10	-	-	-	01/08/2022 22:00
Stopping Time	01/08/2022 22:30	01/08/2022 22:25	-	-	-	01/08/2022 22:35
Capacity	90	63	0	0	0	90

Fig. 12. Graphical user interface of P_2O that provides insights needed for decision-making. (Reproduced from Batarseh et al. 2022, with permission.)

model, and, as time passes, i.e., as more data are collected, the results can keep improving.

Decision-Making Using P_2O

P_2O provides a web-based real-time monitoring interface (best viewed in colors), as shown in Fig. 12. It provides real-time insights into prediction, protection, and optimization. The first plot on the top left indicates wastewater level in the tunnel. In this plot, the horizontal dashed line is sea level, the top line represents the water-level prediction (from the prediction module), and the bottom line represents the simulated water level after following the optimal actions recommended by the optimization module. The target safety level decided by a WWTP is the sea surface level (dotted line), and P_2O aims to prevent rising the wastewater level above this defined level. Thus, the red zone above the dotted line represents this risk associated with operating above the level. The vertical dashed line of different colors indicates the optimal pump operation markers (six pumps), indicating the start and stop times for running the pumps. These details are presented at the bottom right, indicating the time and capacity of the six pumps used in the WWTP. The plot on the top right indicates the percentage of different operational scenarios based on the SMOD data set, which may occur during everyday operations. This plot includes 15 operational situations, and the percentage for every situation is calculated by the protection module. Details of the operational situations can change based on the data collection from the WWTPs. The plot in the lower right corner details the important sensors or variables that affect the wastewater-level predictions. These insights are also derived from the prediction module and explained in terms of their importance. In this way, the pump operators can use these insights to support their decisions by using P_2O as a decision-support solution.

Conclusion and Future Work

The framework presented in this paper explores AI's role in preventing wastewater overflow and in detecting security threats. To achieve these objectives, P_2O is proposed and developed. Three decision-tree-based (RF, LightGBM, and XGBoost) and two NN-based (FF-ANN and LSTM) models were developed to constitute a prediction module in P_2O. The results showed that the LSTM model predicts tunnel water levels better than the other AI models used in the experiments. The LSTM model with a 24 h input sequence with a 2 h output sequence is selected as the best model for the protection module based on RMSE (0.036), RSR (0.260), and NSE (0.739) evaluation metrics. SHAP analysis is also performed, which revealed that the top-five important variables that affect the prediction the most are water level sensor data, overflow indicator sensor, total water flow sensor, pump five, and wastewater treatment flow sensor. Further, the SMOD data set is used to develop the P_2O's protection module for detecting security threats at WWTPs. For this purpose, two experiments focused on intention classification and attack situation detection are performed. These experiments are executed using LSTM and GRU models. For the intention classification, the LSTM model showed 94% accuracy, while the GRU model showed 96% accuracy in identifying intentional attacks. Further, the LSTM model misclassifies about 4% of intentional attacks as outlier events, but the misclassification rate for the GRU model is only about 0.5%. The LSTM model misclassified three attack scenarios for attack situation detection as normal operations, while the GRU model misclassified only two attacks as normal operations. These results revealed that the LSTM model showed higher misclassification than the GRU model. These experiments conclude that the GRU model is the best suitable for detecting security threats considering the accuracy and severity of not detecting an attack at WWTP. Finally, the simulation results of the optimization module indicate a reduction in the amount of influent directed to the wet-weather treatment plant by 23% while preventing overflow incidents under extremely wet weather conditions based on five years of data.

In the future, we would like to focus on three objectives for improving the framework: context; AI assurance; and attention-based modeling. In the first objective, we would like to understand the effect of the utilization of weather variables (e.g., snow, air temperature, humidity) and demographic data on the models, as a "context" for improved water-level predictions. In the second objective, we would like to evaluate the AI models further against implicit bias and cyberattacks with minimum perturbations such as adversarial networks, especially via threat-detection solutions. Finally, in the third objective, we would like to use an attention-based model to understand the effect of existing and new variables on water-level predictions, especially for predictions during wet seasons.

Data Availability Statement

The code for P_2O will be made available based on requests to the corresponding author. The SWaT (Goh et al. 2016) (https://itrust.sutd.edu.sg/itrust-labs_datasets/) and SMOD (Laso et al. 2017) are open-source data sets that can be obtained by contacting the original owners of the data. The third data set cannot be shared due to a nondisclosure agreement with the WWTP.

Notation

The following symbols are used in this paper:

A_c = accuracy of detecting cyberattacks;
B = number of trees;
C = pump's operation capacity;
d = danger level threshold;
e_m = error;
$\hat{F}$ = ensembled model;
$\hat{F}^B_{rf}$ = decision tree model;
$g(..)$ = activation function;
$I(l)$ = water level to mass conversion interpolation function;
k = hidden layers;
n = total data points in the data set;
P = pump;
p_i = predicted value;
$\bar{p}$ = average of predicted values;
R = pump's run time;
S = pump's start time;
$T(x;\theta_b)$ = forest of b trees;
w = weight values;
wl_p = wastewater-level prediction;
x = input parameter to the model;
y_i = observed value;
$\bar{y}$ = average of observed values;
α = weight;
β = bias; and
∞ = infinity.

Supplemental Materials

There are Supplemental Materials associated with this paper online in the ASCE Library (www.ascelibrary.org).

References

Abdi, H., and L. J. Williams. 2010. "Principal component analysis." *Wiley Interdiscip. Rev. Comput. Stat.* 2 (4): 433–459. https://doi.org/10.1002/wics.101.

Adadi, A., and M. Berrada. 2018. "Peeking inside the black-box: A survey on explainable artificial intelligence (XAI)." *IEEE Access* 6 (Sep): 52138–52160. https://doi.org/10.1109/ACCESS.2018.2870052.

Adepu, S., and A. Mathur. 2016a. "Introducing cyber security at the design stage of public infrastructures: A procedure and case study." In *Complex systems design & management Asia*, 75–94. Cham, Switzerland: Springer.

Adepu, S., and A. Mathur. 2016b. "An investigation into the response of a water treatment system to cyber attacks." In *Proc., 2016 IEEE 17th Int. Symp. on High Assurance Systems Engineering (HASE)*, 141–148. New York: IEEE.

Adepu, S., and A. Mathur. 2018. "Distributed attack detection in a water treatment plant: Method and case study." *IEEE Trans. Dependable Secure Comput.* 18 (1): 86–99. https://doi.org/10.1109/TDSC.2018.2875008.

Alanazi, M., A. Mahmood, and M. J. M. Chowdhury. 2022. "SCADA vulnerabilities and attacks: A review of the state-of-the-art and open issues." *Comput. Secur.* 125 (Feb): 103028.

Albahar, M. A., R. A. Al-Falluji, and M. Binsawad. 2020. "An empirical comparison on malicious activity detection using different neural network-based models." *IEEE Access* 8 (Mar): 61549–61564. https://doi.org/10.1109/ACCESS.2020.2984157.

Alin, A. 2010. "Multicollinearity." *Wiley Interdiscip. Rev. Comput. Stat.* 2 (3): 370–374. https://doi.org/10.1002/wics.84.

Ardabili, S., A. Mosavi, M. Dehghani, and A. R. Várkonyi-Kóczy. 2020. "Deep learning and machine learning in hydrological processes climate change and earth systems a systematic review." In Vol. 101 of *Engineering for sustainable future. INTER-ACADEMIA 2019. Lecture notes in networks and systems*, edited by A. Várkonyi-Kóczy. Berlin: Springer.

Asadollahfardi, G., M. Afsharnasab, M. H. Rasoulifard, and M. Tayebi Jebeli. 2022. "Predicting of acid red 14 removals from synthetic wastewater in the advanced oxidation process using artificial neural networks and fuzzy regression." *Rend. Lincei Sci. Fis. Nat.* 33 (1): 115–126. https://doi.org/10.1007/s12210-021-01043-8.

Asadollahfardi, G., H. Zangooi, M. Asadi, M. Tayebi Jebeli, M. Meshkat-Dini, and N. Roohani. 2018. "Comparison of box-Jenkins time series and ANN in predicting total dissolved solid at the Zāyandé-Rūd River, Iran." *J. Water Supply Res. Technol. AQUA* 67 (7): 673–684. https://doi.org/10.2166/aqua.2018.108.

Batarseh, F. A., L. Freeman, and C.-H. Huang. 2021. "A survey on artificial intelligence assurance." *J. Big Data* 8 (1): 1–30. https://doi.org/10.1186/s40537-021-00445-7.

Batarseh, F. A., M. O. Yardimci, R. Suzuki, M. N. K. Sikder, Z. Wang, and W. Mao. 2022. "Realtime management of wastewater treatment plants using AI." Accessed March 15, 2023. https://www.waterrf.org/news/2022-intelligent-water-systems-challenge.

Bennett, N. D., et al. 2013. "Characterising performance of environmental models." *Environ. Modell. Software* 40 (Feb): 1–20. https://doi.org/10.1016/j.envsoft.2012.09.011.

Bergstra, J., and Y. Bengio. 2012. "Random search for hyper-parameter optimization." *J. Mach. Learn. Res.* 13 (2): 281–305. https://doi.org/10.5555/2503308.2188395.

Biewald, L. 2020. "Experiment tracking with weights and biases." Accessed March 2, 2023. https://www.wandb.com/.

Bruce, P., A. Bruce, and P. Gedeck. 2020. *Practical statistics for data scientists: 50+ essential concepts using R and Python*. Sebastopol, CA: O'Reilly Media.

Chai, T., and R. R. Draxler. 2014. "Root mean square error (RMSE) or mean absolute error (MAE)?—Arguments against avoiding RMSE in the literature." *Geosci. Model Dev.* 7 (3): 1247–1250. https://doi.org/10.5194/gmd-7-1247-2014.

Charu, C. A. 2018. *Neural networks and deep learning: A textbook*. New York: Springer.

Chawla, N. V., K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. 2002. "Smote: Synthetic minority over-sampling technique." *J. Artif. Intell. Res.* 16 (Jun): 321–357. https://doi.org/10.1613/jair.953.

Chicano, F., A. M. Sutton, L. D. Whitley, and E. Alba. 2015. "Fitness probability distribution of bit-flip mutation." *Evol. Comput.* 23 (2): 217–248. https://doi.org/10.1162/EVCO_a_00130.

Collier, K. 2021. "50,000 security disasters waiting to happen: The problem of America's water supplies." Accessed March 2, 2023. https://www.nbcnews.com/tech/security/hacker-tried-poison-calif-water-supply-was-easyentering-password-rcna1206.

Corominas, L., M. Garrido-Baserba, K. Villez, G. Olsson, U. Cortés, and M. Poch. 2018. "Transforming data into knowledge for improved wastewater treatment operation: A critical review of techniques." *Environ. Modell. Software* 106 (Aug): 89–103. https://doi.org/10.1016/j.envsoft.2017.11.023.

Das, B., and D. Chakrabarty. 2016. "Newton's divided difference interpolation formula: Representation of numerical data by a polynomial curve." *Int. J. Math. Trend Technol.* 35 (3): 197–203.

Dokeroglu, T., E. Sevinc, T. Kucukyilmaz, and A. Cosar. 2019. "A survey on new generation metaheuristic algorithms." *Comput. Ind. Eng.* 137 (Nov): 10–40. https://doi.org/10.1016/j.cie.2019.106040.

Doshi-Velez, F., and B. Kim. 2017. "Towards a rigorous science of interpretable machine learning." Preprint, submitted February 28, 2017. https://arxiv.org/abs/1702.08608.

Faramondi, L., F. Flammini, S. Guarino, and R. Setola. 2021. "A hardware-in-the-loop water distribution testbed dataset for cyber-physical security testing." *IEEE Access* 9 (Aug): 122385–122396. https://doi.org/10.1109/ACCESS.2021.3109465.

Fernández, A., S. Garcia, F. Herrera, and N. V. Chawla. 2018. "Smote for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary." *J. Artif. Intell. Res.* 61 (Apr): 863–905. https://doi.org/10.1613/jair.1.11192.

Feurer, M., and F. Hutter. 2019. "Hyperparameter optimization." In *Automated machine learning. The springer series on challenges in machine learning*, edited by F. Hutter, L. Kotthoff, and J. Vanschoren. Berlin: Springer.

Flynn, M. J. 2020. "Civilians 'defending forward' in cyberspace." *Cyber Defense Rev.* 5 (1): 29–40.

Forest, J. J. F. 2006. Vol. 3 of *Homeland security: Critical infrastructure*. Westport, CT: Greenwood Publishing Group.

Gilpin, L. H., D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal. 2018. "Explaining explanations: An overview of interpretability of machine learning." In *Proc., 2018 IEEE 5th Int. Conf. on data science and advanced analytics (DSAA)*, 80–89. New York: IEEE.

Goh, J., S. Adepu, K. N. Junejo, and A. Mathur. 2016. "A dataset to support research in the design of secure water treatment systems." In *Proc., Int. Conf. on Critical Information Infrastructures Security*, 88–99. Berlin: Springer.

Hassanzadeh, A., A. Rasekh, S. Galelli, M. Aghashahi, R. Taormina, A. Ostfeld, and M. K. Banks. 2020. "A review of cybersecurity incidents in the water sector." *J. Environ. Eng.* 146 (5): 03120003. https://doi.org/10.1061/(ASCE)EE.1943-7870.0001686.

Hettiarachchi, H., and T. Ranasinghe. 2019. "Emoji powered capsule network to detect type and target of offensive posts in social media." In *Proc., Int. Conf. on Recent Advances in Natural Language Processing (RANLP 2019)*, 474–480. Varna, Bulgaria: INCOMA Ltd.

Hindy, H., D. Brosset, E. Bayne, A. Seeam, and X. Bellekens. 2019. "Improving SIEM for critical SCADA water infrastructures using machine learning." In *Proc., Computer Security: ESORICS 2018 Int. Workshops, CyberICPS 2018 and SECPRE 2018, Barcelona, Spain, September 6–7, 2018, Revised Selected Papers 2*, 3–19. Berlin: Springer. https://doi.org/10.1007/978-3-030-12786-2.

Hingston, P. F., L. C. Barone, and Z. Michalewicz. 2008. *Design by evolution: Advances in evolutionary design*. Berlin: Springer.

Hyndman, R. J., and G. Athanasopoulos. 2018. *Forecasting: Principles and practice*. Melbourne, Australia: OTexts.

Ilyas, A., L. Engstrom, A. Athalye, and J. Lin. 2018. "Black-box adversarial attacks with limited queries and information." In Vol. 80 of *Proc., Int. Conf. on Machine Learning, in Proceedings of Machine Learning Research*, 2137–2146. Washington, DC: Machine Learning Research.

James, G., D. Witten, T. Hastie, and R. Tibshirani. 2013. Vol. 112 of *An introduction to statistical learning*. Boston: Springer.

Kabir Sikder, M. N., F. A. Batarseh, P. Wang, and N. Gorentala. 2022. "Model-agnostic scoring methods for artificial intelligence assurance." In *Proc., 2022 IEEE 29th Annual Software Technology Conf. (STC)*, 9–18. New York: IEEE. https://doi.org/10.1109/STC55697.2022.00011.

Katoch, S., S. S. Chauhan, and V. Kumar. 2021. "A review on genetic algorithm: Past, present, and future." *Multimedia Tools Appl.* 80 (5): 8091–8126. https://doi.org/10.1007/s11042-020-10139-6.

Khan, N. A., S. U. Khan, S. Ahmed, I. H. Farooqi, M. Yousefi, A. A. Mohammadi, and F. Changani. 2020. "Recent trends in disposal and treatment technologies of emerging-pollutants-a critical review." *TRAC Trends Anal. Chem.* 122 (Jan): 115744. https://doi.org/10.1016/j.trac.2019.115744.

Khan, N. A., V. Vambol, S. Vambol, B. Bolibrukh, M. Sillanpaa, F. Changani, A. Esrafili, and M. Yousefi. 2021. "Hospital effluent guidelines and legislation scenario around the globe: A critical review." *J. Environ. Chem. Eng.* 9 (5): 105874. https://doi.org/10.1016/j.jece.2021.105874.

Kochenderfer, M. J., and T. A. Wheeler. 2019. *Algorithms for optimization*. Cambridge, MA: MIT Press.

Komorowski, M., D. C. Marshall, J. D. Salciccioli, and Y. Crutain. 2016. "Exploratory data analysis." In *Secondary analysis of electronic health records*, 185–203. Cham, Switzerland: Springer.

Kuhn, M., and K. Johnson. 2013. Vol. 26 of *Applied predictive modeling*. New York: Springer.

Kulkarni, A., D. Chong, and F. A. Batarseh. 2020. "Foundations of data imbalance and solutions for a data democracy." In *Data democracy*, 83–106. Amsterdam, Netherlands: Elsevier.

Laso, P. M., D. Brosset, and J. Puentes. 2017. "Dataset of anomalies and malicious acts in a cyber-physical subsystem." *Data Brief* 14 (Oct): 186–191. https://doi.org/10.1016/j.dib.2017.07.038.

LeCun, Y., Y. Bengio, and G. Hinton. 2015. "Deep learning." *Nature* 521 (7553): 436–444. https://doi.org/10.1038/nature14539.

Li, J.-H. 2018. "Cyber security meets artificial intelligence: A survey." *Front. Inf. Technol. Electron. Eng.* 19 (12): 1462–1474. https://doi.org/10.1631/FITEE.1800573.

Lim, B., and S. Zohren. 2021. "Time-series forecasting with deep learning: A survey." *Philos. Trans. R. Soc. A* 379 (2194): 202–209.

Lunardi, A. 2018. "Real interpolation." In Vol. 16 of *Interpolation theory. CRM series*. Berlin: Springer.

Lundberg, S. M., and S.-I. Lee. 2017. "A unified approach to interpreting model predictions." In *Advances in neural information processing systems*, 30. San Mateo, CA: Morgan Kaufmann Publishers.

Luo, Y., W. Guo, H. H. Ngo, L. D. Nghiem, F. I. Hai, J. Zhang, S. Liang, and X. C. Wang. 2014. "A review on the occurrence of micropollutants in the aquatic environment and their fate and removal during wastewater treatment." *Sci. Total Environ.* 473 (Mar): 619–641. https://doi.org/10.1016/j.scitotenv.2013.12.065.

Miller, T., A. Staves, S. Maesschalck, M. Sturdee, and B. Green. 2021. "Looking back to look forward: Lessons learnt from cyber-attacks on industrial control systems." *Int. J. Crit. Infrastruct. Prot.* 35 (Dec): 100464. https://doi.org/10.1016/j.ijcip.2021.100464.

Moradbeikie, A., K. Jamshidi, A. Bohlooli, J. Garcia, and X. Masip-Bruin. 2020. "An IIOT based ICS to improve safety through fast and accurate hazard detection and differentiation." *IEEE Access* 8 (Nov): 206942–206957. https://doi.org/10.1109/ACCESS.2020.3037093.

Moriasi, D. N., J. G. Arnold, M. W. Van Liew, R. L. Bingner, R. D. Harmel, and T. L. Veith. 2007. "Model evaluation guidelines for systematic quantification of accuracy in watershed simulations." *Trans. ASABE* 50 (3): 885–900. https://doi.org/10.13031/2013.23153.

Owolabi, T. A., S. R. Mohandes, and T. Zayed. 2022. "Investigating the impact of sewer overflow on the environment: A comprehensive literature review paper." *J. Environ. Manage.* 301 (Jan): 113810. https://doi.org/10.1016/j.jenvman.2021.113810.

Park, J., W. H. Lee, K. T. Kim, C. Y. Park, S. Lee, and T.-Y. Heo. 2022. "Interpretation of ensemble learning to predict water quality using explainable artificial intelligence." *Sci. Total Environ.* 832 (Aug): 155070. https://doi.org/10.1016/j.scitotenv.2022.155070.

Perrone, P., F. Flammini, and R. Setola. 2021. “Machine learning for threat recognition in critical cyber-physical systems.” In *Proc., 2021 IEEE Int. Conf. on Cyber Security and Resilience (CSR)*, 298–303. New York: IEEE.

Peters, P. E., and D. H. Zitomer. 2021. “Current and future approaches to wet weather flow management: A review.” *Water Environ. Res.* 93 (8): 1179–1193. https://doi.org/10.1002/wer.1506.

Picek, S., M. Golub, and D. Jakobovic. 2012. “Evaluation of crossover operator performance in genetic algorithms with binary representation.” In *Bio-iInspired computing and applications. ICIC 2011. Lecture notes in computer science*, edited by D. S. Huang, Y. Gan, P. Premaratne, and K. Han, 223–230. Berlin: Springer.

Radanliev, P., D. De Roure, M. Van Kleek, O. Santos, and U. Ani. 2021. “Artificial intelligence in cyber physical systems.” *AI Soc.* 36 (3): 783–796. https://doi.org/10.1007/s00146-020-01049-0.

Rahnama, E., O. Bazrafshan, and G. Asadollahfardi. 2020. “Application of data-driven methods to predict the sodium adsorption rate (SAR) in different climates in Iran.” *Arabian J. Geosci.* 13 (21): 1–19. https://doi.org/10.1007/s12517-020-06146-4.

Ratner, B. 2009. “The correlation coefficient: Its values range between + 1/- 1, or do they?” *J. Targeting Meas. Anal. Mark.* 17 (2): 139–142. https://doi.org/10.1057/jt.2009.5.

Reardon, R. D. 2005. “Clarification concepts for treating peak wet weather wastewater flows.” In *Proc., WEFTEC 2005*, 4431–4444. Clermont, FL: Florida Water Resources Journal.

Robison, M. 1991. “National pollutant discharge elimination system (NPDES) permit application requirement for storm water discharges.” In *Army environmental hygiene agency aberdeen proving ground MD*. Washington, DC: USEPA.

Sahu, A., Z. Mao, P. Wlazlo, H. Huang, K. Davis, A. Goulart, and S. Zonouz. 2021. “Multi-source multi-domain data fusion for cyberattack detection in power systems.” *IEEE Access* 9 (Aug): 119118–119138. https://doi.org/10.1109/ACCESS.2021.3106873.

Sanders, K. T., and M. E. Webber. 2012. “Evaluating the energy consumed for water use in the united states.” *Environ. Res. Lett.* 7 (3): 034034. https://doi.org/10.1088/1748-9326/7/3/034034.

Schütze, M., A. Campisano, H. Colas, W. Schilling, and P. A. Vanrolleghem. 2002. “Real-time control of urban wastewater systems-where do we stand today?” In *Global solutions for urban drainage*, 1–17. Reston, VA: ASCE.

Sojobi, A. O., and T. Zayed. 2022. “Impact of sewer overflow on public health: A comprehensive scientometric analysis and systematic review.” *Environ. Res.* 203 (Jan): 111609. https://doi.org/10.1016/j.envres.2021.111609.

Thompson, N. C., K. Greenewald, K. Lee, and G. F. Manso. 2020. “The computational limits of deep learning.” Preprint, submitted July 10, 2020. https://arxiv.org/abs/2007.05558.

Throneburg, M., P. Amico, and M. Labitzke. 2014. “An optimization planning framework for cost-effective wet-weather planning.” In *Proc., Collection Systems Conf. 2014*. Richmond, VA: Water Environment Federation.

Tuptuk, N., P. Hazell, J. Watson, and S. Hailes. 2021. “A systematic review of the state of cyber-security in water systems.” *Water* 13 (1): 81. https://doi.org/10.3390/w13010081.

Wang, Z., H. Song, D. W. Watkins, K. G. Ong, P. Xue, Q. Yang, and X. Shi. 2015. “Cyber-physical systems for water sustainability: Challenges and opportunities.” *IEEE Commun. Mag.* 53 (5): 216–222. https://doi.org/10.1109/MCOM.2015.7105668.

Willemain, T. R. 2013. “Practical time series forecasting: A hands-on guide, by Galit Shmueli.” *Foresight: Int. J. Appl. Forecasting* 1 (29): 43–44.

Yu, T., and H. Zhu. 2020. “Hyper-parameter optimization: A review of algorithms and applications.” Preprint, submitted March 12, 2020. https://arxiv.org/abs/2003.05689.

Yu, Y., X. Si, C. Hu, and J. Zhang. 2019. “A review of recurrent neural networks: LSTM cells and network architectures.” *Neural Comput.* 31 (7): 1235–1270. https://doi.org/10.1162/neco_a_01199.

Zanzotto, F. M. 2019. “Human-in-the-loop artificial intelligence.” *J. Artif. Intell. Res.* 64 (Feb): 243–252. https://doi.org/10.1613/jair.1.11345.

Zhong, J., X. Hu, J. Zhang, and M. Gu. 2005. “Comparison of performance between different selection strategies on simple genetic algorithms.” In Vol. 2 of *Proc., Int. Conf. on Computational Intelligence for Modelling, Control and Automation and Int. Conf. on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC’06)*, 1115–1121. New York: IEEE.