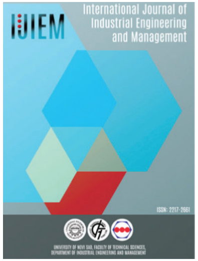
ISSN 2683-345X

journal homepage: http://ijiemjournal.uns.ac.rs/



International Journal of Industrial Engineering and Management

Volume 14 / No 2 / June 2023 / 138 - 146

Original research article

Development of a Machine Learning Model for Predicting Hardness in the Water Treatment Pharmaceutical Industry

A. A. Siahaan, M. Asrol*

Industrial Engineering Department, BINUS Graduate Program - Master of Industrial Engineering, Bina Nusantara University, Jakarta, 11480, Indonesia

ABSTRACT

The pharmaceutical industry has a water treatment process for production needs, and the softener process reduces the content of Ca2, Mg2. Few studies have been conducted to predict hardness in water. Some related studies have been undertaken to indicate lake water quality, water sulfur content, and water content in reverse osmosis output in factory water systems. This study aims to determine the prediction of hardness in water treatment systems using machine learning random forest regression and long short-term memory. The dataset is from Programmable Logic Controller records and daily sampling data from pharmaceutical factory laboratories. Machine learning models developed hyperparameter tuning processes to get the most optimal results. The best machine learning model is RFR with R2 Train 0.990 and R2 Test 0.960, while LSTM with R2 Train 0.946 and R2 Test 0.917.

ARTICLE INFO

Article history:

Received November 13, 2022
Revised May 2, 2023
Accepted May 4, 2023
Published online May 11, 2023

Keywords:
Hardness;
Long short-term memory (LSTM);
Pharmaceutical;
Random forrest regression (RFR);
Water system

*Corresponding author:
Muhammad Asrol
muhammad.asrol@binus.edu

1. Introduction

The water treatment industry generally includes multimedia filters, carbon filters, duplex softeners, reverse osmosis, and mixed beds [1]. The water treatment process in the softener tank serves to reduce water hardness. A softener tank is a tank with an anion cation resin. Hard water is water that contains anions other than bicarbonate ions. For example, it can be in the form of Cl-, NO3- and SO42- ions, which means that the dissolved compounds can be Calcium Chloride (CaCl2), Calcium Nitrate (Ca(No3)2), Calcium Sulphate (CaSO4), Magnesium Chloride (MgCl2), Magnesium Nitrate (Mg(NO3)2) and Magnesium Sulfate (MgSO4) [2].

High water hardness when passing through Reverse Osmosis (RO) filtration im pacts the membrane fouling process and the formation of scale in the piping, which causes a decrease in the flow rate so that the production of clean water will decrease. The softener process requires regeneration to reactivate the saturated or inactive Cation Anion Exchange due to removing water hardness. Currently, the pharmaceutical industry performs softener regeneration manually based on estimated time. According to the

Published by the University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia.

http://doi.org/10.24867/IJIEM-2023-2-329

World Health Organization, the hardness limit for drinking water is 500 mg/l, while for industrial water systems, the softener limit is 30 ppm [3]. Another study found a correlation between flow velocity and increased water hardness [4].

Machine learning has the potential to be applied in the industry through the use of big data and fast computing processes in decision-making. If the data processed by machine learning is done correctly in critical actions, it can be helpful information to contribute to decision-making [5]. In this case, machine learning has the opportunity to be applied in predicting the softener tank regeneration process in the pharmaceutical water treatment industry.

Previous research has conducted the use of machine learning in predicting permeate flux performance in Reverse Osmosis found that Artificial Neural Networks (ANN) have a good ability to predict the model compared to the Multiple Linear Regression (MLR) model [6]. In another study, prediction models for groundwater quality have been conducted using machine learning: Deep Learning (DL), Random Forest (RF), Gradient Boosting (GB), and Artificial Neural Networks (ANN). The best model is using deep learning with an accuracy of $R^2 = 0.996$ [7].

To solve the problem of the regeneration process of anion cation exchangers in the water treatment industry. This study is proposed to predict water hardness in softener systems. Objectives of this study:

(1) What are the attributes that affect water hardness?

(2) What is an accurate machine learning model to predict softener hardness?

The proposed machine learning prediction model utilizes Long Short Term Memory (LSTM), Random Forest Regression (RFR). Since the softener system runs in a closed system, indirect observations are made by utilizing sensors and instrumentation in the water treatment system.

This paper is organized as follows: In the first section, the background of this study is elaborated. In the second section, we outline related theory. In the third section, we detail our research methodology. In the fourth section, the result of this study is reported. Finally, in the last section, we conclude our study and discussion.

2. Related works

2.1 Recurrent Neural Network (RNN) type Long Short- Term Memory (LSTM)

RNNs are very effective in recognizing patterns in sequential data such as time series, word sequences, speech recognition, and genetic information in DNA[10]. Basic Neural Network types such as feed-forward are unsuitable for processing sequential data. This is because, in a feed-forward neural network (FFNN), data is fed separately so that most attributes of both the input and output have no dependency on each other. In addition, the flow of FNNN data processing from input to output is one-way [11] In contrast, RNNs have the memory to process samples simultaneously on each element that has a relationship with the previous calculation process.

The structure of an RNN can be seen in Figure 1. An element of an RNN module can be decomposed into a complete sequence of network processes. The parameters U, V, and W are the weight matrices used in the training data. The RNN performs an iterative function on each input to produce an output stored in memory. Inputs in a state and outputs from previous processes are considered the final decision of the entire sequence. The RNN structure can be written in the following mathematical equation [12]:

$$t_i = W_{hx}X_i + W_{hh}X_{i-1} + b_h \tag{1}$$

$$h_i = \sigma(t_i) \tag{2}$$

$$s_i = W_{oh}h_i + b_y \tag{3}$$

$$\hat{o} = g(s_i) \tag{4}$$

where x_i is the input variable at step t, W_{hx}, W_{hh} and W_o are the weight matrices at the hidden state, b_h and by are the let vectors, σ and g are the sigmoid functions or activation functions. t_i, h_i, and s_i are the temporal

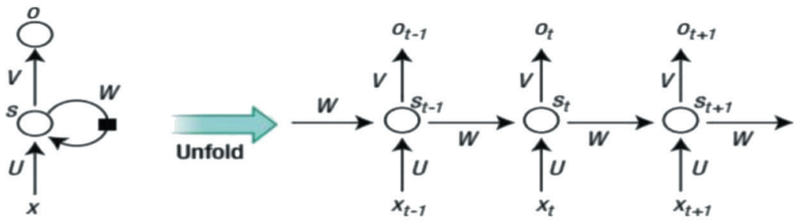


Figure 1. RNN model structure [11]

hidden variables. While $\hat{o}$ is the resulting output.

One of the problems that arise in the RNN algorithm is the failure to understand long term and dependencies if the sequence is too long due to vanishing gradient and exploding gradient [12]. Vanishing gradient occurs when changes in a parameter tend to keep shrinking over time until finally the network cannot understand changes in output [13]. While exploding gradient occurs when the input weights continue to grow during backpropagation training, thus slowing down the computational process [14].

To overcome the vanishing gradient and exploding gradient problems, a modification of RNN called Long Short-Terma Memory (LSTM) was developed. The idea of LSTM is to involve sequential data in a cell memory, but still controlled so that the gradient descent and objective function do not disappear in the training process [15]. This cell memory is what makes LSTM able to understand information in the short and long term. In LSTM, information entering the memory cell is controlled by three gates, namely the input gate, forget gate and output gate to overcome the vanishing gradient and exploding gradient problems. Figure 2 illustration of LSTM architecture.

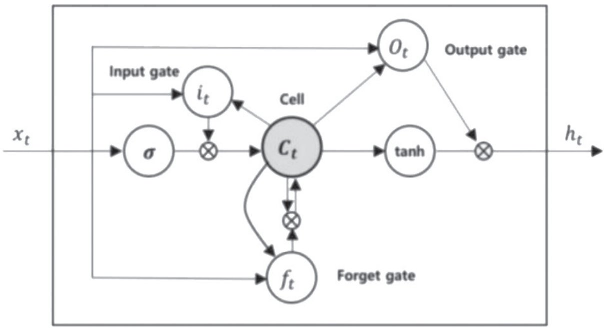


Figure 2. LSTM architecture with three gates [16]

The schematic of the LSTM and the relationship between the three gates are explained in the following steps. The input gate controls which input values can enter into the memory cells. The mathematical model for input values and candidate memory cells uses equations 5-10 [16].

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{5}$$

$$\widetilde{C_t} = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{6}$$

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{7}$$

$$C_t = i_t x \widetilde{C_t} + f_t\, C_{t-1} \tag{8}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + V_o C_t + b_o) \tag{9}$$

$$h_t = o_t . \tanh(C_t) \tag{10}$$

where I_t is the input value $\widetilde{C_t}$ is the memory cell candidate value at time t. h_{t-1} is the output value from time t-1. W_i, W_c are the input values at time t, and cell c, respectively. U_i, U_c are the weights of the output values at time t-1, and cell c-1, respectively. b_i, b_c are biases on the input gate value and cell c, respectively. σ is a sigmoid function, and tanh is a hyperbolic tangent function. f_t is the value of forget gate. W_f is the weight for the input value at time t. U_f is the weight for the output value at time t-1. b_f is the bias on the forget gate. C_t is the memory cell value. C_{t-1} is the memory cell state value at the previous cell computation. where o_t is the value of the output gate. W_o is the weight for the input value at time t. U_o is the weight for the output value from time t-1. b_o is the bias at the output gate. h_t is the value of the final output.

2.2 Random Forest Regression (RFR)

Random Forest is a supervised learning algorithm that uses ensemble learning techniques. The ensemble learning in Random Forest combines multiple aggregate decision trees to predict or classify the output of a variable [17]. The Random Forest step starts by selecting some random samples (bootstrap) from the training data subset through a process called bagging. Bagging is a technique of replacing training data by re-sampling randomly selected samples without removing the selected data from the input that will be used by the next subset [13]. The Random Forest method developed by [18] includes a set of classifier trees {h(x,Θk), k = 1,...,} from the training set, where Θk is an identically distributed random vector on the tree run for k, and x is the input vector. After several tree nodes are sequentially formed, a voting process is performed on each tree sample. The classification of the tree that gets the most votes is the output of the model. Decision function for majority voting.

$$H(x) = arg\ max \sum_{i=1}^{k} I(h_i(x) = Y) \tag{11}$$

Where $H(x)$ is some combination of classifier models, hi is each tree node, Y is the output variable, and I is the indicator function.

Random sampling allows the data to be used more than once in the training sequence of another subset, making the prediction results more stable and robust especially when facing input data variations. Data that is not selected in the bagging process for training at each tree node becomes part of another subset in the form of Out-Of Bag (OOB), which is used to predict classifier performance.

Besides being used for classification problems, Random Forest can also be used for regression problems, called Random Forest Regression (RFR). RFR estimators produce continuous-valued outputs, in contrast to classification tasks that produce categorical or binary variables (VanderPlas, 2016). RFR uses the same algorithmic steps as the Random Forest Classifier. The RFR model works by forming a tree that depends on a random variable Θ, relative to the category class, so that the predictor tree h(X, Θk) produces a continuous-valued output.

3. Methodology design

3.1 Dataset

This process explains the details of the data processing method, from raw data to data ready to be inputted into the machine learning algorithm. Before running the machine learning algorithm, preparations are made, namely the collection of sensor data downloaded from the Programmable Logic Controller (PLC). Figure 3 show a list of all of the instrument and describe the variable. Moreover, daily record data carried out in the 2016-2022 range. This data has 791 rows with ten attributes shown in Table 1. The target variable is the hardness of the softener tank obtained from the laboratory instrument analysis of the output sample from the softener.

Figure 3 shows the maximum and minimum values of the hardness target, 0.01 and 4.29. The maximum tolerance for the acceptability of hardness values is one if it exceeds the tolerance value limit. A regeneration process is required. From the data, it also knows that at a certain time after the tolerance value limit is reached, the regeneration process is not carried out so that the value read reaches 4.29.

Table 1. List of sensor variables used

Sensor	Sensor Type
PT-401	Pressure Transmitter inlet
PT-402	Pressure Transmitter outlet
DeltaPT	Differential pressure inlet and outlet
CE-401	Conductivity Transmitter inlet
CE-402	Conductivity Transmitter Outlet
TT-00	Temperature Transmitter inlet softener
ORP-01	Oxidation reduction Potential
pH-401	pH transmitter outlet Softener
TT-01	Temperature Transmitter outlet Softener

3.2 Research Flow

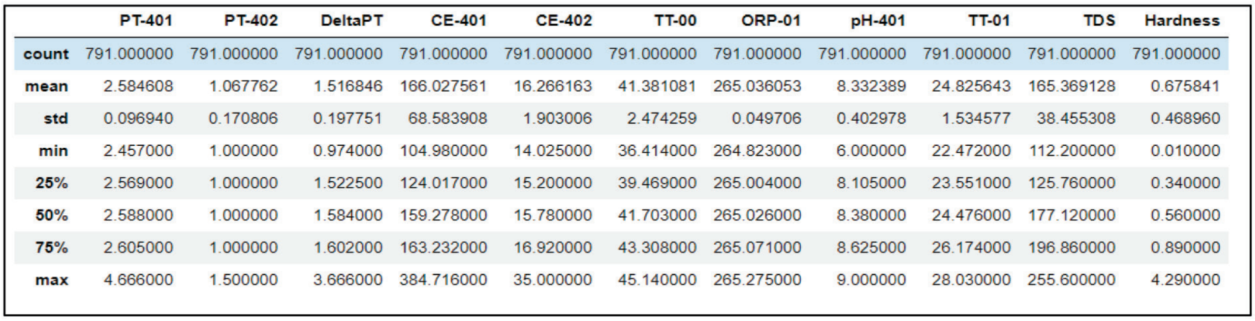
In the supervised learning algorithm process, it is necessary to separate data for the training process and model prediction. In this research, the data is divided into three parts, 70% training data, 10% model validation and 20% testing. The resulting model will be evaluated using three performance metrics for regression tasks, namely mean squared error (MSE), root mean squared error (RMSE), and coefficient of determination (R^2). The evaluation follows equations 12-13 [19], 14 [20] as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{12}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{13}$$

$$R^2 = \frac{\sum(\hat{y}_i - \bar{y})^2}{\sum(y_i - \bar{y})^2} \tag{14}$$

If R2 = 1, the regression prediction fits the actual data points perfectly. On the other hand, R2 =0 implies that the prediction around the data points' mean



	PT-401	PT-402	DeltaPT	CE-401	CE-402	TT-00	ORP-01	pH-401	TT-01	TDS	Hardness
count	791.000000	791.000000	791.000000	791.000000	791.000000	791.000000	791.000000	791.000000	791.000000	791.000000	791.000000
mean	2.584608	1.067762	1.516846	166.027561	16.266163	41.381081	265.036053	8.332389	24.825643	165.369128	0.675841
std	0.096940	0.170806	0.197751	68.583908	1.903006	2.474259	0.049706	0.402978	1.534577	38.455308	0.468960
min	2.457000	1.000000	0.974000	104.980000	14.025000	36.414000	264.823000	6.000000	22.472000	112.200000	0.010000
25%	2.569000	1.000000	1.522500	124.017000	15.200000	39.469000	265.004000	8.105000	23.551000	125.760000	0.340000
50%	2.588000	1.000000	1.584000	159.278000	15.780000	41.703000	265.026000	8.380000	24.476000	177.120000	0.560000
75%	2.605000	1.000000	1.602000	163.232000	16.920000	43.308000	265.071000	8.625000	26.174000	196.860000	0.890000
max	4.666000	1.500000	3.666000	384.716000	35.000000	45.140000	265.275000	9.000000	28.030000	255.600000	4.290000

Figure 3. Research Data Description

explains none of the data points' variability. A value outside the range of 0 to 1 occurs when the model fits the data worse than the mean horizontal hyperplane (mean for each dimension). This could indicate that the model is not an appropriate fit for the data. The MSE measures the average of the errors squared, where the error is the difference between the actual data point and the data point generated by the model. Fig 4 shows research flow of this research.

4. Methodology application

4.1 Hyperparameter Settings LSTM and RFR

Hyperparameter tuning is performed to find the most optimal set of hyperparameters for the algorithm to perform the learning process. Hyperparameter tuning is done to find the best-fitting model to improve the evaluation of machine learning. The method used in hyperparameter tuning in Random Forest Regression is the randomized grid search CV, which tries hyperparameter inputs that provide random iteration limits. The best output is determined based on the metrics used in the algorithm. This study used a number of 300 iterations with five cross-validations, resulting in 2240 fittings from 448 candidates. Table 2 shows the hyperparameter settings in this study.

While in the LSTM hyperparameter tuning process, two optimization methods are carried out, namely the Adam and Rmsprop methods; due to the stochastic nature of LSTM, it is necessary to do model fitting for 100 repetitions to get objective results. The output score obtained is the median score of the model-fitting repetition.

5. Result

5.1 Features analysis

A Pearson correlation analysis is used to investigate the relationships between hardness and the nine potential features. Only the training dataset is used to ensure that the model is not biased [21]. The correlation (R) between each potential feature and the response is shown in the correlation heat map Figure 5. The selection of features from the correlation heat map follows two principles In this process [21].

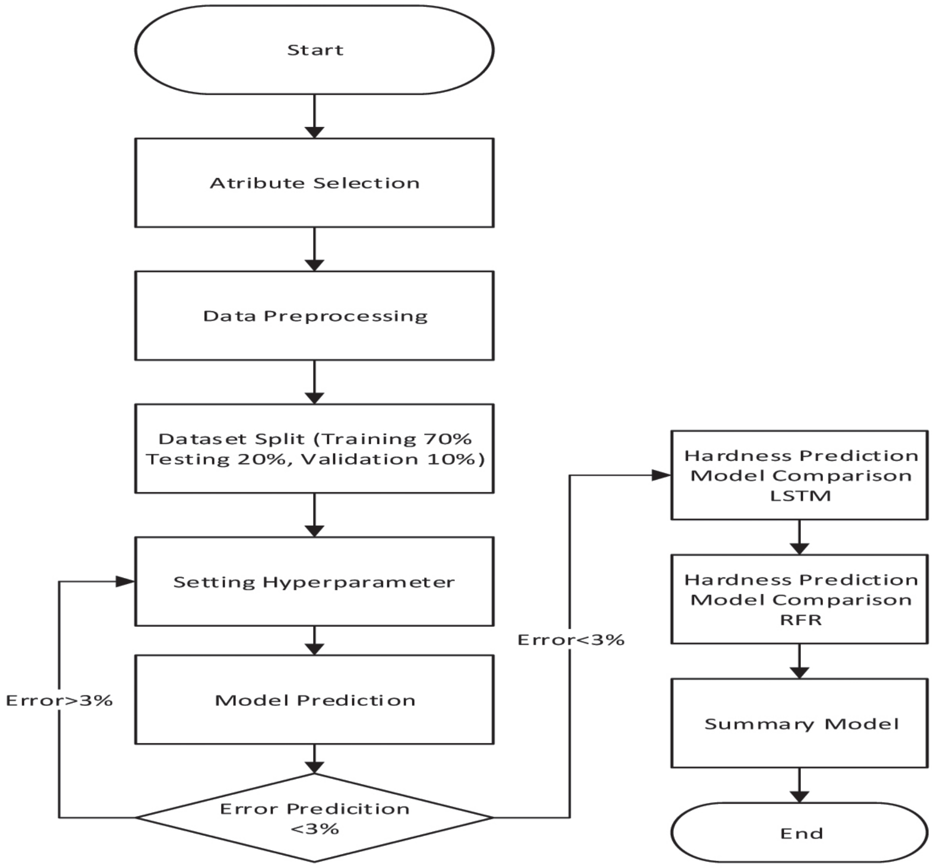

Start
Atribute Selection
Data Preprocessing
Dataset Split (Training 70% Testing 20%, Validation 10%)
Setting Hyperparameter
Model Prediction
Error>3%
Error Predicition <3%
Error<3%
Hardness Prediction Model Comparison LSTM
Hardness Prediction Model Comparison RFR
Summary Model
End

Figure 4. Research flow

Table 2. Hyperparameter setting for RFR and LSTM model

Random Forest Regression			**Long Short Therm Memory**			
Hyperparameter	Candidate Value	Optimized Value	Hyperparameter	Value	Hyper Parameter	Value
N estimator	50, 100, 150, 200, 250, 300	50	Time Lag	n=32	Dropout	0
Max features	0.3, 0.6, 0.8, 1	0.6	Imput Layer	2	Epoch	100
Max depth	20, 50, 80, 100	20	Output Layer	1	Learning Rate	0.1, 0.01, 0.001, 0.0001
Min Samples leaf	1, 5, 10, 15	1	Hidden Neuron	128	Weight decay	0
Bootstrap	True, False	TRUE	Batch Size	16	Optimizer	ADAM, Rmsprop

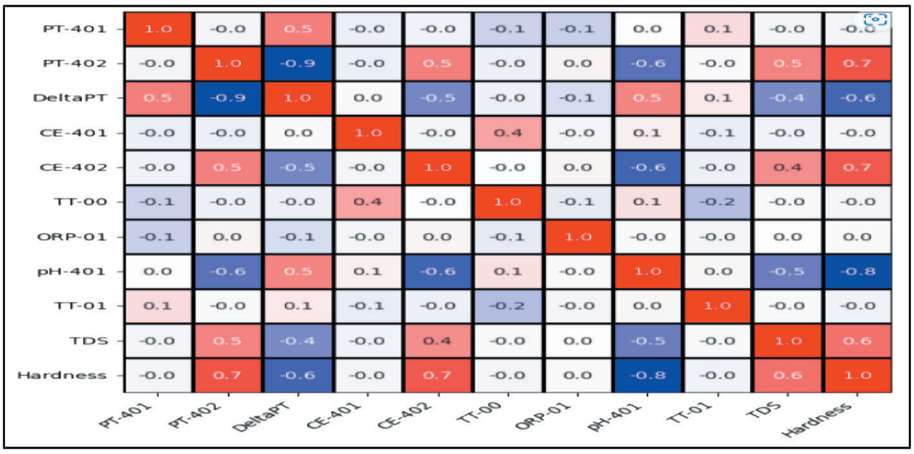


Figure 5. Correlation matrix variables and target hardness

Features that have a correlation matrix in the range of <-0.5 or >0.5 will be selected as predictor variables PT-402 (0.7), DeltaPT (0.6), CE-402 (0.7), pH-401 (0.8), and TDS (0.6). Based on the correlation matrix obtained, the features Temperature Transmitter (TT-00), Temperature Transmitter (TT-01), ORP-01, and Conductivity Transmitter (CE-401) will be removed from the predictor variables.

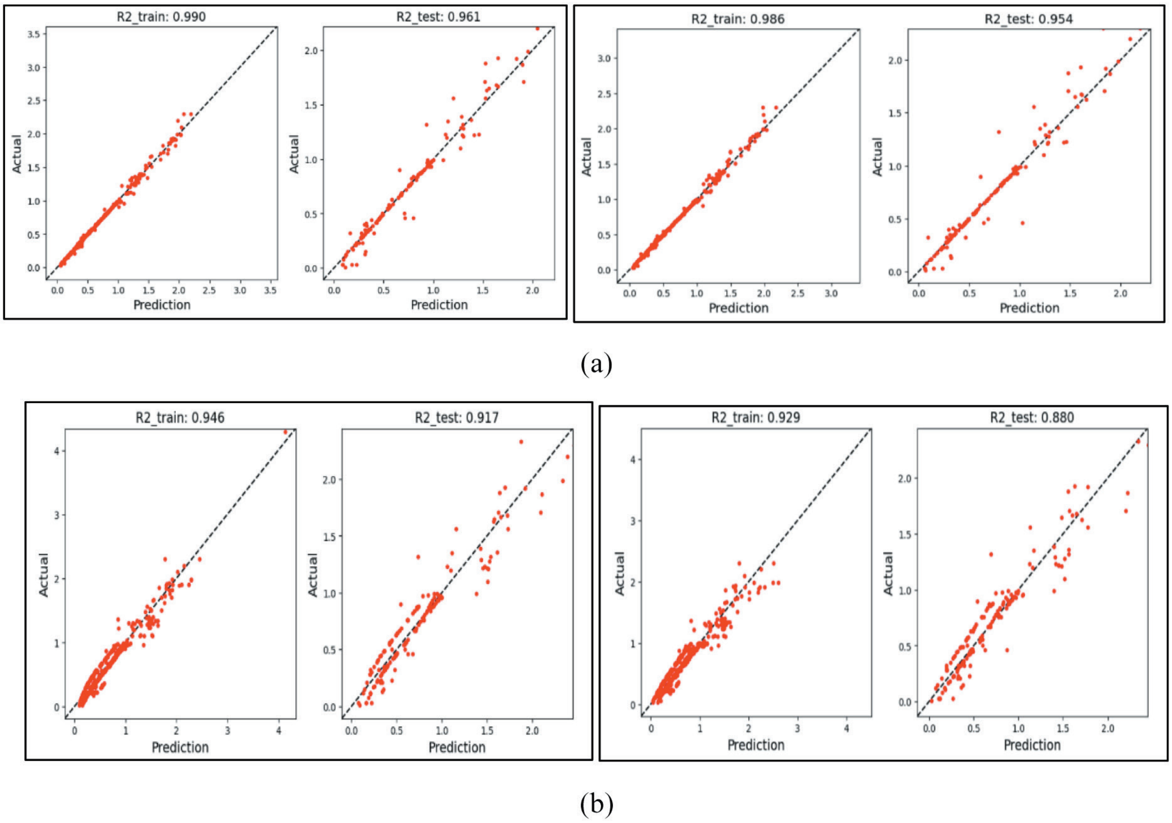
5.2 Prediction Result

In the first model, we develop the RFR baseline model and RFR with hyperparameter tuning. Figure 4 shows R2 RFR after hyperparameter tuning. For the second model, we develop LSTM with a baseline model and LSTM with a hyperparameter. Figure 6. Show result R2 LSTM and RFR before and after hyperparameter tuning.

Based on the performance of the results of the machine learning model shown in Tabel 3. the best model is random forest regression, this is because the dataset used is not a time series, and the shape of the data is tubular, so the LSTM machine learning cannot display the best performance. Research has been done by comparing machine learning models Artificial Neural Network (ANN), Genetic Programming (GP), and Support Vector Machine (SVM) to predict water turbidity in a macro-tidal coastal bay. If the data used is in the form of time series, the machine learning that produces the best performance is ANN [22]. Another research to predict phosphate in water system reservoirs has been carried out by comparing four machine learning models, namely Artificial neural network (ANN), Support vector machine (SVM), Random Forrest (RF), and Boosted tree (BT) [23]. ANN model is the best for predicting phosphate content in water system reservoirs if the data used are time series.

5.3 Transfer Learning LSTM

The result shows that all models did not suffer overfitting or underfitting based on the plots in Figure 7. Therefore, the generalization capability of all models is assured. From the plotting results of Figure 7. it can be observed that if using the optimizer



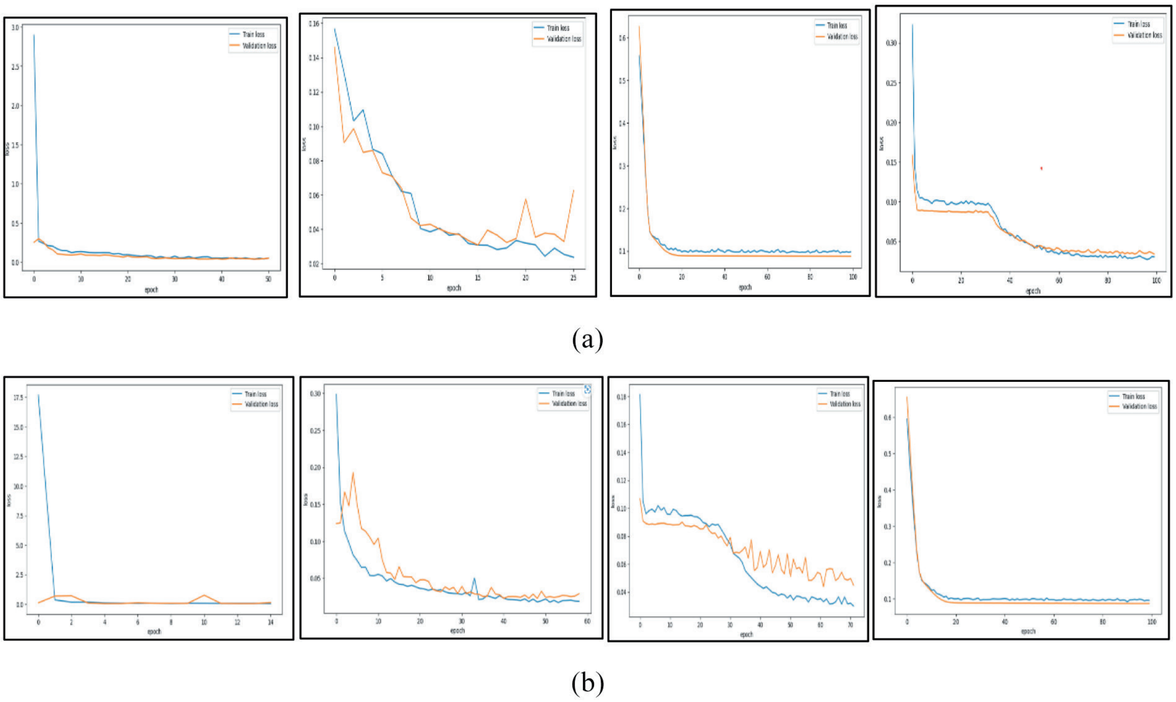
(a)

(b)

Figure 6. Result R² Based and hyperparameter RFR (a) and R² Based and Hyperparameter LSTM (b)

Table 3. Base model performance comparison with after hyperparameter tuning

	MSE	RMSE	R² train	R² test
LSTM base model	0.2536	0.3022	0.929	0.880
LSTM hyperparameter	0.1123	0.1717	0.946	0.917
RFR base model	0.0447	0.1069	0.986	0.954
RFR Hyperparameter	0.0463	0.0981	0.990	0.961



(a)

(b)

Figure 7. Learning rate with Adam Optimizer (a) from l rate 0.1-0.0001 and (b) with Rmsprop from l rate 0.1-0.0001

Table 4. Performance Optimizer model LSTM

Optimizer	I rate	MSE	RMSE	Optimizer	I rate	MSE	RMSE
ADAM	0.1	0.1761	0.2443	Rmsprop	0.1	0.3067	0.3933
	0.01	0.1511	0.2500		0.01	0.281	0.1705
	0.001	0.1210	0.1839		0.001	0.1446	0.2116
	0.0001	0.2470	0.2952		0.0001	0.2480	0.2950

method 'Adam,' the best learning rate is at a value of 0.001 and 100 epochs because it produces the smallest mean squared error value in training and validation. Meanwhile, when using the 'Rmsprop' optimizer method, the best learning rate is at an I rate of 0.01 and 59 epochs which produces the slightest mean square error of 0.1705.

6. Discussion and conclusion

Hardness is one of the crucial parameters in measuring the water quality status of water system pharmaceuticals. This study focuses on predicting models based on machine learning to capture the Hardness level in pharmaceutical water systems. The result reveals that Random forest exhibits the highest accuracy in predicting the hardness. Nine input parameters, such as Conductivity, Total dissolve solute, pressure, different pressure, and pH, have been selected as input to the proposed models. The result revealed that the Random forest regression model exhibited the highest accuracy compared with other models where RMSE= 0.0981, MSE = 0.0463, and R2 test = 0.961. Due to the absence of research on hardness in pharmaceutical water systems, other prediction methods are needed that can be applied to test the accuracy of the machine in predicting results.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] D. Askenaizer and M. Watson Engineers, “Drinking Water Quality and Treatment,” 2001.

[2] S. Sulistyani, A. Fillaeli, U. Negeri, and Y. K. Malang, “Uji kesadahan air tanah di daerah sekitar pantai kecamatan rembang propinsi jawa tengah,” 2012.

[3] H. Elfil and A. Hannachi, “Reconsidering water scaling tendency assessment,” AIChE Journal, vol. 52, no. 10, pp. 3583–3591, Oct. 2006, doi: 10.1002/aic.10965.

[4] A. Sharjeel, S. Anwar, A. Nasir, and H. Rashid, “Design, development and performance of optimum water softener,” Earth Sciences Pakistan, vol. 3, no. 1, pp. 23–28, Jan. 2019, doi: 10.26480/esp.01.2019.23.28.

[5] A. Sircar, K. Yadav, K. Rayavarapu, N. Bist, and H. Oza, “Application of machine learning and artificial intelligence in oil and gas industry,” Petroleum Research, vol. 6, no. 4. KeAi Publishing Communications Ltd., pp. 379–391, Dec. 01, 2021. doi: 10.1016/j.ptlrs.2021.05.009.

[6] J. Jawad, A. H. Hawari, and S. Zaidi, “Modeling of forward osmosis process using artificial neural networks (ANN) to predict the permeate flux,” Desalination, vol. 484, Jun. 2020, doi: 10.1016/j.desal.2020.114427.

[7] S. Singha, S. Pasupuleti, S. S. Singha, R. Singh, and S. Kumar, “Prediction of groundwater quality using efficient machine learning technique,” Chemosphere, vol. 276, Aug. 2021, doi: 10.1016/j.chemosphere.2021.130265.

[8] A. Bannoud, “The electrochemical way of removing the hardness of water,” 1993.

[9] A. Mahvi, N. Dariush, V. Forugh, and S. Nazmara, “Teawaste as An Adsorbent for Heavy Metal Removal from Industrial Wastewaters,” Am. J. Appl. Sci., vol. 2, Jan. 2005, doi: 10.3844/ajassp.2005.372.375.

[10] C. C. Aggarwal, Neural Networks and Deep Learning. Springer International Publishing, 2018. doi: 10.1007/978-3-319-94463-0.

[11] P. Goyal, S. Pandey, and K. Jain, “Unfolding Recurrent Neural Networks,” in Deep Learning for Natural Language Processing, Apress, 2018, pp. 119–168. doi: 10.1007/978-1-4842-3685-7_3.

[12] Z. Zhao, W. Chen, X. Wu, P. C. Y. Chen, and J. Liu, “LSTM network: A deep learning approach for Short-term traffic forecast,” IET Intelligent Transport Systems, vol. 11, no. 2, pp. 68–75, Mar. 2017, doi: 10.1049/iet-its.2016.0208.

[13] A. Saxena and T. R. Sukumar, “Predicting bitcoin price using lstm And Compare its predictability with arima model,” Int. J. Pure Appl. Math., vol. 119, no. 17, pp. 2591–2600, Feb. 2018, doi: 10.13140/RG.2.2.15847.57766.

[14] N. K. Manaswi, “RNN and LSTM BT - Deep Learning with Applications Using Python : Chatbots and Face, Object, and Speech Recognition With TensorFlow and Keras,” N. K. Manaswi, Ed. Berkeley, CA: Apress, 2018, pp. 115–126, doi: 10.1007/978-1-4842-3516-4_9.

[15] A. Sherstinsky, “Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network,” Phys. D Nonlinear Phenom., vol. 404, p. 132306, 2020, doi: 10.1016/j.physd.2019.132306.

[16] H. Chung and K. S. Shin, “Genetic algorithm-optimized long short-term memory network for stock market prediction,” Sustainability (Switzerland), vol. 10, no. 10, Oct. 2018, doi: 10.3390/su10103765.

[17] R. Couronné, P. Probst, and A. L. Boulesteix, “Random forest versus logistic regression: A large-scale benchmark experiment,” BMC Bioinformatics, vol. 19, no. 1, Jul. 2018, doi: 10.1186/s12859-018-2264-5.

[18] L. Breiman, “Random Forests,” 2001.

[19] G. Shmueli, P. C. Bruce, I. Yahav, N. R. Patel, and K. C. Lichtendahl Jr., Data mining for business analytics: concepts, techniques, and applications in R. Wiley, 2017.

[20] D. A. Lind, W. G. Marchal, and S. A. Wathen, Statistical techniques in business & economics. McGraw-Hill, 2017.

[21] T. Beuzen and K. Splinter, “Machine learning and coastal processes,” in Sandy Beach Morphodynamics, D. W. T. Jackson and A. D. B. T.-S. B. M. Short, Eds. Elsevier, 2020, pp. 689–710, doi: 10.1016/b978-0-08-102927-5.00028-x.

[22] Y. Wang, J. Chen, H. Cai, Q. Yu, and Z. Zhou, “Predicting water turbidity in a macro-tidal coastal bay using machine learning approaches,” Estuar Coast Shelf Sci, vol. 252, May 2021, doi: 10.1016/j.ecss.2021.107276.

[23] S. D. Latif et al., “Development of prediction model for phosphate in reservoir water system based machine learning algorithms,” Ain Shams Engineering Journal, vol. 13, no. 1, Jan. 2022, doi: 10.1016/j.asej.2021.06.009.