Farid Amirkhani[1,*]
Amir Dashti[1]
Hossein Abedsoltan[2]
Amir H. Mohammadi[3,4,*]

Estimation of CO_2 Absorption by a Hybrid Aqueous Solution of Amino Acid Salt with Amine

Four developed machine learning algorithms are proposed to prognosticate the CO_2 solubility in amino acid salt solutions, blended with amine solutions as additives, in broad ranges of temperature and pressure. From literature 375 experimental data points for CO_2 solubility were collected. The results from the applied algorithms indicated that the CO_2 solubility is estimated acceptably close to the experimental values. In the best case, the developed network estimates CO_2 solubility in the stated solutions with an average relative deviation of 6.53 % and a correlation coefficient of 0.9892.

Keywords: Amine, Amino acid salt, Artificial intelligence, CO_2 capture, Global warming

Received: October 02, 2022; *revised:* July 15, 2023; *accepted:* September 05, 2023

DOI: 10.1002/ceat.202200469

Supporting Information available online

1 Introduction

The detrimental consequences of climate change have increased concerns regarding the conditions of both human and nature lives. Mainly, the climate change has been attributed to the energy generation of fossil fuels and the release of greenhouse gases. Carbon dioxide (CO_2) is one of the major greenhouse gases that affects the climate change genuinely [1]. Various approaches have been proposed and applied to sufficiently eliminate acid gases. Absorption [2], adsorption [3], cryogenics [4], and membrane [5, 6] based separation methods are among the methods of reducing CO_2 emissions which have been investigated earlier. Chemical solvents can capture CO_2 and decrease its concentrations considerably in different gas streams. For physical solvents, the absorption capacity is independent of the stoichiometric calculations – a positive characteristic in the absorption process [7]. If a chemical solvent and a physical solvent get mixed, then the formed mixture may possess a combination of advantages. An optimized mixture can be made to remove acid gases completely in the specified partial pressure ranges [8].

There are various processes to separate CO_2 from exhaust gases. Chemical absorption is a prominent approach for CO_2 separation, i.e., appropriate, versatile, and relatively low-priced. In this absorption process, alkanolamines are the conventional solvents to absorb the gas. However, amino acid salt (AAS) solutions have recently shown that they could be potential alternatives as chemical absorbents due to remarkable chemical properties they possess such as high chemical reactivity, high stability against oxidation, and low viscosities [9]. In a chemical absorption process, the challenge is focused on the experimental methods for gas separation, which could be risky, tedious, and hard to carry out. Thus, it is necessary to apply models that are able to prognosticate the phase behavior of these mixtures in absorbing the gas. This can ensure the efficacy of process design problems [10].

The literature has already addressed the utilization of empirical correlations [11], equations-of-state (EOSs) [12], and molecular descriptors [13] for the investigation and modeling of gas solubility. These methods typically have limitations, being applicable only to specific systems and compositions within defined temperature and pressure ranges. EOSs have gained significant attention in predicting the solubility of nitrous oxide (N_2O) in ionic liquids (ILs). Shiflett and Yokozeki [14] investigated the Redlich-Kwong (RK) EOS to correlate N_2O solubility data in ILs composed of imidazolium and ammonium families. Revelli et al. [15] assessed the performance of the Peng-Robinson (PR) EOS in estimating the solubility of N_2O in specific imidazolium-based ILs.

Additionally, Pereira et al. [16] employed the statistical associating fluid theory (SAFT), an advanced theory, to model the

[1]Farid Amirkhani https://orcid.org/0000-0002-0950-4454 (faridamirkhani.uk@gmail.com),
Amir Dashti https://orcid.org/0000-0002-3937-7353
Department of Chemical Engineering, Faculty of Engineering, University of Kashan, Kashan, Iran.

[2]Dr. Hossein Abedsoltan https://orcid.org/0000-0001-5754-4808
Department of Chemical Engineering, University of Toledo, Toledo, OH, USA.

[3]Prof. Amir H. Mohammadi (amir_h_mohammadi@yahoo.com)
Institut de Recherche en Génie Chimique et Pétrolier (IRGCP), Paris Cedex, France.

[4]Prof. Amir H. Mohammadi
Discipline of Chemical Engineering, School of Engineering, University of KwaZulu-Natal, Howard College Campus, King George V Avenue, Durban, 4041, South Africa.

solubility of N_2O in certain imidazolium ILs. These EOSs have shown acceptable accuracy in their respective applications. However, in general, EOSs have limitations as they require specific interaction parameters, making them challenging to develop predictive frameworks for estimating the solubility of gas across a wide range of solvents like ILs.

To overcome these limitations, artificial intelligence (AI) and machine learning (ML) approaches have emerged as effective and robust means of calculating gas solubility in solvents. These approaches involve data from a broad range of materials under diverse thermodynamic conditions, particularly when a comprehensive database is employed for model development [17].

Recently, a few intelligent algorithms have been applied to study the solubility trends in various gas-liquid solutions. An artificial neural network (ANN) modeling method was proposed, based on two optimization methods, i.e., particle swarm optimization (PSO) and back propagation (BP), to model hydrogen sulfide (H_2S) solubility in ILs [18]. In another study, H_2S solubility in ILs was modeled by the optimized algorithm of genetic algorithm (GA) along with the least squared support vector machine (LSSVM) [19].

In a specific study, an adaptive neuro-fuzzy inference system (ANFIS) was employed to develop a temperature-dependent intelligent model for accurately correlating the CO_2 loading capacity of AAS solutions. The results demonstrated satisfactory performance, indicating that the developed ANFIS structure can be utilized for precise estimation of CO_2 loading capacities in AAS solutions [20]. Sedghamiz et al. [21] investigated the estimation of acid gases' solubility in various ILs using three different approaches: ANN, PR EOS, and empirical correlation. The ANN model demonstrated high accuracy in estimating the solubility of acid gases in ILs compared to the other two methods.

In this study, the following input parameters were selected to build our proposed AI models: temperature, solution density, apparent molecular weight, CO_2 partial pressure, and solute concentration. Here, the apparent molecular weight and the solution density were considered as the parameters to define the solution type. New data were extracted from literature to build the AI models for different chemical solvents. The results indicate that the developed algorithms can prognosticate CO_2 solubility in amino acid salt/amine solution.

2 Methodology

2.1 Multilayer Perceptron Neural Network (MLP-NN)

The MLP-NN algorithm has a similar configuration like the RBF-NN model. The only difference lies in the internal calculation trend. The hidden layer of MLP-NNs can be more than one and it is feasible to specify a few values of hidden layers for these networks. Every neuron in the hidden layer admits the input vector. Then, it multiplies the input vector with a weight vector while summing the result simultaneously with a bias factor [22].

To enhance the efficiency of the improved MLP-NN model, one should optimize the weight terms of the neurons that are positioned in the hidden layer. The weight terms were adjusted by applying a back propagation (BP) model. Also, MSE was defined as a cost function between the model estimations and the experimental data [23]. The weight terms will have optimum values when the MSE value is minimized. This is the process where the network will be trained by adjusting the weight terms and biases.

2.2 Adaptive Network-Based Fuzzy Inference System (ANFIS)

ANFIS was first proposed by Jang and Mizutani by combining ANN and fuzzy inference system (FIS) to overcome the drawbacks of the two models [24]. ANFIS is categorized as rule-based adaptive models, developing the rules while its structure is going through the learning procedure [25].

An ANFIS algorithm that has two input variables (x_1 and x_2) and one output variable (f) is assumed. In ANFIS system configuration, the relation between inputs and output is defined by applying of if-then fuzzy rules, i.e., Takagi and Sugeno's type. In this case, the algorithm is described as follows [26]:

Rule **1**: IF (x_1 is A_1) and (x_2 is B_1) then $f_1 = p_1 x_1 + q_1 x_2 + r_1$

Rule **2**: IF (x_1 is A_2) and (x_2 is B_2) then $f_2 = p_2 x_1 + q_2 x_2 + r_2$

ANFIS has five layers in its configuration [27]. More details regarding the ANFIS method can be found elsewhere [28].

2.3 Particle Swarm Optimization Algorithm (PSO)

PSO is an algorithm that randomly performs to optimize, considering the population. Its concept was originated from animal social behavior such as bird flocking [29, 30]. PSO resembles in some of its attributes to evolutionary calculation methods such as genetic algorithm (GA). The PSO algorithm mechanism works with a population of stochastic solutions, known as particles, and features the optimum solution by updating generations. However, PSO has no evolution operators such as mutation and cross-over, which is available in GA [31]. In PSO, particles (our solutions) are in motion on the problem space by tracking the present optimum particles. When there is a bundle of particles, it is known as a swarm. The phrases "particle" and "swarm" are equal to "individual" and "population", applied in evolutionary algorithms such as GAs [32].

The PSO algorithm was developed on the basis of a social psychological metaphor. Hence, the particles in a swarm will not only interact with each other both in the swarm, but they will also interact with their whole community. This means each particle is affected by its neighboring particles [33]. This is an external function that supports a particle with its neighbor of a given type. Mainly, there are three forms of a neighborhood: physical, social, and queen. The neighborhood types are explained in details in the literature [33].

The particle i position at iteration t can be defined as $x_{id}^t = \{x_{i1}^t, \ldots, x_{iD}^t\}$. Similarly, the velocity vector for this particle with the stated iteration can be specified by

$v_{id}^t = \{v_{i1}^t, \ldots, v_{iD}^t\}$. Let us assume P_{id} as the previous optima (position), obtained by the i-th particle. P_{gd} indicates the global optima, found up to now. The i-th particle velocity is updated by Eq. (1) [34]:

$$v_{id}^{t+1} = wv_{id}^t + c_1 n_1 \left(P_{id}^t - x_{id}^t\right) + c_2 n_2 \left(P_{gd}^t - x_{id}^t\right), \quad d = 1, 2, \ldots, D \tag{1}$$

in which n_1 and n_2 are stochastic numbers in a range of 0 to 1, c_1 and c_2 are personal and social learning rates, t is the time, and w is the inertia weight [34].

It can be concluded from Eq. (1) that the velocity has three components: cognitive, social, and inertia [35]. The cognitive component, which involves c_1, pulls back the particles to already-determined best positions and specifies a velocity in this direction. The component that is involving c_2 is the social component, quantifying the efficiency of a particle by comparing it with its neighbors. It keeps the particles in a manner that they can move together through the space. The inertia component $\left(wv_{id}^t(t)\right)$ performs as a memory of the previous flight direction, inducing the particle to move in a route where it was going at iteration t. Therefore, particles regulate their path by considering these three velocity components in the search space. Then, the new particle position is updated by adding the new velocity to the former position, as indicated in Eq. (2) [34]:

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1}, \quad d = 1, 2, \ldots, D \tag{2}$$

Eventually, both the position and the velocity are iteratively updated for every particle.

2.4 Least Squares Support Vector Machine (LSSVM)

Support vector machines (SVMs) have provided solutions for a range of complicated problems in areas such as pattern classification and nonlinear function estimation [36, 37]. Literature has addressed how the SVM performs. Also, the equations that are applied for nonlinear functions to do approximations have been explained. The LSSVM is a modified form of the SVM. It was developed to reduce the SVM complexity and to enhance its convergence. Opposed to the SVM training approaches, equality limitations are preferred to be employed in the LSSVM approaches [38]. This reformulation facilitates the training procedure by presenting a set of equations that are linear and are able to be solved iteratively [39]. Thus, the LSSVM is applicable in big-scale problems where training has to be carried out by considering a large dataset. Here, the important factors are time and accuracy [37].

We applied an LSSVM model, which was developed and reported in the literature, to train the CO_2 solubility data [40]. We also improved our model efficiency during the training procedure by applying a coupled simulated annealing (CSA) model. The CSA algorithm optimized two parameters in the model, namely, the regularization constant (γ) and Kernal parameter (σ^2).

3 Data Collection

A proper amount of experimental data on CO_2 solubility in amino-acid salt solutions blended with amine solutions has to be collected to establish the models. In this study, 375 data points were gathered from the literature [41]. Previously, reported data was classified into three parts: trained, validated, and tested [42]. The trained data had 263 data (70 % of the collected reported data points from the literature). The tested data encompasses 112 points. The same solutions for each part that was reported in the literature were applied, but our operational conditions are completely different. The tested data are gathered to evaluate the efficiency of the suggested network. Details of employed data are illustrated in Tab. 1.

It is important to define a basis to specify the solution type in input features of the suggested models. CO_2 solubility in solutions is controlled by the following factors: solution type, overall solute mass concentration, CO_2 partial pressure in the system, and temperature. The solution property is its apparent molecular weight. These properties change by replacing the solution type and the solution concentration. The apparent molecular weight was selected because it is explicit and available in calculations. Solution density was chosen because it can be measured simply. Therefore, the CO_2 solubility capacity (α) can change with respect to the overall solute mass concentra-

Table 1. Details of employed data in this study.

No.	Solution	Temperature [K]	CO_2 partial pressure [kPa]	Overall conc. [wt %]	Apparent molecular weight [g mol^{-1}]	No. data	Ref.
1	Potassium glycinate/AMP	293.15–323.15	8.9000–1397.4	1.00–10.0	18.85–25.37	69	[43]
2	Potassium alaninate/AMP	293.15–293.15	427.30–1421.6	3.00–4.00	20.53–21.24	6	[43, 44]
3	Potassium prolinate/AMP	293.15–323.15	16.300–2008.1	1.00–10.0	20.79–21.50	92	[43, 44]
4	Potassium lysinate/AMP	293.15–293.15	403.90–1413.8	3.00–4.00	21.10–21.81	6	[43, 44]
5	Potassium taurate/AMP	293.15–293.15	418.40–1416.5	3.00–4.00	20.89–21.60	6	[43, 44]
6	Potassium prolinate/PZ	293.15–323.15	10.100–1852.1	1.00–10.0	19.03–25.50	110	[45]
7	Potassium glycinate/PZ	293.15–323.15	15.900–1949.6	1.00–10.0	18.83–25.10	86	[41]
Total	–	–	–	–	–	375	–

tion, solution temperature, and CO_2 pressure in the solution, and the apparent molecular weight.

These features were chosen as the inputs, as indicated in Eq. (3):

$$\alpha = F\left(T, M_{app}, X, P_{CO_2}\right) \tag{3}$$

The solution apparent molecular weight (M_{app}) is obtained by Eq. (4):

$$M_{app} = \sum_{i=1}^{NC} x_i MW_i \tag{4}$$

MW_i is the molecular weight of component i.

4 Results and Discussion

4.1 Model Development

An MLP network can have just one hidden layer. ANNs are normally able to model a nonlinear function with just one hidden layer sufficiently. Hence, it was supposed that our MLP structure has one hidden layer to decrease the required time for computations. The proposed MLP-ANN algorithm has one neuron in the input layer and nine neurons in the output layer. This means that there are one input and three outputs. The algorithm validity was assessed by rising the number of neurons from 1 to 50, which are positioned in the determined hidden layer. It was observed when the MLP consists of nine neurons in the hidden layer, the test and total data will have the lowest value of MSE and % average absolute relative deviation (%AARD), indicating the highest efficiency.

Two approaches were combined and applied to train the fuzzy inference system (FIS) initially. The first approach, coming from the Hybrid-ANFIS function in MATLAB, applies a hybrid approach to adjust the beginning FIS. ANFIS based on FCM included 14 rules; the model properties are indicated in Tab. S1 (Supporting Information). The model with the minimum test error was selected as the main one. A hybrid learning method was used for parameter prediction. The second method was carried out while it was considered that the FIS learning procedure is an optimization problem. The parameters of the MFs were used as the adjusting factors, optimized by applying the PSO algorithm [32]. Tab. S2 (Supplementary Information) indicates PSO-ANFIS model properties used to estimate CO_2 solubility in the solutions. In addition, a schematic exhibition of the MFs of the hybrid and PSO-ANFIS structures are depicted in Figs. S1 and S2 (Supplementary Information), respectively.

The two important parameters of γ and σ^2 are needed to be computed to design the LSSVM model [46]. They control the accuracy and the globalization capability of our suggested LSSVM model. These two factors were optimized by the CSA concept [41, 47–49]. The obtained values of γ and σ^2 were 40630220244.2 and 432.9714, respectively.

4.2 Accuracy of the Proposed Model and Validation

Some statistical parameters and graphical approaches were applied to check the precision of the proposed model. Fig. S3 (Supplementary Information) shows the trend plots for three phases of trained and tested data points of the developed model. This figure clarifies the precision and the enhancement of the improved model. It also exhibits the anticipations of the executed model track and the trend of reported data with a satisfactory accuracy. This is concluded from the high overlay between the reported data and the anticipated ones by the models.

Fig. 1 illustrates a regression plot, comparing prognosticated and actual data. By looking at the vertical axis, one can observe that the prognosticated values and the actual data are located on the horizontal axis. Also, the points are mostly situated in a concentrated form around the unit slope line (the 45° line), indicating the precision and the enhancement of the proposed model. Hence, there is a satisfactory consistency between the reported and the anticipated data.

Fig. S4 (Supplementary Information) represents the relative deviations of the model results versus the reported data amounts. It also affirms the accuracy of the proposed model because the relative error data points are located mostly adjacent the zero-error line. In addition, the maximum relative error of the proposed model is below 6 %. Most of the data points are located in an area where the relative errors are constrained to ± 6 %.

The applied statistical factors in this research are: correlation factor (R^2), average absolute relative deviation (AARD), standard deviation (STD), and mean squared error. Eqs. (5)–(8) present the formula of these parameters.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left[x_i^{predicted} - x_i^{experimental}\right]^2}{\sum_{i=1}^{n}\left[x_i^{predicted} - x^m\right]^2}, \quad x_m = \frac{\sum_{i=1}^{n} x_i^{experimental}}{n} \tag{5}$$

$$\%\ \mathrm{AARD} = \frac{100}{n}\sum_{i=1}^{n}\frac{\left|x_i^{predicted} - x_i^{experimental}\right|}{x_i^{experimental}} \tag{6}$$

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(x_i^{experimental} - x_i^{predicted}\right)^2 \tag{7}$$

$$\mathrm{STD} = \sqrt{\sum_{i=1}^{n}\left(\frac{\left(x_i^{predicted} - x_m\right)^2}{n}\right)} \tag{8}$$

Tab. 2 shows the obtained amounts of these factors that were applied to estimate the reported data. From Tab. 2, the enhanced model expresses high values of R^2 and low values of MSE, AARD, and STD for the sets of trained, tested, and overall data, thus confirming that the model replicates the reported

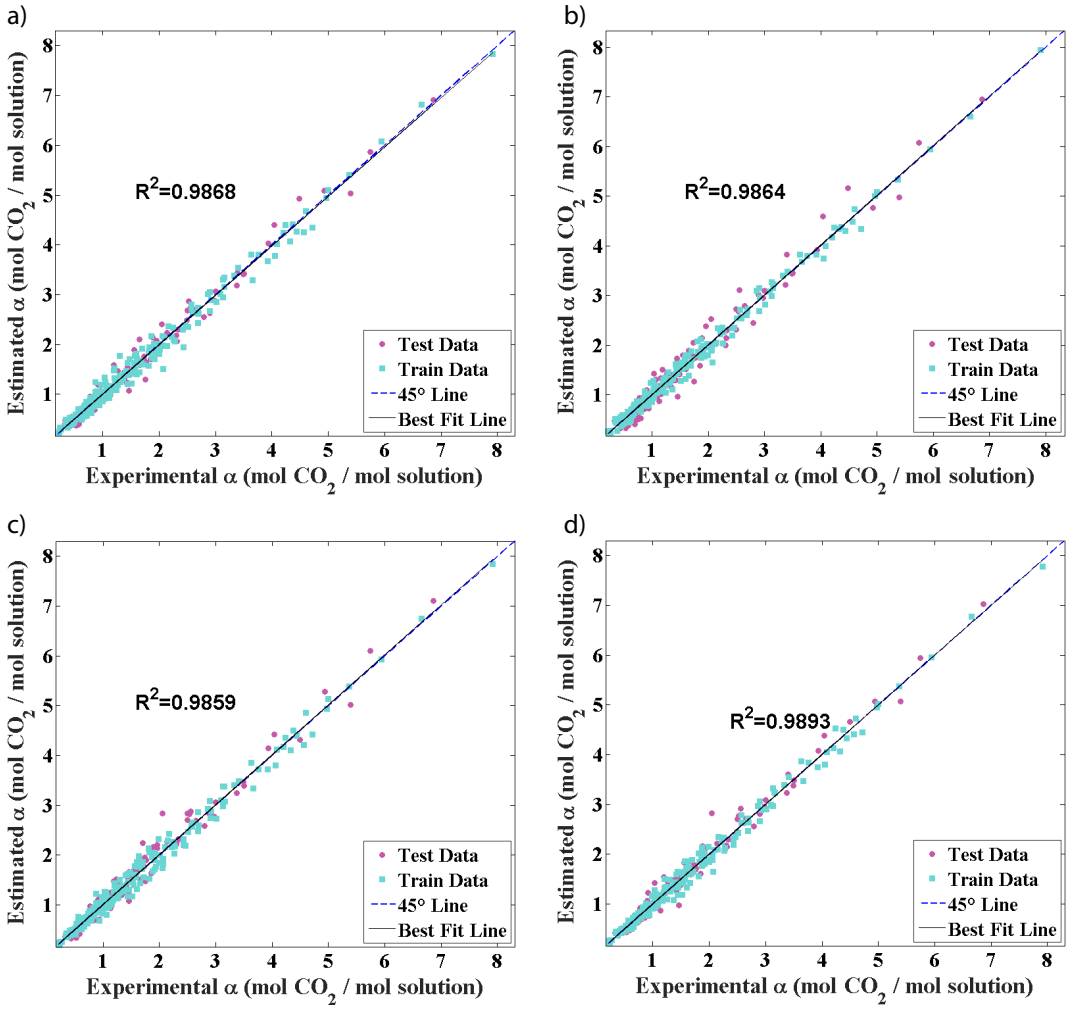

Figure 1. Regression plot for the solubility of CO_2 prognostication by (a) MLP-ANN, (b) Hybrid-ANFIS, (c) PSO-ANFIS, (d) CSA-LSSVM.

data with a satisfactory accuracy. The values of R^2 and %AARD for the tested dataset of LSSVM are 0.9848 and 7.9739, respectively. It can also be concluded from this table that the LSSVM model indicates high values of R^2 and low values of RMSE, AARD, and STD for the tested data, expressing that the model reproduces the experimental data with satisfying accuracy.

Tab. 3 provides the %AARD values obtained with different models for various blends. Consistently, Fig. 2 evaluates the stated parameters and confirms the high efficiency of models to estimate the CO_2 solubility. The LSSVM model has the best performance among these models due to its highest accuracy.

Figs. 3 and 4 illustrate that the CO_2 solubility rises with the increase of the CO_2 pressure in the solution at a specified temperature and concentration. An raise in pressure causes the reproduction of the collisions among the gas molecules on the

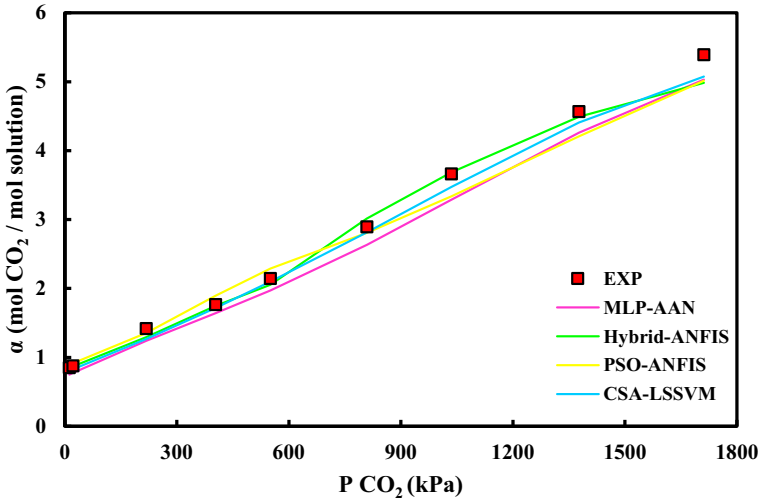

Figure 2. Solubility of CO_2 in a solution of 0.5 wt % potassium prolinate blended with 0.5 wt % PZ at 293.15 K.

Table 2. Accuracy of different models.

Models	Test	Train	Data set
Parameters			
MLP-ANN			
R^2	0.9882	0.9842	0.9868
MSE	0.1650	0.0241	0.0188
STD	1.1711	1.2296	1.1894
% AARD	7.80	8.75	8.08
Hybrid-ANFIS			
R^2	0.9926	0.9746	0.9864
MSE	0.0103	0.0418	0.0197
STD	1.1769	1.2596	1.2027
% AARD	6.48	12.20	8.19
PSO-ANFIS			
R^2	0.9886	0.9818	0.9859
MSE	0.0160	0.0307	0.0204
STD	1.1772	1.2546	1.2015
% AARD	7.07	9.11	7.68
CSA-LSSVM			
R^2	0.9917	0.9848	0.9893
MSE	0.0116	0.0244	0.0154
STD	1.1760	1.2398	1.1960
% AARD	5.92	7.97	6.53

solution surface, which eventually facilitate the CO_2 molecules' diffusion into the solution in a higher extent [50]. Fig. 3 exhibits the temperature effect on the solubility. Absorption is an exothermic phenomenon, and by considering the Le Chaterlier's principle, a temperature increase will cause in reduction of the CO_2 absorption [51]. Hence, the temperature has a converse effect on the CO_2 solubility. Fig. 4 demonstrates the impact of the added piperazine (PZ) on the CO_2 solubility. Its addition into the aqueous solution of potassium prolinate will result in an increase of the CO_2 solubility [45].

4.3 Sensitivity Analysis

Here, a thorough analysis was performed to study the connection between the output (impure CO_2 solubility) and the input (independent) variables of the model [52]. To quantify this dependency, the Pearson correlation coefficient (r) was applied, which is calculated by Eq. (9) [53]:

$$r(V_j,\ \text{loading}) = \frac{\sum_{i=1}^{n}\left(V_{j,i}-\overline{V_j}\right)(L_i-\bar{L})}{\sqrt{\sum_{i=1}^{n}\left(V_{j,i}-\overline{V_j}\right)^2\sum_{i=1}^{n}(L_i-\bar{L})^2}} \quad (9)$$

L_i and $\bar{L}$ are the i-th and the mean value of CO_2 solubility, and $V_{j,i}$ and $\overline{V_j}$ are the i-th and mean value of the j-th input variable, respectively. The higher absolute r value, based on the relationship between the dependent (output) and one of the independent parameters shows the higher significance and effect of that variable on CO_2 solubility. Moreover, r indicates change direction that an input variable may have on the output value.

Fig. 5 illustrates the results that confirm clearly the negative effect of T on CO_2 solubility, i.e., the reduction trend of CO_2 solubility with temperature increase. Among the parameters, pressure is the most effective parameter on the solubility, as conceived from the reported data in literature. Meanwhile, the least effective parameter on solubility is temperature. The overall mass concentration of the solute and the apparent molecular weight proved to have a reduction effect on the solubility.

5 Conclusions

The capability of some AI models to prognosticate the CO_2 solubility in amino-acid salt solutions blended with amine solutions was investigated. The enhanced AI models can be applied in a broad range of performing situations and for numerous kinds of amino-acid salts. An overall of 375 data points for CO_2 solubility was applied to enhance and optimize the AI models. To the best of our knowledge, this is the first time that MLP-NN, Hybrid-ANFIS, CSA-LSSVM, and PSO-ANFIS were enhanced on the basis of the CO_2 solubility data values in the specified conditions. The temperature range was between 293.15 and 323.15 K, the CO_2 partial pressure range between 8.9 and 1949.6 kPa, and the overall mass concentration ranged

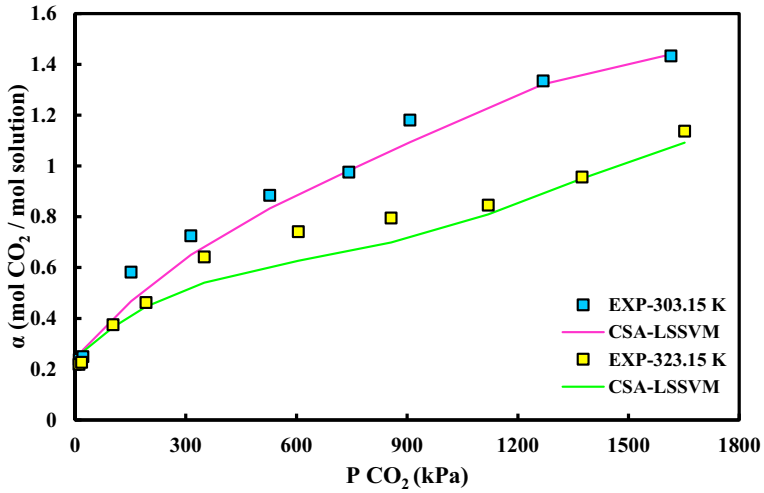

Figure 3. Solubility of CO_2 in a solution of 1.0 wt % potassium prolinate blended with 3.0 wt % PZ at 303.15 K and 323.15 K.

Table 3. AARD% of each blended solution of the developed model.

Solution [wt %]	% AARD			
	MLP-ANN	Hybrid-ANFIS	PSO-ANFIS	CSA-LSSVM
(0.5 %) Potassium glycinate / (0.5 %) AMP	14.64	10.34	11.21	10.88
(1 %) Potassium glycinate / (3 %) AMP	10.59	7.00	8.39	7.22
(1 %) Potassium glycinate / (9 %) AMP	7.65	11.19	5.32	5.78
(1 %) Potassium alaninate / (2 %) AMP	13.21	8.00	22.43	20.85
(1 %) Potassium alaninate / (3 %) AMP	4.40	0.67	8.62	7.15
(0.5 %) Potassium prolinate / (0.5 %) AMP	4.71	5.74	7.44	5.10
(1 %) Potassium prolinate / (2 %) AMP	16.68	3.58	13.24	14.10
(1 %) Potassium prolinate / (3 %) AMP	9.04	9.54	8.89	8.22
(1 %) Potassium prolinate / (9 %) AMP	9.61	14.11	8.51	5.27
(1 %) Potassium lysinate / (2 %) AMP	8.79	17.11	11.00	16.29
(1 %) Potassium lysinate / (3 %) AMP	14.04	20.65	8.46	8.00
(1 %) Potassium taurate / (2 %) AMP	19.52	11.72	19.03	23.41
(1 %) Potassium taurate / (3 %) AMP	14.58	9.41	13.18	9.642
(0.5 %) Potassium prolinate / (0.5 %) PZ	6.81	4.77	4.80	4.55
(1 %) Potassium prolinate / (3 %) PZ	8.48	8.43	7.16	6.07
(1 %) Potassium prolinate / (9 %) PZ	8.05	11.70	7.59	6.67
(0.5 %) Potassium glycinate / (0.5 %) PZ	5.47	6.06	6.58	5.90
(1 %) Potassium glycinate / (3 %) PZ	4.52	5.20	7.97	4.04
(1 %) Potassium glycinate / (9 %) PZ	5.94	5.93	4.71	3.76
Total	8.08	8.19	7.68	6.53

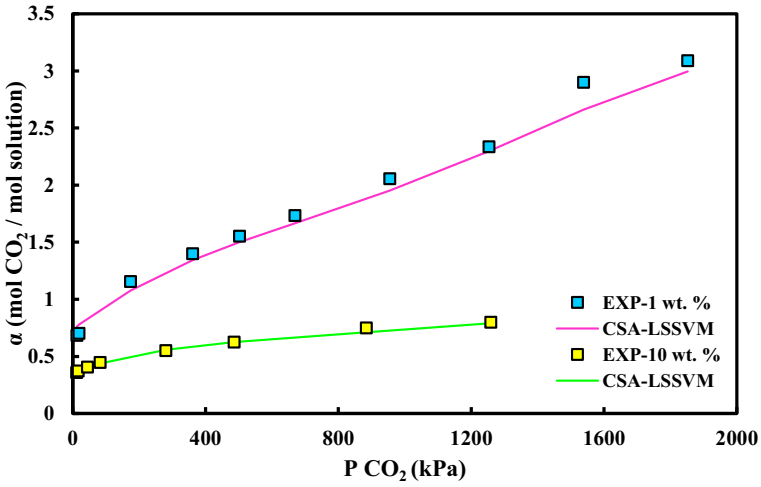

Figure 4. Solubility of CO_2 in a solution of 0.5 wt % potassium prolinate blended with 0.5 wt % PZ and 1 wt % potassium prolinate blended with 9 wt % PZ at 323.15 K.

from 1.0 to 10.0 wt %. Moreover, the enhanced AI models in this work can be applied to predict the CO_2 solubility in numerous conventional and new types of amino-acid salt solutions, blended with amines.

Supporting Information

Supporting Information for this article can be found under DOI: https://doi.org/10.1002/ceat.202200469.

The authors have declared no conflict of interest.

Symbols used

$A_{1,2}$	fuzzy set of input variables x_1
$B_{1,2}$	fuzzy set of input variables x_2
$c_{1,2}$	acceleration constants (PSO)
F	function
$\bar{L}$	mean value of CO_2 solubility

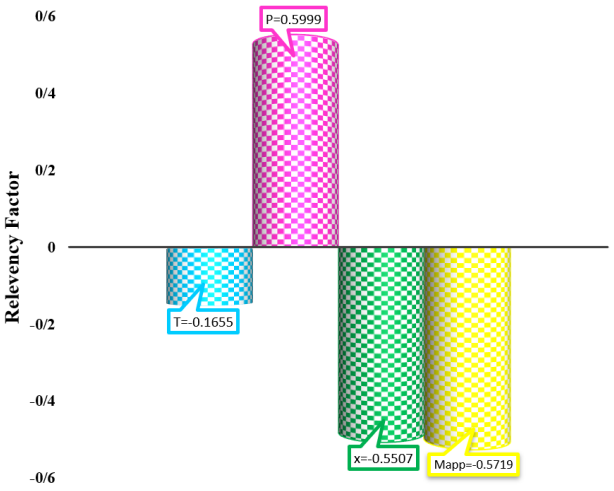

Figure 5. Relevancy factor for all inputs.

L_i	i-th value of CO_2 solubility
M_{app}	apparent molecular weight of solution
M_{wi}	molecular weight of component i
n	number of data
$n_{1,2}$	positive vectors (PSO)
NC	number of component
P	pressure
$p_{1,2}$	adjustable parameters (ANFIS)
P^t_{gd}	global best solution (PSO)
P^t_{id}	local best solution (PSO)
$q_{1,2}$	adjustable parameters (ANFIS)
r	correlation coefficient
$r_{1,2}$	adjustable parameters (ANFIS)
R^2	correlation factor
T	temperature
$V_{j,i}$	i-th value of input variable
$\overline{V_j}$	mean value of input variable
v^t_{id}	velocity of the particle (PSO)
w	weight for expert (PSO)
X	overall solute mass concentration
$x_{1,2}$	input feature values (ANFIS)
x_i	mol fraction of component i, also property
x^t_{id}	location of the particle (PSO)

Greek letters

α	solubility capacity
γ	regularization constant (LSSVM)
σ^2	Kernal parameter (LSSVM)

Abbreviations

AARD	average absolute relative deviation
AAS	amino acid salt
AI	artificial intelligence
AMP	2-amino-2-methyl-1-propanol
ANFIS	adaptive neuro-fuzzy inference system
ANN	artificial neural network
BP	back propagation
CLM	couple local minimizer
CSA	coupled simulated annealing
EOS	equation-of-state
EXP	experimental
FCM	fuzzy C-means
FIS	fuzzy inference system
GA	genetic algorithm
GCM	group contribution method
GEP	gene expression programming
IL	ionic liquid
LSSVM	least squared support vector machine
MF	membership function
ML	machine learning
MLP	multilayer perception
MSE	mean squared error
MW	molecular weight
NN	neural network
PR	Peng-Robinson
PSO	particle swarm optimization
PZ	piperazine
RBF	radial basis function
RK	Redlich-Kwong
RMSE	root mean square error
SA	simulated annealing
STD	standard deviation
SV	support vector
SVM	support vector machine

References

[1] M. Mosadegh, F. Amirkhani, H. R. Harami, M. Asghari, M. J. Parnian, *Sep. Purif. Technol.* **2020**, *247*, 116981. DOI: https://doi.org/10.1016/j.seppur.2020.116981
[2] Y. Shen, C. Jiang, S. Zhang, J. Chen, L. Wang, J. Chen, *Appl. Energy* **2018**, *230*, 726–733. DOI: https://doi.org/10.1016/j.apenergy.2018.09.005
[3] L. Shao, Y. Li, J. Huang, Y. N. Liu, *Ind. Eng. Chem. Res.* **2018**, *57 (8)*, 2856–2865. DOI: https://doi.org/10.1021/acs.iecr.7b04533
[4] S. M. Safdarnejad, J. D. Hedengren, L. L. Baxter, *Appl. Energy* **2015**, *149*, 354–366. DOI: https://doi.org/10.1016/j.apenergy.2015.03.100
[5] F. Amirkhani, H. R. Harami, M. Asghari, *Polym. Test.* **2020**, *86*, 106464. DOI: https://doi.org/10.1016/j.polymertesting.2020.106464
[6] F. Amirkhani, M. Mosadegh, M. Asghari, M. J. Parnian, *Polym. Test.* **2020**, *82*, 106285. DOI: https://doi.org/10.1016/j.polymertesting.2019.106285
[7] A. L. Kohl, R. Nielsen, *Gas purif.*, Elsevier, **1997**.
[8] F.-Y. Jou, A. Mather, *Fluid Phase Equilib.* **2005**, *228*, 465–469. DOI: https://doi.org/10.1016/j.fluid.2004.10.004
[9] A. Dashti, F. Amirkhani, A.-S. Hamedi, A. H. Mohammadi, *ACS omega* **2021**, *6 (19)*, 12459–12469. DOI: https://doi.org/10.1021/acsomega.0c06158
[10] P. F. Arce, P. A. Robles, T. A. Graber, M. Aznar, *Fluid Phase Equilib.* **2010**, *295 (1)*, 9–16. DOI: https://doi.org/10.1016/j.fluid.2010.03.030
[11] M. Mesbah, E. Soroush, M. Momeni, S. Shahsavari, M. Mofidi, S. Soltanali, *Can. J. Chem. Eng.* **2020**, *98 (2)*, 441–452. DOI: https://doi.org/10.1002/cjce.23604

[12] H. Al-fnaish, L. Lue, *Fluid Phase Equilib.* **2017**, *450*, 30–41. DOI: https://doi.org/10.1016/j.fluid.2017.07.008

[13] X. Kang, J. Qian, J. Deng, U. Latif, Y. J. Zhao, *J. Mol. Liq.* **2018**, *265*, 756–764. DOI: https://doi.org/10.1016/j.molliq.2018.06.113

[14] M. B. Shiflett, A. Yokozeki, *Ind. Eng. Chem. Res.* **2005**, *44 (12)*, 4453–4464. DOI: https://doi.org/10.1021/ie058003d

[15] A. L. Revelli, F. Mutelet, J. N. Jaubert, *J. Phys. Chem. B* **2010**, *114 (24)*, 8199–8206. DOI: https://doi.org/10.1021/jp103734c

[16] L. M. Pereira, V. Martins, K. A. Kurnia, M. B. Oliveira, A. M. Dias, F. Llovell, L. F. Vega, P. J. Carvalho, J. A. Coutinho, *J. Supercrit. Fluids* **2016**, *110*, 56–64. DOI: https://doi.org/10.1016/j.supflu.2015.12.006

[17] M. N. Amar, M. A. Ghriga, A. Hemmati-Sarapardeh, *J. Taiwan Inst. Chem. Eng.* **2020**, *117*, 63–74. DOI: https://doi.org/10.1016/j.jtice.2020.11.029

[18] A. Shafiei, M. A. Ahmadi, S. H. Zaheri, A. Baghban, A. Amirfakhrian, R. Soleimani, *J. Supercrit. Fluids* **2014**, *95*, 525–534. DOI: https://doi.org/10.1016/j.supflu.2014.08.011

[19] M.-A. Ahmadi, B. Pouladi, Y. Javvi, S. Alfkhani, R. Soleimani, *J. Supercrit. Fluids* **2015**, *97*, 81–87. DOI: https://doi.org/10.1016/j.supflu.2014.11.009

[20] E. Soroush, M. Mesbah, N. Hajilary, M. Rezakazemi, *J. Environ. Chem. Eng.* **2019**, *7 (1)*, 102925. DOI: https://doi.org/10.1016/j.jece.2019.102925

[21] M. A. Sedghamiz, A. Rasoolzadeh, M. R. Rahimpour, *J. CO2 Util.* **2015**, *9*, 39–47. DOI: https://doi.org/10.1016/j.jcou.2014.12.003

[22] S. Haykin, *Neural networks*, Prentice Hall, New York **1994**.

[23] S. Sabzevari, M. Moosavi, *Fluid Phase Equilib.* **2014**, *361*, 135–142. DOI: https://doi.org/10.1016/j.fluid.2013.10.044

[24] J. S. R. Jang, E. Mizutani, *Levenberg-Marquardt method for ANFIS learning*, IEEE **1996**.

[25] E. Heidari, S. M. Ghoreishi, *J. Supercrit. Fluids* **2013**, *82*, 158–167. DOI: https://doi.org/10.1016/j.supflu.2013.07.006

[26] A. Dashti, M. Jokar, F. Amirkhani, A. H. Mohammadi, *J. Mol. Liq.* **2020**, *300*, 111797. DOI: https://doi.org/10.1016/j.molliq.2019.111797

[27] F. Amirkhani, A. Dashti, H. Abedsoltan, A. H. Mohammadi, A. G. Chofreh, F. A. Goni, J. J. Klemeš, *Fuel* **2022**, *323*, 124292. DOI: https://doi.org/10.1016/j.fuel.2022.124292

[28] J. S. Jang, *ANFIS: Adaptive-networkbased fuzzy inference system*, IEEE **1993**, *23 (3)*, 665–685. DOI: https://doi.org/10.1109/21.256541

[29] R. Eberhart, J. Kennedy, *A new optimizer using particle swarm theory*, IEEE **1995**, 39–43.

[30] B. K. Panigrahi, Y. Shi, M.-H. Lim, *Handbook of swarm intelligence: concepts, principles and applications*, Springer Science & Business Media, **2011**. DOI: https://doi.org/10.1007/978-3-642-17390-5

[31] L. Cervantes, O. Castillo, P. Melin, *Intelligent control of nonlinear dynamic plants using a hierarchical modular approach and type-2 fuzzy logic*, Springer, **2011**, Part II 10, 1–12. DOI: https://doi.org/10.1007/978-3-642-25330-0_1

[32] J. E. Onwunalu, L. J. Durlofsky, *Comput. Geosci.* **2010**, *14 (1)*, 183–198. DOI: https://doi.org/10.1007/s10596-009-9142-1

[33] A. Sharma, G. Onwubolu, in *Hybrid self-organizing modeling systems*, Springer, **2009**, *211*, 193–231. DOI: https://doi.org/10.1007/978-3-642-01530-4_5

[34] M. Y. Chen, *Inf. Sci.* **2013**, *220*, 180–195. DOI: https://doi.org/10.1016/j.ins.2011.09.013

[35] S. Yuhui, R. Eberhart, *A modified particle swarm optimizer*, IEEE **1998**, 69–73.

[36] N. Cristianini, J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*, Cambridge university press, **2000**. DOI: https://doi.org/10.1017/CBO9780511801389

[37] J. A. K. Suykens, J. Vandewalle, *Neural processing letters* **1999**, *9 (3)*, 293–300. DOI: https://doi.org/10.1023/A:1018628609742

[38] H. Wang, D. Hu, *Comparison of SVM and LS-SVM for regression*, IEEE **2005**, 279–283.

[39] S. Smola, Statistics, Computing, *Statistics and Computing* **2004**, *14 (3)*, 199–222. DOI: https://doi.org/10.1023/B:STCO.0000035301.49549.88

[40] ;A. Dashti, A. Bahrololoomi, F. Amirkhani, A. H. Mohammadi, *J. CO2 Util.* **2020**, *41*, 101256. DOI: https://doi.org/10.1016/j.jcou.2020.101256

[41] M. E. Hamzehie, H. Najibi, *J. CO2 Util.* **2016**, *16*, 64–77. DOI: https://doi.org/10.1016/j.jcou.2016.06.003

[42] M. Valipour, S. M. Mousavi, R. Valipour, E. Rezaei, *A new approach for environmental crises and its solutions by computer modeling.* **2013**.

[43] M. E. Hamzehie, H. Najibi, *RSC Adv.* **2016**, *6 (67)*, 62612–62623. DOI: https://doi.org/10.1039/C6RA09600J

[44] M. E. Hamzehie, H. Najibi, *J. Nat. Gas Sci. Eng.* **2016**, *34*, 356–365. DOI: https://doi.org/10.1016/j.jngse.2016.07.004

[45] M. E. Hamzehie, H. Najibi, *Thermochim. Acta* **2016**, *639*, 66–75. DOI: https://doi.org/10.1016/j.tca.2016.07.019

[46] A. Dashti, F. Amirkhani, M. Jokar, A. Mohammadi, K.-W. Chau, Technology, *Int. J. Environ. Sci. Technol.* **2021**, *18*, 1773–1784. DOI: https://doi.org/10.1007/s13762-020-02912-9

[47] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi, *Sci.* **1983**, *220 (4598)*, 671–680. DOI: https://doi.org/10.1126/science.220.4598.671

[48] F. Amirkhani, A. Dashti, H. Abedsoltan, A. H. Mohammadi, K.-W. Chau, *J. Taiwan Inst. Chem. Eng.* **2021**, *127*, 109–118. DOI: https://doi.org/10.1016/j.jtice.2021.07.032

[49] S. Xavier-de-Souza, J. A. Suykens, J. Vandewalle, D. Bollé, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, **2009**, *40 (2)*, 320–335. DOI: https://doi.org/10.1109/TSMCB.2009.2020435

[50] M. Wong, M. Bustam, A. Shariff, *Int. J. Greenhouse Gas Control* **2015**, *39*, 139–147. DOI: https://doi.org/10.1016/j.ijggc.2015.05.016

[51] H.-J. Song, S. Lee, S. Maken, J.-J. Park, J.-W. Park, *Fluid Phase Equilib.* **2006**, *246 (1–2)*, 1–5. DOI: https://doi.org/10.1016/j.fluid.2006.05.012

[52] A. Dashti, O. Mazaheri, F. Amirkhani, A. H. Mohammadi, *Energy* **2021**, *217*, 119292. DOI: https://doi.org/10.1016/j.energy.2020.119292

[53] G. Chen, K. Fu, Z. Liang, T. Sema, C. Li, P. Tontiwachwuthikul, R. Idem, *Fuel* **2014**, *126*, 202–212. DOI: https://doi.org/10.1016/j.fuel.2014.02.034

