Desalination 516 (2021) 115233

Contents lists available at ScienceDirect

Desalination

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

Deep learning for pH prediction in water desalination using membrane capacitive deionization

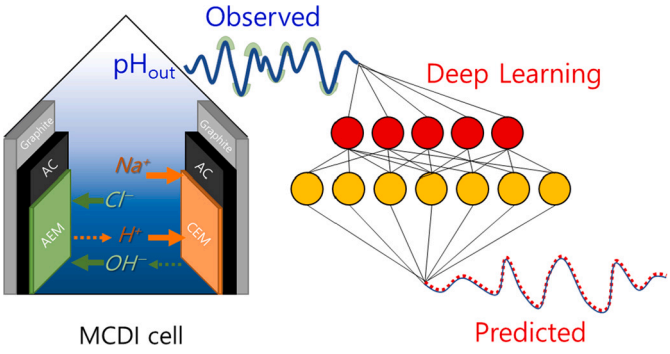
Moon Son [a], Nakyung Yoon [a], Kwanho Jeong [a], Ather Abass [a], Bruce E. Logan [b], Kyung Hwa Cho [a,*]

[a] School of Urban and Environmental Engineering, Ulsan National Institute of Science and Technology, UNIST-gil 50, Ulsan 44919, Republic of Korea
[b] Department of Civil and Environmental Engineering, The Pennsylvania State University, University Park, PA 16802, USA

HIGHLIGHTS

- A CNN-LSTM model was developed for effluent pH prediction in MCDI.
- The model accurately predicted effluent pH of the MCDI process ($R^2 \geq 0.998$).
- The voltage was the factor that most affected the effluent pH.
- Effluent pH prediction using only current and voltage variables was conducted.

GRAPHICAL ABSTRACT



ARTICLE INFO

Keywords:
Deep learning
Neural networks
Water desalination
Membrane capacitive deionization
pH

ABSTRACT

The pH of a solution has a large influence on the ion removal efficiency of the membrane capacitive deionization (MCDI) process, an electrochemical ion separation process. We developed a convolutional neural network linked with a long short-term memory (CNN-LSTM) model based on an artificial intelligence algorithm to predict the effluent pH of MCDI, as effluent pH is difficult to predict using conventional numerical modeling. The model accurately predicted effluent pH ($R^2 \geq 0.998$) based on the analysis of five input variables (current, voltage, influent conductivity and pH, and effluent conductivity) under standard operating conditions of MCDI using either constant-current or constant-voltage conditions. The developed model predicted effluent pH using only limited input variables, current and voltage, with high accuracy ($R^2 \geq 0.997$). Thus, the CNN-LSTM model can be used in practical applications as only the current and voltage of MCDI cells are often monitored in field applications.

* Corresponding author.
E-mail address: khcho@unist.ac.kr (K.H. Cho).

https://doi.org/10.1016/j.desal.2021.115233
Received 23 April 2021; Received in revised form 3 June 2021; Accepted 6 June 2021
Available online 9 July 2021

1. Introduction

Capacitive deionization (CDI), which is an electric ion separation process, has been widely used for desalination and water treatment [1–4]. The CDI process consists of repeating cycles of charging (ion adsorption) of the electrodes to remove salt and discharging (ion desorption) to release the salt into a brine [4]. Ionic substances in feed water are electrically adsorbed to the electrode used in the CDI process (primarily composed of carbon) during the charging phase, where cations are adsorbed to the cathode and anions to the anode [4]. The adsorbed ions are desorbed during the discharging phase, in which current and voltage flow in opposite directions, and the electrode is regenerated so that ions can be adsorbed on the electrode surface during the next charging phase [5]. Therefore, the amount of ion adsorption of CDI is closely related to the adsorption capacity of the electrode. Consequently, to fully utilize this capacity, the desorption of co-ions (ions of equal polarity as the electrode) must be properly excluded during discharging [6]. To address the co-ion desorption phenomenon, a membrane CDI (MCDI) process has been developed based on using an ion exchange membrane (IEM) on the electrode surface (anion exchange polymer on the anode and cation exchange polymer on the cathode) [6]. As the MCDI process can effectively eliminate the desorption of co-ions owing to the presence of IEMs, the ion adsorption efficiency of MCDI is superior to that of CDI [5].

Ion adsorption in the MCDI or CDI processes is greatly affected by the solution pH [7–9]. In particular, for substances whose dissociation forms vary depending on the pH of the solution, such as phosphorus [10], copper [11], boron [12], or carbonate forms [8], the charge form varies according to the pH of the solution, which could greatly affect the adsorption (or removal) rate. In addition, ions generated at a high operating voltage, such as H^+ and OH^-, may be adsorbed to the electrode before the target compounds, thereby lowering the removal rate of the target compounds [13,14]. Therefore, to better understand the ion removal rate of the MCDI process and optimize the process operation, the effluent pH according to the electrochemical reaction must be predicted and reflected.

Although several studies were conducted to predict the effluent concentration of MCDI (or CDI) through numerical modeling [2,15,16], studies on predicting effluent pH are limited [17]. This is likely because the factors affecting solution pH in MCDI are considerably diverse. Moreover, because the charge of electrodes and ion concentrations in the solution changes at every moment, it is difficult to predict all the variables using a conventional modeling approach. It is known that the Faradaic reactions affecting solution pH include anodic oxidations (carbon, chloride, and water oxidations) and cathodic reductions (oxygen and carbon reductions) [18]. Recently, a theory that included the Faradaic reactions, such as the reduction of water, was developed to predict the solution pH in CDI using numerical modeling [17]. This advanced numerical model for pH prediction in CDI considers the reduction of water, which could affect pH fluctuations during operation. In addition, the numerical model can provide pH information for the effluent solution, anolyte, and catholyte. However, there were still notable differences between the measured and predicted values [17], which appeared to be difficult to address using numerical approaches. For example, the predicted pH value was a maximum of pH≈7.8 during the charging process, but the measured value was pH≈8.8. Similarly, the predicted pH value during the discharging process was pH≈7.5 (on average), but the measured value was pH≈8.5 (on average) [17]. To the best of our knowledge, no studies using numerical modeling approaches have predicted the pH of MCDI effluents with a high accuracy. Therefore, to improve the accuracy of pH prediction, an alternative approach, such as artificial intelligence, which relies on the relationship between data rather than theoretical analysis was used to describe pH variations in the system.

Deep learning, which has been used with great success in a variety of application domains in the past few years, should be a more useful approach to predict effluent pH of an MCDI process because it has been shown to be useful for processing nonlinear data for water treatment [19–21]. Deep learning is a system that processes input data by weighting it through several hidden layers and outputs the resulting value [22]. Depending on the structure of these hidden layers, it can be approximately classified in different ways such as deep neural networks, recurrent neural networks (RNNs), convolutional neural networks (CNNs), long short-term memory (LSTM) [22,23]. In particular, the CNN approach demonstrates superior capabilities in processing local changes in features [24], which results in the minimization of data noise from the correlation of multiple variables. This is because, during the data processing stages of CNNs, greater weight is given to points with large inflection points, while a filter of a specific size scans the entire dataset. Moreover, LSTM is advantageous for processing time-series data because the embedded memory gates can classify the importance of data through time [25]. Based on these advantages, research has recently been conducted to process and predict multivariate time-series data by linking CNN and LSTM (CNN-LSTM) [26].

In this study, we developed a CNN-LSTM model to predict the pH of MCDI effluent using a deep learning algorithm. The MCDI process for data collection was performed for brackish water conditions (20 mM NaCl) using standard operating conditions of either constant current (CC) or constant voltage (CV) [27]. Six types of data (current, voltage, influent conductivity, influent pH, effluent conductivity, and effluent pH) were collected, and five of these (current, voltage, influent conductivity, influent pH, and effluent conductivity) were used to predict effluent pH (Fig. 1). Because current and voltage data are often monitored in practical applications, a further study was conducted to predict the effluent pH using only these two measured parameters. In addition, a sensitivity analysis was conducted by only using each input variable to identify the factors that have the greatest influence on the change in effluent pH. A comparison of the CNN-LSTM model was also made to just the LSTM model to determine the relative accuracy of the CNN-LSTM model.

2. Materials and methods

2.1. MCDI cell assembly

To assemble the electrode, spacer, and current collector, a custom-made cell with an effective area of 7 cm^2 was used (~3 cm in diameter). The two carbon electrodes used in the custom-made cell were obtained by disassembling a commercial MCDI module (E−40, Siontech Co., South Korea). Both carbon electrodes (activated carbon; P-60, Kuraray Chemical Co., Japan) were coated by ion exchange polymers, either cation ($-SO_3H$, ion exchange capacity of 1.7–1.8 meq g^{-1}) or anion polymer ($-NR_4Br$, ion exchange capacity of 1.4–1.5 meq g^{-1}) [28]. The electrode had a surface area of 1503 m^2 g^{-1} (Brunauer–Emmett–Teller surface area), a cumulative volume of 0.19 cm^3 g^{-1} (Barret–Joyner–Halenda desorption cumulative volume of pores), and an average pore diameter of 3.34 nm (Barret–Joyner–Halenda desorption average pore diameter), which were measured using the Micromeritics Tristar II 3020 V1.03 analyzer [28]. The space between the two electrodes contained a polymer spacer (250 μm thickness and 34% open area), and graphite foil (99.8% purity, LS538884, Goodfellow Cambridge Ltd., UK) was used as a current collector.

The cell was operated in up-flow mode and fed with water using silicone tubing (L/S 14, Masterflex) with 1.6 mm (inner diameter) connected to a feed tank and a gear pump (EMP–600A, EMS Tech., South Korea). All experiments were conducted using a fixed volume of 2 L, which contained a feed solution of 20 mM NaCl fed at a fixed flow rate of 4 mL min^{-1}. Considering relatively large reservoir volume and slow flow rate, the operation mode could be interpreted as a semi single-pass process [29].

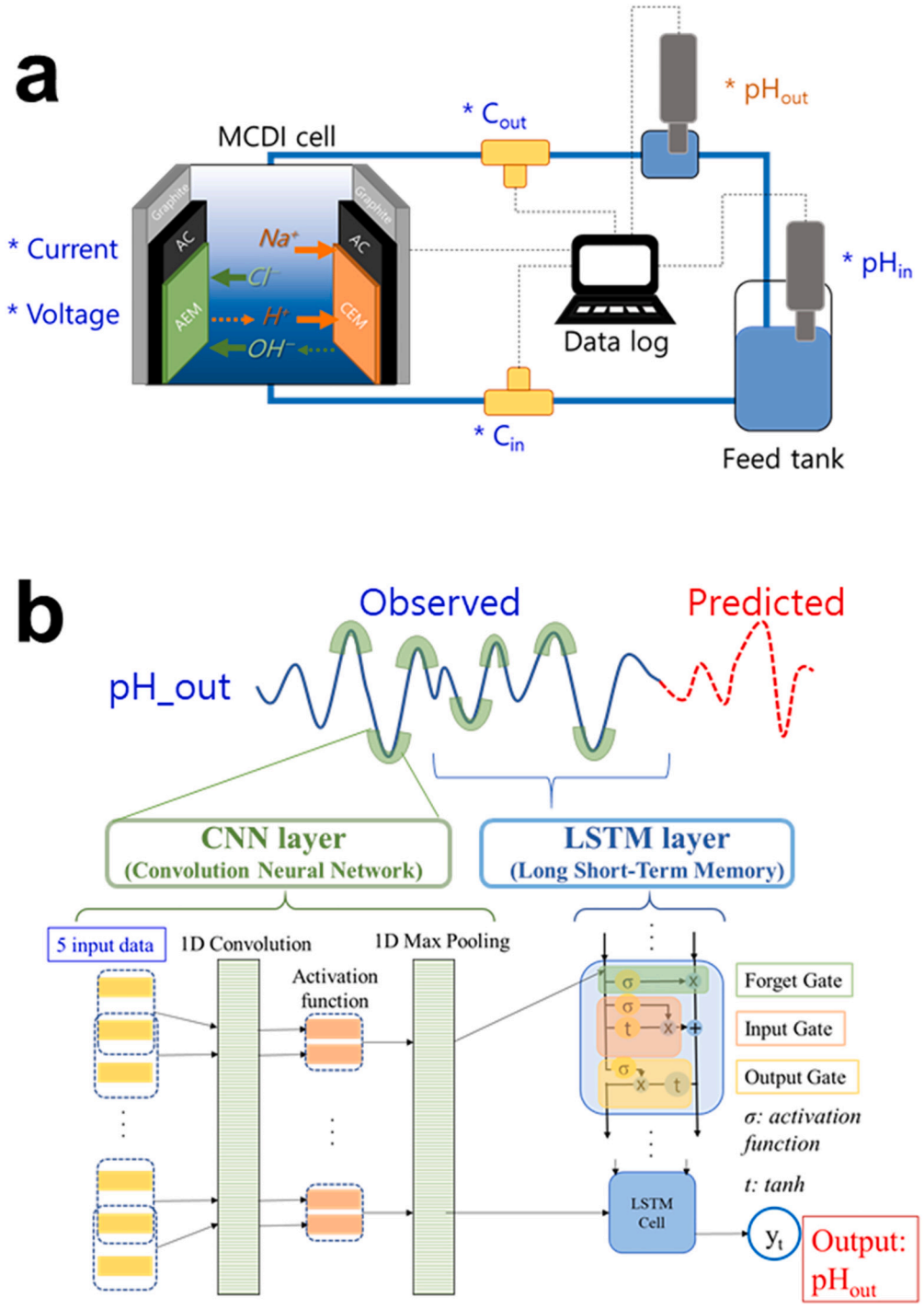


Fig. 1. Schematic illustration of membrane capacitive deionization (MCDI) operation and proposed convolutional neural network (CNN) linked with long short-term memory (LSTM) model. (a) Data collections of five variables (current, voltage, C_{in}, C_{out}, and influent pH (pH_{in})) and effluent pH (pH_{out}). "C_{in}" and "C_{out}" represent influent and effluent conductivity, respectively. (b) Structure of a CNN linked with LSTM (CNN-LSTM) model. While the CNN and LSTM layers scan the same data point, the convolution filter of CNN extracts mostly the local features of the data, whereas LSTM processes time-series data to output results. Note that a 1D CNN and two layers of LSTM were combined and used.

2.2. Electrochemical tests

Four scenarios, including the CC and CV modes, were tested. First, stepwise increments in current during the CC operation were performed from ±14 to ±28 mA (with ±3.5 mA step increases) within a voltage window of ±2 V (CC-wide). A narrow voltage window, +2 to 0 V, was then tested with a smaller current window from ±21 to ±28 mA (with a ±3.5 mA step increase) (CC-narrow). The CV tests were conducted with a voltage window of either ±1.2 to ±2.0 V (CV-high) or ±0.8 to ±1.4 V (CV-low) for 10 min of each voltage point.

2.3. Data collection

Six parameters (voltage, current, influent conductivity, effluent conductivity, influent pH, and effluent pH) were continuously recorded at a time interval of 1 s. Then, the current and voltage were transmitted and stored in real time to a computer connected through a battery cycler (WBCS3000S, WonATech Co., Ltd., South Korea) connected to the cell. The flow-through conductivity probes (ET908, eDAQ, Australia) and pH probes (ET908, eDAQ, Australia) were placed in the feed and effluent hoses to collect the data. To minimize delay in the responses, the conductivity (~2 cm away from the cell) and pH probes (~4 cm away from the cell) were located as closely as possible to the MCDI cell. Under the

fixed flow rate of 4 mL min^{-1} that was used in this study, a response delay of <5 s was found for both probes. To correct for these discrepancies in response time, the start time was set to be equal for each dataset. For example, the timing of the first changes in conductivity and pH were recorded after cell charging was initiated (voltage and current changes). Note that none of the reference electrodes were used to correlate the pH variations with the electrode potential. The total number of data points collected for CC-wide was 6601, CC-narrow was 1264, CV-high was 6000, and CV-low was 4860.

2.4. Data processing

Through the trained models for each condition, the effluent pH (pH_{out}) was predicted using five input data: current, voltage, influent conductivity, effluent conductivity, and influent pH. The data were first preprocessed by normalizing values in the [0, 1] range, which is a method known to improve the accuracy of the models [30]. Then, the data were divided into three subsets for different purposes: training, validating, and testing datasets.

Sixty-four percent of the total data were selected as the training dataset, which trained models, and 16% of the total data were used as the validation dataset to validate the model during the training (8:2 ratio for training versus validation). The remaining 20% of data were used as the testing dataset for the prediction. In other words, 80% of the entire dataset were used for training and validation, while the remaining 20% were used for prediction. This 8:2 or 7:3 ratio is commonly found in contemporary deep learning studies on environmental applications [30,31]. Although standardized rules have not yet been developed, these ratios are known to prevent deep learning models from being trained with too little data or from consuming too much data for training. As chosen temporal resolutions is also closely related to the total size of datasets, further assessment of the accuracy of the deep learning model when varying the ratio and resolution of data points is desirable. The data were randomly distributed to each dataset without overlapping to prevent overfitting of the model [32,33]. Unlike data processing in general electrochemical processes, the entire data was used from the first cycle in this study. This is because the model might be trained in a biased manner if the information about the initial solution pH (even before the first charging step was initiated) is excluded. For example, the initial value of the influent pH in the first cycle was slightly below pH = 7, whereas that of the second cycle varied from pH≈3.5 to pH≈10.0 (Fig. 2).

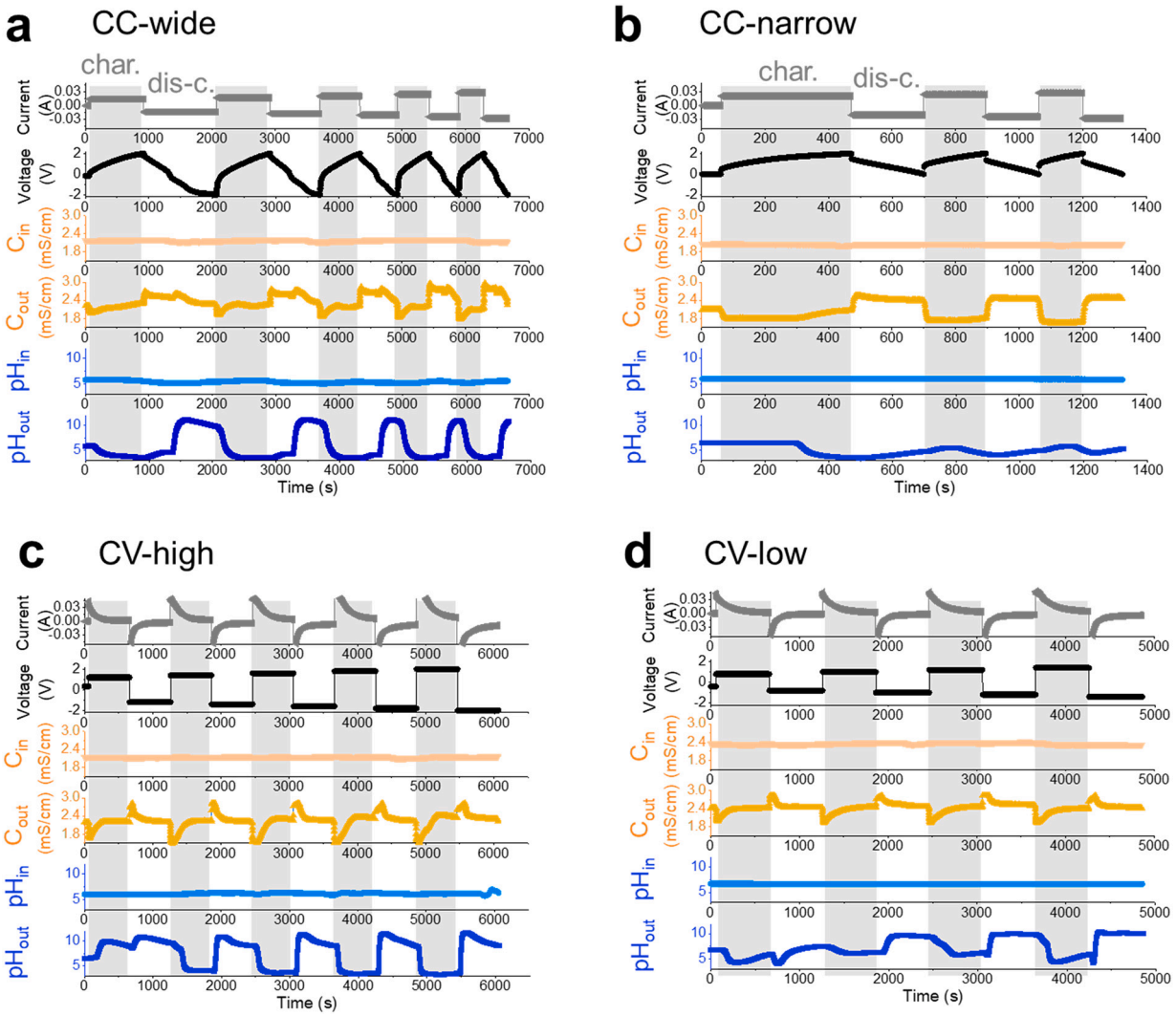


Fig. 2. Input variables collected from MCDI operation. Profiles of current, voltage, C_{in}, C_{out}, and pH_{in} during (a and b) constant-current (CC) or (c and d) constant-voltage (CV) operations. For comparison with the predicted value, the measured value of pH_{out} is also included at the end of each profile. For each operation, the charging phases were marked as gray boxes. CC-wide: ±14 to ±28 mA (±3.5 mA step increments) with cutoff voltage of ±2 V; CC-narrow: ±21 to ±28 mA (±3.5 mA step increments) with cutoff voltage of +2 to 0 V; CV-high: ±1.2 to ±2.0 V for 10 min each with ±0.2 V step increments; CV-low: ±0.8 to ±1.4 V for 10 min each with ±0.2 V step increments.

2.5. Deep learning model of CNN-LSTM

The CNN-LSTM hybrid model was employed to improve the insufficient prediction of the stand-alone LSTM due to the complexity of the MCDI experimental data. In particular, features of the MCDI data can be better reflected in deep learning models because the CNN is more suitable for extracting local features from time-series data [24]. The proposed model (CNN-LSTM) combines two different types of networks: a CNN layer followed by an LSTM layer (Fig. 1b).

In the convolutional layer, the features are first extracted through convolution operations on time-series data with convolution filters (also known as kernels). Briefly, the convolution filter extracts mostly the local features of the time-series data; moreover, an activation function is applied to the output of the CNN to add/incorporate nonlinearity.

The purpose of the CNN before LSTM is to employ a CNN to extract local features in time-series data, which might be overlooked by the LSTM. Then, LSTM layers were sequentially integrated with the extracted features, generating the time-series prediction, while remembering the irregular trend factor of data [26,34]. It has been reported that the combination of LSTM with CNN has the advantage of recognizing patterns from data more effectively than using the LSTM individually when the time-series data contains sufficient temporal variances [35]. More detailed information about the CNN-LSTM model can be found in the Supporting information.

2.6. Deep learning model of LSTM

LSTM is a special form of the RNN, which has the ability to process sequential data owing to the presence of the LSTM cell and gating mechanism. The LSTM networks were originally designed to process long sequential data by introducing three gates in the LSTM cell: input (i_t), forget (f_t), and output (o_t) gates [25]. LSTM networks solve the limitations of the RNN due to the long time dependencies of the sequence by removing and adding information through the interaction between these three gates [36]. The input gate allows new information to be stored in the memory state (c_t) and the forget gate determines which information must be remembered or forgotten in the memory state. Subsequently, the output gate comprehensively considers all the calculated results and generates the output of the LSTM cell [37,38]. The operations of the three gates are summarized in Eqs. (1)–(6). The calculations of the hidden states (h) and cell states (c) were performed using Eqs. (3), (4), and (6) as follows:

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

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2}$$

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

$$c_t = f_t \times c_{t-1} + i_t \times \overline{c}_t \tag{4}$$

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

$$h_t = o_t \times tanh(c_t) \tag{6}$$

where w and b denote the weight and bias of each gate, respectively, which is determined during model training, and σ is the sigmoidal nonlinearity (also known as activation function) [39]. It should be noted that the LSTM model had two stacked layers (LSTM layer 1 and LSTM layer 2).

2.7. Hyperparameter optimization

In deep learning, hyperparameters (such as the learning rate, amount of historical data used to predict the next value, and so on) exist, and their optimization is significantly important because these parameters are directly related to the performance of the model [40]. We optimized the hyperparameters of the models by using the Bayesian optimization method, which is based on a probabilistic model [41]. The purpose of Bayesian optimization is to determine the hyperparameters on designated subsets that globally minimize the loss function. Bayesian optimization utilizes the Gaussian process as a surrogate function owing to its descriptive power and analytic tractability [42].

The validation MSE was used as the objective function during hyperparameter optimization, as presented below:

$$MSE = \frac{\left[\sum_{i=1}^{n}\left(Y_i - \widehat{Y}_i\right)^2\right]}{n} \tag{7}$$

where Y_i is the observed data, $\widehat{Y}_i$ is the predicted data, and n is the size of the test set.

To check the reliability of the optimization step, the convergence plot was plotted using the loss values in each hyperparameter optimization iteration. The optimized hyperparameters were chosen when the minimum validation MSE was observed during iterations [43].

2.8. Model evaluation

The prediction results of the models were evaluated using the coefficient of determination (R^2), normalized root-mean-square error (NRMSE), root-mean-square error (RMSE), and mean absolute error (MAE) (Eqs. (8)–(11)), as presented below:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(\widehat{Y}_i - Y_i\right)^2}{\sum_{i=1}^{n}\left(Y_i - \overline{Y}\right)^2} \tag{8}$$

$$RMSE = \sqrt{\frac{\left[\sum_{i=1}^{n}\left(Y_i - \widehat{Y}_i\right)^2\right]}{n}} \tag{9}$$

$$NRMSE = \frac{RMSE}{Y_{max} - Y_{min}} \tag{10}$$

$$MAE = \frac{\sum_{i=1}^{n}\left|Y_i - \widehat{Y}_i\right|}{n} \tag{11}$$

where Y_{max} and Y_{min} are the maximum and minimum values among the test data, respectively. $\overline{Y}$ is the mean of the test data.

3. Results

3.1. Membrane capacitive deionization (MCDI) operation under different scenarios

In practical MCDI applications both CC and CV operation modes are widely used [27]. Thus, both CC and CV conditions were tested to obtain different trends in the pH of the effluent solutions (pH_{out}) (Fig. 2). Detailed operating conditions are provided in the Materials and methods section. Briefly, two different cut-off voltages were used for CC operation, with a wide cut-off voltage of ±2 V (CC-wide, from ±14 to ±28 mA) or a narrow cut-off voltage of +2 to 0 V (CC-narrow, from ±21 to ±28 mA). For CV operation, a higher applied voltage of ±1.2 to ±2.0 V (CV-high) was compared to operation with a lower voltage of ±0.8 to ±1.4 V (CV-low). As the higher current or voltage conditions more ions were absorbed, resulting in a more rapid change in the effluent conductivity (CC-wide versus CC-narrow, or CV-high versus CV-low).

For CC-wide conditions, conducted with a symmetric voltage window (±2 V), the solution pH also was symmetric, which indicated that anodic oxidation and cathodic reduction occurred predominantly at similar rates during charging (oxidation) and discharging (reduction). For example, the solution pH was varied from pH = 3.6 (charging) to pH

= 10.7 (discharging) (Fig. 2a), indicating that, based on the neutral pH (pH = 7), the amount of H^+ ions generated upon charging was almost proportional to that of OH^- ions generated during discharging. Unexpectedly, there were two regions in the effluent conductivity profiles (C_{out}; dark orange line in Fig. 2a) during the discharging phases, under CC-wide conditions. As this phenomenon occurred when the cell voltage reached 0 V, it was likely due to an abrupt change in the direction of ion migrations inside the cell, which was instigated when the voltage applied was reversed.

When a narrow voltage window of +2 to 0 V was used (CC-narrow), a flatter and more acidic pH was measured throughout the operation. This flatter trend in pH was likely owing to the reduced Faradaic reactions that occurred, particularly during the discharge step since the narrowed voltage window (from +2 to 0 V) was used. The more acidic pH variations under the CC-narrow conditions observed throughout the operation could be interpreted to be the dominant anodic Faradaic reactions, such as oxygen evolution and carbon oxidation. For the same reason (the larger voltage window, more faradaic reactions occurred), a larger change in solution pH was observed for CV-high due to the higher voltage window used (±1.2 to ±2.0 V), when compared to CV-low (±0.8 to ±1.4 V).

Unlike the CC condition, in the CV operation modes the solution pH rapidly changed when the discharging phase started as the ions adsorbed during charging were quickly desorbed (Fig. 2c and d). In the case of CV-high, the solution pH became alkaline at the first charging phase, whereas the solution pH became acidic during the charging phase in other cases. The mechanisms driving the pH changes toward alkaline values during the first charging phase were unclear. For example, under the same CV-high operation, pH changes toward acidic values were observed from the 2nd cycle. In addition, the solution pH became acidic when a lower voltage was applied (CV-low). Thus, more investigations should be carefully carried out to elucidate the mechanisms driving this unusual trend.

3.2. Deep learning model optimization

Two deep learning models (CNN-LSTM and LSTM) were prepared to predict the effluent pH by utilizing the operation data of MCDI. Briefly, the LSTM model contains an LSTM cell, consisting of input (i_t), forget (f_t), and output (o_t) gates, which was developed to process time-series data (note that two LSTM layers were stacked). The developed CNN-LSTM model consists of a CNN layer (1D convolutional and 1D max-pooling layers) followed by LSTM layers. The hyperparameters (learning speed, the amount of data processed at one time, etc.) were

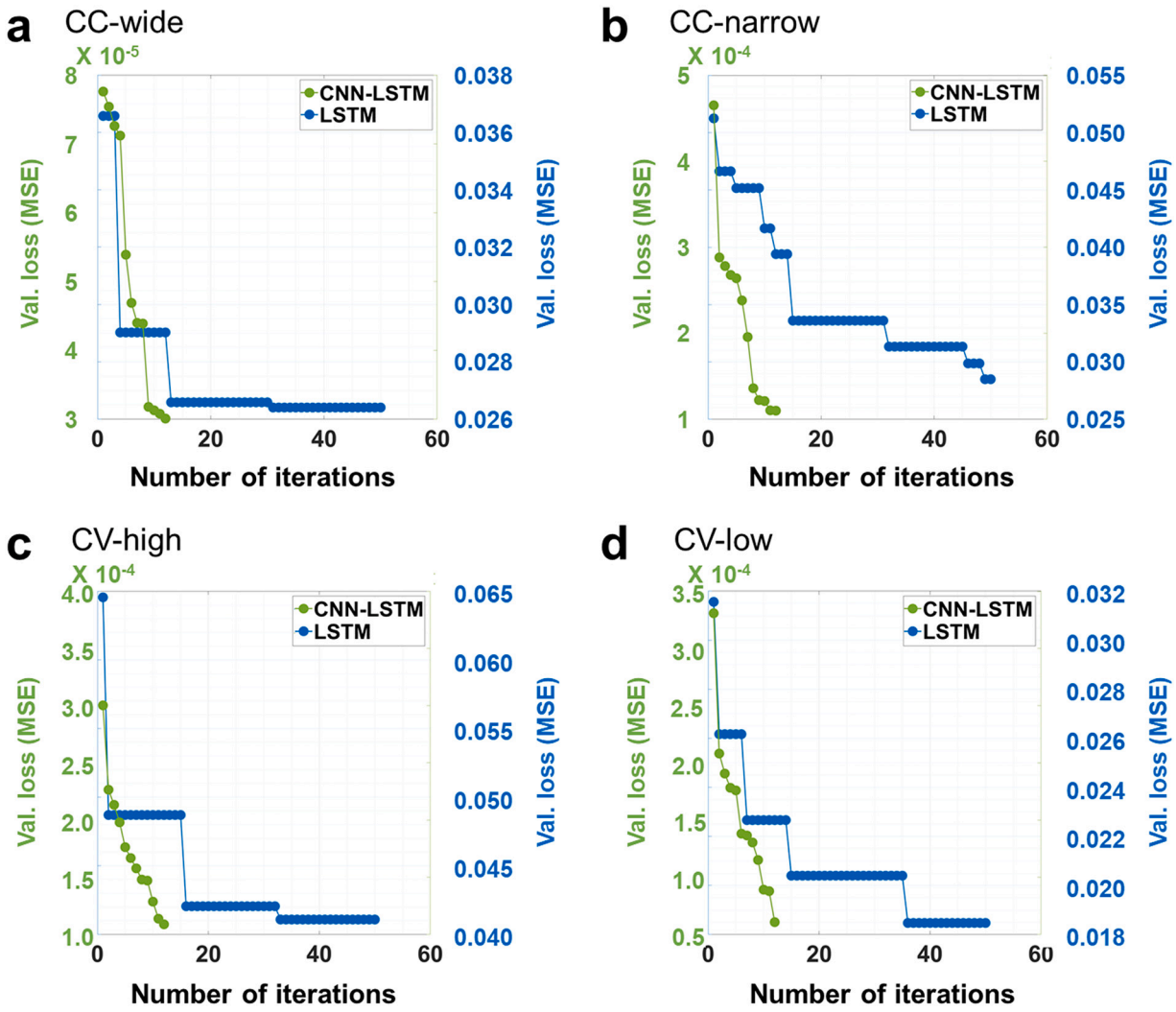


Fig. 3. Validation loss during hyperparameter optimization. Convergence plot for (a) CC-wide, (b) CC-narrow, (c) CV-high, and (d) CV-low. The validation loss of the CNN-LSTM (green, left) and LSTM (blue, right) were plotted as a function of the number of iterations (hyperparameter optimizations). The hyperparameters optimized in this step include batch size, lookback, learning rate, activation functions (refer to Supplementary materials for details). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

optimized using the Bayesian optimization method, which is based on a probabilistic model [41] (details in the Materials and methods). The hyperparameters were determined for particular values that minimized the loss function by considering the interactions with other hyperparameters. Diagonal plots indicate the influence of the change by one hyperparameter when the other parameters are fixed (solid red lines) (Figs. S1–S8). Off-diagonal plots show the results of the interactions between the hyperparameters. The black points represent the tested sets, and the color was contoured according to the calculated loss value (partial dependence). A brighter color (blue to yellow) indicated a lower loss value. The red-star points (marked with red circles) in the off-diagonal plots indicate the optimized hyperparameter values. For example, when we observed the plot for the CC-wide prediction by LSTM (Fig. S1), the optimum value of the batch size was determined to be eight, as indicated by the red dashed line (in the first row) and red stars (in the first column).

The number of iterations during hyperparameter optimization was set to 12 times for CNN-LSTN and to 50 times for LSTM due to the orders of magnitude of lower validation loss in the CNN-LSTM (Fig. 3). For example, in the case of CC-wide, the validation loss of CNN-LSTM was $\sim 3 \times 10^{-5}$ MSE mean squared error (MSE; after 12 iterations), whereas it was $\sim 3 \times 10^{-2}$ MSE for LSTM (after 50 iterations) after hyperparameter optimization. These optimizations choose the most suitable values of hyperparameters for both CNN-LSTM and LSTM (Tables S1–S3). The CNN-LSTM model was sensitive to the lookback (the number of previous time steps used as input), except for the CV-high operation, but not to the learning rate and kernel size (Figs. S1–S4). Conversely, the LSTM model was predominantly sensitive to the learning rate, but not to the number of LSTM cells (Figs. S5–S8).

3.3. pH prediction using convolutional neural network linked with long short-term memory (CNN-LSTM)

In the CNN-LSTM model, the number of learning in the deep learning model (epochs) ended before 300 times although it was set to 500 times (Fig. S9a, c, e, and g). This early stop in the epochs was determined to shorten the time, as the loss was insignificantly changed even if the training was continued, as reported in the previous literature [44]. For all cases, validation losses between prediction and observation were at least in an order of magnitude lower than that of the LSTM. For example, in the case of the simplest form of CC-wide, a low loss value of <0.0005 MSE was maintained from the 10th epoch, which was in orders of magnitude lower than that of LSTM ($\sim$0.1 MSE). Thus, the regression curves for the training and test were in great agreement for all cases in the CNN-LSTM (Fig. S9b, d, f, and h). Consequently, the CNN-LSTM model predicted pH_{out} with an accuracy of a coefficient of determination of $R^2 \geq 0.998$ for all scenarios, likely due to the pretreatment of the CNN (Table 1 and Fig. 4), which can excellently reflect the inflection point of the data. The accuracy of the CNN-LSTM model was significantly higher than that reported in the numerical modeling [17], although the experimental conditions and setup were slightly different. For example, under the same CC mode, the developed CNN-LSTM model showed an accuracy of $R^2 = 0.999$, normalized root-mean-square error of NRMSE = 0.006, and mean absolute error of MAE = 0.029 (CC-wide), whereas the numerical model exhibited an accuracy of $R^2 = 0.309$, NRMSE = 0.301, and MAE = 0.384 (detailed calculations for the numerical model in the Supplementary materials) [17]. Note that, similar to this study, the experiment for the numerical modeling was conducted at a CC mode using the feedwater of 20 mM NaCl with a batch reservoir.

Table 1
Coefficient of determination (R^2), normalized root-mean-square error (NRMSE), and mean absolute error (MAE) during training and test (prediction) using convolutional neural network linked with long short-term memory model (CNN-LSTM).

MCDI operation	R^2 (training)	R^2 (test)	NRMSE (training)	NRMSE (test)	MAE (test)
CC-wide	0.999	0.999	0.006	0.006	0.029
CC-narrow	0.999	0.998	0.011	0.013	0.028
CV-high	0.999	0.999	0.061	0.011	0.062
CV-low	0.999	0.999	0.008	0.008	0.033

3.4. pH prediction using long short-term memory (LSTM)

The number of learning (epochs) in the LSTM was set to 500, but the training of all cases was stopped before this end point was reached. An early stopping technique was employed to avoid overfitting of the results during the training. The training was stopped when the validation loss was increasing, rather than decreasing (Fig. S11a, d, g, and j) [43,45]. For the CC-wide case, where the regression curve of the training and test was in relatively good agreement (Fig. S11b, e, h, and k), the developed LSTM model accurately predicted pH_{out} because of the simple trends in pH (Fig. S11b and c). In this case, the accuracy of the LSTM model was higher than that reported in the numerical modeling [17]. For example, under the same CC mode, which was also the operating condition used in the numerical modeling study, the developed LSTM model showed an accuracy of $R^2 = 0.843$ (CC-wide), whereas the numerical model exhibited an accuracy of $R^2 = 0.309$ [17].

An increase in the complexity of the trends (CC-narrow and CV-high) resulted in a decrease in the prediction accuracy. For example, the predicted CC-wide values, which demonstrated a periodical repetition with similar patterns, was relatively accurate, with $R^2 = 0.843$ (Table S4). However, when the peak value of the pattern was changed (CC-narrow) or the pattern was not constant (CV-high), the accuracy decreased to $R^2 = 0.680$ (CC-narrow) or $R^2 = 0.692$ (CV-high). Even though the trends in pH_{out} were relatively complex, the prediction accuracy was relatively high with $R^2 = 0.807$ in the CV-low case (Fig. S11l and Table S4), where training and validation losses were relatively low at <0.04 of the MAE (Fig. S11j). Overall, it was observed that the use of the LSTM model had certain limitations even after optimization, although it showed superior predictive accuracy to the numerical modeling for a simple pattern ($R^2 = 0.843$ for LSTM and $R^2 = 0.309$ for numerical modeling). Unlike the LSTM model, the CNN-LSTM model showed excellent accuracy regardless of the complexity of data used ($R^2 \geq 0.998$ for all scenarios); thus, the CNN-LSTM was chosen for further investigations such as the prediction with limited input variables.

3.5. pH prediction based on only using current and voltage

The developed CNN-LSTM model has consisted of CNN layer (for preprocessing the changes in the local features) followed by LSTM layer (for processing time-series data to output results). During the hyperparameter optimization, there were no large changes in the CNN layer when only two input variables were used, but there were slight changes in the LSTM layer (Tables S2 and S3). First, only "leaky version of a rectified linear unit (leakyrelu)" or "rectified linear unit (relu)" was selected as the activation function when only two input variables were used, unlike the prediction using five input variables where "tanh" was set as the activation function in certain cases. Second, in the case of CC-wide and CC-narrow in particular, the learning rate and lookback were decreased by choosing a small number of input variables.

When only the two input variables of current and voltage were used in CNN-LSTM, the loss values during training and validation were maintained at low values (Fig. S10a, c, e, and g), similar to when five input variables were used. In addition, the regression curves showed that the predicted and observed values were in great agreement (Fig. S10b, d, f, and h). Consequently, the prediction accuracy was high with $R^2 \geq 0.997$ (Fig. 5 and Table S5) when predicting pH_{out} using CNN-LSTM with limited information of current and voltage, instead of five input variables. This accuracy was remarkably higher than that of the LSTM using five input variables ($R^2 \geq 0.680$). Thus, both the NRMSE and MAE of the CNN-LSTM model with two input variables were

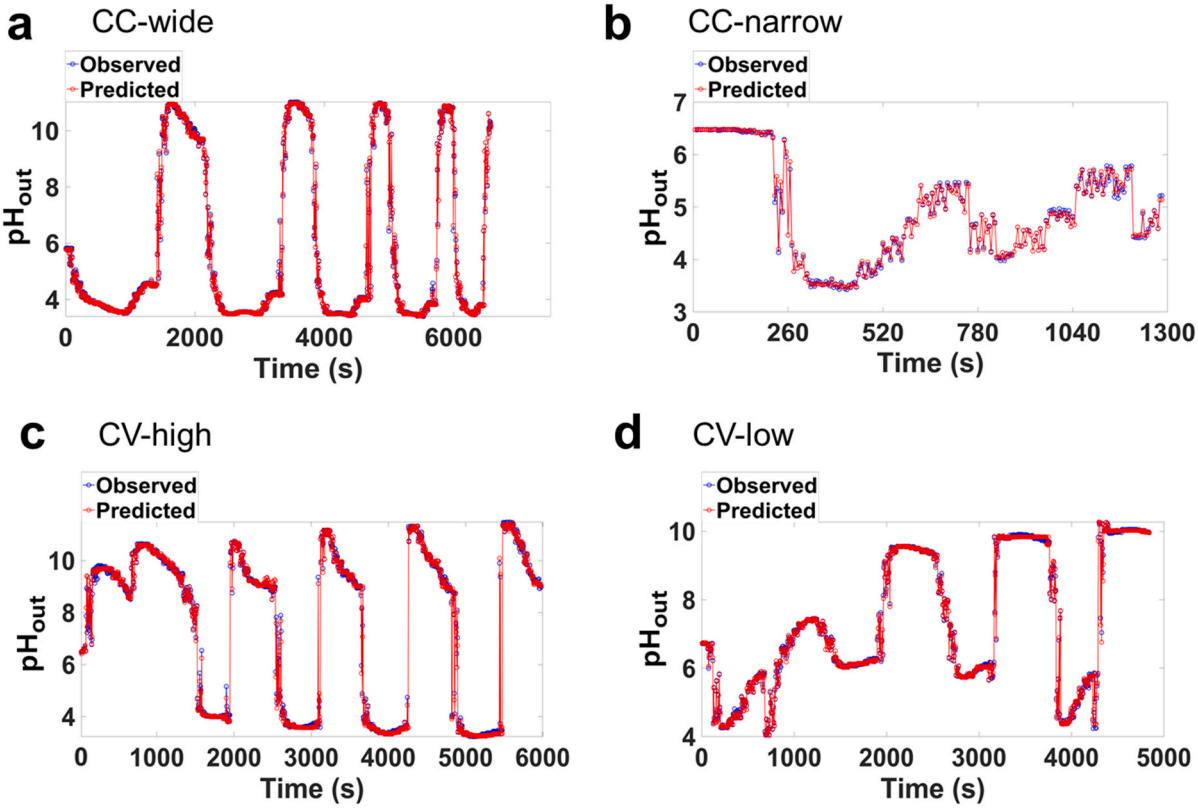


Fig. 4. Effluent pH predictions using CNN-LSTM for (a) CC-wide, (b) CC-narrow, (c) CV-high, and (d) CV-low modes. The observed values (blue) are plotted together with the predicted values (red). Five input variables (current, voltage, C_{in}, C_{out}, and pH_{in}) were used for the predictions. Other relevant data such as training/validation losses and regression curves can be found in the Supporting information (Fig. S9). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

significantly lower than those of the LSTM with five input variables (Tables S5 and S2). For example, in the case of CV-high, the NRMSE of CNN-LSTM with two input variables (NRMSE = 0.023) was in an order of magnitude smaller than that of the LSTM with five input variables (NRMSE = 0.211).

3.6. Contributions of each input variable

To analyze the contribution of different factors, each input variable was sequentially introduced into the developed CNN-LSTM model. The input variable of current was fixed and used together with each input variable for the CC mode because there were no large changes in current over time. Based on the same principle, the voltage was fixed in the CV mode and the other variables were used sequentially and analyzed. The number of epochs was reduced from 500 to 300 for a more explicit comparison between the predictions using the limited condition.

Under CC conditions (CC-wide and CC-narrow), the voltage had the greatest influence on the prediction of pH (Figs. 6a, b, and S12). When the prediction was performed using only voltage, the accuracy was almost the same as that when using all five input variables (Fig. 6a and b). In both CC-wide and CC-narrow operations, the C_{in} factor had the lowest impact on the prediction accuracy (Table S6). In particular, in the case of CC-wide, when the prediction was performed using only C_{in}, R^2 = 0.691, NRMSE = 0.215, and MAE = 0.968 were determined, resulting in a decrease in the prediction accuracy when compared to other factors (Fig. 6a).

Unlike the CC modes, in the CV modes, the difference in sensitivity for each input variable was less distinctive, as shown by the R^2 values (Fig. 6c and d). The prediction using current showed high accuracies (R^2 = 0.980 for CV-high and R^2 = 0.996 for CV-low); however, the differences when compared to C_{out} were marginal (R^2 = 0.996 for CV-high and R^2 = 0.993 for CV-low) (Table S7). In the CV modes, C_{in} and pH_{in} still showed the lowest sensitivity (highest error values). However, a relatively high prediction accuracy of $R^2 > 0.912$ was measured in all cases regardless of the choice of input variable when compared to the CC modes.

4. Discussion

The CNN-LSTM model showed superior predictive accuracy of effluent pH for all scenarios ($R^2 \geq 0.998$) when compared to the stand-alone LSTM model, which showed higher accuracy than the previously reported numerical modeling method (R^2 = 0.843 for LSTM versus R^2 = 0.309 for numerical modeling under CC mode) [17]. This can be interpreted as the fact that the CNN-LSTM model showed strength in simple time-series data processing due to the inclusion of the CNN for superior capability in processing local features. Thus, more weight was given to the interval in which data changes by inclusion of the CNN model. For example, in the case of CC-wide, where the change is relatively monotonous, the LSTM model also showed a relatively high prediction accuracy of R^2 = 0.843. However, in the case of CV-high, which has similar shapes of pH_{out} when compared to CC-wide, the LSTM model predicted the general trend but showed a decrease in the prediction accuracy (R^2 = 0.692). In particular, it was observed that the LSTM model insufficiently predicted the early period of CV-high, where the change in pH_{out} was rapid. This limitation of the use of LSTM implies that LSTM is suitable for processing the data showing pattern regularity over time. Unlike the LSTM model, the developed CNN-LSTM model accurately predicted not only the change in effluent pH but also the absolute value for all cases ($R^2 \geq 0.998$). Therefore, the use of the CNN-LSTM model provided an excellent prediction for time-series data processing with rapid changes, such as the effluent pH of MCDI used in this

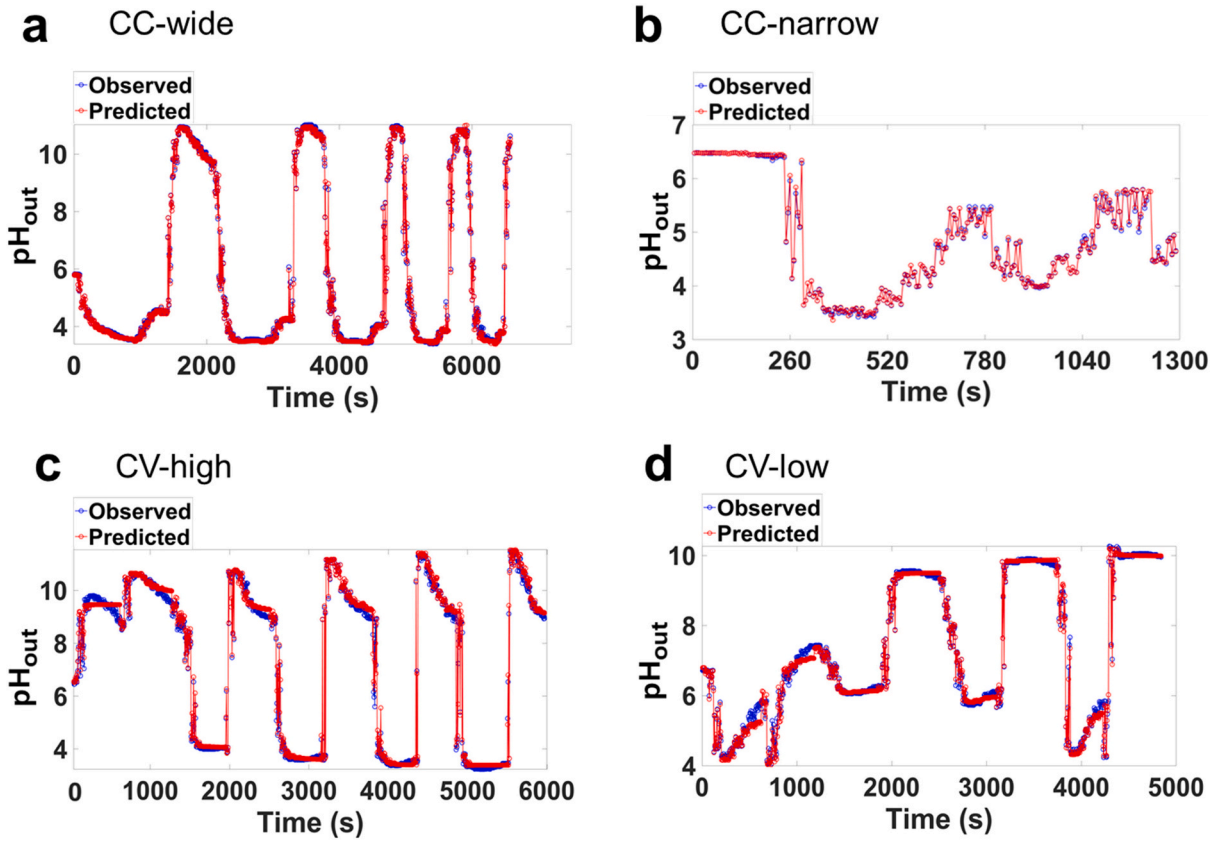


Fig. 5. Effluent pH prediction with limited information for (a) CC-wide, (b) CC-narrow, (c) CV-high, and (d) CV-low modes. The observed values (blue) were plotted with the predicted values (red). Only two input variables (current and voltage) were used for the predictions. Other relevant data such as training/validation losses and regression curves can be found in the Supporting information (Fig. S10). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

study.

In the initial pH_{out} prediction using CNN-LSTM, a total of five input variables including influent/effluent conductivity were used. However, in an actual MCDI process operation, only limited input variables, such as current and voltage, are often recorded. Therefore, for practical applications of the CNN-LSTM model developed in this study, prediction using only current and voltage as input variables was examined. The results showed excellent prediction of pH_{out} using the CNN-LSTM model with only these two input variables ($R^2 \geq 0.997$). These results imply that the CNN-LSTM model can be applied in practical applications where only current and voltage are typically recorded [27].

Because deep learning models are not based on any assumptions under ideal conditions, they are known to have advantages in processing nonlinear environmental data when compared to numerical modeling [19]. However, it is also known to be less explainable because of the presence of hidden layers [46,47]. To partially compensate for this, a sensitivity analysis was performed by sequentially using the input variables. Note that the sensitivity analysis was applied differently for each CC and CV mode. The current was used together with an input variable for the CC mode because a fixed current was used during the operation, whereas the voltage was used together with an input variable for the CV mode, where a fixed voltage was used. In the case of the CC mode, when only voltage was used, the accuracy was almost the same as that when using all five input variables ($R^2 = 0.999$ for voltage, $R^2 \geq 0.998$ for all). Other factors, particularly the influent conductivity, were not helpful in the prediction ($R^2 = 0.691$ for CC-wide). The peculiar thing was that the CV mode showed a high prediction accuracy of $R^2 \geq 0.912$, irrespective of which input variable was added. Collectively, it was confirmed that the voltage of the MCDI process was the factor that most affected the effluent pH. This is in line with previous studies showing that Faradaic reactions, which are typically driven by voltage, are the most important factors in determining the pH of the MCDI process [18].

Accurate prediction of the solution pH can contribute to a more efficient water purification process using MCDI. Because most ionic substances change their dissociation form according to the solution pH, it is possible to optimize the removal rate of the target substance by properly predicting the pH according to the operating conditions. For example, pharmaceutical compounds have different ionic properties depending on their pKa values [48]. Using the CNN-LSTM model developed in this study, operating parameters such as current and voltage can be properly set, thereby enhancing the removal of these ionic compounds. For future investigations, fundamental desalination performances such as salt adsorption capacity (*SAC*) and specific energy consumption (*SEC*) were calculated for the representative condition (constant current charging at +21 mA; cut-off voltage of 2 V). The results showed reasonable *SAC* of 4.4 mg-NaCl g-electrode^{-1} and *SEC* of 0.002 kWh m^{-3} (see the Supporting information).

In contrast to numerical modeling, the CNN-LSTM method used in this study can theoretically predict the effluent pH irrespective of the operating conditions (voltage, current, operation time), size, and type of MCDI cells or electrodes, and the properties of the influent. Moreover, apart from effluent pH prediction, most MCDI studies where large amounts of data are available, such as ion selectivity experiments or desalination performance, could be investigated using similar CNN-LSTM approaches. This is because deep learning performs prediction by focusing only on the correlation of data, not the reaction mechanism of the MCDI cell. However, similar to other deep learning techniques, the CNN-LSTM model must be trained prior to the prediction by securing data for training. For example, to employ the CNN-LSTM model in long-term operations with real brackishwater, full operation data must first

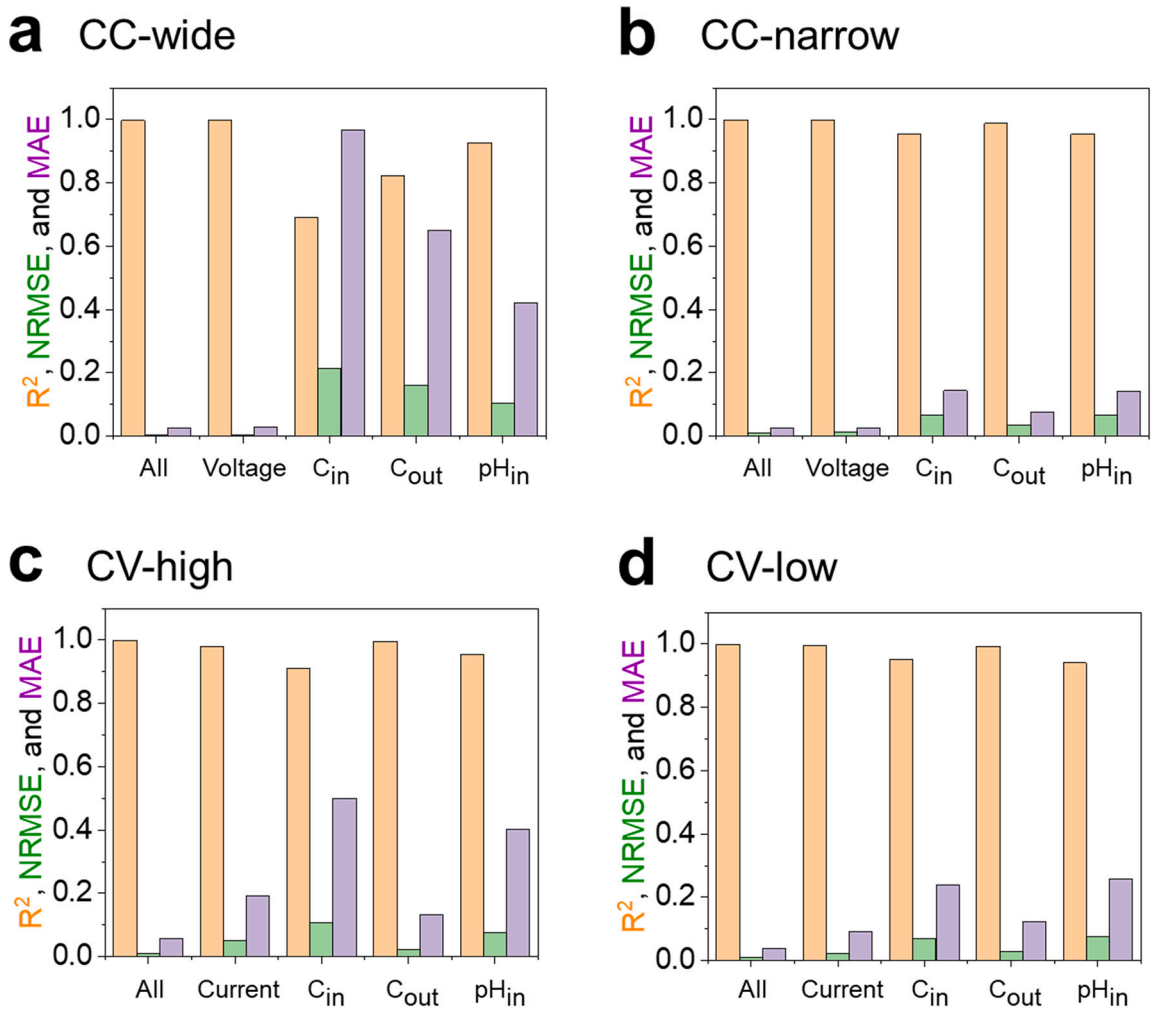


Fig. 6. Sensitivity analysis based on the input variables. Differences (Coefficient of determination (R^2), normalized root-mean-square error (NRMSE), and mean absolute error (MAE)) in pH_{out} between the prediction and observation were evaluated for (a) CC-wide, (b) CC-narrow, (c) CV-high, and (d) CV-low. The values of R^2 (left bar; orange), NRMSE (middle bar; green), and MAE (right; purple) were calculated based on Eqs. (8)–(11). Among the five input variables (voltage, current, C_{in}, C_{out}, and pH_{in}), each input variable was used one by one to determine the dominant input variable during pH_{out} prediction. A fixed input variable of current was used together with each input variable for CC modes. In contrast, a fixed input variable of voltage was used together with each input variable for CV modes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

be acquired, and some portions of the data (i.e., 80% of the total data) should be used to train the model. In addition, for more practical applications, it is necessary to verify the versatility of the deep learning model in future studies which test it under more diverse conditions. Moreover, a proper pre-processing of data such as sequential data preparation for LSTM has to be conducted prior to apply the CNN-LSTM model.

5. Conclusion

The developed CNN-LSTM model exhibited superior predictive accuracy of pH_{out} for various scenarios of MCDI operation ($R^2 \geq 0.998$) when compared to the previously reported numerical modeling method ($R^2 = 0.309$). When the change in pH_{out} is relatively monotonous, the stand-alone LSTM also showed a relatively high prediction accuracy of $R^2 = 0.843$. Results of the sensitivity analysis confirmed that Faradaic reactions, which are typically driven by voltage, are the most important factors in determining the pH of the MCDI process. Moreover, the developed CNN-LSTM model can be applied in practical applications because its excellent prediction of pH_{out} by using only current and voltage as input variables ($R^2 \geq 0.997$).

CRediT authorship contribution statement

Moon Son: Conceptualization, Methodology, Investigation, Visualization, Writing – Original Draft, Writing – Review & Editing, Funding acquisition.: **Nakyung Yoon:** Methodology, Investigation, Visualization, Writing – Review & Editing.: **Kwanho Jeong:** Methodology, Writing – Review & Editing.: **Ather Abass:** Investigation, Visualization, Writing – Review & Editing.: **Bruce E. Logan:** Visualization, Writing – Review & Editing.: **Kyung Hwa Cho:** Conceptualization, Methodology, Writing – Review & Editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

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

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.

2020R1A4A1019568 and No. 2021R1C1C2005643). This work was supported by Korea Environmental Industry and Technology Institute (KEITI) through Industrial Facilities & Infrastructure (Desalination) Research Program, funded by Korea Ministry of Environment (MOE) (146847).

Appendix A. Supplementary data

Details on the numerical modeling that used for comparison, CNN-LSTM deep learning model, hyperparameter optimizations, desalination performances, LSTM results, and sensitivity analysis. The computer algorithm for the developed CNN-LSTM algorithm used in this study can be shared upon reasonable request to the corresponding author (K. H. Cho). Supplementary data to this article can be found online at https://doi.org/10.1016/j.desal.2021.115233.

References

[1] H.Y. Yang, Z.J. Han, S.F. Yu, K.L. Pey, K. Ostrikov, R. Karnik, Carbon nanotube membranes with ultrahigh specific adsorption capacity for water desalination and purification, Nat. Commun. 4 (2013) 2220.
[2] S. Porada, R. Zhao, A. van der Wal, V. Presser, P.M. Biesheuvel, Review on the science and technology of water desalination by capacitive deionization, Prog. Mater. Sci. 58 (2013) 1388–1442.
[3] F.A. AlMarzooqi, A.A. Al Ghaferi, I. Saadat, N. Hilal, Application of capacitive deionisation in water desalination: a review, Desalination 342 (2014) 3–15.
[4] Y. Oren, Capacitive deionization (CDI) for desalination and water treatment — past, present and future (a review), Desalination 228 (2008) 10–29.
[5] H. Li, L. Zou, Ion-exchange membrane capacitive deionization: a new strategy for brackish water desalination, Desalination 275 (2011) 62–66.
[6] P.M. Biesheuvel, A. van der Wal, Membrane capacitive deionization, J. Membr. Sci. 346 (2010) 256–262.
[7] X. Zhang, K. Zuo, X. Zhang, C. Zhang, P. Liang, Selective ion separation by capacitive deionization (CDI) based technologies: a state-of-the-art review, Environ. Sci. Water Res. 6 (2020) 243–257.
[8] L. Legrand, Q. Shu, M. Tedesco, J.E. Dykstra, H.V.M. Hamelers, Role of ion exchange membranes and capacitive electrodes in membrane capacitive deionization (MCDI) for CO2 capture, J. Colloid Interface Sci. 564 (2020) 478–490.
[9] J. Landon, X. Gao, A. Omosebi, K. Liu, Emerging investigator series: local pH effects on carbon oxidation in capacitive deionization architectures, Environ. Sci. Water Res. 7 (2021) 861–869.
[10] Y. Bian, X. Chen, Z.J. Ren, pH dependence of phosphorus speciation and transport in flow-electrode capacitive deionization, Environ. Sci. Technol. 54 (2020) 9116–9123.
[11] X. Zhang, F. Yang, J. Ma, P. Liang, Effective removal and selective capture of copper from salty solution in flow electrode capacitive deionization, Environ. Sci. Water Res. 6 (2020) 341–350.
[12] E. Avraham, M. Noked, A. Soffer, D. Aurbach, The feasibility of boron removal from water by capacitive deionization, Electrochim. Acta 56 (2011) 6312–6317.
[13] J. Chang, F. Duan, C. Su, Y. Li, H. Cao, Removal of chloride ions using a bismuth electrode in capacitive deionization (CDI), Environ. Sci. Water Res. 6 (2020) 373–382.
[14] E. Zhu, X. Hong, Z. Ye, K.S. Hui, K.N. Hui, Influence of various experimental parameters on the capacitive removal of phosphate from aqueous solutions using LDHs/AC composite electrodes, Sep. Purif. Technol. 215 (2019) 454–462.
[15] R. Zhao, P.M. Biesheuvel, A. van der Wal, Energy consumption and constant current operation in membrane capacitive deionization, Energy Environ. Sci. 5 (2012) 9520–9527.
[16] R. Zhao, O. Satpradit, H.H.M. Rijnaarts, P.M. Biesheuvel, A. van der Wal, Optimization of salt adsorption rate in membrane capacitive deionization, Water Res. 47 (2013) 1941–1952.
[17] J.E. Dykstra, K.J. Keesman, P.M. Biesheuvel, A. van der Wal, Theory of pH changes in water desalination by capacitive deionization, Water Res. 119 (2017) 178–186.
[18] C. Zhang, D. He, J. Ma, W. Tang, T.D. Waite, Faradaic reactions in capacitive deionization (CDI) - problems and possibilities: a review, Water Res. 128 (2018) 314–330.
[19] L. Li, S. Rong, R. Wang, S. Yu, Recent advances in artificial intelligence and machine learning for nonlinear relationship analysis and process control in drinking water treatment: a review, Chem. Eng. J. 405 (2021), 126673.
[20] S. Hwangbo, R. Al, G. Sin, An integrated framework for plant data-driven process modeling using deep-learning with Monte-Carlo simulations, Comput. Chem. Eng. 143 (2020), 107071.
[21] K.B. Newhart, R.W. Holloway, A.S. Hering, T.Y. Cath, Data-driven performance analyses of wastewater treatment plants: a review, Water Res. 157 (2019) 498–513.
[22] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444.
[23] M.Z. Alom, T.M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M.S. Nasrin, B.C. Van Esesn, A.A.S. Awwal, V.K. Asari, The History Began From Alexnet: A Comprehensive Survey on Deep Learning Approaches, arXiv preprint, 2018, 1803.01164.
[24] Z. Xu, J. Yu, C. Yu, H. Shen, Y. Wang, H. Yang, CNN-based feature-point extraction for real-time visual SLAM on embedded FPGA, in: 2020 IEEE 28th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), 2020, pp. 33–37.
[25] S. Siami-Namini, N. Tavakoli, A.S. Namin, A comparison of ARIMA and LSTM in forecasting time series, in: 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), 2018, pp. 1394–1401.
[26] T.-Y. Kim, S.-B. Cho, Predicting residential energy consumption using CNN-LSTM neural networks, Energy 182 (2019) 72–81.
[27] J.E. Dykstra, S. Porada, A. van der Wal, P.M. Biesheuvel, Energy consumption in capacitive deionization – constant current versus constant voltage operation, Water Res. 143 (2018) 367–375.
[28] J. Shim, N. Yoon, S. Park, J. Park, M. Son, K. Jeong, K.H. Cho, Influence of natural organic matter on membrane capacitive deionization performance, Chemosphere 264 (2021), 128519.
[29] J. Lee, P. Srimuk, R. Zwingelstein, R.L. Zornitta, J. Choi, C. Kim, V. Presser, Sodium ion removal by hydrated vanadyl phosphate for electrochemical water desalination, J. Mater. Chem. A 7 (2019) 4175–4184.
[30] N. Yoon, J. Kim, J.-L. Lim, A. Abbas, K. Jeong, K.H. Cho, Dual-stage attention-based LSTM for simulating performance of brackish water treatment plant, Desalination 512 (2021), 115107.
[31] J. Shim, S. Park, K.H. Cho, Deep learning model for simulating influence of natural organic matter in nanofiltration, Water Res. 197 (2021), 117070.
[32] A. Khan, E. Huerta, S. Wang, R. Gruendl, E. Jennings, H. Zheng, Deep learning at scale for the construction of galaxy catalogs in the dark energy survey, Phys. Lett. B 795 (2019) 248–258.
[33] A. Madani, R. Arnaout, M. Mofrad, R. Arnaout, Fast and accurate view classification of echocardiograms using deep learning, NPJ Digit. Med. 1 (2018) 6.
[34] J. Wang, L.-C. Yu, K.R. Lai, X. Zhang, Dimensional sentiment analysis using a regional CNN-LSTM model, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2016, pp. 225–230.
[35] A. Ramachandran, D.I. Oyarzun, S.A. Hawks, P.G. Campbell, M. Stadermann, J. G. Santiago, Comments on “Comparison of energy consumption in desalination by capacitive deionization and reverse osmosis”, Desalination 461 (2019) 30–36.
[36] C.-J. Huang, P.-H. Kuo, A deep cnn-lstm model for particulate matter (PM2. 5) forecasting in smart cities, Sensors 18 (2018) 2220.
[37] Y. Li, H. Cao, Prediction for tourism flow based on LSTM neural network, Procedia Comput. Sci. 129 (2018) 277–283.
[38] I.E. Livieris, E. Pintelas, P. Pintelas, A CNN–LSTM model for gold price time-series forecasting, Neural Comput. Applic. 32 (2020) 17351–17360.
[39] X. Wang, L. Gao, J. Song, H. Shen, Beyond frame-level CNN: saliency-aware 3-D CNN with LSTM for video action recognition, IEEE Signal Process. Lett. 24 (2017) 510–514.
[40] J. Wang, J. Xu, X. Wang, Combination of Hyperband and Bayesian Optimization for Hyperparameter Optimization in Deep Learning, 2018 arXiv preprint arXiv: 1801.01596.
[41] J. Snoek, H. Larochelle, R.P. Adams, Practical Bayesian Optimization of Machine Learning Algorithms, 2012 arXiv:1206.2944.
[42] A. Klein, S. Falkner, S. Bartels, P. Hennig, F. Hutter, Fast bayesian optimization of machine learning hyperparameters on large datasets, in: Artificial Intelligence and Statistics, PMLR, 2017, pp. 528–536.
[43] J. Bergstra, R. Bardenet, Y. Bengio, B. Kégl, Algorithms for hyper-parameter optimization, in: 25th Annual Conference on Neural Information Processing Systems (NIPS 2011), Neural Information Processing Systems Foundation, Granada, Spain, 2011.
[44] H. Liang, S. Zhang, J. Sun, X. He, W. Huang, K. Zhuang, Z. Li, Darts : Improved Differentiable Architecture Search With Early Stopping, 2019 arXiv preprint arXiv: 1909.06035.
[45] F. Chollet, Deep Learning Mit Python Und Keras: Das Praxis-Handbuch Vom Entwickler der Keras-Bibliothek, MITP-Verlags GmbH & Co. KG, 2018.
[46] D. Gunning, Explainable Artificial Intelligence (XAI) 2, Defense Advanced Research Projects Agency, 2017.
[47] Z. Yang, A. Zhang, A. Sudjianto, Enhancing explainability of neural networks through architecture constraints, IEEE Trans. Neural Netw. Learn. Syst. (2020) 1–12.
[48] H. Wan, A. Holmén, M. Någård, W. Lindberg, Rapid screening of pKa values of pharmaceuticals by pressure-assisted capillary electrophoresis combined with short-end injection, J. Chromatogr. A 979 (2002) 369–377.