Delving into machine learning modeling of catalytic reactor system: a case study of steam methane reforming

Hyeon Yang,Chanhee You,Chanmok Kim,and Jiyong Kim\*

*School of Chemical Engineering, Sungkyungkwan University, 16419, Republic of Korea*

\*Corresponding Author’s E-mail: jiyongkim@skku.edu

Abstract

Navigating the complexity of reactions is one of the fundamental tasks in the field of chemical engineering. Within this domain, identifying reactions in catalytic systems is one of the challenges due to the intrinsic interplay between process variables and reaction mechanisms. In recent years, an increasing number of researchers have examined the application of machine learning in the development of prediction models for identifying reaction mechanisms. Although, their applications in reaction modeling involving kinetics have remained confined, leading to unreliable predictions and limited insights into reaction systems. To address these challenges, we propose a graph ensemble deep learning approach for multiscale modeling that predicts catalytic reaction systems with process variables (e.g., pressure, temperature, reactor size, and flow rate). Our approach includes a tree-based deep learning model to predict conversion of reaction systems governed by distinct mechanisms. To distinguish the mechanism, our model is ensembled with graphical neural networks for inferring correlation between feature and target, such as chemical distribution and reaction condition profiles. We demonstrated the approach through a case study of Ni-based steam methane reforming under varying conditions using a process simulation dataset derived from kinetic equations. The proposed model has optimized hyperparameters by nested k-fold cross-validation. The prediction result of reaction conversion and selectivity shows that our approach can estimate outcomes and effectively explore undiscovered reaction spaces. Furthermore, our preliminary findings illustrate that the proposed approach is applicable to a wide range of reactions involving complex mechanisms without requiring extended experiments for kinetic study.

**Keywords**: Machine learning, Deep ensemble learning, Chemical process, Steam methane reforming

* 1. Introduction

In the chemical industry, modeling of reaction or reactor systems is crucial for designing, scaling up, controlling, and optimizing processes. Modeling has gained significant attention over the past century because it provides a better understanding of the underlying mechanisms, thereby improving the economics and development of chemical process. (Glassey, 2018). Machine learning (ML) approaches are increasingly being utilized in chemical engineering to improve understanding of complex chemical systems.

One prominent approach in the development of ML is the Cross-Industry Standard Process for Data Mining (CRISP-DM) (Shearer, 2000). CRISP-DM comprises six distinct phases (i.e., business understanding, data understanding, data preparation, modeling, evaluation, and deployment) and provides a comprehensive and systematic strategy for addressing many challenges inherent in data-driven modeling. CRISP-DM is an inherently iterative process, as insights gained during the modeling process can lead to a redefinition of objectives and approaches, as shown in Figure 1.



**Figure 1** Cross-industry standard process for data mining (CRISP-DM) (Shearer, 2000)

In the deployment of a model, it is crucial to consider generalization, which refers to the ability of the model to accurately predict outcomes using data not previously encountered during development (Kim, 2017). Furthermore, the development of model requires consideration of the risk of overfitting. Overfitting arises when a model becomes too closely tuned to the specific noise or errors in the training dataset, leading to inferior performance on new data (Srivastava et al., 2014). Based on these concepts, developing reliable models in chemical process modeling are requires ensuring consistent performance of the model on both the utilized dataset and the unseen data.

However, while models such as the artificial neural network (ANN), XGBoost (XGB), and ExtraTreesRegressor (ETR) demonstrate high performance in various chemistry-related areas (Wu et al., 2018, Shinya et al., 2021), these studies do not sufficiently address for model capability in aspect of generalization and overfitting. Therefore, our study evaluates model not only in aspect of accuracy but also rigorously examines generalization and resistance to overfitting. By doing so, we aim to provide comprehensive insights into the development and application of ML model in the field of chemical engineering.

* 1. Methodology

2.1 Data preparation

In this study, we developed model to predict catalytic reaction systems with handling operating conditions (e.g., pressure, temperature, reactor size, and flow rate). Since it is difficult to collect reaction data in the chemical industry, we collected data through simulation which was performed in Aspen Plus V12.0. For case study of Ni-based steam methane reforming, the reaction is simulated with experimentally validated kinetics (Oliveira et al., 2009) and the reforming reactor is assumed as tubular reactor. We reorganized the simulated data with eight input variables and one output variable. The input variables include (1) pressure (*P*), (2) temperature (*T*), (3) H2O ratio in the feedstock, (4) reactor diameter (*D*), (5) the number of tube (*NTUBE*) in reactors, (6) volumetric flow rate (*Q*) (7) gas hourly space velocity (*GHSV*) and (8) Linear velocity (*LV*) driven by Equation 1 and the output variable is CH4 conversion (XCH4).

$$\begin{array}{c}LV= \frac{Q}{πD^{2}\*NTUBE}\#\left(1\right)\end{array}$$

2.2 Data preprocessing

As shown in Equation 1, the variable of reactor size (e.g., *D, NTUBE*) and the flowrate variable of streams (*Q, GHSV, LV*) are highly correlated, and reactor size variables increase exponentially thereby making skewed distribution. To address the skewness of flowrate variables, we first performed a log transformation on the flow variables.

Also, the correlation coefficient within log-transformed flowrate variables is higher than 0.9 in Figure 2. (a), which mean highly intercorrelated features.

To consider these correlations in variables, we applied principal component analysis (PCA). PCA is a technique for dimension reduction, which linearly transform the data into new coordinates system with fewer dimensions, conserving the information about the variation in the data. PCA are used to five variables among the input variables (i.e., *D, NTUBE, Q, GHSV, LV*) for make three variables (*PCA1, PCA2, PCA3*) in new coordinates systems based on the explained variance as shown in Figure 2. (b). For all the cases, 80% of the simulated data were used as the training dataset to optimize the model hyperparameters, while the rest of 20% simulated data were treated as the *test* dataset for evaluation the accuracy of model prediction.

Furthermore, we prepare another *test* dataset called as an *unseen* dataset, which denotes that we regard model performance on not encountered dataset during training as the generalizability and resistance to overfitting. the similar and interpolated boundaries for each variable. An *unseen* dataset is organized similarly with training dataset and has an interpolated boundary. For example, the distribution of XCH4 by temperature in *unseen* data are included with in boundary of training data, as shown in Figure 2. (c).



**Figure 2**. Configuration of dataset about (a) correlation matrix for input and output variables, (b) explained variance ratio of each PCA components, (c) distribution for CH4 conversion of *test* (blue) and *unseen* (purple) data.

2.3 Model development

After preparing the dataset, we generated three ML models based on different algorithm.

**ANN**: The ANN model employs a method of correlating input and output variables by multiplying weights, adding biases, and implementing nonlinear functions, known as activation functions (Goodfellow et al., 2016). To prevent overfitting, the model incorporates a dropout method as a regularization technique, which involves randomly omitting a subset of neurons during the training process (Hinton et al., 2012). This approach reduces co-adaptation among neurons, enhancing the model’s ability to generalize. The model was developed using the TensorFlow Python package (Abadi et al., 2016).

**XGB**: The XGB model is based on decision tree algorithms that divide the input variable space and associating specific regions with output variables. In the context of decision trees, overfitting often occurs when the model creates excessive branches in response to outliers or anomalies in the training data. To address this, the XGB model employs the built-in *max\_depth* parameter which limits the depth of the trees, reduce model complexity and enhancing generalizability. Additionally, the model employs built-in normalization parameters: *reg\_alpha* (L1 lasso regression) and *reg\_lambda* (L2 ridge regression). These parameters introduce penalty terms into the loss function, serving to further regularize the model and prevent overfitting. The development of this model was conducted through the XGBoost Python package (Chen et al., 2016)

**ETR:** The ETR model, while similar to the XGB model in its foundation on decision tree algorithms, extremely randomized algorithm by selecting random subsets of features to split on at each tree node. The parameters (i.e., *max\_depth*, *min\_samples\_leaf and min\_samples\_split*) are used for controlling the complexity of trees, contributing to its resistance to overfitting. This model was created with the ExtraTreesRegressor in scikit-learn(Pedregosa et al., 2011).

By using the nested 5-fold cross validation with Bayesian-optimization, we searched the optimal hyperparameters of each model (Raschka et al., 2018). The search space of hyperparameters is listed in Table 1.

**Table 1**. Search space of hyperparameters for ANN, XGB, and ETR

|  |  |  |
| --- | --- | --- |
| ANN | XGB | ETR |
| n\_layer | [3, 5] | n\_estimators | [500, 1000] | n\_estimators | [500, 1000] |
| n\_node | [25,26, … , 29] | max\_depth | [3, 10] | max\_depth | [3, 10] |
| learning rate | [1e-3, 3e-2] | learning rate | [1e-3, 3e-2] | min\_samples\_leaf | [1, 10] |
| batch size | [26, …, 211] | gamma | [1e-9, 1e-6] | min\_samples\_split | [1, 10] |
| dropout | [0.1, 0.5] | reg\_alpha | [1e-5, 1]  |  |  |
|  |  | reg\_lambda | [1e-5, 1] |  |  |

* 1. Result

The performance of each model (i.e., ANN, XGB, and ETR) was evaluated across *train*, *valid*, *test* and *unseen* datasets based on the Mean Absolute Error (MAE), detailed in Figure 3. To assess the impact of the PCA variables on model performance, we developed two cases: Case 1 consists of eight input variables (i.e., *P, T, H2O ratio, D, NTUBE, Q, LV, GHSV*), Case 2 consists of eight input variables with 3 PCA variables (i.e., *PCA1, PCA2, PCA3)*.

In the *train*, *valid*, and *test* datasets, the XGB model consistently outperformed the other models in both cases both cases, followed closely by the ETR model. The ANN model exhibits the lowest performances among both cases. For instance, the XGB model shows a significantly lower MAE of 0.05% compared to the 3% MAE observed in the ANN model.



**Figure 3.** Comparison of model accuracy (MAE (%)) among the datasets

Regarding *train*, *valid*, and *test* datasets, the narrow gap of MAEs across these datasets suggest that all models adequately generalize and resist overfitting. However, the different pattern is exhibited in the *unseen* dataset. A notable shift was observed in the *unseen* dataset, where the performance of the XGB and ETR models was inferior compared to the ANN model. Specifically, the XGB model display a MAE of approximately 4% indicating the highest error, and ETR model follows with a MAE of 3.7%. This reversal in performance for both XGB and ETR model can be attributed to the inherent traits of tree-based algorithm of both models, leading to higher accuracy on familiar *test* data but diminished performance on *unseen* data. Conversely, the ANN model shows the lowest more consistent results, maintaining an MAE range of 2-3% in the *unseen* dataset. The consistent MAE suggests that the ANN model’s superior generalization capability, less influenced by the training data. These observations suggest two crucial points in modelling for chemical reaction systems where the reliability and predictability of models are paramount. The first point is the importance of not only high accuracy but also generalizability for deploying of model. The second point is consideration of model’s characteristics can be able to improve the model performance.

|  |
| --- |
| R2 gap between Case 2 and Case 1 in R2 of ANN model |
|  | *Test* data | *Unseen* data |
| Range of targetin data of XCH4 (%) | Data density (%) | Case 1 | Case 2 | ΔR2 | Data density (%) | Case 1 | Case 2 | ΔR2 |
| 0 ~ 100 | 100 | 0.98 | 0.99 | 0.01 | 100 | 0.98 | 0.99 | 0.01 |
| 0 ~ 20 | 19.2 | -0.63 | -0.40 | 0.24 | 12.0 | -1.74 | -0.87 | 0.87 |
| 20 ~ 40 | 26.3 | 0.28 | 0.77 | 0.49 | 31.5 | 0.39 | 0.82 | 0.43 |
| 40 ~ 60 | 19.4 | 0.79 | 0.83 | 0.04 | 26.0 | 0.86 | 0.90 | 0.05 |
| 60 ~ 80 | 14.9 | 0.97 | 0.98 | 0.02 | 17.4 | 0.91 | 0.90 | -0.01 |
| 80 ~ 100 | 20.1 | 0.98 | 0.95 | -0.03 | 13.1 | 0.77 | 0.72 | -0.05 |

**Table 2.** Model accuracy based on R2 of ANN model by specific range of target.

As shown in Figure 3, ANN model exhibits the notable decrease in MAE between case 1 and 2 for both the *test* and *unseen* datasets, while XGB and ETR show similar MAE values. The details of ANN model's accuracy are presented as coefficient of determination (R2) by various ranges of target (XCH4) and data density in Table 2. The R2 near 1 indicates high accuracy, whereas R2 near -1 signifies poor performance. Although the minor increase of R2 exhibits 0.01 in all range of target, the increase of R2 by the range of target occurs between 0.02 and over 0.5 excluding the range of 80% to 100%. The improved performance of ANN model is attributed to the use of PCA variables, which are derived from the reactor size and flowrate variables. This result suggests that our approach, which considers the interconnections among process variables in the chemical system, improves model training and emphasizes the importance of in-depth data understanding.

* 1. Conclusion

In this study, we aim to derive the insights for datamining in chemical process industry. We conducted machine learning based modeling of catalytic reaction, including the prediction of steam methane reforming reaction using operating variables. Based on the understanding about interaction in systems, (i.e., the correlation of operating variables in chemical reactions), we transformed data and introduced new PCA feature using dimension reduction method. Also, we prepare *test* and *unseen* datasets, then develop three model (i.e., ANN, XGB, ETR) and evaluate these models based on MAE. As a results ofthe validation, resulting models based on tree algorithms such as XGB and ETR may be susceptible to overfitting against *unseen* dataset. Our findings highlight that we consider for model deployment not only based on prediction accuracy, but also generalizability and susceptibility to overfitting. Also, the understanding of the system and data are significant within cycle of data mining process, the need for a balanced approach to modeling, paving the way for the development of more robust and reliable predictive models in the field of chemical engineering.

References

Glassey, J., & Von Stosch, M., 2018. "Hybrid modeling in process industries," CRC Press.

Shearer, C., 2000. "The CRISP-DM model: the new blueprint for data mining," Journal of Data Warehousing, 5(4), 13-22.

Kim, S., 2017. "MATLAB Deep Learning With Machine Learning, Neural Networks and Artificial Intelligence," 1st ed.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. "Dropout: a simple way to prevent neural networks from overfitting," J. Mach. Learn. Res., 15, 1929–1958.

Wu, Z., Ramsundar, B., Feinberg, E. N., Gomes, J., Geniesse, C., Pappu, A. S., Leswing, K., Pande, V., 2018. "Chem. Sci.," 9, 513–530.

Mine, S., Takao, M., Yamaguchi, T., Toyao, T., Maeno, Z., Siddiki, S. M. A. H., Takigawa, I., 2021. "nalysis of updated literature data up to 2019 on the oxidative coupling of methane using an extrapolative machine‐learning method to identify novel catalysts," ChemCatChem, 13(16)

Oliveira, E. L.G., Grande, C.A., Rodrigues, A.E., 2009. "Steam methane reforming in a Ni/Al2O3 catalyst: kinetics and diffusional limitations in extrudates," The Canadian Journal of Chemical Engineering, 87, 945–956.

Goodfellow, I., Bengio, Y., Courville, A., 2016. "Deep Learning," MIT Press.

Chen, T., Guestrin, C., 2016. "Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining," 785–794.

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., Zheng, X., 2016. "12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16)."

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, É., 2011. "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, 12,

Raschka, S., 2018. "Model evaluation, model selection, and algorithm selection in machine learning."

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580.