Journal of the Brazilian Society of Mechanical Sciences and Engineering (2020) 42:620
https://doi.org/10.1007/s40430-020-02701-y

TECHNICAL PAPER

Deep neural network approach to estimation of power production for an organic Rankine cycle system

İlker Mert[1] · Hasan Huseyin Bilgic[2] · Hüseyin Yağlı[2] · Yıldız Koç[2]

Received: 12 March 2020 / Accepted: 25 October 2020

Abstract

In this study, the possibility of using Stepwise multilinear regression and deep learning models to estimate the behaviour of the organic Rankine cycle (ORC) has been investigated. It was found that a number of parameters affects the performance of the turbine and hence the amount of power obtained by the ORC. Therefore, limited and simulative parameters might not be sufficient to obtain the best prediction expression. In the present study, the data obtained from a 10 kW ORC system was used as the basis for deep learning models. To this end, the variable selection, which represents the inputs to the neural network, is included in the first steps of a stepwise multilinear regression (SMLR). The aim of the deep learning (DL) models is to use the capabilities of dense layers, and then to strengthen SMLR contributions. The main aim here was to estimate the power generation of the expander, which has an important role in deciding the ORC's performance. The present study is intended to act as a crucial resource for defining an active estimation procedure for the ORC system through the use of DL. Therefore, an interoperability framework is proposed to estimate ORC power production using SMLR and DL as a new approach in this study. The interoperability approach for the proposed models (SMLR and DL) was found to be successful.

Keywords Stepwise multilinear regression · Deep learning · RMSprop · Organic Rankine cycle (ORC)

List of symbols

ORC	Organic Rankine cycle
SMLR	Stepwise multilinear regression
DL	Deep learning
ANN	Artificial neural network
RMSE	Root mean square error
MSE	Mean square error
R^2	The coefficient of determination
T_{ei}	Evaporator exhaust inlet temperature
T_{eo}	Evaporator exhaust outlet temperature
m_{b}	Heat source mass flow rate
T_{ri}	Cooling water inlet temperature
T_{ro}	Cooling water outlet temperature
m_{y}	Cooling water mass flow rate
P_{orc}	ORC turbine outlet power
FCDN	Fully connected deep networks
AF	Activation function
ReLU	Rectified linear unit
NN	Neural network
MAE	Mean absolute error

1 Introduction

Today's main interest within the energy sector is finding cheap and environment-friendly alternative power production systems because of increasing environmental problems and decreasing primary energy sources (i.e. fossil fuels) [1–3]. In this context, the main scientific focus is on renewable energy sources such as biofuels [4], wind [5], and solar [6] energies, which are effectively unlimited but currently expensive to convert into usable energy [7]. During these

Technical Editor: Jader Barbosa.

✉ Hüseyin Yağlı
huseyin.yagli@iste.edu.tr

İlker Mert
ilkermert@osmaniye.edu.tr

Hasan Huseyin Bilgic
hhuseyin.bilgic@iste.edu.tr

Yıldız Koç
yildiz.koc@iste.edu.tr

[1] Osmaniye Vocational School, Osmaniye Korkut Ata University, Osmaniye, Turkey

[2] Department of Mechanical Engineering, Faculty of Engineering and Natural Sciences, Iskenderun Technical University, 31200 Iskenderun, Hatay, Turkey

Published online: 10 November 2020

studies, another important area of focus is on improving the efficiency of current power systems by integrating additional systems [8, 9]. Kalina cycles, heat wheels, recuperators, and economizers represent just some of these additional systems [10–12]. In addition to these, another most important approach is organic Rankine cycles (ORC). The organic Rankine cycles use organic fluids as a system fluid instead of steam. Therefore, ORCs can be used to recover low-temperature heat sources (80 °C and above) [13, 14]. In addition to the availability of ORC for a wide range of heat sources, ORC systems have the advantages of easy construction, simple configuration, cheaper investment and almost zero maintenance costs when compared with other low-temperature additional power systems [15, 16]. Therefore, there are many theoretical and experimental studies aimed at improving overall system performance by integrating ORC with various-temperature waste heat sources [16, 17]. In these literature studies, the main objective is one of improving ORC performance is by the use of different working fluids, changing turbine inlet parameters, and using different ORC design configurations (simple, regenerative, dual loop, etc.) [18, 19]. In spite of the various experimental and theoretical studies, it is still not possible to precisely predict how much energy might be recovered and ORC performance due to the presence of various heat sources and dynamic variation of the heat source parameters. At this juncture, the importance of machine learning (ML) and prediction methods become apparent.

Machine learning enables computers to make data-based decisions rather than programming them to perform a specific task [20]. These programs or algorithms are designed for computers that are exposed to new data to learn and improve themselves over time. This learning method takes advantage of the computing power of modern computers, which can easily handle large datasets. Deep learning is a machine learning method. It allows one to train artificial intelligence models to predict outputs with a given dataset [21]. Both supervised and unsupervised learning can be used to train artificial intelligence. Although deep learning is relatively new, it has become an important subfield of artificial intelligence in academic research and industry due to its achievements in many different fields.

Many deep learning methods use architectures called neural networks. Deep learning algorithms can be considered structurally more complex forms of artificial neural networks (ANNs). Classical neural networks consist of two or three layers. However, in deep learning, this number can exceed 100, according to the computing capability of the computer involved. In other words, deep learning is not a single layer with big data, but is a system that works using many layers and performs the calculations used in machine learning at the same time, discovers the parameters that need to be defined in machine learning, and performs evaluations with better parameters. While classical networks can work with human-dependent traits, deep learning works with human-independent traits. Networks with deep learning capabilities can perform prediction, estimation, and classification tasks. In this study, the aim is to model the output power of an ORC system using the estimation ability of the proposed deep learning models.

There are many studies in the literature that have used ANN for the estimation of power system performance as well as ORC. Yilmaz et al. used an ANN method to improve the performance of an ORC using R410a and R407c as a working fluid. In the study, mathematical expressions have been derived to find the efficiency of an ORC for each working fluid [22]. Rashidi et al. optimized an ORC using R717 by use of ANNs and Artificial Bees Colony (ABC) algorithms [23]. Kovacı et al. estimated the thermal efficiency of an ORC by using an adaptive neuro-fuzzy (ANFIS) and ANN method depending on the condenser temperature and the evaporator temperatures [24]. Massimiani et al. optimized a regenerative ORC system with ANNs trained using WEKA [25]. Yang et al. used an ANN method to estimate the performance of and improvements to an ORC-assisted diesel engine in terms of waste heat [26]. Bilgiç et al. developed a mathematical expression to estimate the power production of an ORC by use of an ANN [27]. Kılıç and Arabacı carried out a performance analysis of an ORC using R123, R125, R227, R365mfc, and SES36 as refrigerants. In the study, ANN and ANFIS methods were used to evaluate a prediction expression [28]. Palagi et al. compared the performance predictions for a 20 kW ORC system for feed-forward, recurrent (RNN) and long short-term memory (LSTM) networks. As a result of the study, RNN and LSTM networks showed greater success at predicting the performance of the associated system [29].

Although there are many ANN-based ORC studies, it is clear that the majority of prediction models based on classical neural networks are not sufficient to predict the performance of ORC power systems [30–32] because the parameters that influence the performance of the ORC, such as heat losses and pressure drops, are not considered in the majority of thermodynamic models. Another, and possibly the most important reason for the adequacy of the prediction methods is that the parameters used to evaluate prediction methods are theoretical parameters and commonly ignore ORC component isentropic efficiency and assumes the ORC system to be under steady state conditions. An ORC system consists of evaporator, turbine, condenser, and pump. In order to evaluate the best prediction methods and prediction expressions, it is essential to consider pressure drops and component efficiencies.

By considering these, the novelty of the present manuscript can be listed as follows. (1) The ORC parameters used to derive the prediction expression are taken from an

ORC dataset. Therefore, essential parameters like pressure drops and component efficiencies are taken into account. (2) Since the data obtained from the ORC are dynamic values and recorded for various heat source temperatures, turbine inlet pressures and turbine inlet temperatures, the accuracy of the derived prediction method is as high as possible. (3) Throughout the study, stepwise multilinear regression (SMLR) and deep learning (DL) methods are used and compared. Therefore, the present study makes it possible to compare these two prediction methods, as well as for deciding the best prediction method. The performances of the models created in this study were compared using statistical criteria. For this purpose, root mean square error (RMSE), mean square error (MSE), and the coefficient of determination (R^2), as measures used in statistical analysis, are preferred as the evaluation criteria. (4) The present study aimed to derive an expression to predict the performance of ORC accurately. The results of the present study make it possible to easily predict the performance of an ORC system.

2 Materials and methods

In this study, the data for evaporator exhaust inlet temperature (T_{ei}), evaporator exhaust outlet temperature (T_{eo}), heat source mass flow rate (m_{b}), cooling water inlet temperature (T_{ri}), cooling water outlet temperature (T_{ro}), cooling water mass flow rate (m_y), and ORC turbine outlet power (P_{orc}) were obtained from an organic Rankine Cycle dataset of 10 kW power production capacity. The min, max, and statistical parameters of the dataset used in the study are presented in Table 1.

2.1 Description of the organic Rankine cycle (ORC)

The organic Rankine cycle (ORC) is one of the most efficient methods with which to recover waste heat from low- and medium-grade temperature heat sources. When compared with the conventional Rankine cycle, ORCs have similar system components, but different working fluids. The working fluid of the classical Rankine cycle is steam. However, ORCs use a wide variety of organic-based working fluids [33, 34]. The organic-based working fluids have many superiorities over steam such as low boiling temperatures, low specific heats, etc. [35–37]. In the present study, data obtained from an ORC set up in the laboratory is used to obtain a prediction expression. A simplified scheme of an ORC system is shown in Fig. 1.

The ORC used in the present study uses R245fa as the working fluid, where the maximum capacity of the ORC is 10 kW. The heat is supplied to the ORC by an electric boiler. In the system, the electric boiler first heats the water. The heated water is directed to the evaporator to heat up the working fluids. The working fluid, as a vapour, goes through the turbine and produces mechanical shaft power. After the turbine, the working fluid cools down to the saturated liquid phase and is then pushed through the pump to pressurize it. The isentropic efficiency of the pump and the turbine, as defined in the technical working sheet of the system, is 82% and the heat source fluid (water at 10 bar) can be heated up to 180 °C by the electrical heater. However, these parameters of the ORC fluctuate depending on the working conditions. Throughout the analyses, thermodynamic parameters like pressure, temperature, etc. and system parameters like shaft power, heat input, and mass flow rate are continuously recorded and saved by data recorder. T-type thermocouples (−100 to 300 °C measuring range, ± 0.15 °C accuracy and 0.01 °C resolution for full range), electromagnetic flow meters (0.03–12 m/s flow range, ± 0.2% accuracy) and wattmeters (12–10,400 W measuring range, 225 mW resolution and 0.5% accuracy) could be used to measure ORC system parameters. The data could be recorded by Universal Dataloggers (eight channels, 24-bit resolution with a sampling frequency of 0.1 s). In the present study, the ORC system was run for 1 day and the recorded data used to train the prediction models.

2.2 Stepwise multilinear regression (SMLR)

The SMLR model, unlike the simple linear regression model, considers more than one independent variable and

Table 1 Min, max, and statistical parameters of input and output variables

Parameters	Symbol	Unit	Mean	SD	Min	Max	Skewness	Kurtosis
Cooling water condenser inlet temperature	T_{ei}	°C	0	1.15	6.10	16.30	– 1.88	9.26
Cooling water condenser outlet temperature	T_{eo}	°C	17.13	1.48	7.30	19.70	– 119	5.15
Cooling water mass flow rate	m_{b}	l/h	15.00	0.65	7.30	46.75	28.81	1266.79
Hot water evaporator inlet temperature	T_{ri}	°C	125.85	1.90	112.25	128.98	– 4.42	23.82
Hot water evaporator outlet temperature	T_{ro}	°C	103.02	6.74	88.55	117.46	0.38	– 0.40
Hot water mass flow rate	m_y	l/h	1.38	0.78	0.57	3.24	1.41	0.71
ORC turbine outlet power	P_{orc}	kW	5.38	0.93	3.74	7.99	0.66	0.40

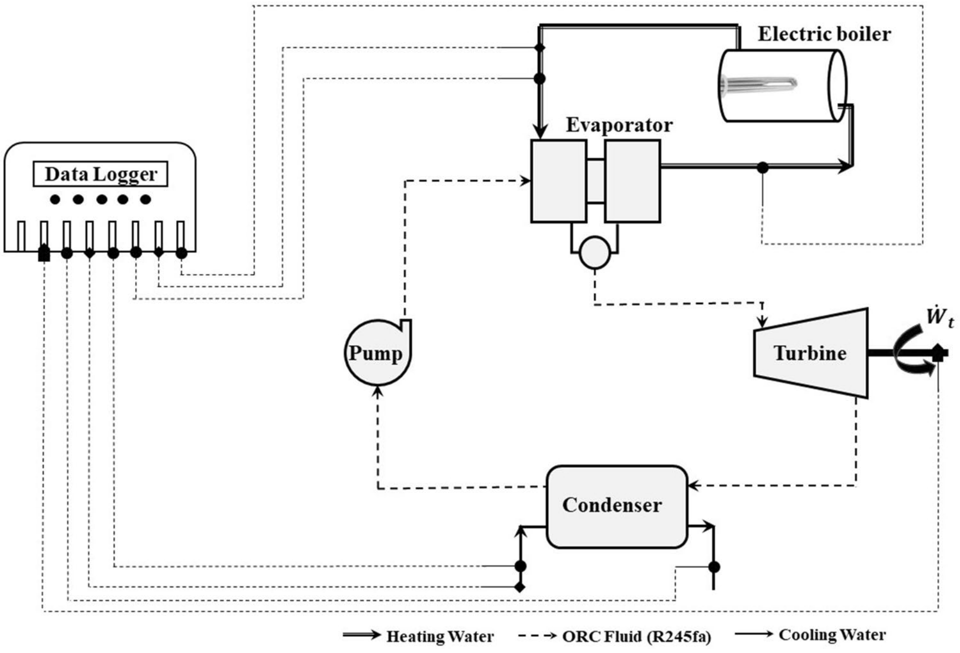


Fig. 1 The simplified scheme of the ORC system

calculates the associated regression model from n dependent variables at the same time.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_j X_n + \varepsilon \tag{1}$$

where X is the explanatory variable and Y is the dependent variable in Eq. 1. Every value of X is associated with another value of the Y. β_0 is the line y-axis intersection, β_1 the regression coefficient (gradient of the line), β_j is the jth parameter and ε is the chance value. The values β_0 and (β_{j-1}) are theoretical values calculated using the entire dataset. The success of the dependent variable Y calculated via X allows the regression model to be evaluated via various performance measures.

SMLR adds or subtracts the independent variable to or from the model to determine a meaningful subset of variables. Stepwise regression generally uses three procedures: standard stepwise (adds and removes terms), forward selection (adds term), and backward elimination (removes terms). Standard stepwise procedure (SSP): by default, this procedure starts with an empty model and then adds or removes a term for each step [38]. This can lead to the optimization of the estimated performance of the system, coupling DL with SMLR.

2.3 Deep learning

Fully connected deep networks (FCDN) are the most essential part of deep learning used for applications [39, 40]. An FC layer is the actual component that performs the discriminative learning in a deep neural network. An FFNN with L layers describes a mapping $f\left(r_0;\theta\right) : \mathbb{R}^{N_0} \mapsto \mathbb{R}^{N_L}$ of inputs $r_0 \in \mathbb{R}^{N_0}$ to outputs $r_L \in \mathbb{R}^{N_L}$ through L iterative processing steps [41].

$$r_l = f_l\left(r_{l-1};\theta_l\right), l = 1, \ldots, L \tag{2}$$

where $r_l = f_l\left(r_{l-1};\theta_l\right) : \mathbb{R}^{N_{l-1}} \mapsto \mathbb{R}^{N_l}$ is mapping carried out by the lth layer. A successful map depends on both the output, r_{l-1} from the previous layer and on a set of parameters θ_l.

The set representing all the parameters of the network is identified by $\theta = \left\{\theta_1, \ldots, \theta_L\right\}$. The lth layer is referred to as dense or fully connected if $f_l\left(r_{l-1};\theta_l\right)$ has the form

$$f_l\left(r_{l-1};\theta_l\right) = \sigma(W_l r_{l-1} + b_l) \tag{3}$$

where $W_l \in \mathbb{R}^{N_l \times N_{l-1}}, b_l \in \mathbb{R}^{N_l}$, and $\sigma(\ldots)$ is an AF that can be defined briefly. The parameter set is $\theta_l = \left\{W_l, b_l\right\}$ for this layer.

The AF, $\sigma(\ldots)$ in (Eq. 3) gives a non-linearity to the network which is necessary to learn the distribution of complex datasets because stacking more layers increases the non-linearity. AF is generally applied independently to each element of the input vector, i.e.$[\sigma(u)]_i = \sigma\left(u_i\right)$. The AF and rectified linear unit (ReLU) used in this study are reported in Table 2 and shown in Fig. 2.

Table 2 AF used in DL models

Name	$[\sigma(u)]_i$	Range
ReLU	max(0,u_i)	(0;∞)

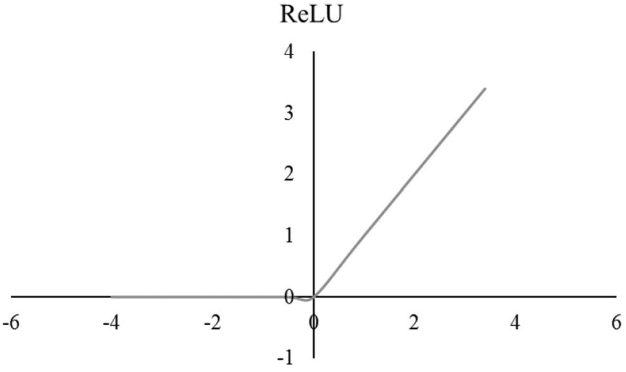


Fig. 2 ReLU AF

As can be seen in Fig. 3, the DNN has L layers. Here, each hidden layer uses the output of the previous layer and transmits it to the input of the next layer.

$$r^l = \sigma\left(W^{(l-1)\intercal} r^{l-1} + b_0^{l-1}\right) = \sigma\left(b_0^{l-1} + \sum_{n=1}^{N^{l-1}} b_n^{l-1} r_n^{l-1}\right) \tag{4}$$

$$y = W^{(L)\intercal} r^L + w_0^L \tag{5}$$

with hidden layer outputs $r^l = \left(r_1^l, \dots, r_{N^{(l)}}^l\right) \in \mathbb{R}^{N_l}$ and parameters $W^l = \left(b_1^l, \dots, b_{N^l}^l\right) \in \mathbb{R}^{N_l \times N_{l-1}}$ for $l = 0, \dots, L-1$ and $W^L \in \mathbb{R}^{N_l}$

NNs are trained using labelled data except for unsupervised learning, i.e. a set of data pairs composed of input/output vectors $\left(r_{0,i}, r_{L,i}^{\circ}\right), i = 1, \dots, S$, where $r_{L,i}^{\circ}$ has targeted outputs when $r_{0,i}$ is used as the input. While the network is training, the goal is to minimize any loss

$$L(\theta) = \frac{1}{S} \sum_{i=1}^{S} l\left(r_{L,i}, r_{L,i}^{\circ}\right) \tag{6}$$

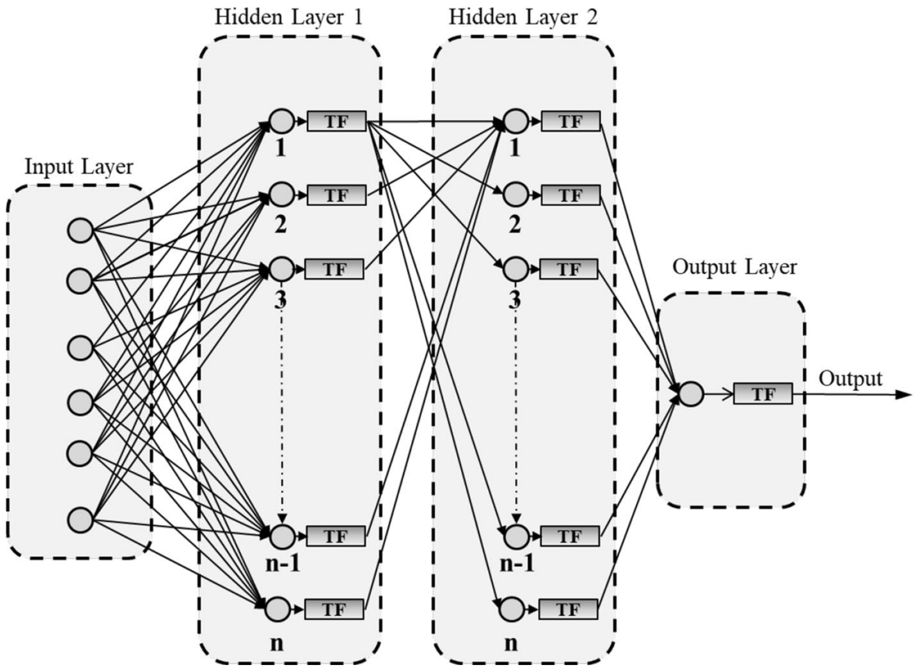
concerning the parameters in θ, where l(u; v): $\mathbb{R}^{N_L} \times \mathbb{R}^{N_L} \mapsto \mathbb{R}^{N_L}$ is the loss function (LF). $r_{L,i}$ and $r_{0,i}$ are the output and input of the network, respectively. Most commonly, LFs are MSE (for regression problems) and categorical cross-entropy (for classification problems). MSE can be given by $u - v_2^2$. A neural network with a large number of parameters requires an optimization technique during the training phase. The most popular algorithms to find good sets of parameters θ are generally variants of classical gradient descent (GD), which starts with some random initial values of $\theta = \theta_0$ and then updates θ iteratively. Such as stochastic gradient descent (SGD) rule is given as θ_{t+1}:

$$\theta_{t+1} = \theta_t - \eta \nabla \tilde{L}\left(\theta_t\right) \tag{7}$$

where η and $\tilde{L}$ are the learning rate and an approximation of the LF which is computed for a random mini-batch of training examples $S_t \subset \{1, 2, \dots., S\}$ of size S_t at each iteration, i.e.

$$\tilde{L}(\theta) = \frac{1}{S_t} \sum_{i \in S_t}^{S} l\left(r_{L,i}, r_{L,i}^{\circ}\right) \tag{8}$$

Fig. 3 Deep neural network with two hidden layers



While still reducing weight update variance, by choosing S_t to be sufficiently small compared to S the computational complexity of the gradient can be significantly reduced. Therefore, the gradient in (14) can be very efficiently computed through the backpropagation (BP) algorithm. GD, which is an optimization method, is generally used in DL models to update the weights of the NN through BP. Optimizers are effective for GD in three ways: (1) modifying the gradient component, $\partial L/\partial W$, (2) modifying the learning rate component, η, or (3) both. For (1), optimizers commonly make use of the moving averages of the gradient (momentum), instead of just taking one value like in 'vanilla' GD. For (2), these optimizers multiply the learning rate by some positive factor such that they become smaller. Optimizers that act on both (3) are like Adam. SGD is a very powerful technique, currently employed to optimize all deep learning models [42]. The number of patterns used to calculate the error includes how stable the gradient used to update the model actually is. It requires $O(n)$, number of computations and memory use, where n is the number of parameters. SGD can be used to avoid this computational problem. At every iteration, we sample a mini-batch, $\mathbb{M}$ of n' examples from the training set. Usually, a variant of GD is used, using noisy subset estimation of the gradient. The simplest update rule for SGD is:

$$W_{t+1} = W_t - \eta \frac{\partial L}{\partial W_t} \tag{9}$$

where w is the optimized parameter (weight). L and η are the error (LF estimation) and the learning rate, respectively. However, the root mean square propagation algorithm (RMSprop) is used in this study because RMSprop also takes away the need to adjust the learning rate by doing it automatically, as different from SGD. The update rule for RMSprop is [43]:

$$W_{t+1} = W_t - \frac{\eta}{\sqrt{S_t}} \frac{\partial L}{\partial W_t} \tag{10}$$

where

$$S_t = \beta S_{t-1} - (1-\beta)\left[\frac{\partial L}{\partial W_t}\right]^2 \tag{11}$$

S_t is initialized to 0. ε, β and α are set to 10^{-6}, 0.9 and 0.001, respectively, by per Hintons' suggestion.

Although actual DL implementations are usually used for speech or image recognition, different dataset applications do also exist. This work is an application of DL to a 10 kW ORC dataset. The results suggest that DL models are also applicable for P_{orc} estimation, and thus that this kind of application of DL increases the novelty of this study.

2.4 Evaluation criteria

In this study, the models were evaluated using a number of popular performance tests: mean squared error (MSE), root mean squared error (RMSE), coefficient of determination (R^2), mean absolute error (MAE) and Mallows' *Cp* (or *Cp-statistic*). Those indicate the degree of correlation between measured and estimated values. Thus, the reliability of the models could be tested because higher reliability means more accurate P_{orc} estimation [44–46].

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} \left|e_i\right| \tag{12}$$

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N} e_i^2 \tag{13}$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\left(\sum_{i=1}^{N} e_i^2\right)} \tag{14}$$

$$R^2 = 1 - \sum y_i - y_{(p,i)} \Big/ \sum y_i - y_m \tag{15}$$

where y_i is the observed value and $y_{(p,i)}$ is the predicted value of y_i, e_i is $y_i - y_{(p,i)}$.

$$\text{Mallows'Cp} = \left(\frac{\text{SS}_{\text{res}}}{\text{MS}_{\text{resres}}}\right) - N + 2p \tag{16}$$

where SS_{res} is the residual sum of squares (SS) for the model with (p-1) parameters when MS_{res} is the residual mean square of all parameters [47]. The best models were determined using statistical software to maximize R^2 and minimize RMSE and Mallows' *Cp*, in a stepwise process.

3 Results

3.1 SMLR results

Step 1 was to regress P_{orc} for each parameter. This simply means to run a regression for each predictor parameter alone versus P_{orc}. Then, the predictor parameter with the lowest p value is added to the model (as long as is there is a predictor variable with a p value < 0.15. T_{ei} has the lowest p value so it is added to the model in step 1. The SMLR marked m_b, and the T_{ro} parameter as inactive and eliminated them according to their p value (they were greater than the '*Alpha-to-Enter*' value of 0.15). Then, the above steps were repeated using models that contained the T_{eo}, m_y and T_{ri} parameters, respectively. Parameters with the lowest p value of less than 0.15 were added to the model in the following steps. Table 3 shows the model that SMLR builds entirely from a 10 kW ORC dataset. In step 4 (the final step), the R^2 is reasonably high, and all the variables have very low p values. The steps used to determine the best model in SMLR and a comparison of models can be seen in Tables 3 and 4. Best values are also highlighted in bold.

Scatter plots for SMLR models can be seen in Fig. 4.

According to Tables 3 and 4, the highest R^2 and lowest RMSE and Mallows' Cp (the best results) with values of 0.9392, 0.2282 kW and 4.4, respectively, were obtained for Model 4, which is composed of T_{ei}, T_{eo}, T_{ri}, and m_y.

Equation for Model 4:

Table 3 Depending on all P_{orc} values, the steps of calculating the optimum model in SMLR process

	Step 1		Step 2		Step 3		Step 4	
	Coef.	*P* value	Coef.	*P* value	Coef.	*P* value	Coef.	*P* value
Constant	– 0.884		2.5304		2.4017		**1.9970**	
T_{ei}	0.4334	0	– 3.3304	0	– 3.2490	0	**– 3.2615**	0
T_{eo}			2.9743	0	2.9114	0	**2.9186**	0
m_y					0.0234	0.004	**0.0238**	0.003
T_{ri}							**0.0037**	0.082
RMSE		0.7782		0.2284		0.2282		**0.2282**
R^2		0.2926		0.9391		0.9392		**0.9392**
Mallows' Cp		5752		11.69		5.43		**4.4**

Table 4 Comparison among the SMLR models for P_{orc}

Model	Type	Input parameters	*Mallows' Cp*	RMSE	R^2	Rank
Model 1	SMLR	T_{ei}	5752	0.7782	0.2926	4
Model 2	SMLR	T_{ei}, T_{eo}	11.69	0.2284	0.9391	3
Model 3	SMLR	T_{ei}, T_{eo}, m_y	5.43	0.2282	0.9392	2
Model 4	**SMLR**	$\boldsymbol{T_{ei}}$, $\boldsymbol{T_{eo}}$, $\boldsymbol{T_{ri}}$, $\boldsymbol{m_y}$	**4.4**	**0.2282**	**0.9392**	**1**

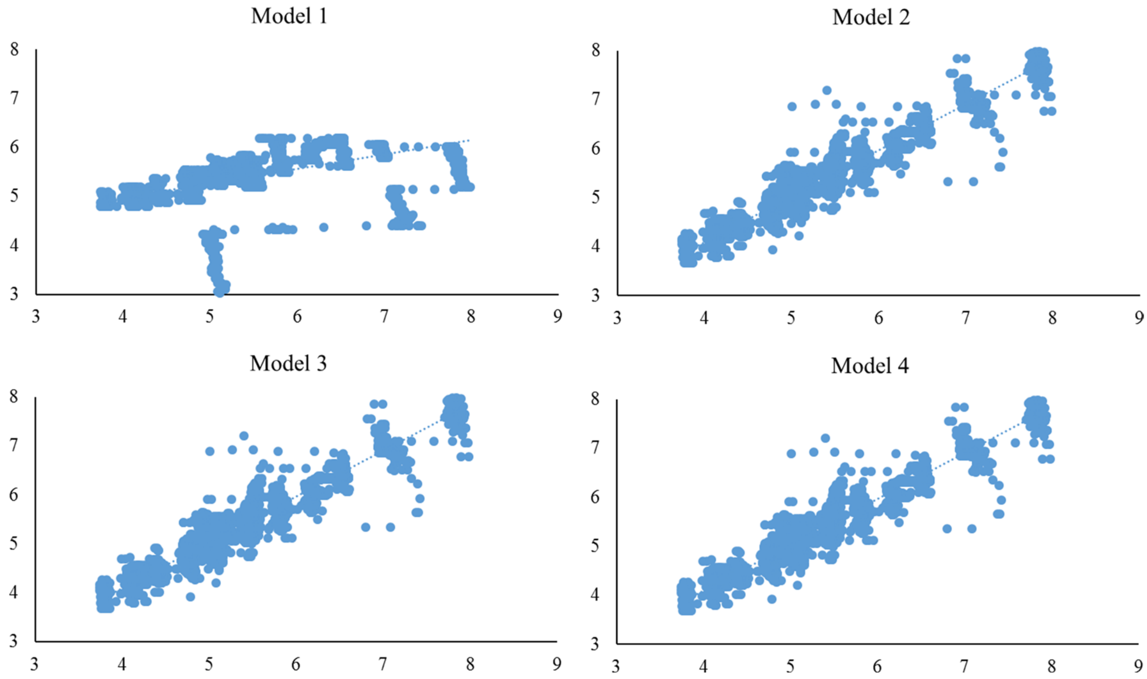


Fig. 4 Scatter plots for SMLR models

$$P_{\mathrm{orc}} = 1.9970 - 3.2615 \cdot T_{\mathrm{ei}} + 2.9186 \cdot T_{\mathrm{eo}} + 0,0037 \cdot T_{\mathrm{ri}} + 0,0238 \cdot m_y$$

3.2 Deep learning results

In this study, DL models with five variations have been trained and tested using all the available parameters, T_{ei}, T_{eo}, m_{b}, T_{ri}, T_{ro}, m_y, to estimate P_{orc}. Models 5–8, as based on the active parameters determined in the SMLR phase as a hybrid approach, have been applied. DL models were developed using Keras, an open-source neural network library written in Python (keras.io). Models were built using the sequential model to obtain the best results. In the sequential model, four densely connected NN layers were utilized. The models were trained and tested according to L, which is the mean squared error, and the RMSprop optimization algorithm. Simulations were run for 100 epochs. DNN configurations in the third layer, as determined by the trial and error method, are reported in Tables 5 and 6. In Table 6, the best models in input combinations are highlighted with bold. The numbers of nodes in all layers of the DL models were determined by the lowest computing power requirement and loss values.

After determining the parameters via the SMLR steps, input combinations given in Table 6 were chosen as the dataset in the training process, where 30% of the data were used to test the DL models.

Figure 5 shows the train results of the DL models according to loss values.

Table 5 Structure of DNN used in the present study

Layer no.	Layer type	AF	Number of neurons
1	Dense (Input layer)	ReLU	50
2	Dense	ReLU	50
3	Dense	ReLU	1–50
4	Dense (output layer)	Linear	1

Table 6 Comparison of DL models for P_{orc}

Model	Type	Input combination	AF	Structure	Loss values for testing phase
Model 5	DL	T_{ei}	ReLU	:	:
				50–50–5–1	0.026422
				50–50–10–1	**0.019872**
				50–50–15–1	0.020467
				:	:
Model 6	DL	T_{ei}, T_{eo}	ReLU	:	:
				50–50–10–1	0.011217
				50–50–15–1	**0.003771**
				50–50–20–1	0.005455
				:	:
Model 7	DL	T_{ei}, T_{eo}, m_y	ReLU	:	:
				50–50–10–1	0.005432
				50–50–15–1	**0.004522**
				50–50–20–1	0.005590
				:	:
Model 8	DL	T_{ei}, T_{eo}, T_{ri}, m_y	ReLU	:	:
				50–50–5–1	0.003159
				50–50–10–1	**0.002214**
				50–50–15–1	0.002566
				:	:
Model 9	DL	T_{ei}, T_{eo}, m_{b}, T_{ri}, T_{ro}, m_y	ReLU	:	:
				50–50–10–1	0.004261
				50–50–15–1	**0.003596**
				50–50–20–1	0.004130
				:	:

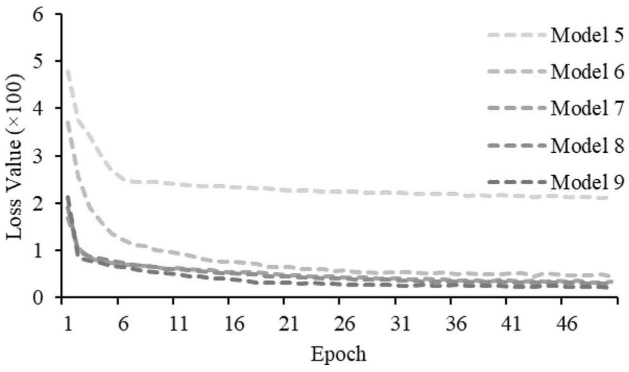


Fig. 5 Comparison of loss values implemented DL models for training phase

After the training process is completed, the DL models are tested with 30% of the data. Overall, the estimations of the DL models match the real data well (except Model 5). To see the estimation precision of the proposed models, test errors and scatter plots are reported in Table 7 and shown in Figs. 6, 7, 8, 9, and 10. In Table 7, the best DL Models are highlighted with bold. The scatter plots are ordered according to the number of neurons in the third dense layer in the DL models. It can be seen that most of the test and train errors are maintained within the range of 0.047053 kW to 0.587180 kW as *RMSE* values. Compared with the 10 kW ORC dataset, the maximum R^2 is 0.9528, as achieved in Model 8; however, the maximum error is less than 0.60 kW, as can be seen in the same table. Therefore, the proposed DL models with strong learning ability and well-generalized performance show good agreement. Model 5 estimates did not match well with the testing data with the lowest R^2 value.

According to Table 7, the highest R^2 and lowest *RMSE* (the best results), with values of 0.9528, 0.047053 kW (in training) and 0.199682 kW (in testing), respectively, were obtained for Model 8, which is composed of T_{ei}, T_{eo},

Table 7 Results of DL models

Model	Type	Best structure	Training RMSE	Testing RMSE	R^2	Rank
Model 5	DL	50–50–10–1	0.140968	0.587180	0.5917	5
Model 6	DL	50–50–15–1	0.061408	0.231881	0.9363	4
Model 7	DL	50–50–15–1	0.067246	0.212472	0.9465	2
Model 8	**DL**	**50–50–10–1**	**0.047053**	**0.199682**	**0.9528**	**1**
Model 9	DL	50–50–15–1	0.059967	0.213325	0.9461	3

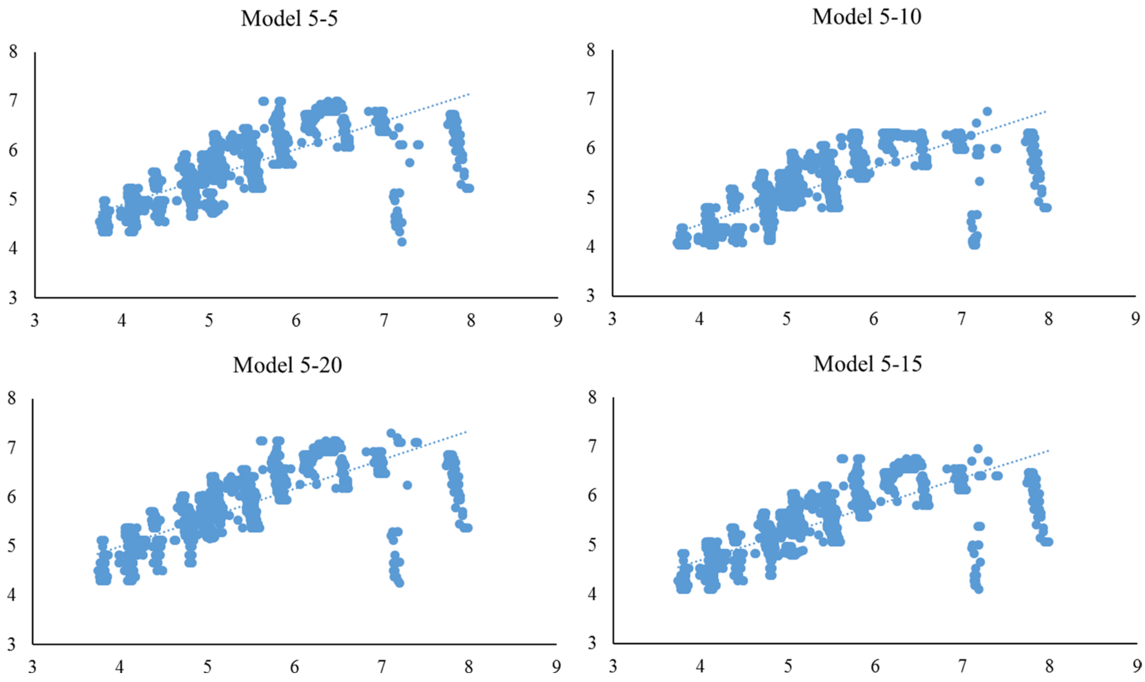


Fig. 6 Scatter plots for model 5

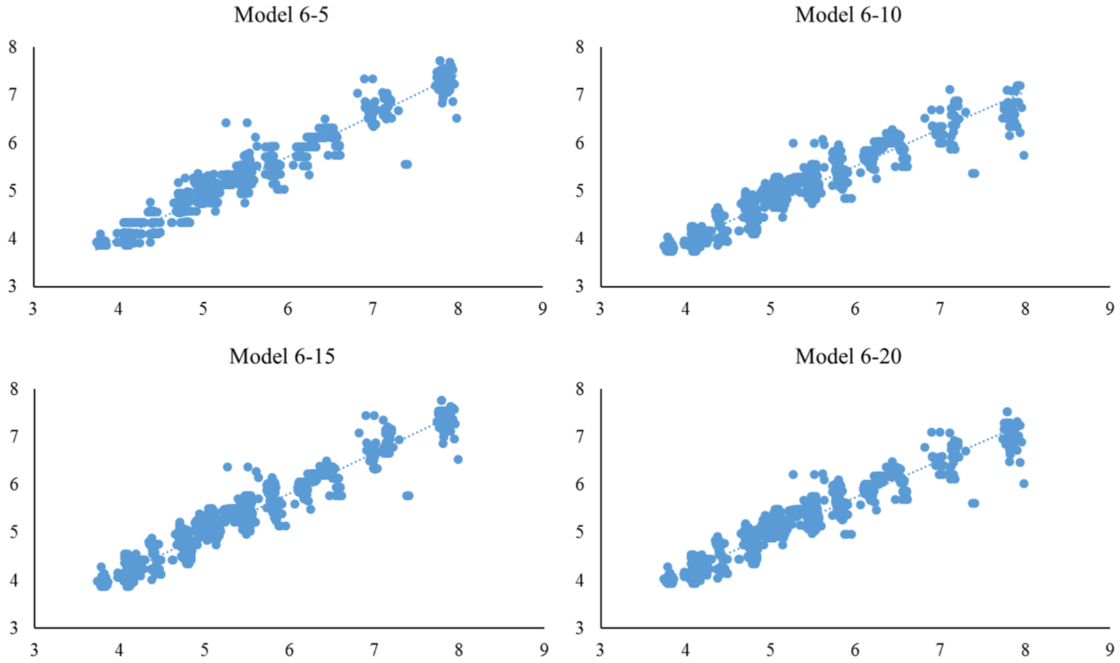


Fig. 7 Scatter plots for model 6

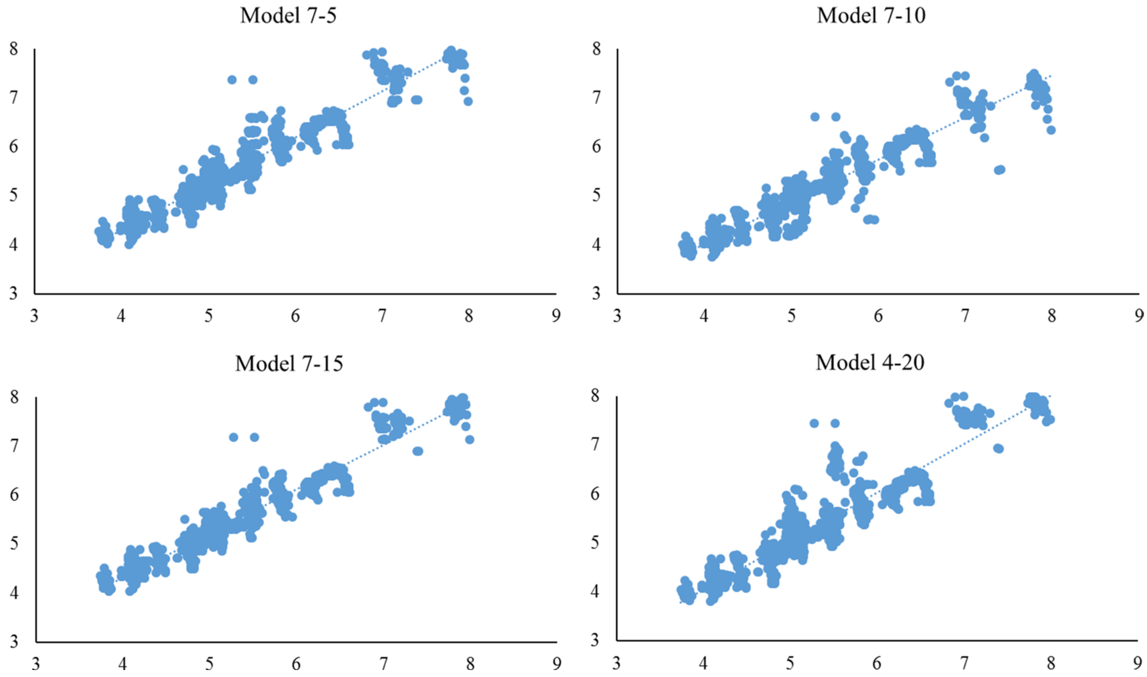


Fig. 8 Scatter plots for model 7

T_{ri}, and m_y. On the other hand, the lowest R^2, the highest RMSE values (the worst results) were obtained as 0.5917, 0.140968 kW (in training) and 0.587180 kW (in testing) for Model 5, which is composed of T_{ei}. Model 7, which is composed of T_{ei}, T_{eo}, and m_y, was the second-most successful combination, with a value of 0.9465 for R^2. Model 9, Model 6, and Model 5 were the third-, fourth-, and fifth-most successful combinations in order of appearance. The model performances were ranked from worst

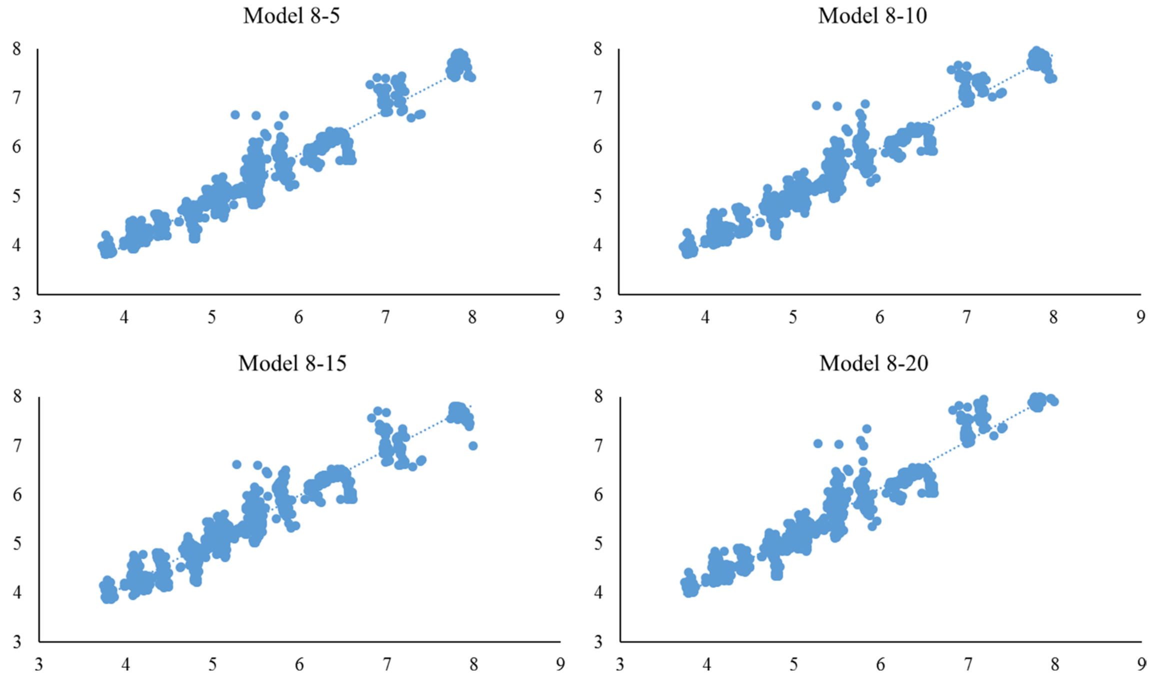


Fig. 9 Scatter plots for model 8

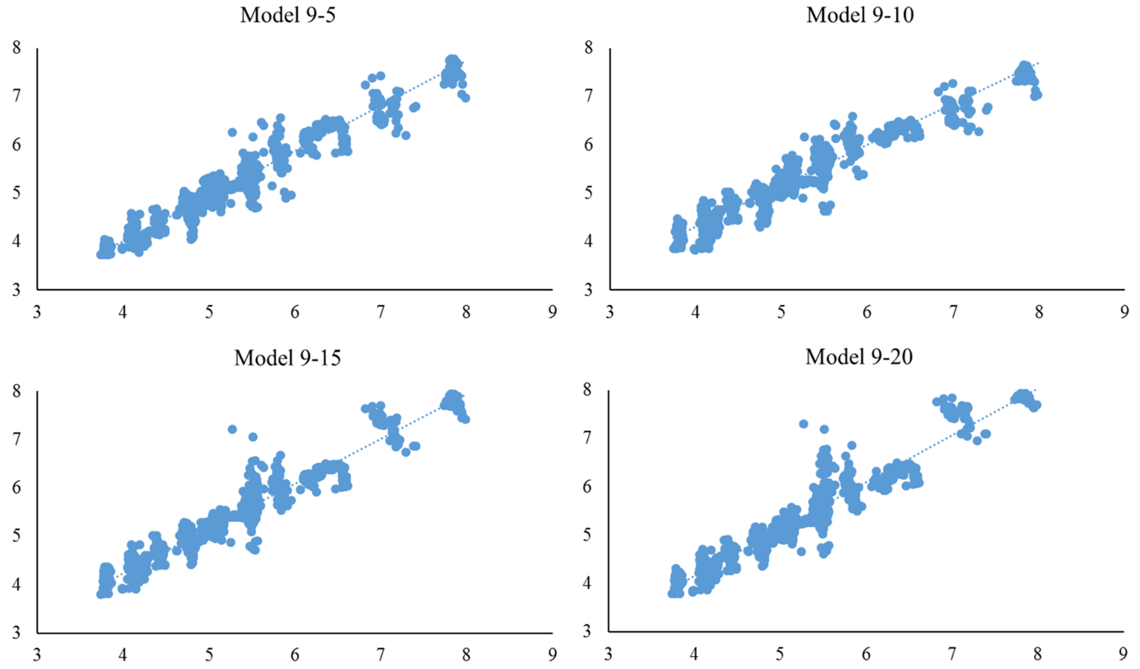


Fig. 10 Scatter plots for model 9

to best as follows: Model 8, Model 7, Model 9, Model 6, and Model 5.

The scatter plots above represent the P_{orc} versus estimation of P_{orc} via DL-SMLR Model 5 (with one input variable). It is clear from the scatter plot that as P_{orc} values increase, deviations from the fitting line (regression line) increase. According to the scatter plots, where it seems that the data do not follow a linear pattern there is no linear correlation. On the other hand, if it seems to be the case that the points follow a linear pattern then it can be said that there is a high linear correlation, as can be seen in Fig. 6.

If the dots look a little scattered along the trend line, then it can be said that there is a moderate linear correlation, as seen in Fig. 7.

Comparing with Model 9 in which all variables were input, the DL Model 8 based on the SMLR had the advantages of decreasing the number of input parameters and the computing process. The model has a higher convergence rate and estimate accuracy, as seen in Figs. 8 and 9 than other models.

According to the results in Table 7 and Fig. 10, Model 9, which has maximum input parameters, is the second choice. It was also clearly seen that T_{ei} and T_{eo} were the most important input parameters because all the combinations containing T_{ei} and T_{eo} produce lower *MSE* and *RMSE* values than any others. Model 8 produced more precise results than Model 9, Model 7, Model 6, and Model 5, implying that it is more important parameter than T_{ri}, $m_{y.}$ It should be the first choice of input attribute when T_{ei} and T_{eo} are unavailable as input.

It can be concluded that when the input number is increased, estimation error is generally decreased. Case summaries (input, activation, and output equations) for Model 8 are given in Table 8.

3.3 Comparison of performance of the models

This study demonstrates the applicability of the DL-SMLR method for estimating specific operating parameters of the ORC system, which is difficult to model mathematically. This paper demonstrates the advantage of using the DL-SMLR model's configuration over the other architectures tested for the estimation of the performance of ORC systems. Table 9 reports the MAE, MSE, and R^2 for Model 4 and Model 8. The MAE, MSE, and R^2 were calculated for test data comparing the value estimated by the trained NN and that taken from the ORC system. As can be seen in Table 9, the maximum estimated MAE value can reach 0.818979 kW, and estimated MSE values are between 0.047053 kW and 0.477598 kW. When Model 8 is considered, the MAE, MSE, and R^2 values for the testing phase are 0.140295 kW, 0.047053 kW, and 0.9528, respectively. The performances with the Model 8, DL based SMLR are better than those with the SMLR and Model 9 (without SMLR steps). The real and estimated P_{orc} values are also plotted in Fig. 11.

Table 8 Case summaries for model 8

Model 8 inputs	$\left[T_{\mathrm{ei}}\ T_{\mathrm{eo}}\ T_{\mathrm{ri}}\ m_y\right]_{1\times4}\times\begin{bmatrix}-0.3165 & -0.0645 & \cdots & -0.2469\\ +0.1347 & +0.2301 & \cdots & +0.1044\\ -0.1207 & -0.1204 & \cdots & +0.2678\\ +0.4035 & +0.0881 & \cdots & -0.3548\end{bmatrix}_{4\times50} + \left[-0.0570\ -0.0347\ -0.0936\ -0.0589\ \cdots\ -0.0762\right]_{1\times50}$						
1st hidden Layer AF	ReLU: $Y_i = \max(0, x_i)$						
2nd hidden layer AF	ReLU: $Y_i = \max(0, x_i)$						
output weights (i)	w_1	w_2	w_3	…	w_8	w_9	w_{10}
1	0.2064	0.2508	− 0.0246	…	− 0.1403	− 0.2579	− 0.1282
2	− 0.1588	0.1425	0.2407	…	0.1484	0.0670	0.3130
3	0.3231	− 0.0332	0.1493	…	− 0.0319	− 0.0294	0.2537
…	…	…	…	…	…	…	…
48	− 0.1911	0.0583	− 0.1210	…	− 0.2323	0.1438	0.0466
49	− 0.1003	− 0.2717	0.3127	…	− 0.0535	− 0.3102	0.0111
50	0.2469	0.1037	0.2680	…	0.0776	0.2936	0.2128
b	0.0016	0.0030	− 0.0252	…	− 0.0467	− 0.0048	− 0.0001
Output Eq.	$P_{\mathrm{orc}}=0.3376\bullet Y_1+0.4277\bullet Y_2-0.5625\bullet Y_3-0.5682\bullet Y_4-0.5729\bullet Y_5-1.0585\bullet Y_6+0.2939Y_7+0.3121\bullet Y_8+0.1959\bullet Y_9+0.2697\bullet Y_{10}+0.0087$						

Table 9 Comparison of best DL-SMLR and SMLR models

Model	Type	Best structure	Test data			Rank
			MAE	MSE	R^2	
Model 8	**DL-SMLR**	**50–50–10–1**	**0.140295**	**0.047053**	**0.9528**	**1**
Model 4	SMLR	T_{ei}, T_{eo}, T_{ri}, m_y	0.818979	0.477598	0.9392	2

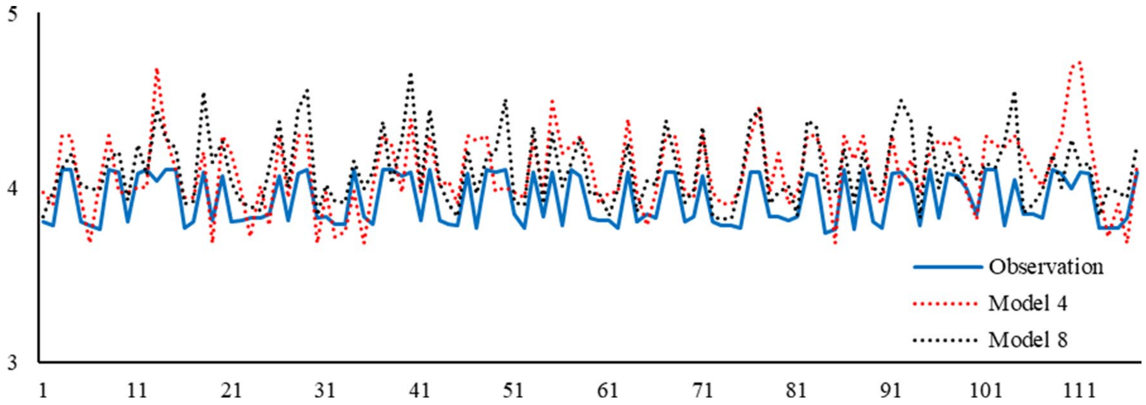


Fig. 11 Comparison of model 4 and model 8

Several studies propose ANN-based models to estimate and optimize the performance of ORC systems. Kovacı et al. estimated the thermal efficiency of ORC systems using the condenser temperature and the evaporator temperature via ANFIS and ANN. They chose R365-mfc and SES32 as fluids. When the performances of the ANN and ANFIS models were compared with actual values, R^2 values were determined to be between 0.97 and 0.99 for SES36 and R365-mfc. It was concluded that the ANN and ANFIS models showed good statistical prediction performance [24]. Kılıç and Arabacı analysed the performance of an ORC system using ANN and ANFIS. They chose R123, R125, $R^2$27, R365mfc, SES36 as refrigerants. Therefore, the efficiency ratio was forecast depending on the steam generator temperature, condenser temperature, subcooling temperature, and superheating temperature. The results of ANN and ANFIS are very satisfactory according to the R^2 values, which ranged from between 0.99670 and 0.99928 [28]. Both studies suggested the use of ANN and ANFIS. However, these models can be challenged when big data needs to be processed and interpreted. In this regard, LeCun et al. suggested the DL approach [48]. Also, in this study, it is seen in Table 9 that using a statistical method such as SMLR for parameter selection developed the designed models. Yang et al. used the ANN method to estimate performance and improvements of an ORC-assisted diesel engine with regard to waste heat. In the study, they compared the prediction accuracy of the ANN model with and without the use of GA. During the analysis, seven key parameters were used to find their effects on the power output of the ORC system. They found working fluid volume flow rate, pump outlet pressure, and expander inlet temperature to be more important, and expander outlet pressure and condenser outlet temperature to be less so (total five of seven parameters), on the power output of the ORC system [26]. However, this study had similar success with four of six parameters. On the other hand, deep learning models are rarely used in performance estimation and optimization of an ORC. When this study was compared with other studies, it was very encouraging. Moreover, the error rates reached are similar. A **c**omparison with other studies can be seen in Table 10.

Although ANN and ANFIS were used in the studies mentioned above, the size of the datasets were limited in terms of both the number of data and the number of parameters used. However, the size of the data and the number of parameters used in the present study required more computing power, but made the DL approach more effective. Especially LeCun et al. inferred that DL method should be used for big datasets [48]. As a conclusion, the DL method was applied in the present study to train the system. As seen in Table 10, Palagi et al. used the DL method to train their network, and they calculated an RMSE error of less than 5%, which is considerably higher than the RMSE error found in the present study. Therefore, the success of the trained network developed in the present paper can be clearly seen.

4 Conclusion

This study investigated the predictability of a 10 kW ORC system using the DL approach, which has a strong ability to deal with nonlinear problems and is highly generalizable. DL and SMLR, with various input combinations, have been compared. It can be seen that the DL models that developed with SMLR steps, optimized with the RMSprop optimization algorithm, and trained with different input variations, demonstrate varying levels of estimation accuracy. Both the DL-SMLR and DL models are capable of estimating the daily P_{orc} with different input combinations. It is also important to note that SMLR has achieved remarkable success.

The results obtained in this study show that:

Table 10 Comparison with other studies

No.	Study	Methods used in the study	Active parameters	Best method	Metric	Results
1	This study	SMLR, DL, DL-SMLR	T_{ei}, T_{eo}, T_{ri}, m_y	DL-SMLR	RMSE	0.0470–0.1997
2	Palagi et al.	Feed Forward neural networks (FFNN), recurrent neural networks (RNN), long short term memory networks (LSTM)	Mass flowrate of the thermal oil at the inlet of the evaporator, temperature of the thermal oil at the inlet of the evaporator, mass flow rate of the working fluid at the inlet of the turbine, temperature of the working fluid at the inlet of the turbine, the pressure of the working fluid at the inlet of the turbine	LSTM	RMSE	0.03–0.08
3	Yang et al.	ANN (FFNN), ANN (FFNN)-genetic algorithm (GA)	Working fluid volume flow rate expander torque expander inlet pressure expander outlet pressure expander inlet temperature condenser outlet temperature Pump outlet pressure	ANN-(FFNN)	Relative error	Less than 5%
4	Kılıç and Arabacı	ANN adaptive neuro-fuzzy (ANFIS)	Steam generator temperature, condenser temperature, subcooling temperature, superheating temperature	ANN	RMSE	0.0023–0.0199
5	Kovacı et al.	ANN adaptive neuro-fuzzy (ANFIS)	Condenser temperature, Evaporator temperature	ANN	RMSE	0.0108

- The three different types of model, SMLR, SMLR-DL, and DL, provide good estimates of the power output of the ORC system.
- According to SMLR, R^2 and Mallows' Cp criteria, it reaches the highest level of consistency in 4 steps with the parameter set composing from T_{ei}, T_{eo}, T_{ri}, and m_y.
- Compared to Model 9, the R^2 of 95.28% achieved by Model 8 means that P_{orc} is better explained by movements in the independent variables T_{ei}, T_{eo}, T_{ri}, and m_y.
- The accuracy of SMLR-DL Model 8 had been improved greatly compared with DL Model 9 without the use of SMLR steps. The *MSE* of P_{orc} was reduced by 83%, and the R^2 was increased by 1.45%. So, the DL Model 8 improved with SMLR in this study could be used as an effective method for P_{orc} estimation with low uncertainties.
- Depending on the availability of the ORC dataset, DL-SMLR Models 6–8 and DL Model 9 can be adopted to estimate the power output of ORC systems where direct measurement of turbine output is not available.
- Statistical inferences based on MSE, RMSE, MAE and R^2 suggest that the DL-SMLR models with three and four input variables are superior to the DL-SMLR and SMLR models with one input.

This study has revealed that through the determination of a proper training set and input parameters by SMLR, the DL-SMLR approach is able to estimate the power generation of ORC systems with acceptable accuracy.

Compliance with ethical standards

Conflict of interest The authors declare no conflict of interest.

References

1. Achinas S, Euverink GJW (2019) Elevated biogas production from the anaerobic co-digestion of farmhouse waste: insight into the process performance and kinetics. Waste Manag Res. https://doi.org/10.1177/0734242X19873383
2. Koç A, Yağlı H, Koç Y, Uğurlu İ (2018) Dünyada ve Türkiye'de Enerji Görünümünün Genel Değerlendirilmesi. Eng Mach Mag 59(692):86–114
3. Cavalcanti EJ, Lima MS, de Souza GF (2020) Comparison of carbon capture system and concentrated solar power in natural gas combined cycle: exergetic and exergoenvironmental analyses. Renew Energy 156:1336–1347
4. Aghbashlo M, Tabatabaei M, Soltanian S, Ghanavati H (2019) Biopower and biofertilizer production from organic municipal solid waste: an exergoenvironmental analysis. Renew Energy 143:64–76
5. Mert I, Karakuş C (2015) A statistical analysis of wind speed data using Burr, generalized gamma, and Weibull distributions in Antakya, Turkey. Turk J Electr Eng Comput Sci 23(6):1571–1586

6. Yağlı H, Karakuş C, Koç Y, Çevik M, Uğurlu İ, Koç A (2019) Designing and exergetic analysis of a solar power tower system for Iskenderun region. Int J Exergy 28(1):96–112
7. Koç Y, Yağlı H (2020) Isı-güç kombine sistemlerinde kullanılan kalina çevriminin enerji ve ekserji analizi. Politek Derg 23(1):181–188
8. Ferreira AC, Silva J, Teixeira S, Teixeira JC, Nebra SA (2020) Assessment of the Stirling engine performance comparing two renewable energy sources: solar energy and biomass. Renew Energy 154:581–597
9. Palacios-Bereche MC, Palacios-Bereche R, Nebra SA (2020) Comparison through energy, exergy and economic analyses of two alternatives for the energy exploitation of vinasse. Energy 197:117231
10. Yağlı H, Koç Y, Koç A, Görgülü A, Tandiroğlu A (2016) Parametric optimization and exergetic analysis comparison of subcritical and supercritical organic Rankine cycle (ORC) for biogas fuelled combined heat and power (CHP) engine exhaust gas waste heat. Energy 111:923–932
11. Yagli H, Koc A, Karakus C, Koc Y (2016) Comparison of toluene and cyclohexane as a working fluid of an organic Rankine cycle used for reheat furnace waste heat recovery. Int J Exergy 19(3):420–438
12. Cavalcanti EJ, Carvalho M, da Silva DR (2020) Energy, exergy and exergoenvironmental analyses of a sugarcane bagasse power cogeneration system. Energy Convers Manag 222:113232
13. Yang A, Su Y, Shen W, Chien IL, Ren J (2019) Multi-objective optimization of organic Rankine cycle system for the waste heat recovery in the heat pump assisted reactive dividing wall column. Energy Convers Manag 199:112041
14. Yan C, Yang A, Chien IL, Shen W, Ren J (2019) Advanced exergy analysis of organic Rankine Cycles for Fischer-Tropsch syngas production with parallel dry and steam methane reforming. Energy Convers Manag 199:111963
15. Köse Ö, Koç Y, Yağlı H (2020) Performance improvement of the bottoming steam Rankine cycle (SRC) and organic Rankine cycle (ORC) systems for a triple combined system using gas turbine (GT) as topping cycle. Energy Convers Manag 211:112745
16. Koc Y, Kose O, Yagli H (2019) Exergy analysis of a natural gas fuelled gas turbine based cogeneration cycle. Int J Exergy 30(2):103–125
17. Koç Y, Yağlı H, Koç A (2019) Exergy analysis and performance improvement of a subcritical/supercritical organic rankine cycle (ORC) for exhaust gas waste heat recovery in a biogas fuelled combined heat and power (CHP) engine through the use of regeneration. Energies 12(4):575
18. Lecompte S, Ntavou E, Tchanche B, Kosmadakis G, Pillai A, Manolakos D, De Paepe M (2019) Review of experimental research on supercritical and transcritical thermodynamic cycles designed for heat recovery application. Appl Sci 9(12):2571
19. Wu D, Zuo J, Liu Z, Han Z, Zhang Y, Wang Q, Li P (2019) Thermodynamic analyses and optimization of a novel CCHP system integrated organic Rankine cycle and solar thermal utilization. Energy Convers Manag 196:453–466
20. Bilgic HH, Guvenc MA, Cakır M, Mistikoglu SA (2019) study on prediction of surface roughness and cutting tool temperature after turning for S235JR steel. Konya Mühendis Bilim Derg 7:966–974
21. Kaba K, Sarıgül M, Avcı M, Kandırmaz HM (2018) Estimation of daily global solar radiation using deep learning model. Energy 162:126–135
22. Yılmaz F, Selbaş R, Şahin AŞ (2016) Efficiency analysis of organic Rankine cycle with internal heat exchanger using neural network. Heat Mass Transf 52(2):351–359
23. Rashidi MM, Galanis N, Nazari F, Parsa AB, Shamekhi L (2011) Parametric analysis and optimization of regenerative Clausius and organic Rankine cycles with two feedwater heaters using artificial bees colony and artificial neural network. Energy 36(9):5728–5740
24. Kovacı T, Şahin AŞ, Dikmen E, Şavklı HB (2017) Performance estimation of organic rankine cycle by using soft computing technics. Int J Eng Appl Sci 9(3):1–10
25. Massimiani A, Palagi L, Sciubba E, Tocci L (2017) Neural networks for small scale ORC optimization. Energy Procedia 129:34–41
26. Yang F, Cho H, Zhang H, Zhang J, Wu Y (2018) Artificial neural network (ANN) based prediction and optimization of an organic Rankine cycle (ORC) for diesel engine waste heat recovery. Energy Convers Manag 164:15–26
27. Bilgiç HH, Yağlı H, Koç A, Yapıcı A (2016). Deneysel bir organik Rankine çevriminde yapay sinir ağları (YSA) yardımıyla güç tahmini. Selcuk Univ J Eng Sci Technol 4(1)
28. Kılıç B, Arabacı E (2019) Alternative approach in performance analysis of organic rankine cycle (ORC). Environ Progress Sustain Energy 38(1):254–259
29. Palagi L, Pesyridis A, Sciubba E, Tocci L (2019) Machine learning for the prediction of the dynamic behavior of a small scale ORC system. Energy 166:72–82
30. Pei G, Li J, Li Y, Wang D, Ji J (2011) Construction and dynamic test of a small-scale organic rankine cycle. Energy 36(5):3215–3223
31. Shu G, Zhao M, Tian H, Wei H, Liang X, Huo Y, Zhu W (2016) Experimental investigation on thermal OS/ORC (oil storage/organic rankine cycle) system for waste heat recovery from diesel engine. Energy 107:693–706
32. Sohani A, Shahverdian MH, Sayyaadi H, Garcia DA (2020) Impact of absolute and relative humidity on the performance of mono and poly crystalline silicon photovoltaics; applying artificial neural network. J Clean Prod 276:123016
33. Yang A, Su Y, Chien IL, Jin S, Yan C, Shen W (2019) Investigation of an energy-saving double-thermally coupled extractive distillation for separating ternary system benzene/toluene/cyclohexane. Energy 186:115756
34. Yang A, Wei R, Sun S, Wei SA, Shen W, Chien IL (2018) Energy-saving optimal design and effective control of heat integration-extractive dividing wall column for separating heterogeneous mixture methanol/toluene/water with multiazeotropes. Ind Eng Chem Res 57(23):8036–8056
35. Chatzopoulou MA, Lecompte S, De Paepe M, Markides CN (2019) Off-design optimisation of organic rankine cycle (ORC) engines with different heat exchangers and volumetric expanders in waste heat recovery applications. Appl Energy 253:113442
36. Salim MS, Kim MH (2019) Multi-objective thermo-economic optimization of a combined organic rankine cycle and vapour compression refrigeration cycle. Energy Convers Manag 199:112054
37. Koç A, Yağlı H, Bilgic HH, Koç Y, Özdemir A (2020) Performance analysis of a novel organic fluid filled regenerative heat exchanger used heat recovery ventilation (OHeX-HRV) system. Sustain Energy Technol Assess 41:100787
38. Perform stepwise regression for Fit Regression Model. https://support.minitab.com/en-us/minitab/18/help-and-how-to/modeling-statistics/regression/how-to/fit-regression-model/perform-the-analysis/perform-stepwise-regression/. Accessed 10 Mar 2020
39. Mahmud M, Kaiser MS, Hussain A, Vassanelli S (2018) Applications of deep learning and reinforcement learning to biological data. IEEE Trans Neural Netw Learn Syst 29(6):2063–2079

40. Üstün İ, Üneş F, Mert İ, Karakuş C (2020) A comparative study of estimating solar radiation using machine learning approaches: DL, SMGRT, and ANFIS. Energy Sources Part A Recov Util Environ Effects. https://doi.org/10.1080/15567036.2020.1781301
41. O'Shea T, Hoydis J (2017) An introduction to deep learning for the physical layer. IEEE Trans Cogn Commun Netw 3(4):563–575
42. Cui X, Zhang W, Tüske Z, Picheny M (2018) Evolutionary stochastic gradient descent for optimization of deep neural networks. In: Advances in neural information processing systems, pp 6048–6058
43. Hinton G, Srivastava N, Swersky K (2012) Lecture 6d-a separate, adaptive learning rate for each connection. Slides of lecture neural networks for machine learning
44. Mert İ, Karakuş C, Üneş F (2016) Estimating the energy production of the wind turbine using artificial neural network. Neural Comput Appl 27(5):1231–1244
45. Mert İ, Üneş F, Karakuş C, Joksimovic D (2019) Estimation of wind energy power using different artificial intelligence techniques and empirical equations. Energy Sources Part A Recovery Util Environ Effects. https://doi.org/10.1080/15567036.2019.1632981
46. Bilgic HH, Mert İ (2020) Comparison of different techniques for estimation of incoming longwave radiation. Int J Environ Sci Technol. https://doi.org/10.1007/s13762-020-02923-6
47. Mallows CL (1973) Some comments on C p. Technometrics 15(4):661–675
48. LeCun Y, Bengio Y, Hinton G (2015) Deep Learn Nat 521(7553):436–444

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.