Supplementary Materials

Predicting the Crystal Structure and Lattice Parameters of the Perovskite Materials via Different Machine Learning Models Based on Basic Atom Properties

1. Flowchart of the Study

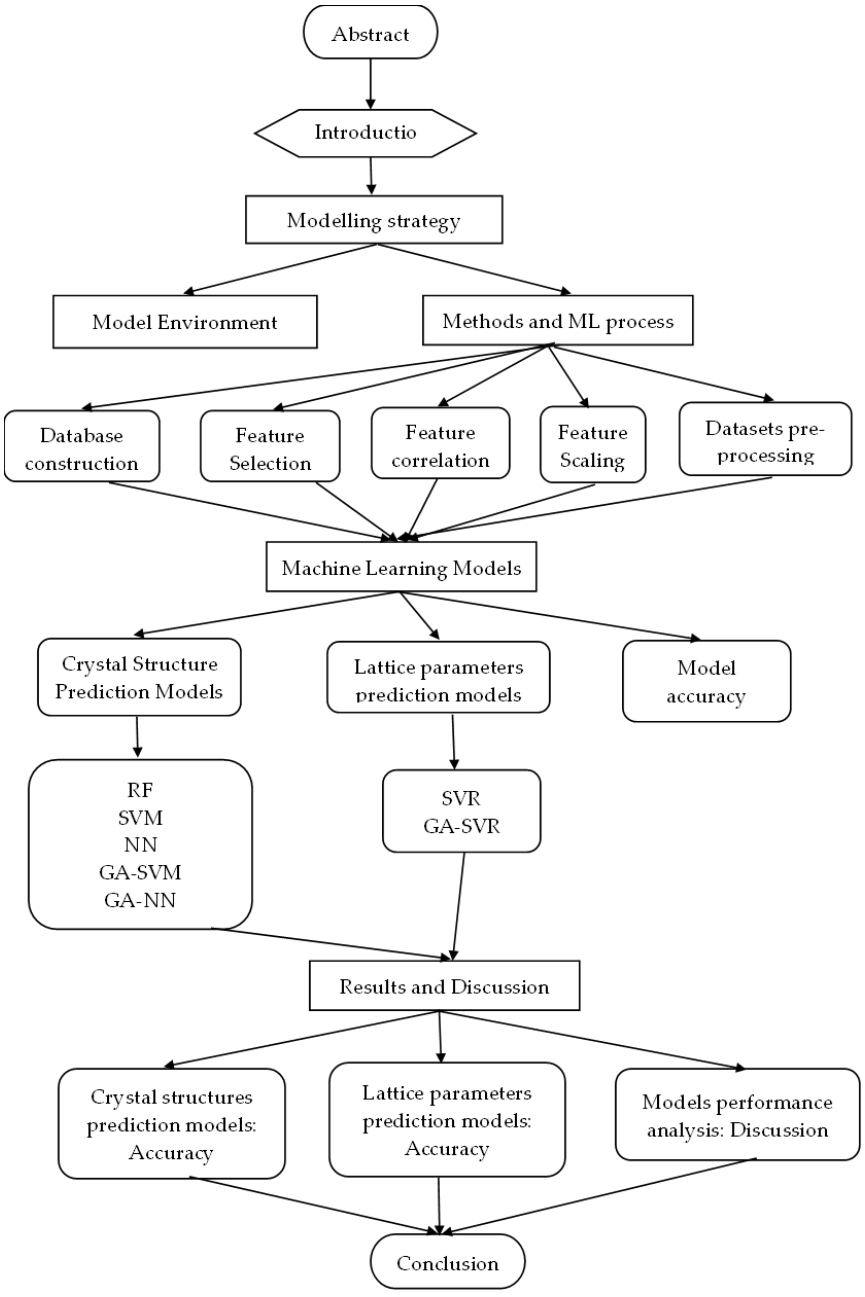

Figure S1. Flowchart of the whole project study.

2. Feature Selection

Feature selection (FS) can find out distinctive features/columns in a descending order of importance, which can be referred as feature importance (FI). Feature importance can be defined as a selection process that selects the most useful features for the predictive modelling problem. Redundant and less importance features can then be removed from the dataset, which avoids overfitting or underfitting. Also, it is reported by some researchers, removing those least important features can help the subsequent tasks to some extent (increase performance by 1–3% in accuracy).

Bootstrap with replacement method selects samples for the construction of the individual decision trees, while out-of-bag (OOB) samples (samples not included in the bootstrapped data) are used to assess the VI for each feature via Equation (S1) [1]:

$$\text{VI}(x\text{^}j) = 1/\text{ntree} \sum\_(i=1)\ \text{^}n\ 〚(\text{err}\ (\text{OOB})\tilde{}\_i\text{^}j - \text{err}\ 〚\text{OOB}〛\_i)〛 \tag{S1}$$

where VI(x^j) is the variable importance of variable x^j; n denotes the number of decision tree in RF; err 〚OOB〛_i is the error of tree i on OOB samples; and err (OOB)˜_i^j is the error of tree i on OOB samples with random permutation of variable x^j with random permutation of variable x^j.

3. Feature Correlation

Pearson correlation coefficient (R) have been calculated using the following formula (Kantardzic, 2011) [2]:

$$R = \frac{cov(X,Y)}{\sqrt{var(X)var(Y)}} \tag{S2}$$

Cov(X,Y) means the covariance of X and Y and var designates variance. In the procedure, R is taken an absolute value. $|R|$ equals 1 means the two features are either positive correlation or negative correlation, so one of them can be deleted from the dataset. Highly correlated features express a much higher value than other features of the model.

The Pearson's correlation coefficient measures the strength of the relationship between a pair of variables. This also helps to find out the most correlated features needs to be identified for an efficient modelling system.

4. Feature Scaling

Normalization typically means rescales the values into a range of [0,1] whereas standardization typically means rescales data to have a mean of 0 and a standard deviation of 1 (unit variance). Normalization can be expressed by the Equation (S3) as following [3]:

$$x' = \frac{x-min}{max-min} \tag{S3}$$

where x and x' are the original and normalized features, respectively; max and min denote the upper and lower limits of the original features.

5. Support Vector Machine Modelling Process

Support Vector Machine (SVM) is a famous machine learning method. The basic idea of SVM is to map the original data X into a feature space F with high dimensionality through a non-linear mapping function and construct an optimal hyperplane in new space. SVM can be applied to both classification and regression. In the case of classification, an optimal hyperplane is found that separates the data into two classes. Whereas in the case of regression a hyperplane is to be constructed that lies close to as many points as possible.

The fundamental scheme behind the SVM model is to find the objective function $f(x)$, expressed by Equation **Error! Reference source not found.**, and this function is simultaneously subjected to the cost function, given by Equations **Error! Reference source not found.**, and several constraints, given by Equation **Error! Reference source not found.** as follows [4]:

$$f(x) = w \cdot x + b \tag{S4}$$

$$\text{minimize} \quad \frac{1}{2}\|w^2\| + C\sum_{i=1}^{n}(\xi_i) \tag{S5}$$

whereas the optimization function of SVM for the regression model is as follows:

$$\text{minimize} \quad \frac{1}{2}\|w^2\| + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \tag{S6}$$

$$\text{subject to} \begin{cases} y_i - w \cdot x_i - b \leq \varepsilon + \xi_i \\ w \cdot x_i + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases} \tag{S7}$$

where w denotes the parameter vector of the SVR model; x is the feature set of samples; b represents the constant number; C stands for the regularization parameter; ξ_i and ξ_i^* are the slack variables, allowing for infeasible samples that exceed the precision of the model; x_i and y_i are the features and labels of the sample i, respectively; and ε represents the precision of the model (i.e. the maximum toleration of predicted value can deviate from actual value). This objective function illustrated the basic idea for a linear model, while the kernel function could be applied for more complex nonlinear problems in the present work. The kernel function transfers the original feature space (x) to a higher dimensionality (x') used for mapping the nonlinear correlation [5]. Thus, it is a critical factor in the generalizability of the SVR model.

$$k(x - x') = \exp\left(-\gamma\|x - x'\|^2\right) \tag{S8}$$

6. GA Supported SVM—Tuning Process

The genetic algorithm (GA) is a well-known optimization methodology known for its robustness. In general, a GA is defined by an iterative process, comprised of six key stages: generating initial population, evaluation, ranking, selection, crossover and mutation. Each GA can differ, but all GAs has a few things in common. Each GA has a population size, which is made up of individuals. Each individual represents a potential solution to the problem the GA is trying to solve. Each individual in the population (plus the newly created individual starting in the second iteration) is given a fitness value in the evaluation step. All the individuals are then ranked based on their fitness. In the selection step, two individuals are selected to reproduce based on their rank. The two selected individuals' genotypes are used to produce a new individual in the crossover step. Finally, the new individual's genes may be mutated (bits are randomly flipped) and the next iteration begins.

7. Neural Network—Modelling Process

A neural network (NN) usually consists of an input layer, one or more hidden layers, and one output layer. The number of input neurons corresponds to the number of parameters that are being presented to the network as inputs. Neurons in hidden layers are responsible primarily for feature extraction. The number of hidden layers and hidden neurons is unknown and can be unlimited. They provide increased dimensionality and accommodate such tasks as classification and pattern regression. However, an optimal network size is necessary to obtain efficiency, accuracy, and excellent results in a finite time. A multilayer neural architecture has one or more layers of nodes/neurons (hidden layers) between the input and output layers.

Let i and j be the different neurons of the network. The input layer neurons (i) distribute the input signals, x_i, to the hidden layer neurons. Each neuron, j, in the hidden layer sums up its input signal, x_i, after weighting them with the respective connection's strength, W_{ji}, from the input layer and computes its output y_j according to Equation (S9) [6].

$$Y_i = f\left(\sum W_{ji}X_i\right) \tag{S9}$$

where f is a nonlinear activation function and assume that it is a logistic sigmoidal function of linearity. Therefore, the output neuron of any layer is given by Equation (S10).

$$f(x) = \frac{1}{1 + e^{-a(x)}} \tag{S10}$$

The change W_{ji} in the weight of a connection between neurons i and j is given by Equation (S11)

$$\Delta W_{ji} = \eta \delta_j x_i \tag{S11}$$

where η is a learning rate parameter and δ_j is a factor depending on whether neuron j is an output neuron or a hidden neuron.

For output neurons:

$$\delta_j = \frac{\delta f}{\delta_{net_j}} (y^t{}_j - y_j) \tag{S12}$$

For hidden output neurons:

$$\delta_j = \frac{\delta f}{\delta_{net_j}} (\sum W_{qj} \delta_q) \tag{S13}$$

In Equation (S12), net_j is the total weighted sum of input signals to neuron j, and $y^t{}_j$ is the target output for neuron j. In the absence of target outputs for hidden neurons in Equation (S13), the difference between the target and actual outputs of a hidden neuron j is replaced by the weighted sum of the δ_q terms already obtained for neurons q connected to the output of j. Next, the squared error, ε_j can be calculated as given by Equation (S14).

$$\varepsilon_j = \frac{1}{2} \sum_{j=1}^{k} [y^t{}_j - y_j]^2 \tag{S14}$$

where k is the total number of output nodes.

8. GA Optimized Neural Network (GA-NN) Modelling Process

GA uses the survival of the fittest principle of nature to search the given objectives. An important part of the genetic algorithm is the generation of new individuals by combining the genomes of two "fit" parents. In order to determine which individuals are more "fit" than others, a formula for calculating fitness is defined. In this work, GA is used for optimizing objective function of the ANN model through it's fitness function.

The objective function is a function which describes the objective of the optimization or search the best input and output based on the given relationship mapping between them. It can also express the minimize or maximum output and corresponding input for a defined system. Here we are going to use our established ANN model input-output relationship model equation which can be written as:

$$f(Z_{j(w,b)}) = O^4 = f\left(W_4 f\left(W_3(X_3 + b_3) f\left(W_2(X_2 + b_2) f(W_1(X_1 + b_1))\right)\right) + b_4\right) \tag{S15}$$

Here, $f(Z)$ represents the objective function for making the fitness function. This function has been created based on the ANN model which has given best performance for the given inputs. O^4 is the output vector of the forth layer, W_1, W_2, W_3 and W_4 represents the weight matrix of the four hidden layers of the established ANN model. The values of b_1, b_2, b_3 and b_4 represent the bias vectors of the four layers where $X_{(i)}$ represents the input vectors of the next corresponding layers. The objective function Y= $f(Z)$ formed by the ANN model cost function have to be optimized through the GA optimization techniques.

9. Model Accuracy

The performance evaluation will show us the efficiency of the model. The way to measure the performance of the trained model is the validation test. In this research, validation of the trained networks is carried out after trained a model then model will be validated to improve the accuracy of the model. To evaluate the performance of the model

several evaluation criteria have been used by different researcher [7]. Therefore, the root means square error (RMSE) and coefficient of determination (R^2) were chosen as the evaluation criteria to quantify the accuracy of SVR-RBF models. The *coefficient of determination (R^2),* is a measurement way which provides the information about how well a model fit with its target values. Mean squared error or MSE presents the average squared difference or error between the estimated values and the predicted values. Root mean squared error or RMSE is the root of the MSE. These calculation methods are expressed in Equation (S16) and Equation (S17):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_{ia} - y_{ip})^2} \quad \text{(S16)}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_{ia} - y_{ip})^2}{\sum_{i=1}^{n}(y_{ia} - \hat{y})^2} \quad \text{(S17)}$$

where n is the number of samples; y_{ia} and y_{ip} are the actual and the predicted output of sample i, respectively; and $\hat{y}$ is the mean value of the label in the testing set. If the model gives less error percentage along with better prediction and less root mean square error (RMSE) with higher correlation coefficient (R^2) value that model is said to be an efficient prediction modelling with good performance.

References

1. Hunter, J.D. Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.* **2007**, *9*, 90. https://doi.org/10.1109/MCSE.2007.55.
2. Kantardzic, M. *Data Mining: Concepts, Models, Methods, and Algorithms*; John Wiley & Sons: Hoboken, NJ, USA,2011.
3. Kotsiantis, S.; Kanellopoulos, D.; Pintelas, P. Data preprocessing for supervised leaning. *Int. J. Comput. Sci.* **2006**, *1*, 111–117.
4. Smola, A.J.; Schölkopf, B. A tutorial on support vector regression. *Stat. Comput.* **2005**, *14*, 199–222. https://doi.org/10.1023/B:STCO.0000035301.49549.88.
5. Burges, C.J.C. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Min. Knowl. Discov.* **1998**, *2*, 121–167. https://doi.org/10.1023/A:1009715923555.
6. Jarin, S. *Modelling and Optimization of Nano Powder Mixed Micro WEDM Process Using Artificial Neural Networks and Genetic Algorithm*; International Islamic University: Kuala Lumpur, Malaysia, 2018. https://doi.org/10.1504/IJMATEI.2019.103614.
7. Kohavi, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. *IJCAI* **1995**, 1137–1145.