SEHS
science, engineering and health studies

Research Article

Science, Engineering and Health Studies
https://li01.tci-thaijo.org/index.php/sehs
ISSN (Online): 2630-0087

Enhancing small dataset prediction of silver nanoparticle size with deep learning and Latin hypercube sampling framework

Chinakrit Akkawong[1], Tanawadee Dechakupt[2], Kulpavee Jitapunkul[1], and Chanin Panjapornpon[1]*

[1] Department of Chemical Engineering, Center of Excellence on Petrochemicals and Materials Technology, Faculty of Engineering, Kasetsart University, Bangkok 10900, Thailand
[2] Department of Industrial Chemistry, Faculty of Applied Science, King Mongkut's University of Technology North Bangkok, Bangkok 10800, Thailand

***Corresponding author**:
Chanin Panjapornpon
fengcnp@ku.ac.th

***Received**: 24 April 2024*
***Revised**: 18 June 2024*
***Accepted**: 2 July 2024*
***Published**: 31 December 2024*

***Citation**:*
Akkawong, C., Dechakupt, T., Jitapunkul, K., and Panjapornpon, C. (2024). Enhancing small dataset prediction of silver nanoparticle size with deep learning and Latin hypercube sampling framework. Science, Engineering and Health Studies, 18, 24020012.

ABSTRACT

Laboratory experiments often face challenges such as inherent complexities, difficulties in data gathering, high costs, and time-consuming procedures. These constraints typically result in a limited amount of experimental data, leading to calculation issues such as overfitting and underfitting. To address these issues, this study applied an integrated framework of deep learning combined with Latin hypercube sampling (LHS) to enhance prediction models based on small datasets. A case study on size prediction in silver nanoparticle synthesis was used to demonstrate the performance of the developed framework. The LHS technique augments the amount of raw data for model development. Consequently, the original raw data and the data generated from LHS were integrated as training data for the development of a deep learning prediction model. This integrated model improved prediction performance, validated by the validation and test dataset R^2 values, which are 0.924 and 0.918, respectively. Additionally, the accuracy of unseen data test results was significantly higher when compared to a model trained on a small dataset, with the value rising from 0.442 to 0.893. The proposed framework enables high-accuracy predictions of silver nanoparticle size using small experimental datasets and other conditions within specified boundaries.

Keywords: Latin hypercube sampling; limited data; deep learning; artificial intelligence

1. INTRODUCTION

Deep learning is being increasingly applied to construct prediction models in various fields, including the chemical industry. However, constructing a high-accuracy prediction model requires quality and extensive data to learn the dataset's characteristics (Dhiman et al., 2023). Small datasets, often found in laboratory experiments, are valuable but lead to insufficient data characteristics, resulting in overfitting and underfitting scenarios that cause inaccuracy and unusable prediction results (Subramanian and Simon, 2013). The challenges of data collection in laboratory experiments, such as inherent complexity, high costs, and time consumption, contribute to this issue (Falk and Heckman, 2009).

One method to address the issue of small datasets is to apply statistical methods to expand the data (Austin et al., 2021). This involves extracting additional value from small data points to construct prediction models, thus improving accuracy and reducing the need for extensive experimental sampling.

Machine learning and deep learning algorithms are commonly used for constructing prediction models. While both aim to learn from data and make predictions, deep learning algorithms are more powerful in handling complex data and extracting relational information (Sharma et al., 2021). However, these algorithms require sufficient information to prevent overfitting and underfitting, which occur when an algorithm performs well on the training dataset but poorly on the test and validation datasets (Pothuganti, 2018).

Recent attempts to construct predictive models with small data have encountered overfitting and suggested data enlargement to solve this issue (Brigato and Iocchi, 2021). Enlarging the data can be achieved by designing independent experiment sampling techniques (Etikan and Bala, 2017) and estimating dependent variables using the correlation between independent and dependent variables (Berndt, 2020). Techniques such as Monte Carlo sampling (MCS) and Latin hypercube sampling (LHS) are used for this purpose. MCS is a simple and universal method but requires a large sample size to achieve stable output probability distributions (Shields and Zhang, 2016). In contrast, LHS provides systematic sampling by dividing probability distributions into equal intervals and is suitable for multidimensional spaces (McKay et al., 1979; Cioppa and Lucas, 2007). However, LHS does not reduce variance in variable interactions (Shields and Zhang, 2016).

The study aimed to (1) enhance the small dataset of silver nanoparticle synthesis using LHS to generate more samples for training the prediction model; (2) implement an alternative method to eliminate overfitting scenarios in small datasets by using generated data from LHS; and (3) construct a high-accuracy prediction model by combining small data and generated data for silver nanoparticle synthesis applications.

2. MATERIALS AND METHODS

This study's methodology aimed to construct a meaningful prediction model from limited data, consisting of four major parts. The first part involved data preparation, which aimed to break the received data into a limited dataset and an unseen dataset. The second part was the application of LHS to expand the limited dataset for training the prediction model. The third part involved constructing the prediction model, and the final part evaluates the prediction model's performance with an unseen dataset. Figure 1 shows the overall process flow diagram for this study.

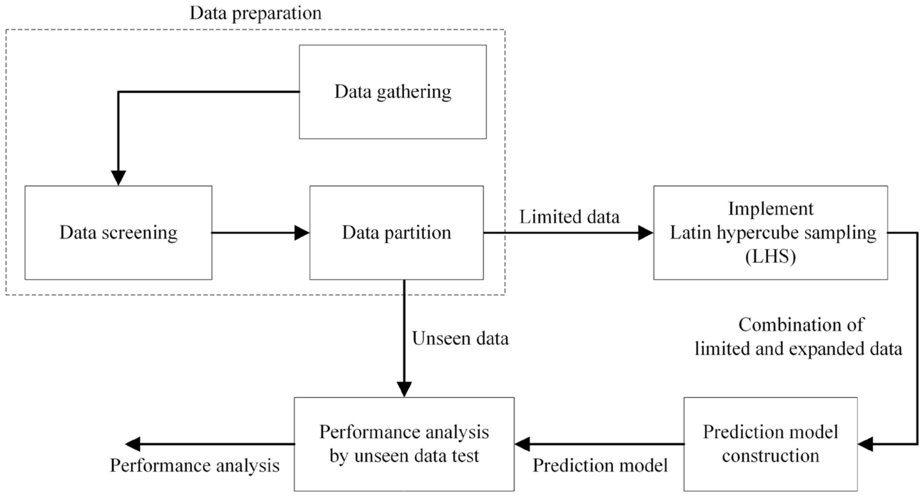


Figure 1. Process flow diagram

2.1 Experiment data preparation

The first step in constructing a prediction model was to manage the raw data. In this study, the experimental data on silver nanoparticle synthesis were obtained from a published article (Shafaei and Khayati, 2020). These published data combine a silver nanoparticle synthesis experiment with a swarm optimization algorithm to predict nanoparticle size as the output. The aim of the experiment was to identify the most effective conditions for achieving the smallest silver nanoparticle size, which significantly affects their properties. This study applied the concept of limited data by using data expansion techniques. The "limited data concept" involves separating the gathered data into two datasets: one for data expansion and another for evaluating model performance with unseen data. The data preparation process includes three steps: data gathering, data screening, and data partitioning (Figure 1).

2.1.1 Data gathering

The data were collected from multiple published articles on silver nanoparticle synthesis experiments using the "green method". This method is environmentally friendly, utilizing plant extracts or eco-friendly substances as stabilizing and reducing agents. In the specific experiment, opium syrup was used as a reducing agent mixed with silver nitrate as a metal salt. The particle size was observed under varying initial conditions that affect the synthesis process within 30 min of the experiment. After the synthesis procedure was completed, the samples were centrifuged three times at 15,000 rpm for 20 min to remove any contaminants adsorbed on the nanoparticle surfaces, yielding dry powder for analysis. The experimental design depended on the volume ratio of various opium syrup concentrations to 100 mL of silver nitrate solution, the feed rate of the mixed solution between opium syrup and silver nitrate, pH, reaction temperature, and agitation speed. These five experimental conditions were considered independent variables, while the size of the silver nanoparticles was the dependent variable. Each independent variable varied within a specific range (Table 1). The gathered data were analyzed using an F-test to rank and visualize the significance of each independent variable (Figure 2).

According to the F-test importance scores ranking, pH and volumetric ratio were significantly higher than the other independent variables, indicating that these two conditions had a high influence on the experimental results despite minor changes in their values (Figure 2). The importance scores for feed rate, agitation speed, temperature, and silver nitrate concentration were respectively lower compared to pH and volumetric ratio. Notably, the importance score for silver nitrate concentration was zero because this independent variable was fixed in every experiment and was not significant in determining silver nanoparticle size.

Table 1. Range of gathered data input variables

Parameter	Range
Volume ratio of $AgNO_3$ to opium syrup	5–20
Feed rate (ml/min)	0.33–10
Agitation speed (rpm)	100–600
pH	5–8
Temperature (°C)	25–65

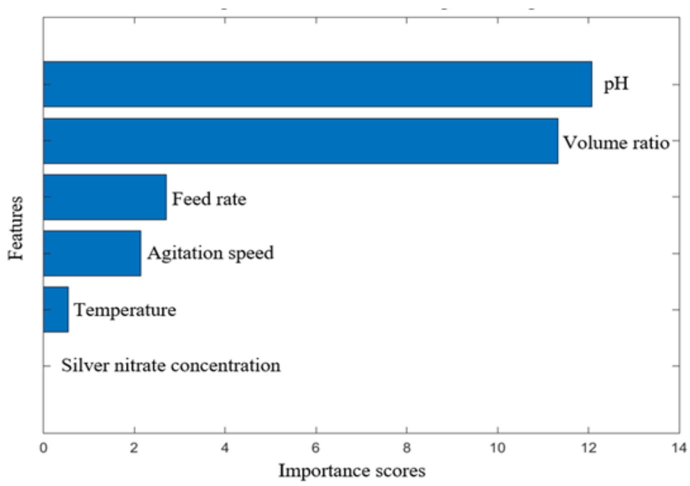


Figure 2. Importance scores sorted using F-test

2.1.2 Data screening and data partition

The reference data consist of the conditions and results of the experiment. Some dataset samplings exhibited repeated experimental conditions and conflicting data, increasing the dataset's complexity. To reduce complexity and enhance model construction effectiveness, data screening was necessary. The dataset initially contained 103 experimental data samplings. Issues arose where some experiments were repeated but yielded different results, or different input conditions produced the same result. Data screening was applied in these cases to reduce complexity and prevent conflicts within the dataset. For experimental data with identical operating conditions but different results, a single sampling was used to represent those samples by averaging their values. Similarly, repeated experiments with identical operating conditions and results were reduced to a single sampling. Consequently, after screening, the dataset was reduced to 94 samples. The screened data were then divided into two datasets: 50 samples for model construction and training, and 44 samples for an unseen test to evaluate model performance. These two datasets were randomized by holdout randomization. The unseen dataset was screened after randomization to ensure that the unseen data fell within the range of the trained dataset. If some samplings in the unseen dataset were out of range, those data were swapped with appropriate data from the trained dataset.

2.2 LHS framework

Small datasets are commonly found in experiments, and LHS is applied to address this problem. LHS uses statistical methods to generate samples from the multidimensional distribution of the data. LHS divides the data distribution into equal probability intervals based on the desired number of samples, ensuring that each interval is sampled once in each dimension (Figure 3).

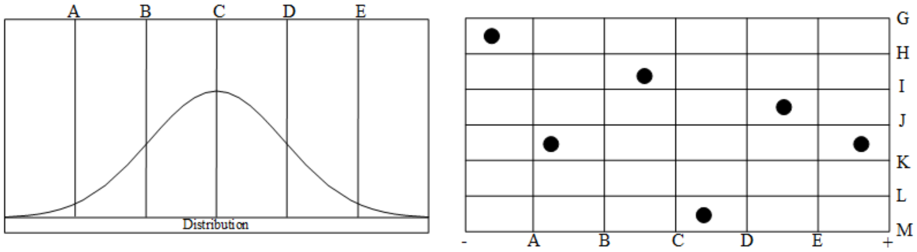


Figure 3. The random pattern of LHS

By using equal probability intervals, LHS reduces data clustering across all samplings. A key feature of LHS is random sampling with equal distribution, leading to a more efficient sample space that covers the entire range of input conditions. Following LHS, the input space is randomized, and a correlation model is used to estimate the output of each generated sample. The correlation model is a statistical mathematical model that analyzes the characteristics between input variables (independent variables) and output variables (dependent variables) to understand how input variables affect output variables. Figure 4 shows the three workflows of the LHS framework: modeling the correlation model, implementing data with LHS, and hypothesis testing, respectively.

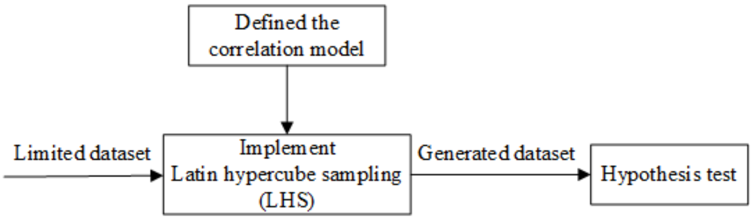


Figure 4. LHS workflow

2.2.1 Defining the correlation model

LHS was the chosen method to expand the data range. However, LHS cannot estimate output results, as it only distributes the input data range into the desired number of samplings. To address this issue, a mathematical model was applied to learn the correlation between independent variables (input data) and dependent variables (output data). For this study, second-order polynomial regression was used for regression analysis because the relationship between the independent and dependent variables was nonlinear. The polynomial regression model captures the characteristics of the data between the input and output variables and estimates the coefficient values, providing insights into the importance of the original data. The second-order polynomial regression equation is shown as Equation 1:

$$Y = \beta_0 + \sum_{i=1}^{n} \beta_i \cdot x_i + \sum_{i=1}^{n} \beta_{ii} \cdot x_i^2 + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \beta_{ij} \cdot x_i \cdot x_j \tag{1}$$

where Y represents the predicted output, β_0 represents the interception term, β_i represents the coefficients for the linear terms of each input variable, β_{ii} represents the coefficients for the quadratic terms of each input variable, represents the coefficients for the interaction terms between pairs of input variables *i* and *j*, and x_i and x_j represent different input variables.

2.2.2 Expanding the dataset with LHS

With limited data, LHS was used to expand the dataset. As previously mentioned, LHS increases the data size by distributing each input variable's data range into equal probability intervals for the required number of samples, then randomizing the sampling within intervals. After enlarging the sample size of the independent variables, the polynomial regression correlation model was applied to estimate the output value of each independent variable. Following regression analysis, the output dataset was estimated using the coefficients from the mathematical correlation model. After expanding the data, an analysis test of statistical samples was performed in the next section to ensure the expanded dataset values were within the range of probability and accuracy.

2.2.3 Hypothesis testing

The generated data is a result of LHS. Before using the generated data to model a neural network, the dataset must be examined for plausibility. To validate the expanded data, a hypothesis test was applied. A hypothesis test analyzes statistical samples—in this case, the generated data from LHS. It makes a statistical inference from sample population data to determine whether there is enough evidence to reject a null hypothesis in favor of an alternative hypothesis. The null hypothesis (H_0) posits that the data has no significant difference in population parameters, while the alternative hypothesis (H_α) posits that there is a significant difference in population parameters, supported by the data's evidence.

2.3 Deep neural network framework

The prediction model was constructed using deep neural networks (DNN). Deep learning, a type of artificial neural network (ANN), is suitable for solving complex problems due to its multiple layers between the input and output layers, enabling the network to learn and address complex issues. The DNN design is shown in Figure 5. These multiple layers allowed the neural network to capture the data's characteristics, enhancing its ability to learn and solve complex problems more effectively than ordinary neural networks. The DNN framework consists of three major parts: the input layer, the output layer, and the layers between them (Figure 5)

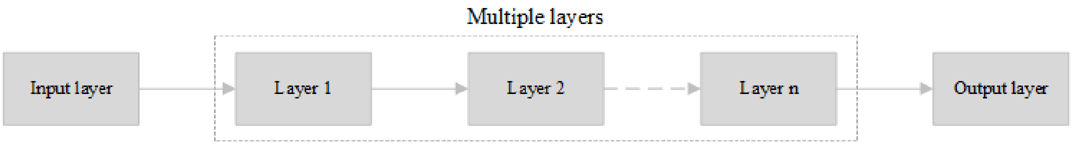


Figure 5. Structure of a deep neural network

In this study, DNNs were constructed as feedforward neural networks. The network included an input layer, an output layer, two fully connected layers, and one dropout layer to learn and train data for the prediction model. The dropout layer was incorporated because the model tends to overfit the training data but fails to perform well on validation and test datasets. Adding a dropout layer would help reduce overfitting by randomly ignoring nodes from the previous layer during training, leading to a more generalized network for the validation and test datasets. Figure 6 illustrates the neural network architecture used in this study's predictive model.

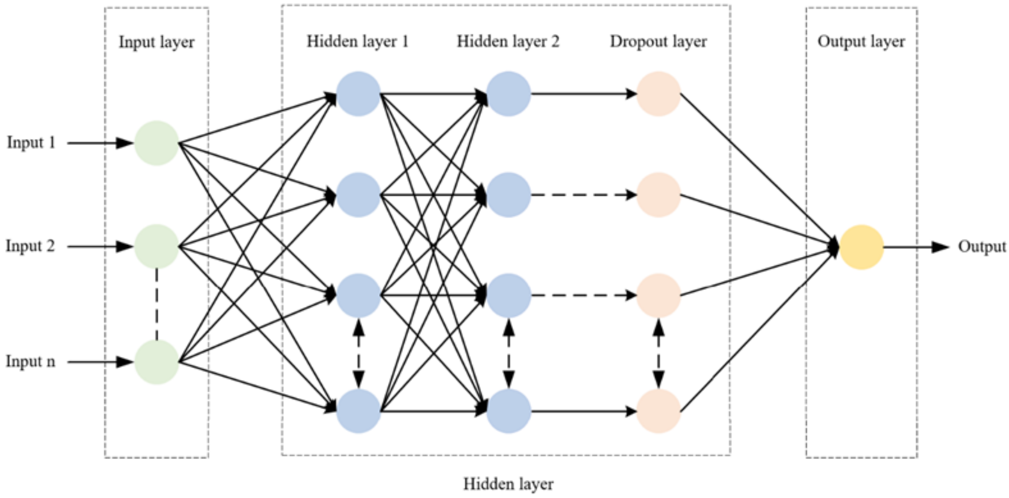


Figure 6. The construction of a prediction network

2.4 Performance analysis

The predicted output was used to evaluate the performance of the prediction model. Various indicators based on error analysis are available. However, four indicators were selected for this study: [1] (R^2 coefficient of determination) to measure the variance of predicted output, [2] (RMSE root mean squared error) to identify the average error's magnitude between predicted and actual output, [3] (MSE mean squared error) to measure the average squared between predicted output and actual output, and [4] (MAE mean absolute error) used to indicate the average absolute difference between predicted and actual output, respectively

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

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{4}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{5}$$

where y_i represents actual output value, $\hat{y}_i$ represents predicted output value, $\bar{y}$ represents mean of actual output, and n represents total number of observations.

3. RESULTS AND DISCUSSION

A meaningful prediction model was successfully constructed by following the steps of data preparation, data expansion, prediction model construction, and performance analysis based on an unseen test. This section provides the consequent output of each process.

3.1 Data propagation result

Firstly, the correlation model was constructed using second-order polynomial regression on the limited dataset, resulting in an R^2 value of 0.786. The implementation of LHS with the limited data to generate expanded data was then evaluated. Figure 7 illustrates the generated input variables from LHS and the deviation range for each input variable, which includes volume ratio, feed rate of opium syrup and silver nitrate, pH, temperature, and agitation speed. The deviation of each input variable was within the range of the gathered data, with random sampling distributed across equal intervals, demonstrating the advantage of LHS. LHS uses the given range to generate expanded input variables, ensuring nearly equal sampling numbers in each interval for every condition, while maintaining a constant silver nitrate concentration. Figure 7 shows that the expanded input variables are within the gathered data range shown in Table 1.

In this study, LHS was used to generate 100 additional datasets of independent variables. The hypothesis test was applied to analyze the plausibility of the generated data for each parameter with a significance level of 0.05 (α = 0.05). Table 2 shows the p-value for each variable from the hypothesis test, where all p-values were higher than the significance level, indicating that the alternative hypothesis was rejected for all generated variables. The p-values for volume ratio and feed rate were relatively small because the received data for these variables had a large gap between the sampling range compared to the generated data. In contrast, pH, temperature, and agitation time had small gaps between the sampling ranges, leading to higher p-values. Additionally, the silver nanoparticle size, which is the dependent variable estimated by the correlation model, also resulted in small p-values. However, since these p-values were still higher than the significance level, the expanded dataset could be used for predictive model training.

3.2 Modelling result

Following the dataset expansion, the prediction model was constructed using the previously mentioned design. The configuration of each layer, learning rate, and epoch is shown in Table 3. The model was trained using a combination of the gathered data and the expanded data. The number of samples in each dataset is summarized in Table 4. Table 5 shows the performance analysis of the prediction model trained with the combined dataset (gathered and expanded data) compared to the model trained with the limited dataset. For performance analysis, the combined and limited datasets were separated into three subsets: 70% for training, 15% for validation, and 15% for testing.

The prediction model with extended data provided slightly higher performance than the model without extended data during training (Table 5). The prediction model without expanded data presented overfitting scenarios, as indicated by an R^2 of 0.484 for the training dataset. This was higher than the test and validation datasets. Conversely, the MSE for the training set was more than that for the test and validation datasets. The R^2 value is used to consider overfitting scenarios because it captures the accuracy of actual, predicted, and mean values of the dataset. MSE is calculated based on the square magnitude of the error and averaged by the number of samples in each dataset (Chicco et al., 2021).

As a result, the prediction model without extended data achieved R^2 values of 0.484, 0.242, and 0.201 for the training, validation, and test datasets, respectively. The MSE values for each dataset were 90.208, 70.851, and 78.094, respectively, because the training dataset is approximately five times larger than the validation and test datasets, leading to a large accumulation of error values. In contrast, the prediction model with expanded data significantly improved performance, resulting in R^2 values of 0.941, 0.924, and 0.918 for the training, validation, and test datasets, respectively. Moreover, the higher accuracy of the prediction model led to a lower MSE for the training dataset. This information suggests that LHS effectively facilitates the expansion of the dataset based on limited gathered data and can reduce overfitting issues associated with small datasets.

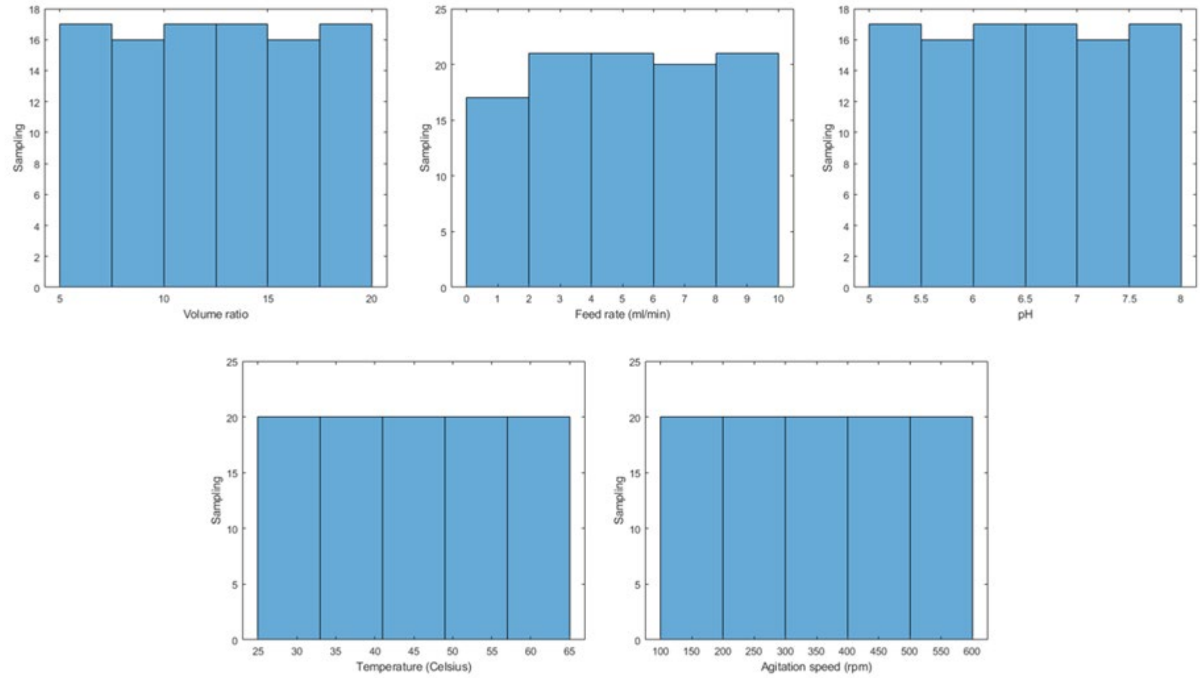


Figure 7. Generated sampling range of each input variable

Table 2. p-value of generated variables

p-value	Volume ratio	Feed rate	pH	Temperature	Agitation speed	Particle size
Value	0.101	0.092	0.818	0.734	0.9136	0.063

Table 3. Neural network structure design

Layer	Value
Fully connected layer	15 nodes
Fully connected layer	15 nodes
Dropout layer	0.2
Learning rate	0.001
Weight of each independent variable	1
Epoch	700

Table 4. Summary of sampling number of each dataset

Dataset	Samplings
Gathered data	94
Limited data	50
Unseen data	44
Generated data	100
Combined data	150

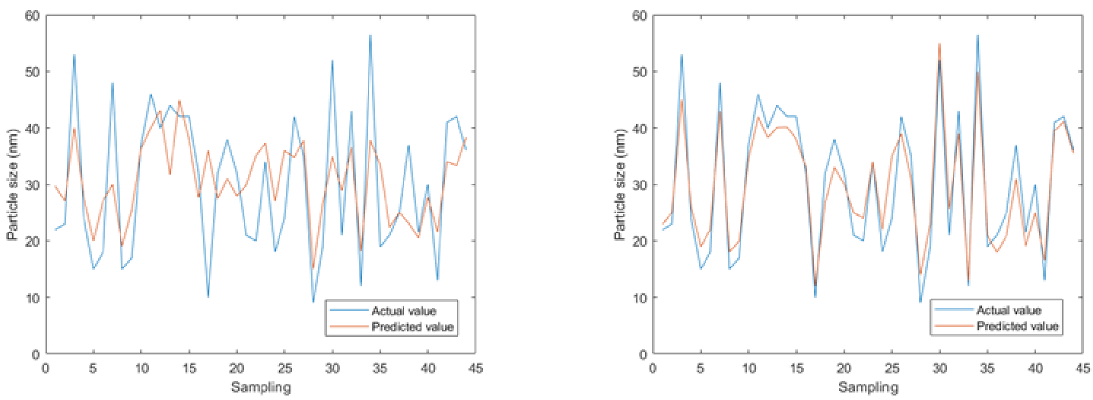
Table 5. Benchmark of a prediction model trained with and without extended data

Error criteria	Without extended data train			With combined data train		
	Train dataset	Validate dataset	Test dataset	Train dataset	Validate dataset	Test dataset
R^2	0.484	0.242	0.201	0.941	0.924	0.918
RMSE	9.948	8.417	8.837	2.431	2.697	2.629
MSE	90.208	70.851	78.094	5.907	7.271	6.915
MAE	7.927	6.793	7.563	1.869	2.149	2.274

3.3 Performance analysis for unseen dataset test

The prediction model's ability to predict the output value of each sample was evaluated using unseen data. Unseen data refers to the 44 samples not used in the model training process, which were set aside during the data preparation phase. Testing the model with unseen data is an effective way to evaluate its performance.

The results of the unseen test (Figure 8) compare the predicted values between the prediction model without expanded data and the prediction model with expanded data using the LHS technique. The prediction model trained with limited data produced accurate predictions for some samples, but most predicted values were imprecise (Figure 8a). Conversely, the prediction model with expanded data was more accurate (Figure 8b), where the predicted values were closer to the actual data, although there were still some differences in certain samples.



(a) Predicted data of limited data trained

(b) Predicted data of combined data trained

Figure 8. Performance analysis of the unseen data prediction
Note: The blue line represents the actual experimental values, while the orange line represents the predicted values from the prediction model

Following the visualization of the unseen test results, the model's performance was evaluated using four criteria: R2, RMSE, MSE, and MAE (Table 6). The desired outcomes for these criteria are the highest R^2 and the lowest RMSE, MSE, and MAE values, indicating better prediction model performance. The integrated model showed a significant increase in R^2 from 0.442 to 0.893, indicating that the predicted output was closer to the actual output. Additionally, the RMSE, MSE, and MAE values decreased significantly by approximately 5.3, 71.1, and 4.2, resulting in new values of 4.099, 16.802, and 3.459, respectively. These results demonstrate that LHS effectively improves the limited data for constructing the prediction model, as this technique expands the data and estimates information based on its characteristics.

Table 6. Performance analysis of unseen dataset between limited data trained network and integrated data with LHS trained network

Error criteria	Limited data	Integratewith LHS
R^2	0.442	0.893
RMSE	9.377	4.099
MSE	87.934	16.802
MAE	7.6239	3.459

This study aimed to enhance the prediction model's performance using limited data and LHS. The reference experimental data were gathered from Shafaei and Khayati (2020), which utilized an ANN with a particle swarm optimization algorithm to construct a prediction model. While this optimization approach leads to higher accuracy, it also requires more computational time. By applying the limited data concept, the received data were divided into small datasets, expanded using LHS, and combined for training the DNN to construct the prediction model. The proposed model achieved good prediction accuracy with an R^2 of 0.893 based on unseen testing with the unseen dataset. Although the DNN model does not provide the highest accuracy, it is simple and powerful in expanding small samples.

4. CONCLUSION

This study utilized limited data as the initial dataset for creating a prediction model. For the model to produce meaningful and accurate outputs, it must be trained on a large dataset, as limited data typically leads to underfitting and overfitting. To address this issue, this study proposed the application of LHS.

As a result, the LHS technique successfully expanded the limited data into a sufficiently large dataset that retained the complete characteristics of the original data. The expanded dataset was then used to train and construct a neural network for prediction. Moreover, the integrated prediction model, trained with a combined dataset of limited and expanded data, yielded more accurate forecasts. Performance analysis of the prediction model showed an increase in R^2 and significant decreases in RMSE, MSE, and MAE, indicating effective training from the combined dataset.

LHS could generate an input dataset from the range of each input variable in the gathered data. The generated dataset was random and equally distributed across intervals. Additionally, the output of each sampling was estimated by a correlation model, which is essential for generating accurate output predictions. The integrated prediction model, trained on the larger dataset, achieved greater prediction accuracy than the model trained on the limited dataset. Performance analysis of the unseen dataset yielded an R^2 of 0.893, an RMSE of 4.099, an MSE of 16.802, and an MAE of 3.459. LHS is a simple yet powerful approach for expanding small datasets, making it an effective alternative for improving prediction model accuracy.

ACKNOWLEDGMENT

The authors would like to acknowledge the support of the Faculty of Engineering, Kasetsart University (Grant No.67/04/CHEM/M.Eng), Center for Advanced Studies in Industrial Technology, and the Center of Excellence on Petrochemical and Materials Technology.

REFERENCES

Austin, P. C., Harrell, F. E. Jr., and Steyerberg, E. W. (2021). Predictive performance of machine and statistical learning methods: Impact of data-generating processes on external validity in the "large N, small p" setting. *Statistical Methods in Medical Research*, 30(6), 1465–1483.

Berndt, A. E. (2020). Sampling methods. *Journal of Human Lactation*, 36(2), 224–226.

Brigato, L., and Iocchi, L. (2021). A close look at deep learning with small data. In *Proceeding of the 25th International Conference on Pattern Recognition (ICPR)*, pp. 2490–2497. Milan, Italy.

Chicco, D., Warrens, M. J., and Jurman, G. (2021). The coefficient of determination r-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Computer Science*, 7, e623.

Cioppa, T. M., and Lucas, T. W. (2007). Efficient nearly orthogonal and space-filling latin hypercubes. *Technometrics*, 49(1), 45–55.

Dhiman, P., Ma, J., Qi, C., Bullock, G., Sergeant, J. C., Riley, R. D., and Collins, G. S. (2023). Sample size requirements are not being considered in studies developing prediction models for binary outcomes: A systematic review. *BMC Medical Research Methodology,* 23(1), 188.

Etikan, I., and Bala, K. (2017). Sampling and sampling methods. *Biometrics & Biostatistics International Journal*, 5(6), 00149.

Falk, A., and Heckman, J. J. (2009). Lab experiments are a major source of knowledge in the social sciences. *Science*, 326(5952), 535–538.

McKay, M. D., Beckman, R. J., and Conover, W. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2), 239–245.

Pothuganti, S. (2018). Review on over-fitting and under-fitting problems in machine learning and solutions. *International Journal of Advanced Research in Electrical Electronics and Instrumentation Engineering*, 7(9), 3692–3695.

Shafaei, A., and Khayati. G. R. (2020). A predictive model on size of silver nanoparticles prepared by green synthesis method using hybrid artificial neural network-particle swarm optimization algorithm. *Measurement*, 151, 107199.

Sharma, N., Sharma, R., and Jindal, N. (2021). Machine learning and deep learning applications-a vision. *Global Transitions Proceedings*, 2(1), 24–28.

Shields, M. D., and Zhang, J. (2016). The generalization of Latin hypercube sampling. *Reliability Engineering & System Safety*, 148, 96–108.

Subramanian, J., and Simon, R. (2013). Overfitting in prediction models – is it a problem only in high dimensions? *Contemporary Clinical Trials*, 36(2), 636–641.