Prediction of Gas Hydrate Formation Condition by Data-Driven Modeling: Different Machine Learning Models with Vector Quantization and Cuckoo Search Algorithm

Ganji, Zahra
University of Bonn, Bonn, GERMANY

Ganji, Hamid*+
Gas Research Division, Research Institute of Petroleum Industry, Tehran, I.R. IRAN

Shokri, Saeid
Technology and Innovation Group, Research Institute of Petroleum Industry, Tehran, I.R. IRAN

ABSTRACT: *Greenhouse gases can be defined as air pollutants that cause global climate warming. In order to reduce their harmful effects, these gases like methane and carbon dioxide can be stored in the form of compact gas hydrates. Prediction of gas hydrate formation conditions is very important for gas hydrate production and storage in industries. The goal of this study is to develop machine learning methods based on support vector regression and adaptive boosting models for predicting gas hydrate formation conditions for CO_2 and natural gas. In this regard, SVR, AdaBoost.R2, VQ-SVR, VQ-AdaBoost.R2, CS-VQ-SVR, and CS-VQ-AdaBoost.R2 models have been developed and compared to obtain a model with the best performance. The cuckoo search optimization algorithm and vector quantization technique have also been utilized to determine the optimal values of the models' hyper-parameters, reduce the computation time, and improve the accuracy and robustness of the models. As a result, since the values of the coefficient of determination and root mean square error for the CS-VQ-SVR model are 0.0215 and 0.9995, respectively, and the best agreement between predicted and actual values in this model's graphs is obtained, it can be concluded that the CS-VQ-SVR model has the best accuracy and robustness among other developed models in predicting gas hydrate formation pressure with time. These results show that machine learning is viable for predicting the conditions of gas hydrate formation and preventing greenhouse gas emissions in industries.*

KEYWORDS: *Gas hydrate formation; Greenhouse emissions; Air pollution; Machine learning; Data-driven models.*

INTRODUCTION

Global warming, defined as the continuous rise in the average temperature of Earth's climate system, causes extreme droughts, wildfires, floods, and tropical storms. The emission of greenhouse gases and subsequently

* *To whom correspondence should be addressed.*
+E-mail address: ganjih@ripi.ir
1021-9986/2023/4/1376-1387 12/$/6.02

increasing their level in the environment is the principal contributor to the change of climate and global warming problems [1]. Reducing greenhouse gas emissions is one of the most critical issues in coping with global warming in the Paris Agreement. Consequently, emission control and the quality of output products in the process units are very crucial. The quality control of the output products is usually done by time and money-consuming lab technics or expensive online analyzers. These approaches are not only expensive but also cause a lot of problems for the process units due to their breakdown. Considering the daily high amounts of generated and saved data in oil and gas industries, we have this opportunity to benefit from artificial intelligence and data-driven models to design and develop a soft sensor alongside Industry 4.0 and the age of digital transformation. This can be an alternative/parallel for a real analyzer to operate more efficiently in the unit.

Two of the most important greenhouse gases are carbon dioxide and methane. These gases can be stored in the form of ice-like compounds known as gas hydrates in order to control their emission and prevent their harmful effects. Gas hydrates are composed of water and a certain number of natural gas molecules under favorable conditions of pressure and temperature. The guest molecules are enclosed in host molecules' cavities that are composed of hydrogen bonding in water. Typical natural gas molecules that can form gas hydrate include CH_4, C_2H_6, C_3H_8, and CO_2 [2].

There are some methods for prediction of gas hydrate formation conditions such as experimental based methods [3-5], but the problem with these methods is that they are time consuming and expensive.

In this regard, alternative methods such as data-driven models can be used. Data-driven modeling is a technique that makes strategic decisions based on data analysis and interpretation without explicit knowledge of the physical behavior of a system. Machine learning techniques [6-9] are one of the main groups of data-driven approaches. There are a large number of these techniques, two of which are Support Vector Machines (SVM) [10-13] and committee machines [14-16].

A committee machine generates an ensemble of predictors and combines the prediction of each committee member to predict the overall prediction for a new input [17]. Committee machines combine estimators so that the general performance improves compared to the performance of each single estimator [18]. One of the most popular committee machine methods is boosting [19]. Boosting algorithms are available in different versions for classification and regression problems. AdaBoost [20], short for adaptive boosting, is a boosting algorithm that was extended by *Freund* and *Schapire* [21] for regression problems under the name AdaBoost.R [17]. Then *Ducker* [22] introduced a modification version of AdaBoost.R called AdaBoost.R2 algorithm. *D.L. Shrestha* and *D.P. Solomatine* [17] introduced a new boosting algorithm called AdaBoost.RT with a view to solve regression problems. In this algorithm, examples which have higher estimation error than the preset threshold value are filtered out. As a result, this algorithm has higher performance than other boosting methods, bagging, artificial neural networks, and a single M5 model tree. *S. Patil*, *A. Patil*, and *V. Phalle* [23] used AdaBoost regressor to predict the Remaining Useful Life (RLU) of rolling element bearing and found that their proposed model has better results than other data-driven methods from the literature.

In recent years, a new learning method called SVM developed by *Vapnik* [24] has become an important topic in machine learning and competed with other methods such as neural networks and decision trees. SVM is a promising technique for both classification and regression problems [25]. The SVM method for regression problems is called the Support Vector Regression (SVR) method [26,27]. The basic idea behind SVR is to find the best hyper plane as a decision function in high-dimensional space [28].

Hyper-parameter optimization, the problem of choosing a set of optimal hyper-parameters for a learning algorithm, helps a machine learning model to optimally solve a problem. One of the most common methods determining optimal values of hyper-parameters is the Grid Search Method (GSM) [28]. In some research, GSM has been adopted to optimize hyper-parameters of AdaBoost regression and SVR models [29, 30]. However, GSM is time-consuming and computationally expensive because it searches over the whole hyper-parameter space. Due to these shortcomings, *X.S. Yang* and *S. Deb* [31] proposed a meta-heuristic algorithm for optimization called the Cuckoo Search (CS) algorithm [32, 33]. CS algorithm has some advantages such as strong and global search with fewer parameters, having a good search path, and solving multi-objective problems powerfully [34]. *Y. Dong*, *Z. Zhang*, and *W.-C. Hong* [35] used SVR model with a seasonal mechanism to forecast electric load with

improved CS algorithm called chaotic cuckoo search. Their proposed model obtains better results than other alternative models.

Since there are large datasets available in process industries, processing the data with high-speed and extensive memory capacities is an important issue. The Vector Quantization (VQ) technique [36,37] compresses the data with the aim of solving this problem and reduces the training time for selecting optimal parameters in a robust system [28].

In recent years there are some research which have used data-driven models to predict hydrate formation condition. *Tan et al*. [38] built a mechanism-based data-driven modeling method to predict hydrate formation. Based on the collected data, including temperature, pressure and components, a data-driven method was introduced to identify the unknown parameters in the mechanism model. Four different component systems were calculated using the mechanism model, empirical model and data-driven mechanism model for comparison. Results show that the average error of the data-driven model is as low as 0.0085 MPa, and this method can overcome the irrationality of prediction caused by only using historical data or mathematical formulas.

Yu et al. [39] employed machine learning to predict the formation condition of natural gas hydrates to overcome the high computation cost and low accuracy. Three data-driven models, Random Forest, Naive Bayes, and Support Vector Regression (SVR) were tentatively used to determine the formation condition of hydrate formed by pure and mixed gases. The comparison of results predicted by Chen–Guo model and machine learning models with the experimental data indicated that the Random Forest model performed better than the Naive Bayes and SVR models on both computation speed and accuracy.

Monday and *Odutola* [40] developed machine learning models after a kinetic inhibitor to predict the gas hydrate formation and pressure changes within the natural gas flow line. Green hydrate inhibitors A, B, and C were obtained as plant extracts and applied in low dosages (0.01 wt.% to 0.1 wt.%) on a 12-meter skid-mounted hydrate closed flow loop. From the data generated, the optimal dosages of inhibitors A, B, and C were observed to be 0.02 wt.%, 0.06 wt.%, and 0.1 wt.% respectively. The data associated with these optimal dosages were fed to a set of supervised machine learning algorithms (Extreme gradient boost, Gradient boost regressor and Linear regressor) and a deep learning algorithm (Artificial Neural Network). The output results from the set of supervised learning algorithms and Deep Learning algorithms were compared in terms of their accuracies in predicting the hydrate formation and the pressure within the natural gas flow line. All models had accuracies greater than 90%.

Sadi et al. [41] developed two artificial intelligence models based on an adaptive neuro-fuzzy inference system (ANFIS) and a support vector machine (SVM) technique to predict the desalination efficiency of produced water through a hydrate-based desalination treatment process. A genetic algorithm as an evolutionary optimization method has been used to determine the optimal values of SVM model coefficients. For the ANFIS model, the coefficient of determination (R^2) and average absolute relative error (AARE) are 0.9927 and 0.58%, respectively. The values of AARE and R2 for the SVM model are obtained at 0.35% and 0.9985, respectively.

Hosseini and *Leonenko* [42] proposed machine learning-based models to predict methane-hydrate formation temperature for a wide range of brines. The results showed that the extremely randomized trees are capable of predicting methane-hydrate formation temperature with good accuracy.

Xu et al. [43] compared five machine-learning to develop prediction tools for the estimation of methane hydrate formation temperature in the presence of salt water. These machine learning algorithms were Multiple Linear Regression, k-Nearest Neighbor, Support Vector Regression, Random Forest, and Gradient Boosting Regression. The experimental data span salt concentrations up to 29.2 wt% and pressures up to 200 MPa. Among these five machine learning methods, Gradient Boosting Regression gave the best prediction with R^2=0.998 and *AARD* = 0.074%.

Ibrahim et al. [44] investigated the applicability of radial basis function networks and support vector machines to predict hydrate formation conditions. Data-based models enable the oil industry to predict the conditions leading to hydrate formation hence preventing clogging of the pipeline and high-pressure buildup that could lead to sudden bursts at the connections.

Kumari et al. [45] discussed the least square support vector machine and artificial neural network models for the prediction of stability conditions of gas hydrates and

the use of Genetic Programming (GP) and Genetic Algorithm (GA) to develop a generalized correlation for predicting equilibrium conditions of gas hydrates.

As can be seen, almost all data driven models have been used to predict hydrate thermodynamic condition. Based on our information, no data driven model have used to predict hydrate formation kinetics i.e. prediction of pressure change with time.

In this paper, in order to predict CO_2 and natural gas hydrate formation pressure with time, different combinations of SVR and AdaBoost.R2 models with CS algorithm and VQ technique have been developed for the first time. The developed models are SVR, AdaBoost.R2, VQ-SVR, VQ-AdaBoost.R2, CS-VQ-SVR, and CS-VQ-AdaBoost.R2. Then, results have been compared with each other to find the most reliable and robust model among others. This machine learning approach can be used to predict the conditions of gas hydrate formation, and it has application in industries to produce gas hydrates for sequestration of greenhouse gasses.

METHODOLOGY

Adaptive boosting regression

Adaptive Boosting (AdaBoost) algorithm is a supervised learning algorithm and the most widely used form of boosting algorithms which combines multiple weak learners into a single strong learner. In AdaBoost algorithm each one of the weak learners is a model slightly better than random guessing, such as a decision tree. AdaBoost can be used for both classification and regression problems. In the present study, AdaBoost.R2 which is one of the boosting algorithms for regression problems has been used, same as Ducker [22]. In AdaBoost.R2, final prediction is a weighted average of predictions given by each weak learner and the functioning of the algorithm is such that the information from previous weak learner is fed to the next one so that the error of previous learner improves [23] and the performance of a particular weak learner depends on the previous one [22]. The first weak learner is trained using equal weighting coefficients, and in subsequent boosting rounds these weighting coefficients will be updated. The weights of poorly predicted examples increase and the weights of well-predicted ones decrease [17]. There are three hyper parameters for AdaBoost.R2 algorithm. The first one is base estimator from which the boosted ensemble is built, and its default estimator is decision tree with maximum depth as three. The second one is number of estimators which is the maximum number of estimators at which boosting is terminated. The last one is learning rate which is a weight applied to each weak learner at each boosting iteration. A higher learning rate increases the contribution of each base learner. In this paper, base estimator has been tuned to its default value, and the number of estimators and the learning rate has been tuned by using cuckoo search optimization algorithm.

Support Vector Regression (SVR)

SVR is one of the most flexible and robust algorithms for regression problems, which falls under the supervised machine learning models category. SVR allows us to model non-linear relationships between variables. Different loss functions can be used in SVR, the most common of which is robust ε-insensitive loss function (L_ε) [28]:

$$L_\varepsilon(f(x)-y)=\begin{cases} |f(x)-y|-\varepsilon & \text{for } |f(x)-y|\geq\varepsilon \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Where ε is a tunable parameter and it determines the width of a tube around the estimated function (hyper plane). The main goal of SVR is to find an optimal hyper plane which reduces the total deviation of all data points to less than or equal to epsilon by putting more data points inside the tube and reducing slack variables. slack variables, ξ and ξ^*, measure the distance from training data values outside the tube and edge values of ε -tube, and they can be tuned by regularization parameter C. C is a penalty of misclassifying a data point. As C increases, algorithm puts more points inside the ε -tube and try to minimize slack variables as much as possible. Therefore, the data can be fitted better. However, it makes our model less robust to outliners, so the risk of over fitting can be increased.

The objective function which should be minimized in SVR algorithm can be formulated as follows:

$$Min\ \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(\xi_i^- + \xi_i^+) \quad (2)$$

With the following constraints:

$$y=\begin{cases} y_i - (\langle w, x_i\rangle + b) \leq \varepsilon + \xi_i \\ (\langle w, x_i\rangle + b) - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases} \quad (3)$$

SVR algorithm can deal with non-linear problems with the use of kernel trick. A kernel is a function which maps a non-linear dataset from original space into a higher dimensional feature space, then constructs a linear

Table 1: CS algorithm parameters

Variable	Value
Initial population size	50
Maximum generation	100
$\boldsymbol{\alpha}$	0.01
$\boldsymbol{p_a}$	0.25
$\boldsymbol{\beta}$	1.5

regression function there. There are different kernel functions such as linear kernel, polynomial kernel, sigmoid kernel, and Gaussian (radial basis function, RBF) kernel. Since the Gaussian kernel has high accuracy and good generalization ability [46], it has been used in this study. The Gaussian kernel equation is as follows:

$$K(x,x') = \exp\left(-\frac{\|x-x'\|^2}{2\sigma^2}\right) \quad (4)$$

Where gamma, σ, is a hyper-parameter for determining how much curvature a decision boundary should have.

In the present study, the tube size, ε , the regularization parameter, C, and the parameter of the RBF kernel, σ, have been specified by using cuckoo search optimization algorithm.

Cuckoo Search (CS) algorithm

Cuckoo Search (CS) inspired by the reproduction strategy of cuckoo birds is a meta-heuristic algorithm. Similar to other evolutionary algorithms, CS begins with primary population of cuckoos. These cuckoos lay their eggs in the nests of other host birds. The host birds can realize that the eggs do not belong to them and either throw them out or abandon the whole nest to build another nest in a new location. However, if the eggs are not recognized by host birds, they can grow up and become mature birds. the term which CS algorithm want to optimize is the position in which more eggs survive. Cuckoo birds constantly lay new eggs and choose a nest around the current best position by Lévy flight behaviors. This process continues until the best position which maximize the eggs survival rate is found, and most of the cuckoo population are gathered there. There are three rules for CS algorithm used by *X.S. Yang* and *S. Deb* [30]: (1) each cuckoo lays one egg at a time in a random nest of a host bird; (2) eggs with the best quality would transfer to the next generation; (3) the number of nests are fixed, and the host bird can detect the stranger egg with a probability $p_a \in [0,1]$. In this case the host bird can either remove it or abandon the whole nest to build a new nest in a new location.

Based on these three rules, the steps of the CS algorithm via Lévy flight algorithm is presented by *Zheng* and *Zhou* [47].

In CS algorithm the new position of cuckoo i is created as follows:

$$nest_i^{t+1} = nest_i^t + \alpha \oplus lévy(\beta) \quad (5)$$

Where alpha, > 0 , is the size of each step and β, $(0 < \beta \leq 2)$, is a constant value given as an input to Lévy flights function. The product $\oplus$ means entry-wise multiplication. Lévy flights provide a random walk with random steps drawn from a Lévy distribution for large steps as follows [47]:

$$Lévy \sim u = t^{-1-\beta} \quad (6)$$

The CS algorithm parameters, which have been applied in this paper to optimize hyper-parameters of AdaBoost.R2 and SVR models, are shown in Table 1.

Vector quantization (VQ)

VQ is an efficient technique for data compression, and it is based on the principle of block coding. In this technique, each training data is mapped to a vector called code vector. This vector is a list of numbers, and the number of input and output attributes in it is as same as the training set. In the algorithm of this technique, an initial codebook which is a finite set of code vectors is needed. The initial codebook can be generated using randomly selected instances from the training set or randomly generated vectors with the same scale as the training data. This algorithm executes in some iterations. In each iteration, the most similar code vector is selected from the code book for each instance in the training dataset. The goal of the algorithm is to find code vectors so that the average pairwise distance between the training vectors and their corresponding code vectors is minimized [48].

As a result, we can conclude that VQ technique comprises of three stages: codebook generation, vector encoding and vector decoding. It quantizes and simplifies a large dataset, so the time for choosing optimal parameters and the training time reduce; moreover, the prediction accuracy and robustness of the system increase [28].

EXPERIMENTAL SECTION

The experimental setup used in this research has been shown in Figure 1 which its main part is hydrate reactor with a volume of approximately 300 cm^3. A cooling

Table 2: Sample of experimental data for variation of pressure and temperature with time during natural gas hydrate formation

Time (min)	Temperature (k)	Pressure (psi)
0	276.2	1400
60	276.2	1380
120	276.2	1355
.	.	.
.	.	.
.	.	.
360	276.8	920
.	.	.
.	.	.
.	.	.
540	276.7	750
.	.	.
.	.	.
.	.	.
1800	276.4	580

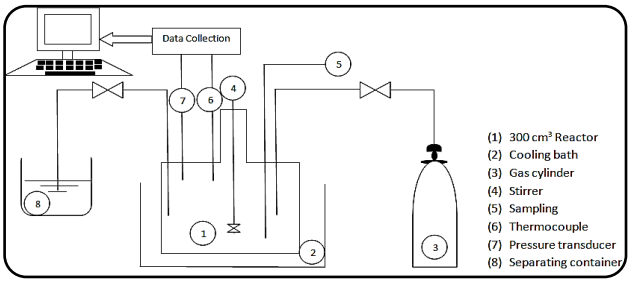

Fig. 1: Schematic of the experimental set-up

medium circulatory system is used to control the temperature, a gas cylinder to inject the gas and a mixer to mix the contents of the hydrate reactor. A pressure transducer with the scale of 0.5 psi (accuracy approximately 0.5%) and a thermocouple with the scale of 0.1ºK (accuracy approximately 0.4%) are used to measure the pressure and temperature and there was a data collection to record them during the process. A computer system with the suitable data acquisition software is used to record and collect experimental data during the time.

The reactor is washed and rinsed with de-ionized water and then 75 cm^3 water is charged into the reactor for each experiment. The reactor is purged with natural gas and two different hydrate formers (CO_2 and natural gas) are used to form hydrate. The reactor is pressurized with hydrate former to 1400 psia at 298.2 K. After reaching equilibrium at the initial temperature and pressure, the system is cooled to the hydrate formation temperature (276.2 K). The mixer is then started at a rate of 200 rpm to initiate hydrate formation. The temperature and pressure changes are recorded during hydrate formation in each 10 second and saved in an excel file. A set of data was shown in table 2 as a sample data for variation of pressure and temperature with time during natural gas hydrate formation. Due to the large number of points, data have been shown every 60 minutes.

MODEL DEVELOPMENT

In this research, models have been developed using Python 3.7.9 programming language, Jupyter Notebook environment, and the scikit-learn library. Experiments have been done on windows 10, processor core i7 and RAM 8 GB to obtain high accuracy and performance.

The SVR, AdaBoost.R2, VQ-SVR, VQ-AdaBoost.R2, CS-VQ-SVR and CS-VQ-AdaBoost.R2 models have been developed for predicting the gas hydrate formation condition and compared to each other. In order to evaluate the performance of these developed models, a data set with 12017 data has been selected. The hydrate formation time, temperature, and types of gases have been selected as models' inputs to determine hydrate formation pressure with time as a target value.

The strategy used to develop each one of the models in the present study has been presented in Figure 2, and according to that, the main steps of model development are as follows:

Step 1: Selecting the train and test datasets; Selecting 9404 samples as training data and 2613 samples as testing data randomly from the original dataset.

Step 2: Data compression: Using VQ technique to change the dataset to a low-dimensional and dense one so that the training and computation time reduce.

Step 3: Performing hyper-parameter optimization: Determining the tube size, ε , the regularization parameter, C, and the parameter of the RBF kernel, σ, in SVR model as well as base estimator and number of estimators in AdaBoost.R2 model using CS optimization algorithm.

Step 4: Training the model: Using training samples and optimized hyper-parameters to train the model before prediction.

Step 5: Model prediction and validation: Using testing samples as inputs of the model to obtain the predictive values and validate the model.

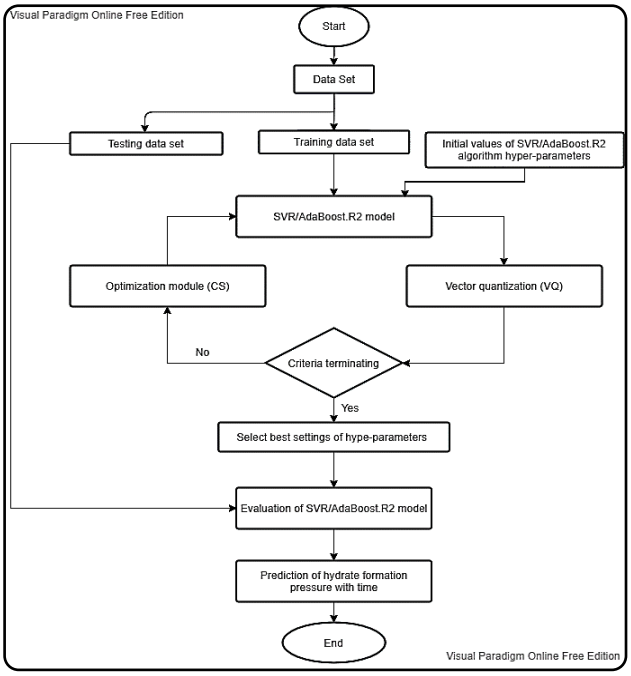

Fig. 2: The procedure of model development

RESULTS AND DISCUSSION

In this research, in order to evaluate and compare the models' performance, graphical representations and statistical analysis methods have been employed. Cross plots and graphs with experimental data have been used as graphical representations and Root Mean Square Error (RMSE) and coefficient of determination (R^2) have been utilized as statistical analysis. The statistical parameters are as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (7)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \quad (8)$$

Where $\hat{y}$, y, $\bar{y}$ and n are values predicted by the model, experimental data, a mean value of experimental data and number of data points, respectively.

Table 2 lists the values of statistical parameters for SVR, AdaBoost.R2, VQ-SVR, VQ-AdaBoost.R2, CS-VQ-SVR and CS-VQ-AdaBoost.R2 models. According to this table, the following assumptions can be made:

1) CS-VQ-SVR model has the best performance, and the AdaBoost.R2 model has the worst performance among other models.
2) The VQ technique improves the performance and accuracy of SVR and AdaBoost.R2 models. It also reduces computational time.

Table 3: Statistical criteria for developed models

Model	Train data		Test data	
	RMSE	R^2	RMSE	R^2
SVR	0.3151	0.9006	0.3518	0.8762
AdaBoost.R2	0.3511	0.8767	0.4033	0.8373
VQ-SVR	0.0949	0.9909	0.0988	0.9902
VQ-AdaBoost.R2	0.0386	0.9613	0.0351	0.9648
CS-VQ-SVR	0.0001	0.9999	0.0215	0.9995
CS-VQ-AdaBoost.R2	0.1435	0.9600	0.1899	0.9639

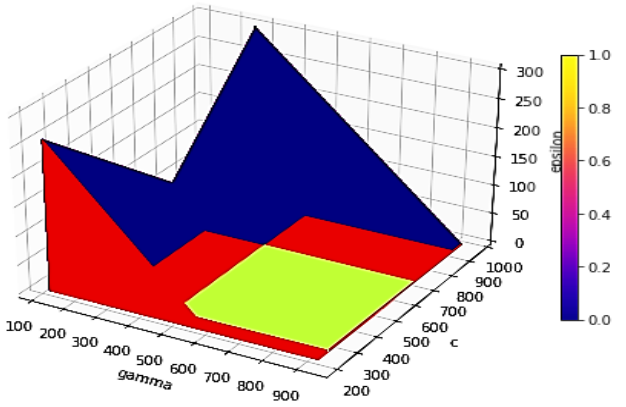

Fig. 3: Hyper-parameters tuning with CS optimization algorithm in CS-VQ-SVR model

3) The CS optimization algorithm improves the performance of models by finding optimized values for hyper-parameters as shown in Figure 3 for the CS-VQ-SVR model. As observed, when the values of C, and σ gradually approach their optimum values, the accuracy of the model for the training dataset improves. The accuracy is shown by different colors from blue for minimum accuracies to green for maximum accuracies on the color bar.
4) The SVR model is more robust and reliable than the AdaBoost.R2 model.

Cross plots of predicted values from model predictions and actual values from experimental data for SVR, AdaBoost.R2, VQ-SVR, VQ-AdaBoost.R2, CS-VQ-SVR and CS-VQ-AdaBoost.R2 models are shown in Fig. 4. As observed, points got closer to the diagonal line as the VQ technique and CS optimization algorithm had been applied to the models and in the SVR model's graph, points are more accumulated around the diagonal line compared to other models' graphs.

Graphs to compare predicted with actual values for every 20 data for SVR, AdaBoost.R2, VQ-SVR, VQ-AdaBoost.R2,

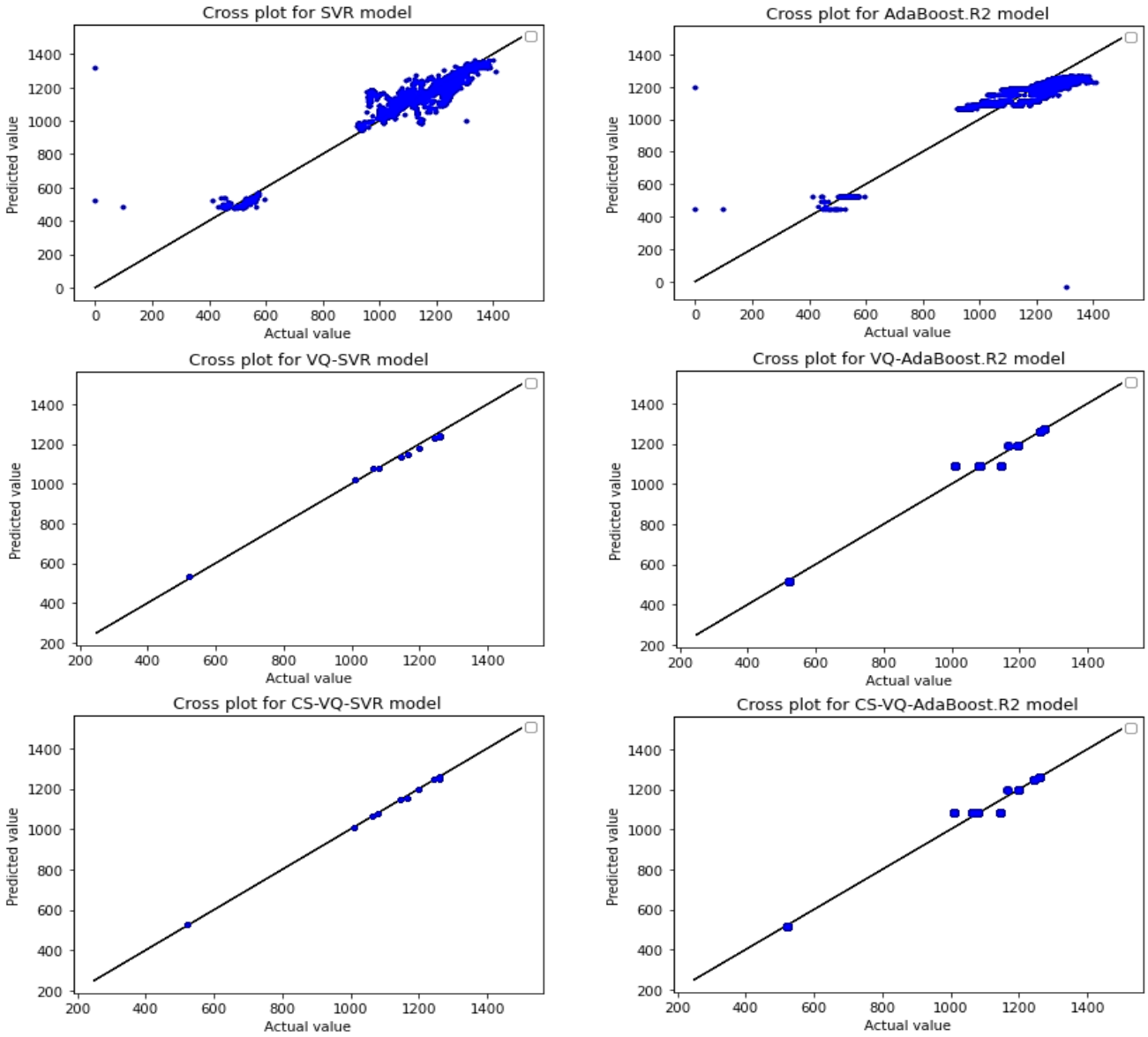

Fig. 4: The cross plots for different models. SVR model. AdaBoost.R2 model. VQ-SVR model. VQ-AdaBoost.R2 model. CS-VQ-SVR model. CS-VQ-AdaBoost.R2 model.

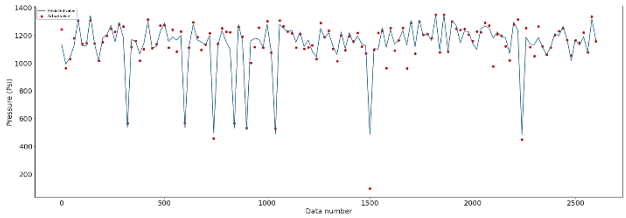

Fig. 5: comparison of actual values with predicted values for the SVR model

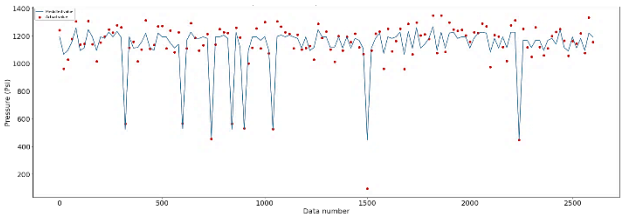

Fig. 6: comparison of actual values with predicted values for the AdaBoost.R2 model

CS-VQ-SVR and CS-VQ-AdaBoost.R2 models are demonstrated in Figures 5, 6, 7, 8, 9 and 10, respectively. These figures also show that the agreement between predicted values and actual values in each model increase with the VQ technique and CS optimization algorithm. According to Fig. 9 and Fig. 6, the best and the worst agreement between

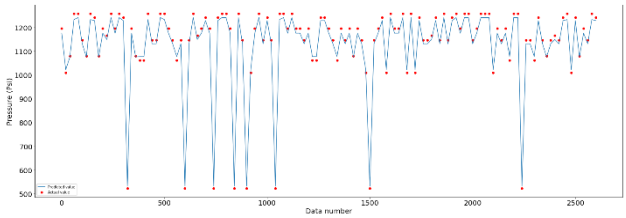

Fig. 7: comparison of actual values with predicted values for the VQ-SVR model

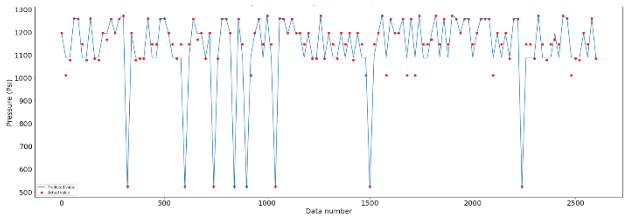

Fig. 8: comparison of actual values with predicted values for the VQ-AdaBoost.R2 model

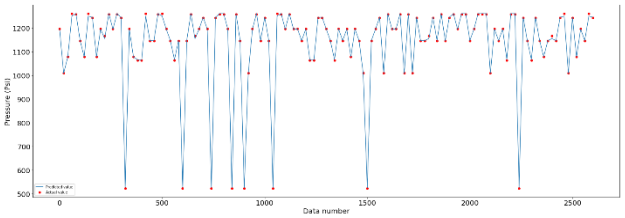

Fig. 9: comparison of actual values with predicted values for the CS-VQ-SVR model

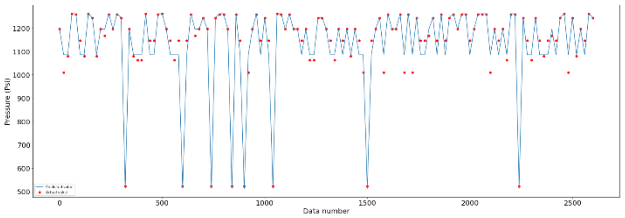

Fig. 10 comparison of actual values with predicted values for the CS-VQ-AdaBoost.R2 model

predicted values and actual values are for CS-VQ-SVR and AdaBoost.R2 models, respectively.

As a result, the above observations confirm that the CS-VQ-SVR model has excellent performance and can be utilized as a reliable, robust and fast method for predicting variation of pressure with respect to time at a given temperature in gas hydrate formation procedure.

CONCLUSIONS

Carbon dioxide and methane, which are two harmful greenhouse gasses, can be stored in gas hydrates to control their release and prevent their harmful effects. Gas hydrate formation conditions have been usually determined experimentally, which is costly and associated with errors. The present study provides data-driven models in the field of machine learning and artificial intelligence to predict gas hydrate formation conditions. In this regard different models based on SVR and AdaBoost.R2 models have been developed to predict the variation of pressure with time at a given temperature in the gas hydrate formation procedure. The results can be used to predict hydrate formation kinetics and reduce the experimental time and cost. The developed models are SVR, AdaBoost.R2, SVR-VQ, AdaBoost.R2-VQ, SVR-VQ-CS, and AdaBoost.R2-VQ-CS. These models have been compared with each other using graphical representations and statistical analysis methods to find one with the best performance. The CS algorithm as a meta-heuristic optimization technique has been applied to models in order to obtain the optimal values for their parameters. In order to make these models more robust and accurate as well as speed up the computation time, the VQ technique has been used. According to the results, the values of R^2 and $RMSE$ for the SVR-VQ-CS model in the testing dataset have been obtained as 0.9995 and 0.0215, respectively, and the graphs associated with this model show the best agreement between predicted and actual values. Therefore, the SVR-VQ-CS model has the best performance among developed models, and it can be utilized for the first time as a reliable, robust and fast method to predict the hydrate formation pressure with time. This Study confirmed that machine learning could be applied to predict gas hydrate formation conditions. It is expected that this study can be applied to utilize gas hydrate in industries to control greenhouse gases' harmful effects on Earth's atmosphere.

Nomenclature

Adaptive boosting	AdaBoost
Bias term	b
Regularization parameter (hyper-parameter)	C
Cuckoo search	CS
Number of experimental data points	n
Probability of detection of strange egg by host bird in CS algorithm (hyper-parameter)	p_a
Coefficient of determination	R^2
Gaussian radial basis kernel function	RBF

Root mean square error	RMSE
Support vector machine	SVM
Support vector regression	SVR
Output value	y
Predicted value of output	$\hat{y}$
Average value of output	$\bar{y}$
Vector quantization	VQ
Weight vector	w
Size of each step in CS algorithm (hyper-parameter)	α
Hyper-parameter in CS algorithm	β
Tube size (hyper-parameter)	ε
Slack variables	ξ
Slack variables	ξ^*
Hyper-parameter of RBF kernel	σ

Received: Now. 09, 2022 ; Accepted: Feb. 06, 2023

References

[1] Yu Y.S., Zhang X., Liu J.W., Lee Y., Li X.S., Natural Gas Hydrate Resources and Hydrate Technologies: A Review and Analysis of the Associated Energy and Global Warming Challenges, *Energy Environ. Sci.*, **14:** 5611-5668 (2021).

[2] Sloan E.D., Koh C.A., "Clathrate Hydrates of Natural Gases", Third ed. Taylor & Francis Group, New York (2008).

[3] Partoon B., Javanmardi J., Effect of Mixed Thermodynamic and Kinetic Hydrate Promoters on Methane Hydrate Phase Boundary and Formation Kinetics, *J. Chem. Eng. Data*, **58:** 501−509 (2013).

[4] Rahimi Mofrad H., Ganji H., Nazari K., Kameli M., Rezaie Rod A., Kakavand M., Rapid Formation of Dry Natural Gas Hydrate with High Capacity and Low Decomposition Rate Using a New Effective Promoter, *J. Pet. Sci. Eng.*, **147:** 756–759 (2016).

[5] Wu Y., Shang L., Pan Z., Xuan Y., Baena-Moreno F.M., Zhang Z., Gas Hydrate Formation in the Presence of Mixed Surfactants and Alumina Nanoparticles, *J. Nat. Gas Sci. Eng.*, **94:** 104049 (2021).

[6] Zhang Z.J., Duraisamy K., "Machine Learning Methods for Data-Driven Turbulence Modeling", *22nd AIAA Computational Fluid Dynamics Conference*, Dallas (2015).

[7] Alsina E.F., Chica M., Trawinski K., Regattieri A., On the Use of Machine Learning Methods to Predict Component Reliability from Data-Driven Industrial Case Studies, *The Int. J. Adv. Man. Tech.*, **94:** 2419–2433 (2018).

[8] Bodendorf F., Merkl P., Franke J., Intelligent Cost Estimation by Machine Learning in Supply Management: A Structured Literature Review, *Comp. & Ind. Eng.*, **160:** 107601 (2021).

[9] Pallonetto F., Jin C., Mangina E., Forecast Electricity Demand in Commercial Building with Machine Learning Models to Enable Demand Response Programs, *Energy and AI*, **7:** 100121 (2021).

[10] Gui G., Pan H., Lin Z., Li Y., Yuan Z., Data-Driven Support Vector Machine with Optimization Techniques for Structural Health Monitoring and Damage Detection, *KSCE J. Civil Eng.*, **21(2):** 523-534 (2017).

[11] G Abo-Khalil A., Lee D.C., MPPT Control of Wind Generation Systems Based on Estimated Wind Speed Using SVR, *IEEE Transactions on Industrial Electronics*, **55(3):** 1489-1490 (2008).

[12] Hong W.C., Dong Y., Chen L.Y., Wei S.Y., SVR with Hybrid Chaotic Genetic Algorithms for Tourism Demand Forecasting, *Applied Soft Computing*, **11(2):** 1881-1890 (2011).

[13] Lee K., Cho S., Asfour S., Web-based Algorithm for Cylindricity Evaluation using Support Vector Machine Learning, *Comp. & Ind. Eng.*, **60(2):** 228-235 (2011).

[14] Guo J.J., Luh P.B., Improving Market Clearing Price Prediction by Using a Committee Machine of Neural Networks, *IEEE Transactions on Power Systems*, **19(4):** 1867-1876 (2004).

[15] Moslem B., Khalil M.O., Diab M., Chkeir A., Marque C., "Combining Multiple Support Vector Machines for Boosting the Classification Accuracy of Uterine EMG Signals", *18th IEEE International Conference on Electronics, Circuits, and Systems*, Beirut, Lebanon, 11-14 December, (2011).

[16] Rustempasic I., Can M., Diagnosis of Parkinson's Disease Using Principal Component Analysis and Boosting Committee Machines, *Southeast Europe Journal of Soft Computing*, **2(1):** 102-109 (2013).

[17] Shrestha D.L., Solomatine D.P., Experiments with AdaBoost.RT, an Improved Boosting Scheme for Regression, *Neural Computation*, **18(7):** 1678-1710 (2006).

[18] Hu Y.H., Hwang J.N., "Handbook for Neural Network Signal Processing", CRC Press, 1st Edition, (2001).
[19] Schapire R.E., The Strength of Weak Learnability, *Machine Learning*, **5:** 197-227 (1990).
[20] Kaur R., Schaye C., Thompson K., Yee D., Zilz R., Sreenivas R.S., Sowers R., Machine Learning and Price-Based Load Scheduling for an Optimal IoT Control in the Smart and Frugal Home, *Energy and AI*, **3:** 100042 (2021).
[21] Freund Y., Schapire R., A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting, *J. Comp. System Sci*., **55(1):** 119-139 (1997).
[22] Drucker H., Improving Regressors Using Boosting Techniques, *Icml*, **97:** 107-115 (1997).
[23] Patil S., Patil A., Phalle V., "Life Prediction of Bearing by Using Adaboost Regressor", *An International Conference on Tribology, TRIBOINDIA-2018, VJTI, Mumbai, India*, (2018).
[24] Vapnik V.N., "The Nature of Statistical Learning Theory", Springer, New York, 1st Edition (1995).
[25] Wang J.Y., "Application of Support Vector Machines in Bioinformatics", Master Thesis, National Taiwan University (2002).
[26] Rajasekaran S., Gayathri S., Lee T.L., Support Vector Regression Methodology for Storm Surge Predictions, *Ocean Engineering*, **35(16):** 1578-1587 (2008).
[27] Yang C.C., Shieh M.D., A Support Vector Regression based Prediction Model of Affective Responses for Product form Design, *Comp. Ind. Eng.*, **59(4):** 682-689 (2010).
[28] Shokri S., Sadeghi M.T., Ahmadi Marvast M., Narasimhan S., Soft Sensor Design for Hydrodesulfurization Process using Support Vector Regression based on WT and PCA, *Journal of Central South University*, **22:** 511−521 (2015).
[29] Vardakas J.S., Zorba N., Verikoukis C.V., A Survey on Demand Response Programs in Smart Grids: Pricing Methods and Optimization Algorithms, *IEEE Communications Surveys & Tutorials*, **17(1):** 152-178 (2015).
[30] Huang Q., Mao J., Liu Y., An Improved Grid Search Algorithm of SVR Parameters Optimization, *IEEE 14th International Conference on Communication Technology*, Chengdu, China, 9-11 November (2012).
[31] Yang X.S., Deb S., Cuckoo Search via Lévy flights, *World Congress on Nature & Biologically Inspired Computing (NaBIC)*, Coimbatore, India, 9-11 December (2009).
[32] Laha D., N.D. Gupta J., An Improved Cuckoo Search Algorithm for Scheduling Jobs on Identical Parallel Machines, *Comp. Ind. Eng.*, **126:** 348-360 (2018).
[33] Valian E., Tavakoli S., Mohanna S., Haghi A., Improved Cuckoo Search for Reliability Optimization Problems, *Comp. Ind. Eng.*, **64(1):** 459-468 (2013).
[34] Jiang M., Luo J., Jiang D., Xiong J., Song H., Shen J., A Cuckoo Search-Support Vector Machine Model for Predicting Dynamic Measurement Errors of Sensors, IEEE Access, **4:** 5030 – 5037 (2016).
[35] Dong Y., Zhang Z., Hong W.C., A Hybrid Seasonal Mechanism with a Chaotic Cuckoo Search Algorithm with a Support Vector Regression Model for Electric Load Forecasting, *Energies*, **11(4):** 1009 (2018).
[36] Buzo A., Gray A., Gray R., Markel J., Speech Coding Based Upon Vector Quantization, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **28(5):** 562-574 (1980).
[37] Krishnamurthy A.K., Ahalt S.C., Melton D.E., Chen P., Neural Networks for Vector Quantization of Speech and Images, *IEEE Journal on Selected Areas in Communications*, **8(4):** 1449-1457 (1990).
[38] Tan C., Yu D., Gao X., Song W., A Mechanism based Data-Driven Model for Prediction of Hydrate Formation. *Proceedings of the 2nd International Conference on Industrial Control Network and System Engineering Research*, pp 84-93 (2020).
[39] Yu Z., Tian H., Application of Machine Learning in Predicting Formation Condition of Multi-Gas Hydrate, *Energies*, **15:** 1-18 (2022).
[40] Monday C.U., Odutola T.O., "Application of Machine Learning in Gas-Hydrate Formation and Trendline Prediction", *SPE Symposium: Artificial Intelligence - Towards a Resilient and Efficient Energy Industry*, *OnePetro*, (2021).
[41] Sadi M., Fakharian H., Ganji H., Kakavand M., Evolving Artificial Intelligence Techniques to Model the Hydrate-based Desalination Process of Produced Water, *J. Water Reuse and Desal*., **9(4):** 372-384 (2019).

[42] Hosseini M., Leonenko Y., A Reliable Model to Predict the Methane-Hydrate Equilibrium: An Updated Database and Machine Learning Approach, *Renewable and Sustainable Energy Reviews*, **173:** 113103 (2023).

[43] Xu P., Xiao Z., Zhang Z., Huffman M., Wang Q., Prediction of Methane Hydrate Formation Conditions in Salt Water using Machine Learning Algorithms, *Comp. Chem. Eng.*, **151:** 107358 (2021).

[44] Ibrahim A.A., Lemma T.A., Kean M.L., Zewge M.G., Prediction of Gas Hydrate Formation using Radial Basis Function Network and Support Vector Machines, *Applied Mechanics and Materials*, **819:** 569-574 (2016).

[45] Kumari A., Madhaw M., Pendyala V.S., Prediction of Formation Conditions of Gas Hydrates using Machine Learning and Genetic Programming, *Machine Learning for Societal Improvement, Modernization, and Progress*, 200-224 (2022).

[46] Ghorbani M., Zargar G., Jazayeri-Rad H., Prediction of Asphaltene Precipitation using Support Vector Regression Tuned with Genetic Algorithms, *Petroleum*, **2(3):** 301-306 (2016).

[47] Zheng H., Zhou Y., A Novel Cuckoo Search Optimization Algorithm Base on Gauss Distribution. *Journal of Computational Information Systems*, **8(10):** 4193–4200 (2012).

[48] Horng M.H., Vector Quantization using the Firefly Algorithm for Image Compression, *Expert Systems with Applications*, **39(1):** 1078-1091 (2012).