Journal of Water Process Engineering 64 (2024) 105674

Contents lists available at ScienceDirect

Journal of Water Process Engineering

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

Machine learning to predict the intrinsic membrane parameters in pressure retarded osmosis for an economic salinity gradient power plant

Nahawand AlZainati [a], Ibrar Ibrar [a], Ali Braytee [b], Ali Altaee [a,*], Mahedy Hasan Chowdhury [a], Senthilmurugan Subbiah [c], John Zhou [a], Adnan Alhathal Alanezi [d], Akshaya K. Samal [e]

[a] Centre for Green Technology, School of Civil and Environmental Engineering, University of Technology Sydney, 15 Broadway, NSW 2007, Australia
[b] School of Computer Science, University of Technology Sydney, 15 Broadway, NSW 2007, Australia
[c] Department of Chemical Engineering, Indian Institute of Technology Guwahati, Guwahati, Assam 781039, India
[d] Department of Chemical Engineering Technology, College of Technological Studies, The Public Authority for Applied Education and Training (PAAET), PO Box 42325, Shuwaikh 70654, Kuwait
[e] Centre for Nano and Material Sciences, Jain University, Ramanagara, Bangalore 562 112, Karnataka, India

ARTICLE INFO

Editor: Ludovic F. Dumée

Keywords:
Pressure retarded osmosis
Water permeability coefficient
Salt permeability coefficient
Membrane structural parameter
And machine learning

ABSTRACT

Pressure retarded osmosis (PRO) is highly investigated in the literature as one of the blue energy techniques. The PRO membrane plays a key role in harvesting the osmotic energy from a salinity gradient resource and process optimization. Therefore, a well-selected membrane will improve the power density generated in the PRO process to meet a designed power density threshold required for an economic salinity gradient power plant. In order to select a proper membrane for the PRO process, it is crucial to know its intrinsic properties, such as the membrane water permeability, salt permeability, and structural parameters, that impact the process performance. Determining the membrane's exact intrinsic parameters in a full-scale PRO module is challenging and time-consuming, and assuming constant parameters will compromise the accuracy of the results and power generation in the PRO process. This study employs artificial neural networks and Boosting-based tree models to predict the intrinsic parameters of the PRO membrane based on the minimum theoretical power density that could be predetermined and was assumed to be 5 W/m^2 in this study. The Random Forest and XGBoost algorithms demonstrate superior predictive power ($R^2 = 0.97$) compared to the other examined machine learning algorithms. The results reveal that machine learning algorithms can provide significant predictive power for the membrane's intrinsic parameters and power density based on the input parameters. Additionally, the algorithms were used to evaluate the feature importance of each input parameter affecting the power density of the pressure retarded osmosis membrane.

1. Introduction

Pressure retarded osmosis (PRO) is one of the blue energy techniques for sustainable energy generation from salinity gradient solutions. The osmotic pressure variation between the two PRO feed solutions is responsible for the water permeation from the low salinity to the high salinity of the PRO semipermeable membrane, where the hydraulic power is generated through the hydro-turbine [1–3]. Recently, the availability and development of promising PRO membranes for producing higher power density of the PRO process have been broadly investigated [4–8]. A PRO membrane of satisfactory water permeability and good salt rejection would ensure the cost-effectiveness of the process.

The membrane efficiency in osmotically driven systems is usually related to the membrane properties such as the membrane water permeability coefficient, the salt permeability coefficient, and the membrane structural parameter [4,9]. The membrane's intrinsic parameters highly influence the PRO process performance. For example, Zhang et al. [10] revealed that the reduction of the salt permeability coefficient caused an enhancement of 58.5 % of the PRO power density. Moreover, the salt permeability coefficient points out the severity of the reverse solute diffusion from the draw to the feed side. Nevertheless, the analytical model in PRO studies [5,11,12] ignored the effect of the reverse salt flux on the process behaviour for simplicity. The reverse salt

* Corresponding author.
E-mail address: ali.altaee@uts.edu.au (A. Altaee).

https://doi.org/10.1016/j.jwpe.2024.105674
Received 21 May 2024; Received in revised form 11 June 2024; Accepted 15 June 2024
Available online 28 June 2024

diffusion can intensify the membrane's internal concentration polarization [11–15] and the possibility of enhancing the membrane fouling [14,15]. These phenomena involve controlling the performance of the PRO process and cannot be underestimated. Recent work by Ruiz-García et al. [16] showed the impact of membrane fouling on the power generation in the PRO process. A 50 % reduction in the water permeability resulted in a 25 % decrease in power generation, whereas a 50 % reduction in the salt permeability showed insignificant impact.

Accurately determining the membrane's intrinsic parameters with the variation of the operating conditions (within the allowable range of the operating conditions from the membrane manufacturer) is necessary for the precise cost estimation of the osmotically driven membrane processes. Determining the membrane parameters affected by the process's operating conditions has been challenging. Many studies in the literature [17–22] assumed constant values of these parameters, which is not the real case. Undoubtedly, the applied hydraulic pressure is a crucial factor affecting the membrane performance, and it could produce a lower permeation rate or enhance the reverse salt flux [23–25]. The water and salt permeability coefficients cannot be constants in this case. For instance, She et al. [26] compared the PRO process outputs with two scenarios predicting the water and solute permeability coefficients. The first scenario relied on the conventional RO method. The second scenario used a new technique in which the membrane transport parameters varied within the applied hydraulic pressure changes. The results showed a preferable prediction of the experimental PRO-specific solute reverse flux when the second scenario determined the membrane transport properties. An interesting study by Ruiz-Carcia et al. [27] revealed that increasing the number of PRO modules in series led to an increase in power generation despite the pressure drop on both sides of the membrane. The study highlighted the significance of using multiple PRO membranes for maximum power generation.

Moreover, analytical PRO models assumed a constant membrane structural parameter [28,29]. Technically, the membrane structural parameter can provide reasonable knowledge of the internal concentration polarization, a significant incident in osmotically driven membrane systems. For instance, the occurrence of internal concentration polarization in the PRO process will attenuate the osmotic driving force at the membrane sides, decreasing the PRO power density [5]. Accordingly, the variation of the membrane structural parameter within the operating conditions of the PRO process should be considered. Another shortcoming of analytical models is that the effect of high salinity is often ignored when estimating membrane intrinsic parameters [30]. For instance, one study [31] reported that the membrane salt permeability coefficient increased 10 times with a 100 % increase in the solute salinity.

Alternatively, to assume constant intrinsic membrane parameters, the determination of these parameters should be addressed. Usually, the conventional RO and the FO processes determine the PRO process membrane parameters (9, 26, 31–33). The water and salt permeability coefficients are predicted using the RO process, where the feed solution is pressurized. On the contrary, the FO test defines the membrane structural parameter. However, this method is inaccurate as there are differences between the RO, FO, and PRO systems [32,33].

Additionally, physical or analytical modelling studies neglect the reverse salt diffusion in the PRO process. Consequently, some studies [32–37] presented different experimental results compared to the modelled results based on predicting the membrane parameters using the conventional technique. This information is not obtainable through traditional analytical models [34]. Several researchers have utilized numerical techniques, such as Computational Fluid Dynamics (CFD) [38], to examine the hydrodynamics and mass transfer within a PRO channel; however, CFD is not practical for large-scale analysis due to its high computational costs [39].

Leveraging ML technology to predict PRO power density using an actual PRO database can minimize the required PRO experimental work, saving the researcher's time and reducing the PRO process's operational cost. It could also be used as a prior experimental step to choose better input conditions for the best power density needed for the PRO process. Notably, the actual economically viable PRO power density deviates according to the process design, including the system capacity, feed solutions salinity, and applied hydraulic pressure on the feed solutions. Additionally, the consumed energy through the pumping and treatment of the PRO feed solutions should be considered to evaluate the net energy generation of the PRO process. The net energy generation would present the best view of the viability of the PRO process. ML models can also aid researchers in exploring novel PRO configurations, materials, and operational strategies by rapidly analyzing and generating insights from vast amounts of experimental and theoretical data.

This study aims to utilize machine learning to predict the intrinsic membrane properties to achieve the assumed threshold power density of 5 W/m^2 for an economic PRO process based on input operating conditions, which can subject the membrane's intrinsic parameters to change. The membrane characteristics, i.e., water and salt permeabilities and the structural parameter, and the power density should be built by techniques that employ factual PRO operating conditions, such as the applied hydraulic pressure, the osmotic pressure of the feed, and the draw solutions, the feed and the draw solution types, and the membrane type. A PRO database will be arranged using the experimental data available in the literature, followed by collecting the membrane parameters by assuming a theoretical power density of 5 W/m^2 for an economic PRO process [1,40]. It should be noted that the theoretical power density threshold could be adjusted to achieve an economic PRO process. The machine learning program will predict the membrane power density based on the membrane characteristics and operating parameters to allow researchers to select or fabricate PRO membranes with A_w, B, and S factors recommended to generate a power density equal to or more than the 5 W/m^2 threshold. These parameters will be predicted by four different machine learning algorithms, namely Artificial Neural Network, CatBoost, XGBoost, and Random Forest, where the results will be compared. The predicted membrane parameters and PRO power density will be compared to the published data.

2. Methods

2.1. The PRO dataset collection

PRO data, termed the PRO dataset, is a comprehensive and systematic collection of data for PRO. The dataset is provided along with the study. The PRO dataset in this study is higher than one thousand points, which is higher than the dataset points of other studies where machine learning modelling applications have been applied [41]. The PRO dataset includes 1190 instances and 24 features. The dataset was collected from 47 published papers. Notably, the conclusion will be based on these studies; however, the experimental work of the PRO system is generally limited, and most studies are from modelling works due to the PRO's expensive infrastructure. The figures were collected from the literature (Data and references provided in supplementary information, S.1). Then the online software "**WebPlotDigitizer**" was applied to extract the data from the figures and plots.

The dataset visualized in Fig. 1 shows the membrane's intrinsic parameters at a power density lower than and higher than the economically viable 5 W/m^2 at various values of the applied pressure difference of the PRO. For instance, Fig. 1a shows the power density lower than 5 W/m^2 in one of the y-axis (black) and the water permeability coefficient in the other y-axis (pink) at different hydraulic pressure values. Also, at a power density of more than 5 W/m^2, the A_w values are shown in Fig. 1b. It is worth noting that around 52 % of the collected power density data are less than 5 W/m^2. The water permeability coefficient, the salt permeability coefficient, and the membrane structural parameter were reported in the studies and considered as they are in the modelling. The data's maximum, minimum, average, median, and standard division were extracted using Python and tabulated in Table 1. The applied

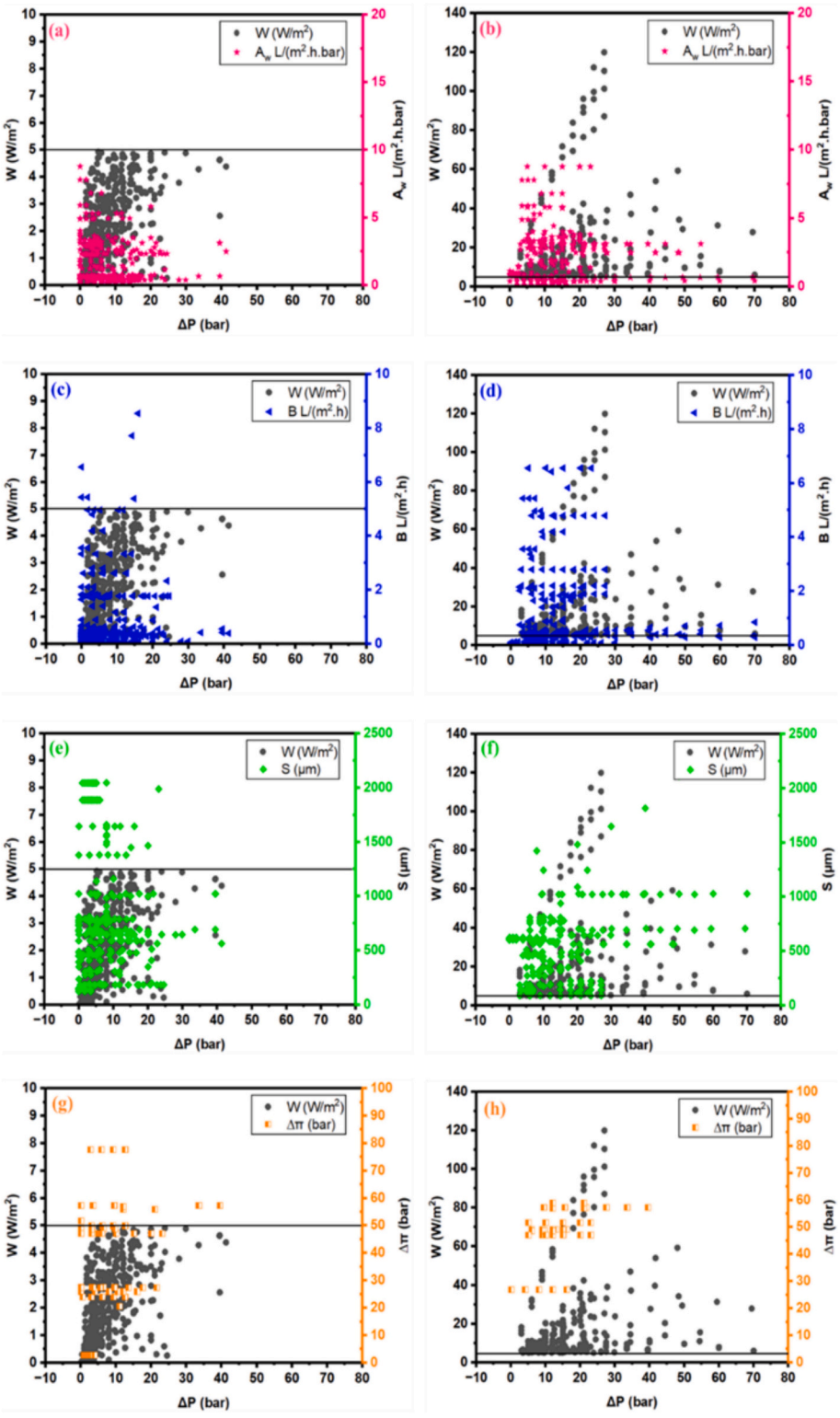


Fig. 1. (a), (b) The A_w values at W lower and higher than 5 W/m^2, respectively, with the variation of the applied hydraulic pressure difference, (c), (d) The B values at W lower and higher than 5 W/m^2, respectively, with the variation of the applied hydraulic pressure difference, (e), (f) The S values at W lower and higher than 5 W/m^2, with the variation of the applied hydraulic pressure difference, and (g), (h) The osmotic pressure difference at W lower and higher than 5 W/m^2, with the variation of the applied hydraulic pressure difference. Table 1 provides an overview of the collected data. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

pressure difference ranges from 0.41 to 70.04 bar (Table 1). Theoretically, modelling higher pressures of more than 70.04 bar can be simulated, requiring less time and no infrastructure. The maximum recorded power density value is 120 W/m^2, while the highest osmotic pressure difference is 176 bar.

Table 1
The maximum, minimum, average, median, and standard division of the numerical database.

Collected data		Maximum	Minimum	Average	Median	Standard deviation
Inputs	Draw flow rate (L/min)	40.00	0.03	2.18	0.50	5.00
	Feed flow rate (L/min)	27.58	0.02	1.33	0.50	2.81
	Applied pressure on the draw side (bar)	29.48	0.72	9.88	8.93	6.33
	Applied pressure on the feed side (bar)	6.63	0.10	1.42	0.50	1.59
	Draw salinity (M)	3.00	0.07	1.00	1.00	0.63
	Feed Salinity (M)	1.00	0.00	0.04	0.01	0.12
	Membrane thickness (m)	0.40	0.000059	0.03	0.00	0.10
	Membrane area (m^2)	650.00	0.00005	9.88	0.00	44.97
	Draw temperature (K)	343.00	283.00	298.18	297.00	8.37
	Feed temperature (K)	343.00	283.00	298.80	298.00	8.58
	The osmotic pressure difference (bar)	176.00	2.80	50.58	49.20	30.72
	The applied pressure difference (bar)	70.04	0.41	10.83	8.62	10.31
	Water flux ($L/m^2 \cdot h$)	148.40	0.11	24.30	16.47	23.47
	Reverse salt flux ($L/m^2 \cdot h$)	459.77	0.00	30.28	0.00	77.08
Outputs	Power density (W/m^2)	120.00	0.02	7.68	3.77	14.27
	Water permeability coefficient ($L/m^2 \cdot h \cdot bar$)	8.78	0.08	2.14	2.27	1.77
	Salt permeability coefficient ($L/m^2 \cdot h$)	8.55	0.01	0.78	0.32	1.21
	Membrane structural parameter (μm)	2877.81	90.00	729.07	645.94	473.51

2.2. Determination of A_w, B, and S in the literature

In the PRO studies, determining the membrane water permeability requires the water flux, the applied hydraulic pressure difference, and the osmotic pressure difference across the membrane (Eq. (1)). These were taken directly from the experimental data for accuracy, where the water flux was calculated according to Eq. (1) in the experimental studies.

$$J_w = A_w(\Delta\pi - \Delta P) \quad (1)$$

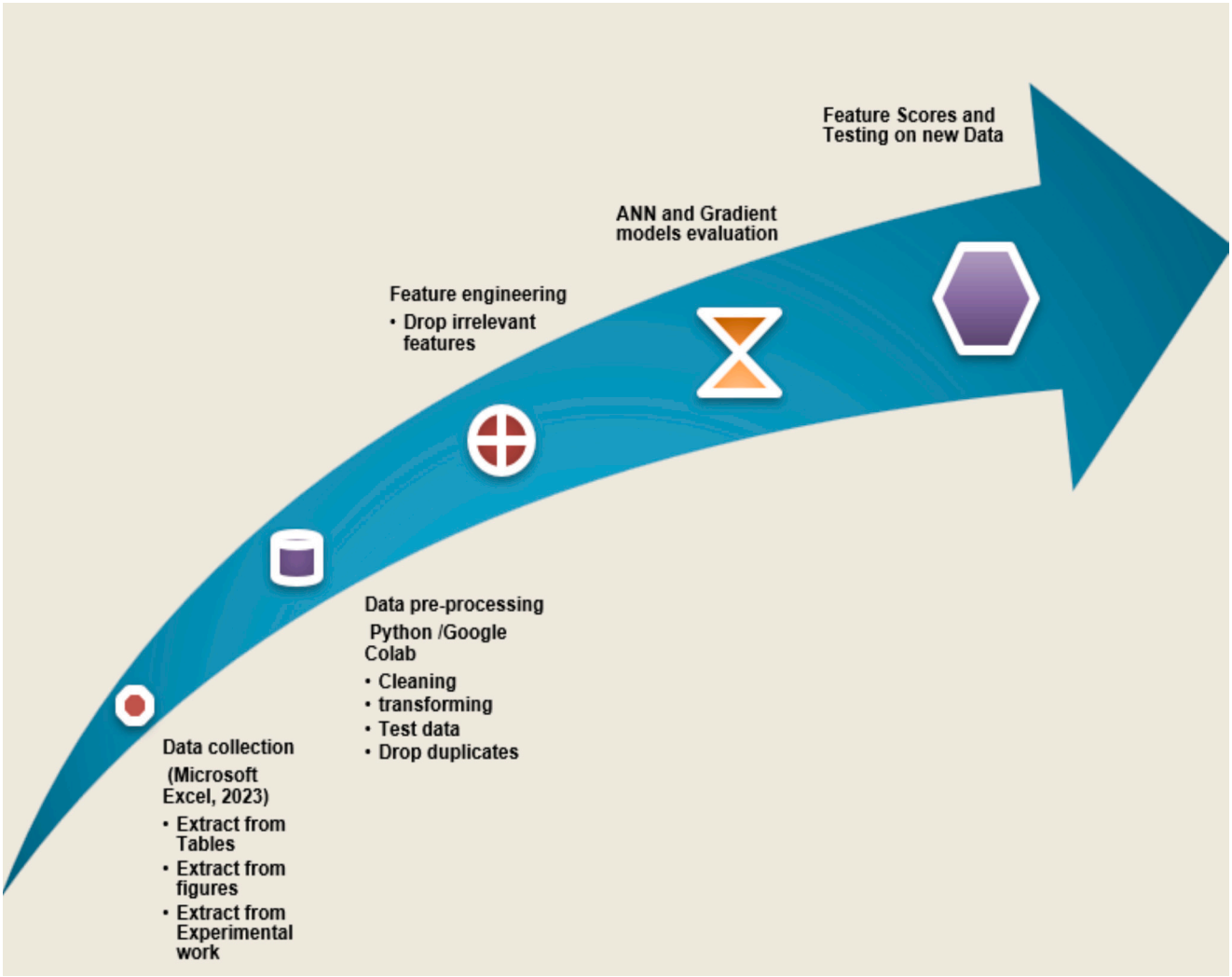


Fig. 2. Study design for machine learning models.

where J_w is the water flux, A_w is the membrane permeability coefficient (L/(m^2·h·bar)), Δπ is the osmotic pressure difference (bar), and ΔP is the applied hydraulic pressure difference (bar), respectively. Power density (W) in W/m^2, the energy generated in the PRO system by the membrane area, is reported in the studies or determined by the following equation.

$$W = JwΔP = A_w(Δπ - ΔP)ΔP \quad (2)$$

Traditionally, in all the data, Eq. (1) estimates the A_w based on the RO test when Δπ is zero [28]. On the other hand, B is estimated by the following equation [28]:

$$B = \frac{A\left(\frac{C_P}{C_F}\right)(\Delta P - \Delta\pi)}{\left(1 - \frac{C_P}{C_F}\right)} \quad (3)$$

where A is the membrane area in (m^2), C_P (M) is the permeate concentration, and C_F (M) is the feed concentration. All the points for B were collected from the literature for the given studies. The membrane structural parameter (S) was also collected from the studies. Data preprocessing will remove any outliers from the data before machine learning modelling can be applied to minimize error due to different estimation methods for A_w, B, S, and power density.

2.3. Machine learning study design and algorithms

2.3.1. Study design

The data attached (Supplementary information, S.1) was collected from PRO studies using figures and tables. The extracted data from the figures and tables was tabulated in Excel and exported to Python. After cleaning, preprocessing, and transforming the data, feature engineering was conducted to drop less relevant features in the collected data (Fig. 2). The data was subsequently fed into the ANN or gradient boosting models and evaluated with the evaluation metrics.

2.3.2. Artificial neural network architecture

A built-in library in Python called "Keras" was employed to scheme a neural network. An artificial neural network (ANN) is a deep learning algorithm built to imitate the human brain's neural networks by developing multi-connections between artificial neurons [42]. It is a subset of machine learning algorithms designed to process and learn from data. The basic processing elements of neural networks are called artificial neurons or nodes, representing features of raw input data. Each neuron in the network is a computational unit that takes the inputs, performs a weighted sum, applies a non-linear function, and generates an output.

We constructed the ANN model with a specific architecture. The training model comprised six dense layers with different numbers of neurons (512, 512, 128, 128, 128, and 64) to learn and represent complex patterns (as shown in Table 2). A rectified linear unit (ReLU) is an activation function that introduces non-linearity into the network. We further apply regularization using dropouts to prevent overfitting and improve the model's generalization ability.

The dropout rate of 0.3 in the first Dropout layer means that during training, 30 % of the input units (neurons) will be randomly set to 0 at each update, which helps prevent the model from relying too heavily on any particular subset of neurons. Similarly, the dropout rate of 0.2 in the second Dropout layer means that 20 % of the input units will be randomly set to 0 during training. The ANN model can become more robust and better generalize unseen data by using dropout. It helps prevent overfitting by forcing the network to learn more robust representations and not relying too much on specific neurons. Finally, the model ends with a dense layer containing a single neuron and an activation layer with a linear activation function suitable for regression tasks.

Table 2
ANN model architecture.

Layer type and activation	Output (shape)	Number of parameters
dense (Dense)	(None, 512)	6656
activation (ReLU)	(None, 512)	0
Dense_1 (Dense)	(None, 512)	262,656
Activation_1 (ReLU)	(None, 512)	0
Dense_2 (Dense)	(None, 128)	65,664
Activation_2 (ReLU)	(None, 128)	0
Dense_3 (Dense)	(None, 128)	16,512
Activation_3 (ReLU)	(None, 128)	0
Dropout (Dropout 0.3)	(None, 128)	0
Dense_4 (Dense)	(None, 128)	16,512
Activation_4 (ReLU)	(None, 128)	0
Dense_5 (Dense)	(None, 64)	8256
Activation_5 (ReLU)	(None, 64)	0
Dropout_1 (Dropout 0.2)	(None, 64)	0
Dense_6 (Linear)	(None,1)	65
Total parameters		
376,321		
Trainable parameters		
376,321		
Non-trainable parameters		
0		

2.3.3. ANN training procedure and hyperparameters

The ANN model was trained using the Adam (Adaptive moment estimation) optimizer with a learning rate of 0.001. The learning rate is a crucial hyperparameter that determines the step size during gradient descent optimization. The Adam optimizer adapts the learning rate during training, helping the model converge faster and potentially escape local minima. The ANN model was trained for 500 epochs (i.e., 500 passes through the entire training dataset), and the batch-size parameter was set to 256, indicating that the training data will be divided into batches of 256 samples during each training iteration. This approach helps optimize memory usage and accelerate training. The MSE was used as a loss function. In general, L2 loss (MSE) tends to provide more precise gradient information for optimization. The pretrained ANN model was loaded to make predictions with ANN, and the new test data was preprocessed similarly to the PRO dataset. Finally, predictions were obtained for the new test data and exported.

2.3.4. Gradient boosting method

Three gradient boosting models were processed in the current study: Categorical Boosting (CatBoost), Extreme Gradient Boosting (XGBoost), and the Random Forest. These models combine multiple decision trees, which leads to strong predictive power. These models are well-known for faster training and improved generalization and have been successfully used in many regression problems. For instance, CatBoost is considered a significant method for the high accuracy of the generalization [43]. Moreover, random forest is determined to be a significantly accurate method and a fast learning method without being affected by the nature of the original dataset [44]. Random Forest as an ML method is a variety of classifiers processed in decision trees, created based on a couple of randomization sources. The decision trees' forming relies on choosing the best split points to minimize the loss [45]. The input variables are the PRO process parameters that impact the water permeability coefficient, the salt permeability coefficient, the membrane structural parameter, or the power density. Furthermore, the output variable is one of the membrane's intrinsic parameters or the power density. A single tree in the gradient boosting models is utilized to build a basis function [46,47] as follows:

$$Fo(x) = arg\,min_{\beta} \sum_{i=1}^{n} Loss(y_i, \beta) \quad (4)$$

where x is the input variable, Fo is the basis function, Loss (yi, β) is the loss function, and β is the split points set for the internal nodes of the

tree. The best β value that guarantees to minimize the loss function is decided using the training data.

2.3.5. Gradient Boosting models architecture and hyperparameters

Gradient Boosting is a popular machine-learning technique used for prediction purposes. It is an ensemble learning method that builds multiple weak learners (i.e., decision trees) where each weak learner is trained sequentially to correct the errors made by the previous ones. CatBoost and XGBoost are powerful gradient-boosting models used in our experiments. RandomForestRegressor is an ensemble learning method that uses a collection of decision trees (forest) to perform regression tasks. Each tree in the forest independently predicts the output and the final prediction is obtained by averaging (for regression) the predictions of all trees. The hyperparameters of CatBoost, XGBoost, and Random Forest regression are listed in Table 3. The parameters iterations (CatBoost) or n_estimators (XGBoost and Random Forest), respectively, are similar and control the number of boosting rounds or iterations, which corresponds to the number of base learners (decision trees) built in the ensemble.

The learning_rate hyperparameter controls the step size at which the models (CatBoost and XGBoost) are updated during each boosting iteration. A lower learning rate allows the model to make more conservative updates, which can improve stability during training. For instance, the learning rate is set to 0.001 for CatBoost, which means the model will make small updates to the parameters at each iteration. The Mean squared error (MSE) was used as a loss function for all the models. In Random Forest, there is no iterative boosting procedure, and the model's parameters are not updated with a learning rate. The trees are trained independently, and their predictions are combined directly without any iterative optimization process. The depth hyperparameter specifies the maximum depth of each decision tree in the ensemble. A deeper tree can capture more complex relationships in the data but may also lead to overfitting. For instance, by setting the depth to 15 for the XGBoost, the code limits each decision tree to a maximum depth of 15 levels. Finally, by setting verbose to False or 0, no progress messages are revealed during training, making the training process silent.

2.3.6. The metrics evaluation of the model performance

Models for the water permeability, the salt permeability, the structural parameter, and the PRO power density were built and evaluated. The main metrics used to evaluate the ML models are the coefficient of determination (R^2), the mean square error (MSE), and the mean absolute error (MAE). The competence of the proposed machine learning models is examined by the coefficient of determination (R^2), between 0 and 1. The higher the R2 value is, the more efficient the model.

$$R^2 = \frac{\sum_{i=1}^{n} (P_i - \bar{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y}_i)^2} \quad (5)$$

where n is the number of data points, P_i is the output predicted from the machine learning model, $\bar{y}_i$ is the mean of the sample data, and y_i is the real output value. To determine how much the model fitting line is to the collected data, the mean square error was evaluated as the following:

Table 3
Hyperparameters used in Gradient boosting models.

Model name	Iterations/ n_estimators	Learning rate	Depth	Verbose	Booster
CatBoost	3000	0.001	10	False	gbtree
XGBoost	10	0.01	15	None	gbtree
Random Forest	100	N/A	15	0	N/A

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - P_i) \quad (6)$$

In Eq. (6), MSE is the mean square error used to evaluate the machine learning model's performance. It has been suggested that the MSE of the testing data governs the machine learning model's predictive performance. However, the importance of the MSE of the training data is to clarify and point out the reliability of the machine learning models in digging for the abnormality in the dataset. The mean absolute error (MAE) of the testing data is also included as the summation of the absolute error between the predicted data and the real output data as follows:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - P_i| \quad (7)$$

The Google Colaboratory platform, through Python 3.9 software and the built-in libraries, was operated to build and run the models. The input data present the independent variables of the machine learning models, while the output layer would be the water permeability, salt permeability, structural parameter, or power density.

2.3.7. Testing the models on new unseen data

Model performance was predicted using unseen data, and a comparison was made to illustrate the predictive powers of the best models for intrinsic membrane parameters and power density.

3. Results

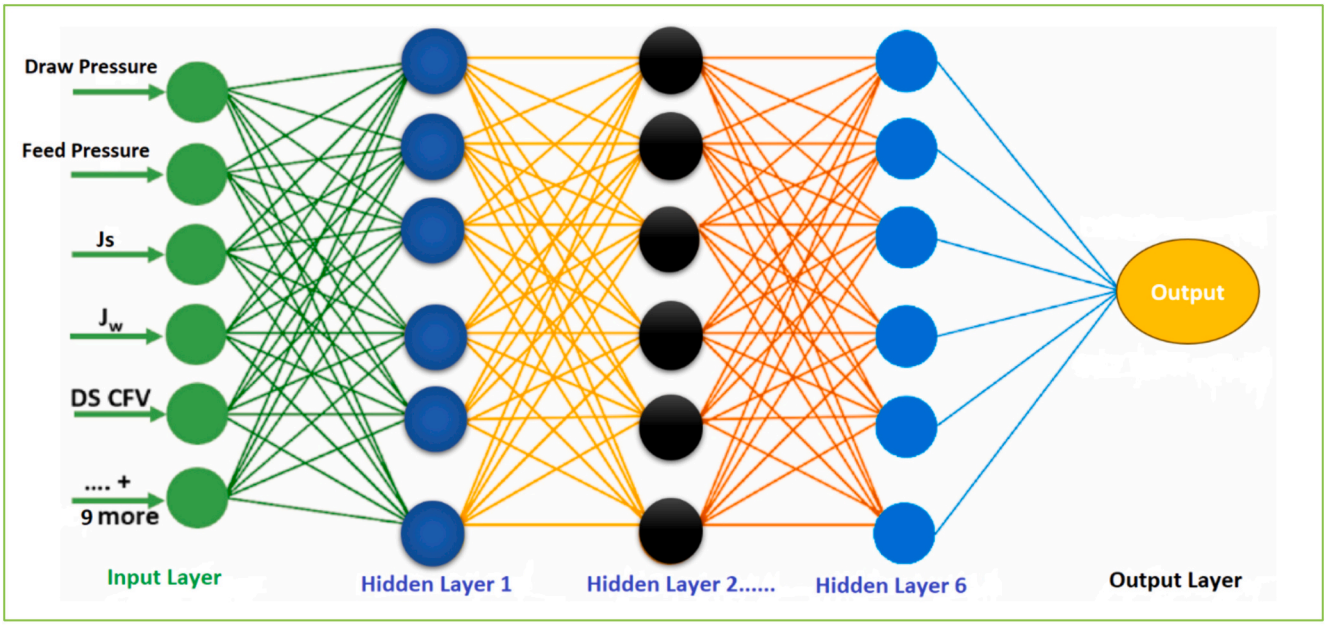
Four machine learning algorithms predict the intrinsic parameters of the PRO membrane. Furthermore, the PRO output represented by the power density is also predicted. The prediction of these parameters and output will be discussed in this section.

3.1. Prediction of the membrane's intrinsic parameters and the PRO power density using ANN

Neural networks are powerful deep learning algorithms that can find hidden patterns in data and accurately predict a process's output. Python was used to design the artificial neural network (ANN) algorithm to predict the water permeability coefficient, the salt permeability coefficient, the structural parameter, and the PRO power density. The PRO dataset was loaded into the software. The categorical variables, such as the types of the feed and draw solutions, the membrane types, or the membrane manufacturing companies, were converted to numerical data by one-hot encoding to form binary variables. Through the design of the ANN algorithm, the layers number, the algorithm learning rate, and the training epochs number were decided precisely (Epochs = 500, learning rate = 0.01) because the PRO input parameters have no direct impact on the process outputs. First, the number of hidden layers was assigned to one, and then, through trial and error, the appropriate number of hidden layers was decided. The optimal number of epochs was determined by plotting the loss function of the training data and validation data against the number of epochs, starting from 100. The input parameters, such as the draw and the feed solutions type, salinity, flow rate, temperature, hydraulic pressure, membrane type, and membrane area, were considered while designing the ANN model. The built ANN model had one input layer, three hidden layers, and one output layer to predict A_w, B, S, and power density (Fig. 3a). The RELU (Rectified linear Unit) activation function was applied to the output of each neuron from the Dense layer.

A significant predictive power of the designed ANN model was displayed for the water permeability coefficient, the salt permeability coefficient, and the structural parameter but not for the power density.

The categorical data were encoded to numbers with the help of the "LabelEncoder" transformer in Python. Early stopping occurred to prevent the model's overfitting and enhance the learning rate. Fig. 3a



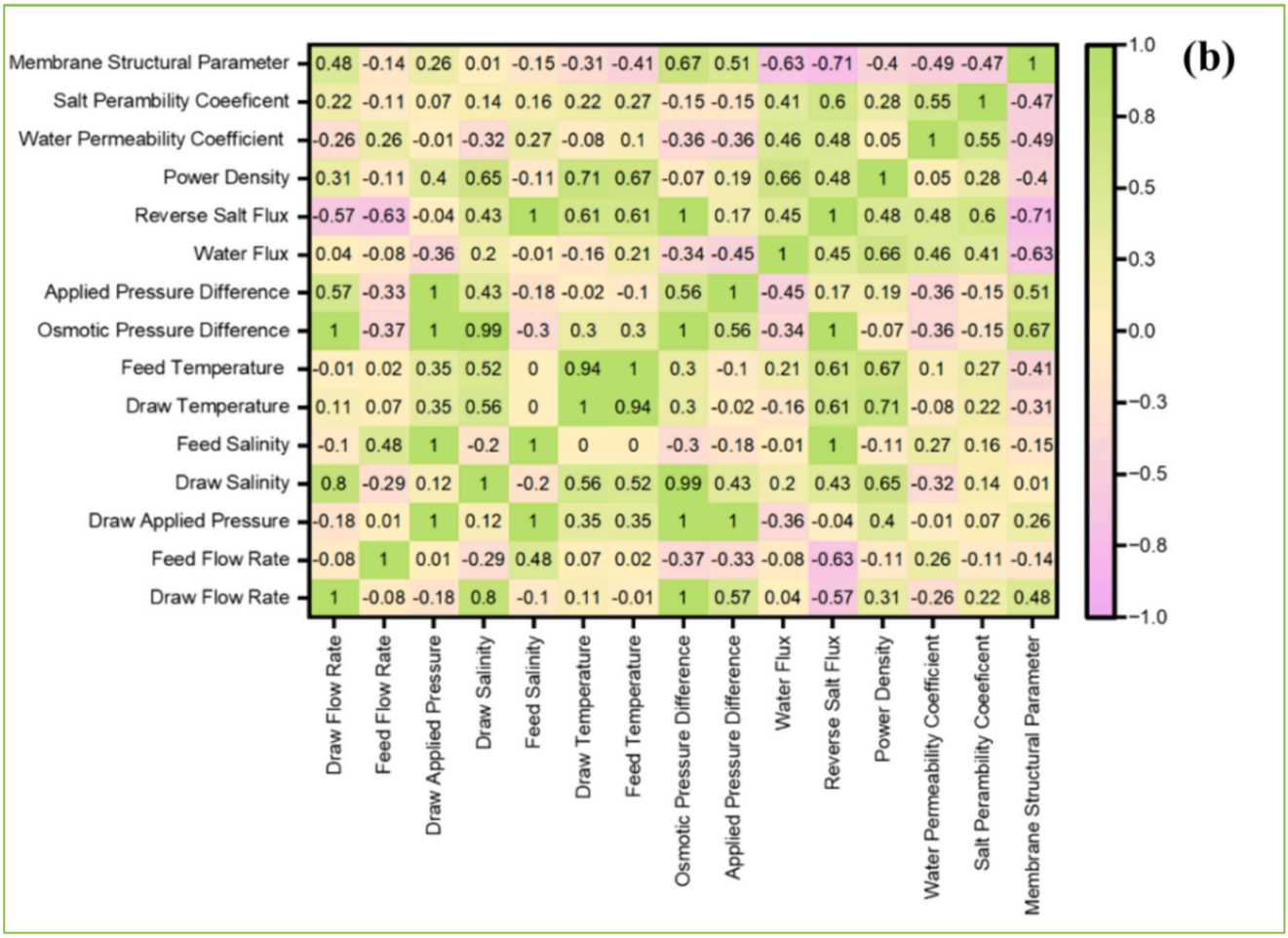


Fig. 3. a) ANN hierarchy b) The correlation matrix of the impact of the PRO input parameters of the water permeability coefficient, the salt permeability coefficient, the structural parameter, and the power density.

represents the hierarchy of the ANN model, and multiple hidden layers make the relation between the inputs and outputs complex. A correlation matrix, using a heat map, was prepared for the water permeability, salt permeability, structural parameters, and power density (Fig. 3b) to investigate further how the input parameters influenced them. A correlation factor of value close to 1 means the parameters have a high correlation and impact each other strongly. Conversely, a correlation factor of a value close to −1 means that the parameters have no or weak impact on each other. A high correlation factor of "1" was found to relate the osmotic pressure difference to the draw flow rate and the reverse salt flux with the feed salinity (Fig. 3b). The training set was around 80 % of the collected input data.

In contrast, the remaining 20 % of the data was used to test the ANN model and predict the output parameters. A larger training set, like an 80–20 split, provided the model with more varied examples, enabling it to learn more robust patterns, leading to slower convergence but potentially lower validation loss, indicating better generalization. Some input data were classified as definite: the type of PRO feed solutions, the membranes type, and the manufacturers. These were encoded with one-hot encoding in Python and normalized.

The Adam optimizer and the learning rate 0.001 simulated the ANN model. The R^2 value was found for the training and testing data, as shown in Fig. 4 for A_W, B, S, and W, and the optimization of the ANN was done using the loss function for the training set against the validation

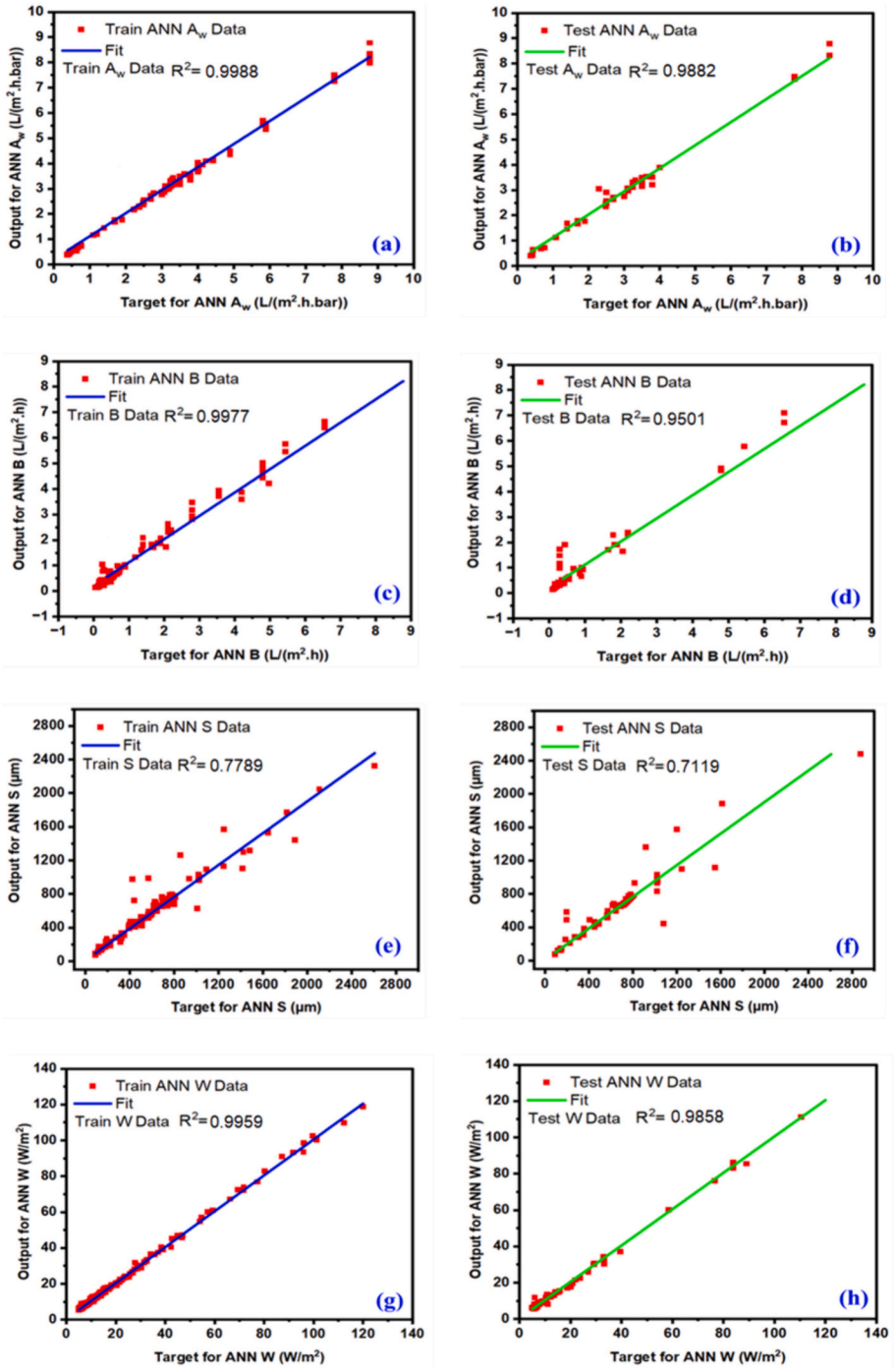


Fig. 4. The ANN R^2 predictive power for the prediction of the training and the testing data of (a) and (b) of the water permeability coefficient, (c) and (d) of the salt permeability coefficient, (e) and (f) for the membrane structural parameter, and (g) and (h) for the PRO power density, respectively.

set. The loss function against the number of epochs plot is presented in Fig. S1 (Supplementary information S1). The R^2 of the training data for the A_w is 0.9988, while the R^2 for the testing data for the water permeability coefficient is 0.9882 (Fig. 4a and b). The high R^2 value exhibits the significant performance of the ANN model for the prediction of A_w, and there are no signs of overfitting. A high R^2 value indicates that the ANN can accurately predict the A_w based on the input conditions. Excellent results of the simulation of the ANN model for the training and testing data are shown for the salt permeability coefficient, B, with R^2 of 0.9977 for the training data (Fig. 4c). Furthermore, the R^2 of the salt permeability coefficient for testing the ANN model is 0.9501 (Fig. 4d). Smaller R^2 values, compared to other output parameters, can be noticed for both training and testing data of the membrane structural parameter of 0.7789 and 0.7119, respectively, as shown in Fig. 4e and f. The coefficient of determination for the training and testing data of the power density is equal to 0.9959 and 0.9858, respectively, which again shows excellent predictive ability.

For further evaluation of the machine learning model, in addition to the R^2 metric, the MSE and MAE metrics were also examined. The MSE results are based on the ANN model for the water permeability coefficient, salt permeability coefficient, structural parameter, and power density, which are 0.03 $L/m^2{\cdot}h{\cdot}bar$, 0.07 $L/m^2{\cdot}h$, 8.78E−8 m, and 3.02 W/m^2, respectively. The results show good predictive power for the ANN testing data using the ANN model of the water and salt permeability coefficients but not for the membrane structural parameter (S) and the power density. Moreover, the MAE results were also evaluated for all the models. The MAE for the models were $A_w = 0.1$ $L/m^2{\cdot}h{\cdot}bar$, B = 0.06 $L/m^2{\cdot}h$, structural parameter = 0.0002 m, and power density = 0.90 W/m^2. According to the results, the ANN model reasonably predicts the water permeability coefficient, salt permeability coefficient, and power density.

The ANN program for the prediction of S was also investigated using the AdamW optimizer instead of the Adam and a variable learning rate instead of a fixed learning rate [48]. AdamW (W stands for "Weight Decay) is a variant of the popular Adam optimizer that introduces weight decay during optimization, which helps with stabilizing training and reducing overfitting. The R^2 value after AdamW optimization increased slightly to 0.7199 compared to 0.7119 for Adam using 500 epochs. We changed the learning rate for the S model to a variable learning rate coupled with AdamW, and the R^2 was improved to 0.86 and MSE to 4.07E−08 m.

3.2. Prediction of the membrane's intrinsic parameters and the PRO power density with the Gradient Boosting models

In the Gradient Boosting model, the previous predictions are updated based on the residual values at every single iteration, which leads to optimizing the loss function [49]. The CatBoost, the XGBoost, and the Random Forest are evaluated in this study to predict the water permeability coefficient, the salt permeability coefficient, the structural parameter, and the power density. The main benefit of using the CatBoost model is that this machine learning algorithm improves its prediction power by learning from its mistakes through simulation. Moreover, it's considered an easy algorithm, stable, requires minimum computations, and has high accuracy [50]. Additionally, it does not require addressing the categorical variables. Generally, boosting models depend on gathering several weak models to design one model with high predictive power in a greedy manner. Accordingly, there is no need for a large dataset for boosting machine learning models to learn from [51]. 80 % of the dataset was used for the training data, while the remaining data points were utilized to validate and test the Gradient Boosting models. The dataset used in the Gradient Tree Boosting models was the same as the one used for the ANN model. The CatBoost model was fitted to the training data, and the input features and target variables (A_w, B, S, and W) were specified. The model iteratively builds an ensemble of decision trees based on the gradient-boosting algorithm. The loss function against the number of iterations plot is presented in Fig. S2 (Supplementary information S1). The R^2 and MSE are determined to present the predictive power of the CatBoost model for the various output parameters.

Fig. 5 displays the performance of the CatBoost model for predicting the water permeability coefficient, the salt permeability coefficient, the structural parameter, and the power density. The R^2 of the training data for the water permeability coefficient is 0.9785, while the R^2 for the testing data for the water permeability coefficient is 0.9012 (Fig. 5a and b). A less significant fitting of the CatBoost model for the training data is shown of the salt permeability coefficient with R^2 of 0.9248 (Fig. 5c). Moreover, the R^2 of the salt permeability coefficient for testing the CatBoost model is 0.8420 (Fig. 5d). Higher R^2 values can be noticed for both training data of the membrane structural parameter of 0.9740 and the R^2 of the testing data is 0.7903, as shown in Fig. 5e and f, respectively.

A significant prediction through the CatBoost model resulted in a power density with R^2 of 0.9772 for the training data and 0.9522 for the testing data. Accordingly, the CatBoost model can predict the power density with high accuracy. Nevertheless, the MAE value of the membrane structural parameter (100.68) and the MSE and MAE values of the PRO power density (23.6 and 3.44, respectively) are relatively high. Based on these results, the CatBoost model presents reasonable prediction behaviour for the water and salt permeability coefficients with the lower predictive power of the membrane structural parameter and the PRO power density. Further investigation through additional ML models should occur for better prediction power. The predictive power of the membrane's intrinsic parameters regarding R^2 value is relatively higher for the ANN model than the CatBoost model. For instance, the predictive power for the salt permeability coefficient based on the training data related to the ANN model is around 7.31 % higher than the CatBoost model. A similar trend can be noticed for the PRO power density, as its predictive power is around 1.88 % higher for the ANN model compared to the CatBoost model.

The MSE results for the water permeability coefficient, the salt permeability coefficient, the structural parameter, and the power density were 0.31 $L/m^2{\cdot}h{\cdot}bar$ 0.35 $L/m^2{\cdot}h$, 3.43E−08 m, and 23.6 W/m^2, respectively. In contrast, the MAE results for the water permeability coefficient, the salt permeability coefficient, the structural parameter, and the power density were 0.43 $L/m^2{\cdot}h{\cdot}bar$, 0.41 $L/m^2{\cdot}h$, 100.68 m, and 3.44 W/m^2, respectively. A similar trend was observed for the MSE results of the power density using the CatBoost to the ANN model, where the MSE values are high. Nevertheless, the R^2, MSE, and MAE values of the water permeability and salt permeability coefficients present significant fitting of the CatBoost model to the testing data but not the membrane structural parameter and the PRO power density.

The Extreme Gradient Boosting model is also investigated to evaluate the membrane's intrinsic parameters and the PRO power density (Fig. 6). XGBoost is an algorithm that relies on the boosted trees, and due to the learning, the boosting procedure is accelerated, which enhances the fitting behaviour of the data [52]. The loss function against the number of iterations plot is presented in Fig. S3 (Supplementary information S1).

The random forest algorithm was also considered in this study, where random forest works on multiple decision trees to evaluate a single output that is valid for classification and regression by using the bagging technique [53]. The output of the Random Forest model was one of the investigated parameters in the current study. The main advantages of the Random Forest model are the possibility of utilizing it for large datasets and its capability of estimating the missing data and averting overfitting, which gives the model better prediction values of the outputs [53]. When using the training data, the Random Forest regression model was trained by calling the ".fit()" function [54]. This process involves adjusting the model's internal settings to find the best patterns and relationships between the input and target variables. The "n_estimators" parameter was set to 100 in this case. It determines the number of decision trees the model will build before calculating the

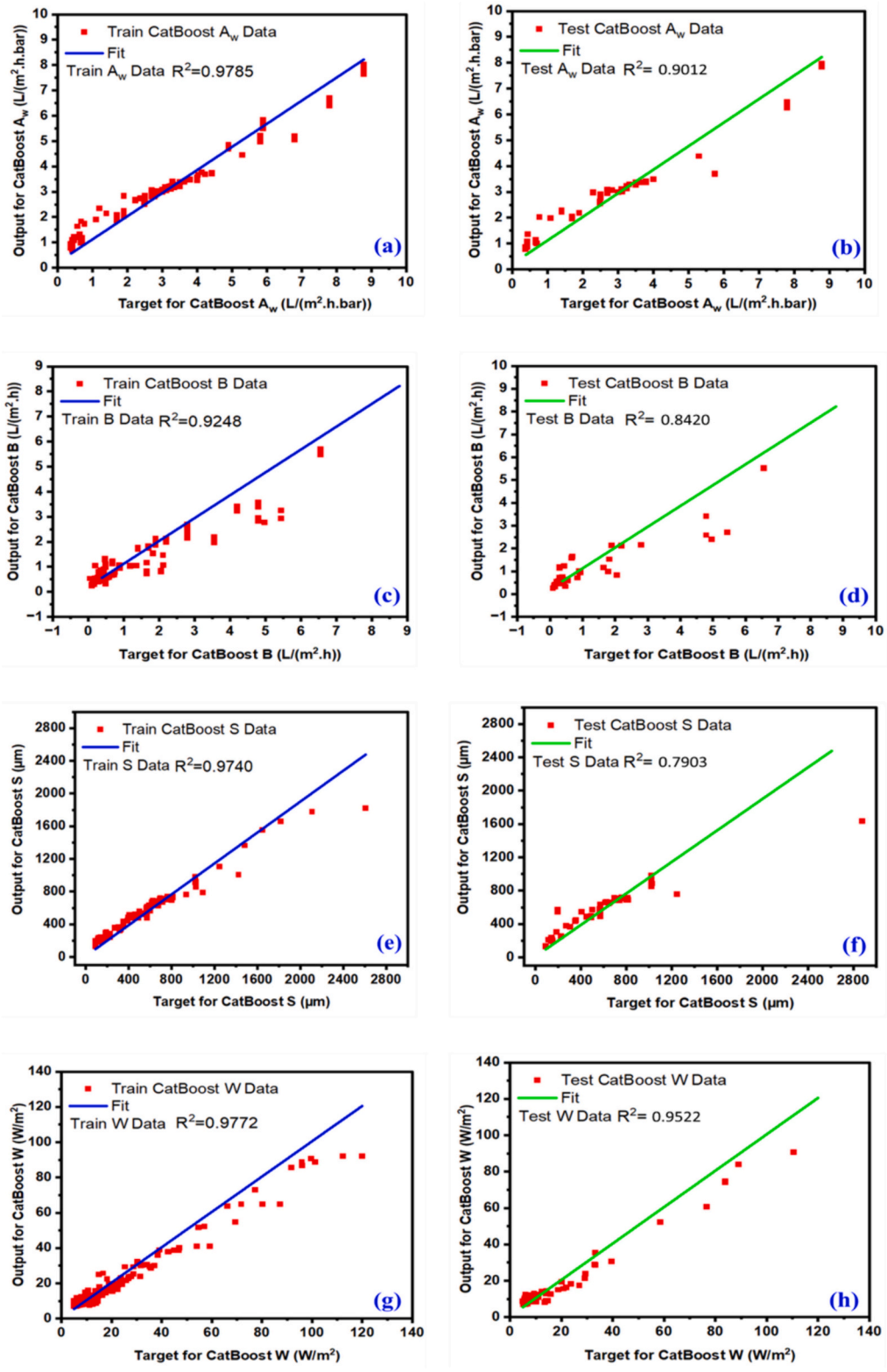


Fig. 5. The CatBoost R^2 predictive power for the prediction of the training and the testing data of (a) and (b) of the water permeability coefficient, (c) and (d) of the salt permeability coefficient, (e) and (f) for the membrane structural parameter, and (g) and (h) for the PRO power density, respectively.

average prediction. Each decision tree is a separate model contributing to the final prediction; setting "n_estimators" to 100 means that the Random Forest model will construct 100 decision trees and then combine their predictions to obtain an average value. This averaging helps to improve the accuracy and stability of the predictions made by the model. The higher the tree number, the better the model performance and the higher the prediction stability of the outputs.

Fig. 7 displays the Random Forest model's performance for predicting the water permeability coefficient, the salt permeability coefficient, the structural parameter, and the power density. The R^2 of the training data for the water permeability coefficient is 0.9958, which is 0.3 % lower than its corresponding value resulting from the ANN model. The

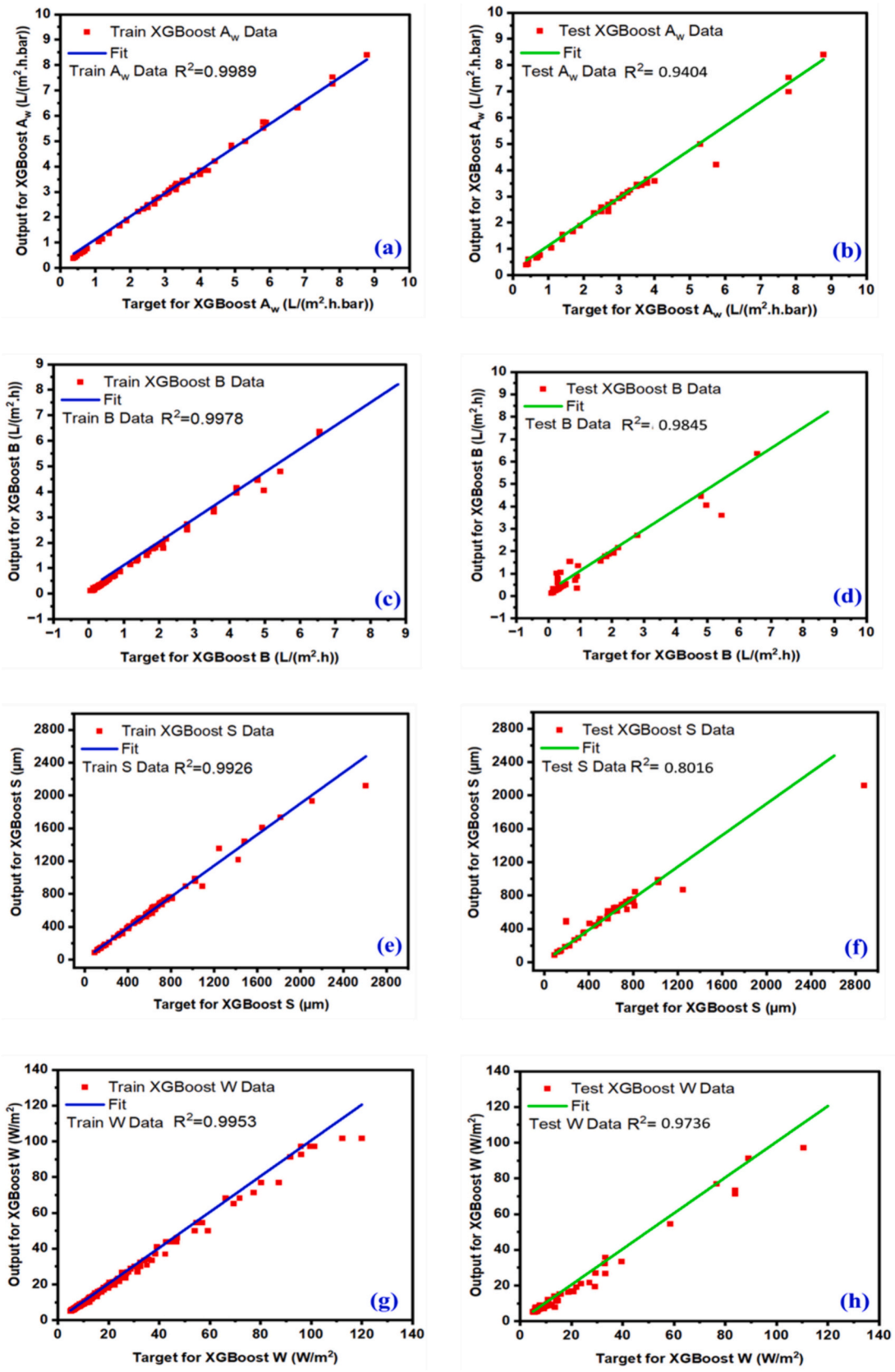


Fig. 6. The XGBoost R^2 predictive power for the prediction of the training and the testing data of (a) and (b) of the water permeability coefficient, (c) and (d) of the salt permeability coefficient, (e) and (f) for the membrane structural parameter, and (g) and (h) for the PRO power density, respectively.

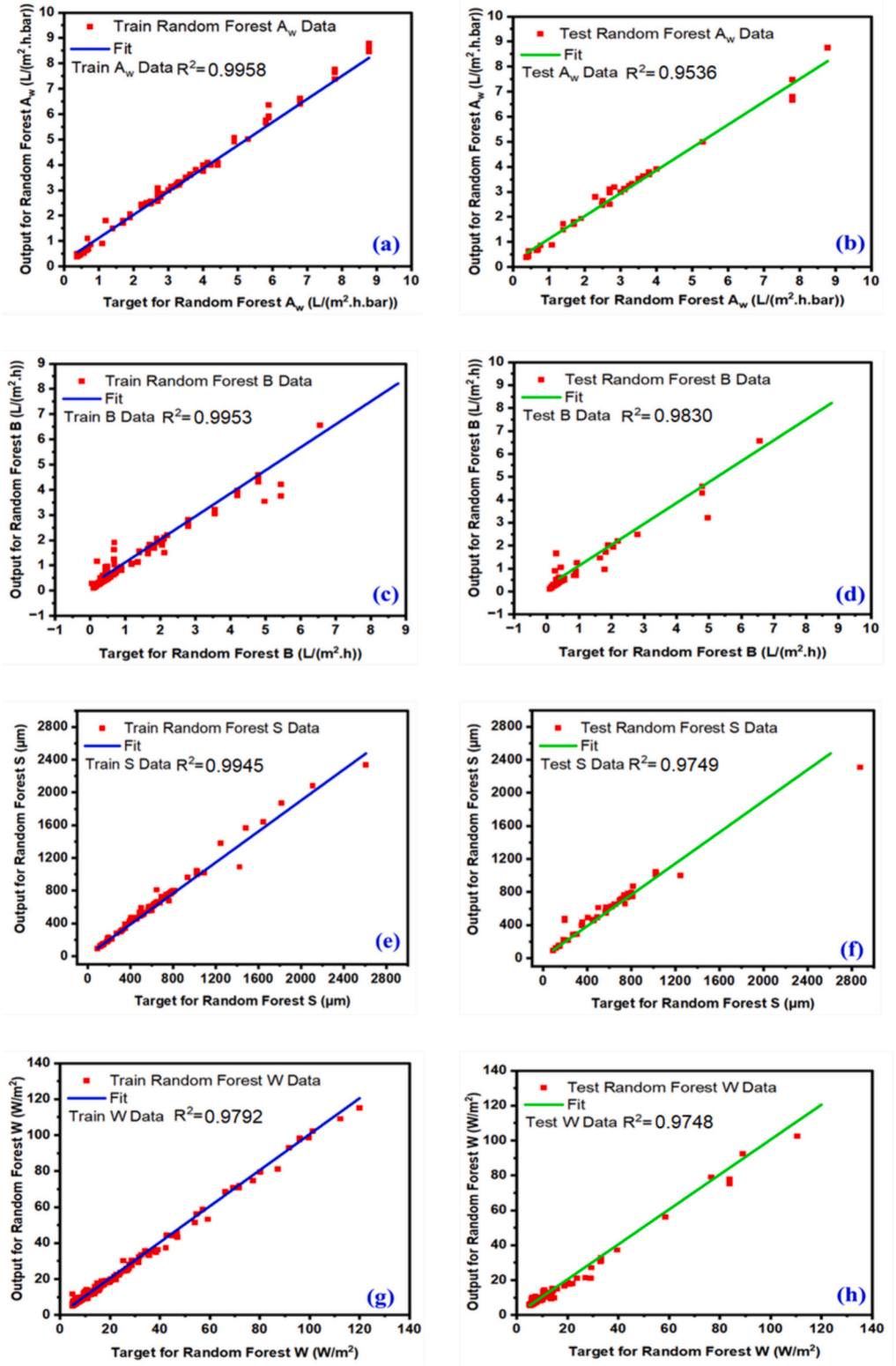


Fig. 7. The Random Forest R^2 predictive power for the prediction of the training and the testing data of (a) and (b) of the water permeability coefficient, (c) and (d) of the salt permeability coefficient, (e) and (f) for the membrane structural parameter, and (g) and (h) for the PRO power density, respectively.

R^2 for the testing data for the water permeability coefficient is 0.9536 (Fig. 7a and b). The fitting of the Random Forest model for the training data is shown of the salt permeability coefficient with R^2 of 0.9953 (Fig. 7c). It can be noticed that the R^2 of the training data for the salt permeability coefficient by the Random Forest model is higher than its corresponding values by the CatBoost model of around 7.19 %. Moreover, the R^2 of the salt permeability coefficient for testing the Random Forest model is 0.9830 (Fig. 7d), which shows the great predictive power of the Random Forest for the prediction of B. The R^2 values for the training and testing data of the membrane structural parameter are 0.9945 and 0.9749, respectively, as shown in Fig. 7e and f.

A significant prediction through the Random Forest model resulted

in the power density with R^2 of 0.9792 for the training data and 0.9748 for the testing data (slightly less than the training data). One can notice that the Random Forest model presents significant prediction behaviour for the water permeability coefficient, the salt permeability coefficient, the structural parameter, and the power density, with R^2 higher than 0.95 value. The MSE results based on the Random Forest model for the water permeability coefficient, salt permeability coefficient, structural parameter, and power density are 0.15 L/m^2·h·bar, 0.02 L/m^2·h, 7.1E−09 m, and 6.06 W/m^2, respectively. On the other hand, the MAE results based on the Random Forest model for the water permeability coefficient, salt permeability coefficient, structural parameter, and power density are 0.13 L/m^2·h·bar, 0.05 L/m^2·h, 0.00028 m, and 1.09 W/m^2, respectively. The Random Forest model's prediction power shows good behaviour for determining the membrane's intrinsic parameters. The R^2 of the training and the testing data of the ML algorithms evaluated in this study and the MSE and MAE results are summarized in Table 4 and discussed more in the following subsection.

3.3. Comparison of the ML models

Four ML models were determined to predict the membrane's intrinsic parameters and the PRO power density, where the R^2, the MSE, and the MAE values of testing data are summarized in Table 4. The higher the R^2 and the lower the MSE and the MAE values, the more significant the model's predictive power. The ANN model is the most significant in predicting the water permeability coefficient. As seen in Table 4, the ANN model achieved the highest R^2 (0.98), the lowest MSE (0.3), and the lowest MAE (0.1). The situation is different for predicting the salt permeability coefficient, in which the Random Forest and the XGBoost models showed the two best predictive behaviours. The resulting MSE value is lower by the XGBoost model compared to the Random Forest model. Slight different trends are presented for the prediction of the membrane structural parameter. The XGBoost and the Random Forest models show the best prediction power; however, the R^2 is higher for the Random Forest, while the MSE is lower for the Random Forest model. Accordingly, the Random Forest model is preferable for predicting the membrane structural parameter. Moreover, the ANN model shows the lowest predictive power of the membrane structural parameter over the other three ML models.

To summarize the results, the Random Forest model shows the best predictive power of the salt permeability coefficient and the membrane structural parameter. In contrast, the XGBoost model shows a significant predictive power of the membrane's intrinsic parameters and power density. Accordingly, the two models will be further discussed in the following subsections.

The execution times and mode sizes for all the programs were also compared. It should be noted that model training time, data manipulation, and cleaning were not considered in execution time. Amongst all the programs, XGboost, followed by Random Forest, was the lightest and fastest (Table 4). CatBoost was the slowest due to a large number of iterations (3000); however, for smaller iterations, CatBoost didn't return any satisfactory results.

While Random Forest might exhibit the best performance in terms of R2, MSE, and MAE, showcasing results from other models like XGBoost and CatBoost adds depth to the analysis. This approach demonstrates thoroughness and allows an understanding of the varying capabilities and limitations of different algorithms. For instance, XGBoost is noted for its exceptional execution speed and small model size, making it highly efficient for real-time applications. However, it might not always rank features identically to Random Forest due to its different algorithmic structure, which emphasizes boosting rather than bagging. Moreover, CatBoost, despite its longer execution time, might provide better generalization for certain datasets due to its superior handling of categorical variables.

Table 4
The R^2, MSE, MAE, execution time, and size comparison of the ML algorithms.

Parameter	Metrics/ training parameters	ANN	CatBoost	XGBoost	Random Forest
A_w (L/ m^2·h·bar)	R^2	0.98	0.90	0.94	0.95
	MSE	0.03	0.31	0.18	0.15
	MAE	0.10	0.43	0.16	0.13
	Execution time (seconds)	86.8	77.1	0.36	1.04
	Model size (megabytes)	4.36	44.7	0.01	1.39
B (L/m^2·h)	R^2	0.95	0.84	0.98	0.98
	MSE	0.07	0.35	0.01	0.02
	MAE	0.06	0.41	0.05	0.05
	Execution time (seconds)	90.3	60.09	0.04	0.47
	Model size (megabytes)	4.38	45.90	0.07	1.39
S (m)	R^2	0.71	0.79	0.80	0.97
	MSE	8.78E−08	3.43E−08	5.49E−08	7.1E−09
	MAE	0.0002	100.68	0.00016	0.00028
	Execution time (seconds)	148	55.3	0.04	0.41
	Model size (megabytes)	4.38	44.79	0.01	1.40
Power density (W/m^2)	R^2	0.98	0.95	0.97	0.97
	MSE	3.02	23.6	7.08	6.06
	MAE	0.90	3.44	1.80	1.09
	Execution time (seconds)	85.5	56.1	0.10	0.55
	Model size (megabytes)	4.38	45.8	0.01	1.39

3.4. Features importance

The feature importance is a metric parameter that can be estimated in the tree-based algorithms to determine further the importance of the input parameters for modelling the outputs. Features with high scores impact the output more than features with low scores. The importance of the different PRO operating parameters can be evaluated with ML models to predict the membrane's intrinsic parameters and the PRO power density. Determining the importance of the feature and selecting the best features can enhance the predictive power of the model and its performance by including the most significant input parameters. Here, the feature importance is defined for the XGBoost and the Random Forest models only since these models exhibited the best predictive powers compared to the CatBoost and the ANN models. Fig. 8 represents the XGBoost, and the Random Forest features the importance of the training data for (a) and (b) the water permeability coefficient, (c) and (d) the salt permeability coefficient, (e) and (f) the structural parameter, and (g) and (h) the PRO power density, respectively. The highest feature importance of the water permeability coefficient is feed type equal to 25 for the XGBoost (Fig. 8a) and 0.035 for the Random Forest (Fig. 8b), followed by water flux, which is 15 for the XGBoost (Fig. 8c) and 0.20 for the Random Forest (Fig. 8d). This observation can reveal the high impact of the feed type and the water flux on the evaluation of the water permeability coefficient. Further, the lowest feature importance resulted in the applied pressure difference, which clarifies that this parameter has the least impact on the water permeability coefficient.

It can be noticed that the highest feature of importance for the salt permeability coefficient is the water permeability coefficient; this can be related to the fact that the water permeability coefficient usually affects the water flux. In return, the water flux impacts the amount of water that permeates through the membrane, affecting the reverse salt flux and the salt permeability coefficient. For instance, Xiaoxiao et al. suggested a direct relationship between the water and the salt permeability

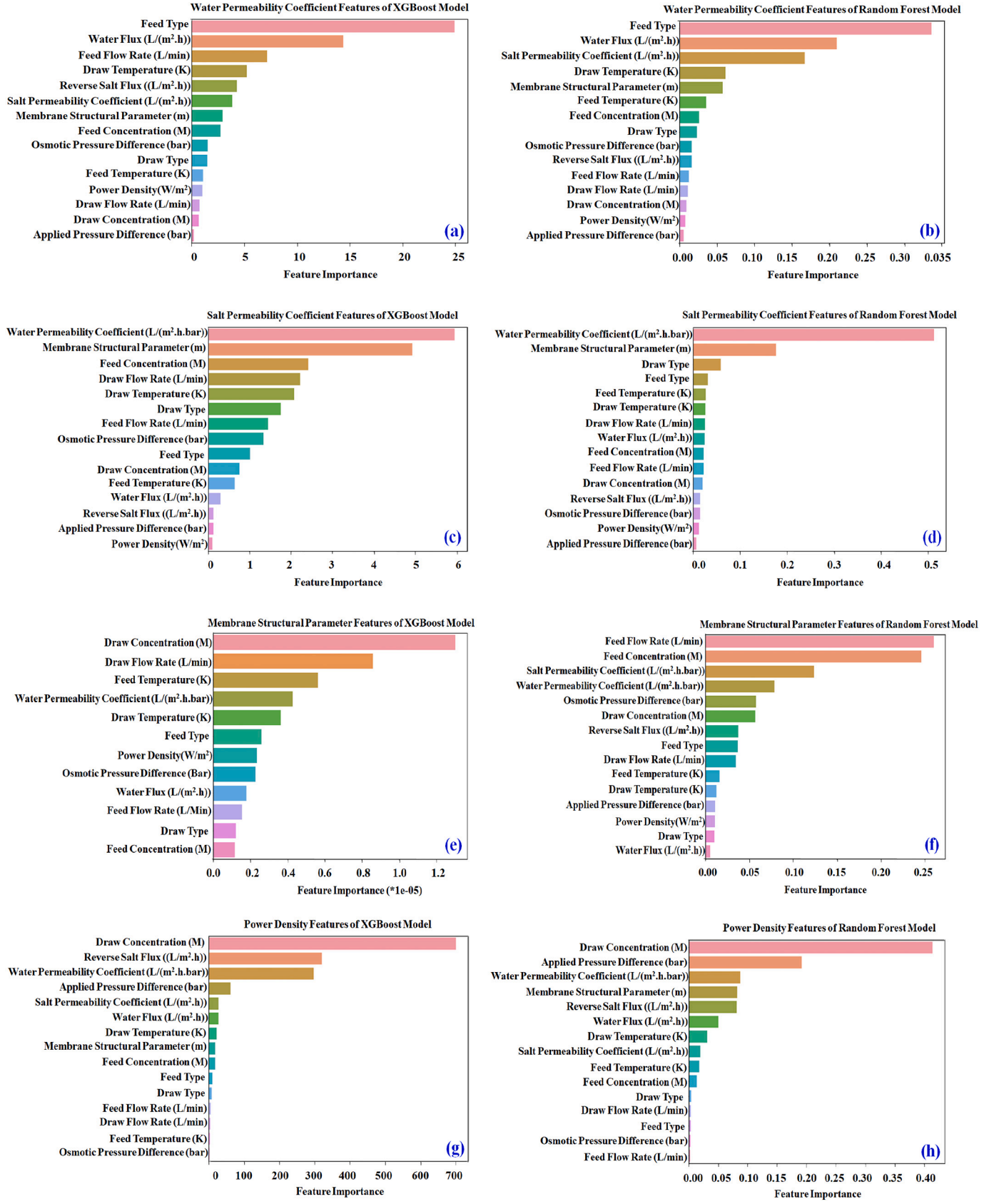


Fig. 8. (a) and (b): The feature importance of the water permeability coefficient of the XGBoost model and the Random Forest model, respectively; (c) and (d): The feature importance of the salt permeability coefficient of the XGBoost model, and the Random Forest model, respectively, (e) and (f): The feature importance of the membrane structural parameter of the XGBoost model, and the Random Forest model, respectively, and (g) and (h): The feature importance of the PRO power density of the XGBoost model, and the Random Forest model, respectively.

coefficients for several thin film nanofiber composite membranes [55]. The lowest feature importance is for the power density and the applied pressure difference (Fig. 8c and d).

The situation is different regarding the membrane structural parameters of the XGBoost and Random Forest models. The XGBoost model shows the highest feature importance for the draw concentration (Fig. 8e). The concentration of the draw solution influences the diffusivity of the solution through the membrane. As the membrane structural parameter is governed by the equation of the solute diffusion resistivity (solute diffusion resistivity = membrane structural parameter/diffusion coefficient of the support layer) [56], it is affected by the variation of the draw concentration, as shown in Fig. 8e. Interestingly, the highest feature importance of the membrane structural parameter found by the Random Forest model is the feed flow rate (Fig. 8f). The flow rate of the feed solution affects the water flux and the permeation rate. Accordingly, the feed flow rate also impacts the solute resistivity to diffusion and the membrane structural parameter [56].

Fig. 8g and h show that the draw concentration is the most important parameter affecting the PRO power density because the concentration of the draw solution governs the osmotic pressure difference across the membrane, and the latter directly relates to the water flux [57]. Thus, the power density, in turn, is directly impacted by the water flux, which explains the feature importance of the draw concentration on the PRO power density.

3.5. New insights for power density modelling

The Random Forest regression is the best model to predict power density accurately. The feature importance scores can provide new insights into unexpected relationships between the input and output parameters. As shown in Fig. 8h, the top six important features for power density predictions are the draw solution concentration, applied pressure difference, water permeability coefficient, membrane structural parameter, reverse salt flux, and water flux. The influence of draw solution concentration and applied pressure difference is well-established in the PRO literature [58]. The effect of the water permeability coefficient on power density is also substantial and agrees with previous work done by Achilli, Cath [59]. However, according to this study's results, the draw solution concentration and applied pressure difference affect the power density more than the pure water permeability value. The reverse salt flux (a feature often neglected by analytical models) has a similar feature importance score as the membrane structural parameter, slightly less than the pure water permeability of the PRO membrane, and a surprisingly higher feature importance score than the water flux of the PRO membrane. Membrane scientists often emphasize improving membrane water permeability and structural parameters, ignoring reverse salt diffusion's impact. The accumulation of draw solution on the membrane surface due to reverse salt flux may cause membrane deformation [60], leading to different A_w, B, and S values for the PRO membrane than predicted with RO tests in analytical models. Therefore, an ideal draw solution in the PRO applications should have high osmotic pressure and low reverse salt flux. As evident from the PRO dataset (e-component file, supplementary), most researchers have investigated NaCl draw solution, which has high reverse salt flux and can impact the output power density. Furthermore, the draw solution temperature has a higher feature score than the feed solution temperature. Brines discharged from thermal desalination or wastewater plants usually have higher temperatures [61], and can be potentially used as a draw solution in the PRO process to optimize the power density in future applications.

3.6. Testing the models on new data

The PRO water flux is determined through the solution diffusion model in the literature. For further consideration, the machine learning models predicted the efficiency of the machine learning prediction power of the water permeability coefficient, the salt permeability coefficient, the membrane structural parameter, and the PRO power density. Then, the prediction results were compared to the experimental results in the literature for different lab-fabricated and commercial membranes. Fig. 9 shows the percentage difference between the experimental and predicted data by the machine learning models of the membrane's intrinsic parameters and the power density. Fig. 9 compares the ANN model, the XGBoost model, and the Random Forest model. The testing data here differs from the original model training dataset. The highest percentage error of prediction A_w by the ANN, the XGBoost, and the Random Forest models is 4.01 %, 5.09 %, and 2.36 %, respectively (Fig. 9a). A higher percentage error was noticed for the prediction of B using the ANN and the Random Forest model with a maximum error of 7.75 % and 3.5 %, respectively, compared to the XGBoost model with the highest error of 0.75 % (Fig. 9b). The Random Forest model shows the best prediction of S compared to the other models, with a percentage error of less than 1.82 % (Fig. 9c).

Overall, the excellent agreement between the modelled water permeability coefficient, the salt permeability coefficient, the membrane structural parameter, and the predicted ones by the Random Forest model showed a percentage error of around 6.61 % only, compared to 6.86 % and 15.4 % for the XGBoost model and the ANN mode, respectively. The results emphasize the reliability of the Random Forest model and its significant fitting with the additional testing data. On the other hand, the ANN model and the XGBoost model show better behaviour than the Random Forest model for predicting the PRO power density (Fig. 9d). The users can predict the membrane A_w, B, and S and whether power density will be economically viable, using simple input initial conditions by providing all the input parameters.

4. Conclusion

While PRO experiments require a lot of infrastructure, modelling studies are abundant in the literature. However, few models have considered the impact of all input parameters on the power density. The outcomes of this study will assist researchers and scientists in determining the characteristics of the PRO membrane from an available dataset collected from literature and could be continuously expanded. The machine learning algorithms assisted in finding the intrinsic membrane parameters so the power density would meet a predesigned threshold for an economic PRO process. Researchers and scientists using the proposed approach in this study can select a suitable membrane for their application based on the type of salinity gradient and required power density.

The prediction of the performance of the PRO process has been conducted through collected PRO experimental data with multiple physical and chemical operating conditions, such as the applied hydraulic pressure and the different types of PRO feed solutions. Four machine learning models were performed to predict the membrane's intrinsic parameters and the PRO power density. The investigated ML models show a high predictive power of the outputs based on the PRO operating conditions. The XGBoost and Random Forest models revealed the best predictive power over the other models for predicting the membrane's intrinsic parameters and the PRO power density. The XGBoost and the Random Forest models achieved a significant R^2 value of 0.97 for the prediction of the power density. The user can predict the membrane's intrinsic parameters and the PRO power density with the membrane's characteristics and input operating conditions. The thing that enhances the design of the PRO membranes with no requirement of performing PRO experiments. It should be mentioned that the more accurate the collected data, the more precise the predictive power of the ML models. Accordingly, the data collection should occur with high accuracy to guarantee the best prediction behaviour through the ML models. Future work should investigate machine learning algorithms for energy optimization and design optimization of PRO.

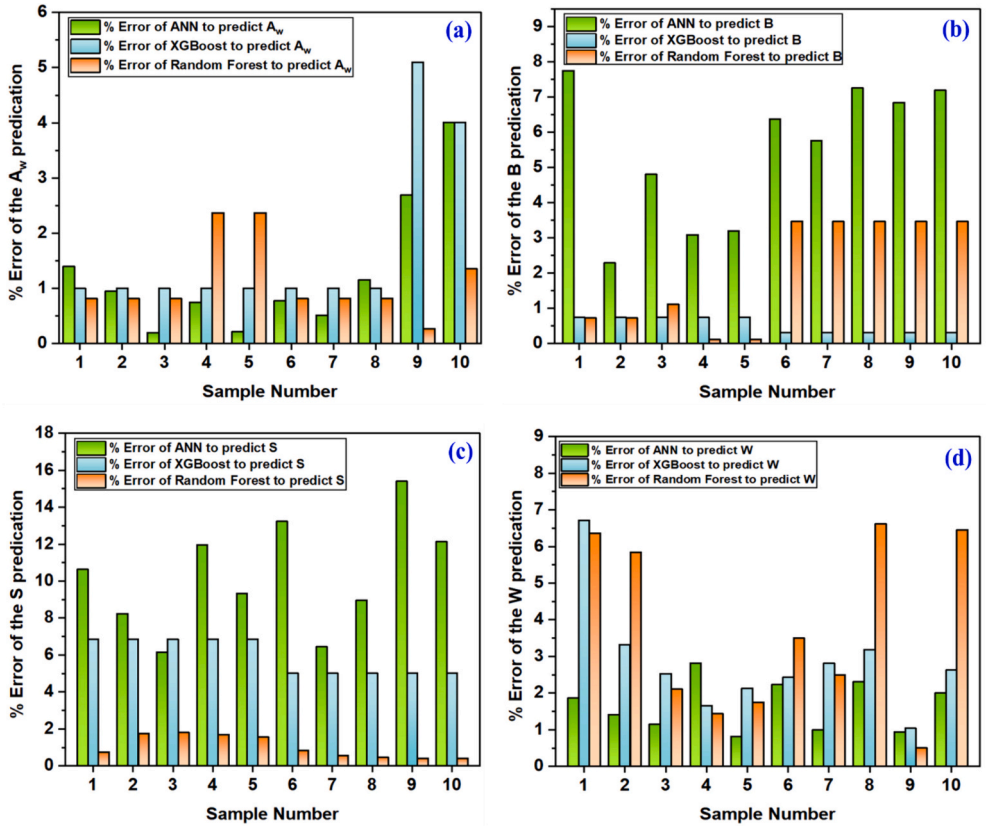


Fig. 9. The percentage error of the ANN, the XGBoost, and the Random Forest models for validation data of prediction (a) the water permeability coefficient, (b) the salt permeability coefficient, (c) the membrane structural parameter, and (d) the power density.

CRediT authorship contribution statement

Nahawand AlZainati: Writing – review & editing, Writing – original draft, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Ibrar Ibrar:** Writing – original draft, Software, Formal analysis, Data curation, Conceptualization. **Ali Altaee:** Writing – review & editing, Writing – original draft, Validation, Supervision, Resources, Methodology, Data curation, Conceptualization. **Mahedy Hasan Chowdhury:** Validation, Formal analysis, Data curation. **Senthilmurugan Subbiah:** Software, Methodology, Data curation. **John Zhou:** Writing – original draft, Supervision, Investigation, Data curation. **Adnan Alhathal Alanezi:** Writing – original draft, Validation, Data curation, Conceptualization. **Akshaya K. Samal:** Writing – original draft, Validation, Formal analysis, Data curation.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jwpe.2024.105674.

References

[1] Nahawand AlZainati, et al., Pressure retarded osmosis: advancement, challenges and potential, J. Water Process. Eng. 40 (2021).
[2] Ali Altaee, et al., Single and dual stage closed-loop pressure retarded osmosis for power generation: feasibility and performance, Appl. Energy 191 (2017) 328–345.
[3] Ali Altaee, et al., Evaluation the potential and energy efficiency of dual stage pressure retarded osmosis process, Appl. Energy 199 (2017) 359–369.
[4] Ngai Yin Yip, et al., Thin-film composite pressure retarded osmosis membranes for sustainable power generation from salinity gradients, Environ. Sci. Technol. 45 (10) (2011) 4360–4369.
[5] Shuren Chou, et al., Thin-film composite hollow fiber membranes for pressure retarded osmosis (PRO) process with high power density, J. Membr. Sci. 389 (2012) 25–33.
[6] Gang Han, et al., High performance thin film composite pressure retarded osmosis (PRO) membranes for renewable salinity-gradient energy generation, J. Membr. Sci. 440 (2013) 108–121.
[7] Shuren Chou, Rong Wang, A.G. Fane, Robust and high performance hollow fiber membranes for energy harvesting from salinity gradients by pressure retarded osmosis, J. Membr. Sci. 448 (2013) 44–54.
[8] Edvard Sivertsen, et al., Pressure retarded osmosis efficiency for different hollow fibre membrane module flow configurations, Desalination 312 (2013) 107–123.
[9] Alberto Tiraferri, et al., Relating performance of thin-film composite forward osmosis membranes to support layer formation and structure, J. Membr. Sci. 367 (1–2) (2011) 340–352.
[10] Sui Zhang, T.-S. Chung, Minimizing the instant and accumulative effects of salt permeability to sustain ultrahigh osmotic power density, Environ. Sci. Technol. 47 (17) (2013) 10085–10092.
[11] Qianhong She, et al., Organic fouling in pressure retarded osmosis: experiments, mechanisms and implications, J. Membr. Sci. 428 (2013) 181–189.
[12] Ngai Yin Yip, M. Elimelech, Performance limiting effects in power generation from salinity gradients by pressure retarded osmosis, Environ. Sci. Technol. 45 (2011) 10273–10282.
[13] Chuyang Y. Tang, et al., Coupled effects of internal concentration polarization and fouling on flux behavior of forward osmosis membranes during humic acid filtration, J. Membr. Sci. 354 (1–2) (2010) 123–133.
[14] Qianhong She, et al., Relating reverse and forward solute diffusion to membrane fouling in osmotically driven membrane processes, Water Res. 46 (7) (2012) 2478–2486.
[15] Shan Zou, et al., The role of physical and chemical parameters on forward osmosis membrane fouling during algae separation, J. Membr. Sci. 366 (1–2) (2011) 356–362.

[16] A. Ruiz-García, F. Tadeo, I. Nuez, Role of permeability coefficients in salinity gradient energy generation by PRO systems with spiral wound membrane modules, Renew. Energy 215 (2023).
[17] Gu Boram, Xiao Yun Xu, C.S. Adjiman, A predictive model for spiral wound reverse osmosis membranemodules: the effect of winding geometry and accurate geometricdetails, Comput. Chem. Eng. 96 (2017) 248–265.
[18] Henrik T. Madsen, et al., Pressure retarded osmosis from hypersaline solutions: investigating commercial FO membranes at high pressures, Desalination 420 (2017) 183–190.
[19] Jung-Gil Lee, et al., Numerical study of a hybrid multi-stage vacuum membrane distillation and pressure-retarded osmosis system, Desalination 363 (2015) 82–91.
[20] Maged Fouad Naguib, et al., Modeling pressure-retarded osmotic power in commercial length membranes, Renew. Energy 76 (2015) 619–627.
[21] Husnain Manzoor, et al., Energy recovery modeling of pressure-retarded osmosis systems with membrane modules compatible with high salinity draw streams, Desalination 493 (2020).
[22] Yuan Xu, et al., Effect of draw solution concentration and operating conditions on forward osmosis and pressure retarded osmosis performance in a spiral wound module, J. Membr. Sci. 348 (2010) 298–309.
[23] Jungwon Kim, et al., Evaluation of apparent membrane performance parameters in pressure retarded osmosis processes under varying draw pressures and with draw solutions containing organics, J. Membr. Sci. 493 (2015) 636–644.
[24] Yunfeng Chen, et al., Identification of safe and stable operation conditions for pressure retarded osmosis with high performance hollow fiber membrane, J. Membr. Sci. 503 (2016) 90–100.
[25] I. Ibrar, et al., Challenges and potentials of forward osmosis process in the treatment of wastewater, Crit. Rev. Environ. Sci. Technol. 50 (13) (2020) 1339–1383.
[26] Qianhong She, et al., Effect of feed spacer induced membrane deformation on the performance of pressure retarded osmosis (PRO): implications for PRO process operation, J. Membr. Sci. 445 (2013) 170–182.
[27] A. Ruiz-García, F. Tadeo, I. Nuez, Simulation tool for full-scale PRO systems using SWMMs, Desalination 541 (2022).
[28] Andrea Achilli, Tzahi Y. Cath, A.E. Childress, Power generation with pressure retarded osmosis: an experimental and theoretical investigation, J. Membr. Sci. 343 (1–2) (2009) 42–52.
[29] Klaus-Viktor Peinemann, et al., Membranes for power generation by pressure retarded osmosis, in: Klaus-Viktor Peinemann, S.P. Nunes (Eds.), Membranes for Energy Conversion, 2008.
[30] Y. Liang, A.V. Dudchenko, M.S. Mauter, Inadequacy of current approaches for characterizing membrane transport properties at high salinities, J. Membr. Sci. 668 (2023) 121246.
[31] Geoffrey M. Geise, et al., Free volume characterization of sulfonated styrenic pentablock copolymers using positronanni hilation lifetime spectroscopy, J. Membr. Sci. 453 (2013) 425–434.
[32] Qianhong She, Jin Xue, C.Y. Tang, Osmotic power production from salinity gradient resource by pressure retarded osmosis: effects of operating conditions and reverse solute diffusion, J. Membr. Sci. 401-402 (2012) 262–273.
[33] Yu Chang Kim, M. Elimelech, Adverse impact of feed channel spacers on the performance of pressure retarded osmosis, Environ. Sci. Technol. 46 (8) (2012) 4673–4681.
[34] Bongchul Kim, Sangyoup Lee, S. Hong, A novel analysis of reverse draw and feed solute fluxes in forward osmosis membrane process, Desalination 352 (2014) 128–135.
[35] Gaetan Blandin, et al., Validation of assisted forward osmosis (AFO) process: impact of hydraulic pressure, J. Membr. Sci. 447 (2013) 1–11.
[36] Xiaoxiao Song, Zhaoyang Liu, D.D. Sun, Energy recovery from concentrated seawater brine by thin-film nanofiber composite pressure retarded osmosis membranes with high power density, Energy Environ. Sci. 6 (4) (2013) 1199–1210.
[37] Alberto Tiraferri, et al., A method for the simultaneous determination of transport and structural parameters of forward osmosis membranes, J. Membr. Sci. 444 (2013) 523–538.
[38] J. Benjamin, et al., Optimizing pressure retarded osmosis spacer geometries: an experimental and CFD modeling study, J. Membr. Sci. 647 (2022) 120284.
[39] Y.Y. Liang, Review of analytical and numerical modeling for pressure retarded osmosis membrane systems, Desalination 560 (2023) 116655.
[40] Ali Altaee, et al., Dual stage PRO process for power generation from different feed resources, Desalination 352 (2014) 118–127.
[41] Ibrar Ibrar, et al., Evaluation of machine learning algorithms to predict internal concentration polarization in forward osmosis, J. Membr. Sci. 646 (2022).
[42] A. Abraham, Artificial neural networks, in: Peter H. Sydenham, R. Thorn (Eds.), Measuring System Design, John Wiley & Sons, Ltd., 2005.
[43] Liudmila Prokhorenkova, et al., CatBoost: unbiased boosting with categorical features, Adv. Neural Inf. Proces. Syst. 31 (2018) 6638–6648.
[44] Manuel Fernández-Delgado, et al., Do we need hundreds of classifiers to solve real world classification problems? JMLR 15 (1) (2014) 3133–3181.
[45] Jian Zhou, et al., Slope stability prediction for circular mode failure using gradient boosting machine approach based on an updated database of case histories, Saf. Sci. 118 (2019) 505–518.
[46] Trevor Hastie, Jerome Friedman, R. Tibshirani, The Elements of Statistical Learning Data Mining, Inference, and Prediction, Springer Link, 2001.
[47] Chester Su HernYeo, et al., Understanding and optimization of thin film nanocomposite membranes for reverse osmosis with machine learning, J. Membr. Sci. 606 (2020).
[48] Timothy V. Bartholomew, Nicholas S. Siefert, M.S. Maute, Cost optimization of osmotically assisted reverse osmosis, Environ. Sci. Technol. 52 (20) (2018).
[49] J.T. Hancock, T.M. Khoshgoftaar, CatBoost for big data: an interdisciplinary review, J. Big Data 7 (2020) 94.
[50] Guomin Huang, et al., Evaluation of CatBoost method for prediction of reference evapotranspiration in humid regions, J. Hydrol. 574 (2019) 1029–1041.
[51] I.H. Sarker, Machine learning: algorithms, real-world applications and research directions, SN Comput. Sci. 2 (2021) 160.
[52] Omer Sagi, L. Rokach, Approximating XGBoost with an interpretable decision tree, Inf. Sci. 572 (2021) 522–542.
[53] Swapan Talukdar, et al., Modeling fragmentation probability of land-use and land-cover using the bagging, random forest and random subspace in the Teesta River Basin, Bangladesh, Ecol. Indic. 126 (2021).
[54] Adam A. Atia, et al., Cost optimization of low-salt-rejection reverse osmosis, Desalination 551 (2023).
[55] S. Xiaoxiao, P. JA, D.D. Sun, Relating water/solute permeability coefficients to the performance of thin-film nanofiber composite forward osmosis membrane, J. Membr. Sci. Technol. 6 (4) (2016).
[56] Nahawand AlZainati, et al., Impact of hydrodynamic conditions on optimum power generation in dual stage pressure retarded osmosis using spiral-wound membrane, Energy Nexus 5 (2022).
[57] Nahawand AlZainati, et al., Experimental and theoretical work on reverse osmosis - dual stage pressure retarded osmosis hybrid system, Desalination 543 (2022).
[58] R.L. McGinnis, J.R. McCutcheon, M. Elimelech, A novel ammonia–carbon dioxide osmotic heat engine for power generation, J. Membr. Sci. 305 (1) (2007) 13–19.
[59] A. Achilli, T.Y. Cath, A.E. Childress, Power generation with pressure retarded osmosis: an experimental and theoretical investigation, J. Membr. Sci. 343 (1–2) (2009) 42–52.
[60] S. Adhikary, et al., Increased power density with low salt flux using organic draw solutions for pressure-retarded osmosis at elevated temperatures, Desalination 484 (2020) 114420.
[61] L.G. Palacin, et al., Evaluation of the recovery of osmotic energy in desalination plants by using pressure retarded osmosis, Desalin. Water Treat. 51 (1–3) (2013) 360–365.