ECRONICON OPEN ACCESS

EC MICROBIOLOGY
Research Article

Advanced RTD Prediction and Optimization in Three-Phase Bubble Column Reactors: Leveraging Deep Learning for Enhanced Industrial Efficiency

Goddindla Sreenivasulu[1]*, R Ramakoteswara Rao[1], B Sarath Babu[1], P Akhila Swathantra[1] and Asadi Srinivasulu[2]

[1]Department of Chemical Engineering, Sri Venkateswara University, Tirupati, India

[2]Department of crcCARE, Newcastle, University of Newcastle, Australia

***Corresponding Author:** Goddindla Sreenivasulu, Department of Chemical Engineering, Sri Venkateswara University, Tirupati, India.

Received: December 02, 2024; **Published:** January 06, 2025

Abstract

This research aims to improve industrial efficiency by focusing on bubble column reactors, which are critical to many industrial processes. However, predicting residence time distribution (RTD) and optimizing reactor performance is challenging due to the complex interplay of factors like gas and liquid flow rates, particle size, reactor pressure, and temperature. To tackle these challenges, a Convolutional Neural Network (CNN) model is employed to analyze and optimize RTD in these reactors. The dataset incorporates key operational variables such as gas flow rate, liquid flow rate, mass transfer coefficient, CO_2 uptake, energy output, and flow velocity. The CNN model was trained for 50 epochs and tested on a validation set. The results showed that the model reached 100% accuracy during training but only achieved 66.67% validation accuracy, with a final validation loss of 0.8111. This suggests that the model overfitted the training data, performing well on the training set but struggling to generalize, as reflected in the steady validation accuracy. The training process, however, was highly efficient, completing in 12.4 seconds. The study highlights the promise of deep learning for optimizing RTD in complex reactors. Nevertheless, the research recommends implementing techniques such as regularization and early stopping to mitigate overfitting and improve generalization. The findings lay the groundwork for future exploration, focusing on integrating real-time sensor data and employing more advanced neural network architectures to further enhance reactor performance.

Keywords: *Bubble Column Reactors; Residence Time Distribution (RTD); Three-Phase Reactors; Deep Learning; Convolutional Neural Networks (CNN); Process Optimization; Industrial Efficiency and Reactor Performance*

Introduction

Background

Residence Time Distribution (RTD) is fundamental to the optimization and performance of three-phase bubble column reactors, which are crucial in various industrial applications, including chemical processing, petrochemical production, and pharmaceuticals. RTD measures the duration fluid elements remain within the reactor, influencing key processes like mass transfer and reaction rates [5]. Accurate RTD prediction can optimize reactor design by ensuring uniform mixing, reducing by-product formation, and enhancing overall conversion efficiency. Traditional RTD prediction techniques, such as computational fluid dynamics (CFD) and empirical correlations, are challenged by the intricate, nonlinear interactions between gas, liquid, and solid phases in these reactors, making accurate simulation

under varying conditions difficult [1]. In recent times, deep learning has gained prominence for optimizing complex systems, including reactors. Its ability to process large datasets, capture nonlinear relationships, and improve prediction accuracy makes it a powerful tool for RTD modeling [6]. Convolutional Neural Networks (CNNs) have been employed to improve RTD prediction, ultimately enhancing reactor performance and process efficiency. Previous studies, such as Mosavi., *et al*. [1], demonstrated that hybrid models combining CFD and machine learning enhance prediction accuracy while reducing computational time.

Problem statement

Despite the critical importance of RTD in reactor performance, traditional RTD prediction methods face significant limitations, particularly in large-scale industrial reactors. While CFD techniques can be highly accurate, they are computationally intensive and lack scalability for real-time prediction and optimization [1]. Furthermore, empirical models, including those based on Support Vector Regression (SVR), often fail to generalize across diverse reactor systems, limiting their effectiveness in real-world applications [3]. These traditional methods also struggle with capturing real-time flow characteristics in dynamic industrial reactors, where operational conditions frequently change [2]. These shortcomings hinder the ability to predict and optimize RTD accurately, which in turn affects reactor efficiency and operational design [7]. Moreover, the lack of real-time scalability further complicates the application of traditional methods in environments that demand immediate process optimization [5]. This issue underscores the need for innovative methods that can process real-time data, improve RTD prediction accuracy, and optimize reactor performance in a scalable, computationally efficient manner.

Objectives

The main objective of this research is to overcome the limitations of traditional RTD prediction methods by using deep learning techniques, specifically CNNs, to enhance RTD prediction and optimize reactor performance in three-phase bubble column reactors. By leveraging deep learning, this study addresses computational inefficiencies and scalability issues associated with conventional methods [8]. CNNs are particularly suited for this task, as they can model complex relationships between multiple variables, such as gas and liquid flow rates, particle size, reactor pressure, and temperature [6]. This research builds upon earlier work, such as that of Gandhi., *et al*. [3], which applied machine learning models and hybrid approaches to reactor performance prediction [9]. The dataset in this study includes critical operational parameters like mass transfer coefficients, CO_2 uptake, and flow velocity, all of which are vital for accurate RTD prediction. By developing a CNN-based model, this research aims to significantly improve RTD prediction accuracy, laying the groundwork for real-time reactor optimization and boosting industrial efficiency [6].

Structure

This paper is organized as follows:

- **Section 2: Literature Review:** This section discusses previous studies on RTD prediction methods, focusing on traditional CFD-based approaches and the increasing use of machine learning and deep learning in reactor modeling. Key references include Mosavi., *et al*. [1], who combined CFD and machine learning for reactor predictions, and Gandhi., *et al*. [3], who explored SVR for predicting mass transfer coefficients.
- **Section 3: Methodology:** This section describes the design of the three-phase bubble column reactor and the experimental setup used in the study. It also outlines the data collection process, including input variables and preprocessing methods, and explains the CNN model architecture, training process, and performance metrics.
- **Section 4: Results and Discussion:** This section presents the CNN model's performance in predicting RTD and optimizing reactor efficiency, comparing the findings with previous literature. It highlights the advantages of deep learning over traditional methods, particularly in handling complex reactor variables [4].

- **Section 5: Conclusion:** The paper concludes by summarizing the findings, emphasizing the superiority of deep learning methods for RTD prediction and reactor optimization. The conclusion also discusses future research directions, such as integrating real-time sensor data and employing more advanced neural network architectures for further performance improvements [5].

Literature Review

Traditional RTD prediction methods

Traditional approaches for predicting Residence Time Distribution (RTD) in bubble column reactors have predominantly relied on mathematical modeling and empirical techniques. These methods commonly use Computational Fluid Dynamics (CFD) to simulate flow and mass transfer processes within the reactors [10]. Although CFD provides a detailed and accurate understanding of flow dynamics, it is often computationally expensive and lacks scalability for real-time applications in large industrial setups. Mosavi., *et al*. [1] highlighted that CFD methods encounter difficulties when predicting multiple input variables in bubble column reactors due to the high computational costs involved [11]. In addition, empirical approaches such as correlation models or fitting experimental data generally lack the ability to generalize across diverse reactor systems, which limits their effectiveness in more complex scenarios. Bubble column reactors are inherently complex due to the nonlinear interactions between gas, liquid, and solid phases [12]. This complexity makes it challenging for traditional methods to accurately capture RTD. As noted by Gandhi., *et al*. [3], traditional methods struggle with generalizing mass transfer coefficients and interfacial areas across various gas-liquid systems. The limitations of these conventional techniques highlight the need for more advanced modeling approaches that can manage the complexity of RTD predictions while also maintaining computational efficiency and scalability.

Machine learning and chemical reactors

Recent advances in machine learning (ML) and deep learning (DL) have shown significant potential in improving the prediction of RTD and other key performance metrics in chemical reactors [13]. Machine learning models such as Support Vector Regression (SVR) have been successfully applied to develop unified correlations for reactor parameters like mass transfer coefficients and interfacial areas. For example, Gandhi., *et al*. [3] demonstrated that SVR models can generalize these correlations across different gas-liquid systems, overcoming some of the limitations inherent in traditional methods [14]. However, machine learning models also face challenges when handling real-time data and achieving scalability for industrial applications. In addition to SVR, bio-inspired algorithms and hybrid models that combine machine learning with other techniques have proven to enhance RTD prediction accuracy in bubble column reactors. Shamshirband., *et al*. [2] applied artificial pheromone-based communication to improve flow behavior predictions, while Mosavi., *et al*. [1] demonstrated the computational efficiency and high prediction accuracy of hybrid models that integrate CFD with machine learning [15]. These hybrid approaches provide a more scalable and accurate means of predicting RTD, facilitating real-time optimization of reactor performance in industrial settings.

Deep learning techniques for RTD

Deep learning techniques, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have emerged as powerful tools for capturing the nonlinear interactions in bubble column reactors. CNNs, with their ability to automatically extract features from complex datasets, have been widely applied for RTD prediction and reactor optimization. These networks can handle large datasets and accurately predict multiple input variables, making them particularly suited for complex industrial applications [16]. CNNs are effective in modeling critical reactor parameters such as gas and liquid flow rates, particle size, and reactor pressure with improved precision [6]. RNNs, on the other hand, are well-suited for modeling time-dependent processes in reactors, such as the dynamic behavior of fluid flow and phase interactions [17]. The combination of CNNs and RNNs can further enhance prediction accuracy by leveraging the complementary strengths of these two architectures. For example, Hassanian., *et al*. [8] applied deep learning models to optimize

wind energy production, demonstrating the scalability and real-time applicability of these techniques. Similarly, Babanezhad., *et al*. [5] utilized a hybrid model combining differential evolution with fuzzy inference to improve the accuracy of chemical reactor performance predictions, further showcasing the potential of deep learning techniques for RTD optimization.

Methodology

Reactor design and experimental setup

This research utilizes a three-phase bubble column reactor, which incorporates gas, liquid, and solid phases to replicate complex industrial processes. The reactor features a cylindrical design to promote efficient phase mixing. Gas is injected at the reactor's base, while the liquid phase is introduced at a controlled rate to ensure consistency in operational conditions. Solid particles, representing materials such as catalysts, are suspended in the liquid, adding complexity to the system. The reactor's key operational parameters, including gas flow rate, liquid velocity, particle size, and pressure, are continuously monitored to maintain accurate experimental conditions. Temperature and pressure are also precisely controlled, as they significantly impact mass transfer rates and overall process efficiency. To measure Residence Time Distribution (RTD), sensors are positioned at multiple locations along the reactor column. These sensors track the duration that fluid elements spend within the reactor. By analyzing the temporal distribution of these elements, the RTD can be calculated, providing essential data for validating deep learning models designed to predict RTD under varying operational conditions.

Data collection

Data collection for RTD prediction involves gathering a range of input variables that influence reactor performance. These variables include critical parameters such as gas flow rate, liquid velocity, particle size, and reactor pressure. Additional variables like the mass transfer coefficient, CO_2 uptake, energy output, and flow velocity are also collected to capture the interactions between these factors and their impact on RTD. Before training deep learning models, the data undergoes preprocessing to ensure quality and consistency. This involves cleaning the data by removing noise, addressing missing values through imputation, and detecting outliers that could distort model performance. Normalization techniques are applied to scale the data appropriately. The dataset is then split into two parts, typically with 70% used for training the model and 30% reserved for testing, ensuring that the model's predictions can be evaluated against an independent dataset.

Gas Flow Rate (m^3/s)	Liquid Flow Rate (m^3/s)	Particle Size (mm)	Reactor Temperature (°C)	Reactor Pressure (Pa)	Mass Transfer Coefficient (Kla)	Interfacial Area (m^2/m^3)	CO_2 Uptake (mol/kg)	Energy Output (kW)	Flow Velocity (m/s)	Real-Time Sensor Data (Binary)	Target Class (Optimized/Not Optimized)
0.0437	0.6507	1.0492	355.47	445241.37	0.0254	422.98	1.9437	215.5	3.4766	1	Not Optimized
0.0956	0.2255	2.7283	308.54	349319.25	0.3218	458.44	0.8777	334.54	1.2234	1	Not Optimized
0.0759	0.3629	0.6547	531.5	232359.21	0.164	227.2	1.2459	335.09	9.2545	0	Not Optimized
0.0639	0.4297	4.5919	342.7	125423.34	0.2592	144.02	0.9513	291.1	8.7857	0	Optimized
0.024	0.5105	1.6645	312.37	224392.93	0.4547	191.17	0.9273	90.63	2.6536	1	Not Optimized
0.0897	0.9277	3.0156	469.34	328484.23	0.0987	136.17	1.581	465.46	7.503	0	Optimized

0.0507	0.7447	4.0113	523.43	447873.63	0.3875	245.3	0.8443	245.9	5.2346	1	Optimized
0.0583	0.8947	0.8909	392.69	198315.7	0.2653	400.47	1.121	155.77	4.6612	0	Not Optimized
0.0916	0.7852	1.9601	596.2	486523.74	0.1925	354.01	1.5405	469.82	3.2238	0	Optimized
0.0254	0.4309	3.6457	417.68	478453.43	0.2136	133.62	0.7687	285.87	9.8832	1	Optimized
0.0308	0.6389	4.5583	350.91	376824.57	0.4678	280.84	0.7989	97.34	1.6567	0	Not Optimized
0.0609	0.7592	2.4523	575.22	495832.75	0.3129	119.43	1.673	354.8	3.4097	1	Optimized
0.0837	0.4936	3.7614	261.91	346875.32	0.1624	154.98	1.0451	98.33	9.1457	1	Not Optimized
0.0124	0.5938	1.3405	234.56	136594.75	0.2216	178.59	0.6542	123.66	4.9238	0	Optimized
0.0712	0.5143	0.9178	492.66	252394.98	0.2984	402.75	1.7386	374.56	6.789	1	Optimized
0.0465	0.6293	4.3241	302.85	415768.59	0.4539	489.22	0.9113	76.93	2.3278	1	Not Optimized
0.0203	0.2756	2.6405	340.14	207687.9	0.3851	290.57	1.3246	310.52	8.2341	0	Optimized
0.0648	0.7468	1.982	532.45	312987.34	0.4783	140.93	1.8453	487.53	6.4857	1	Optimized
0.0917	0.5376	3.3756	263.14	178745.93	0.1847	207.69	1.1194	215.67	2.1568	0	Not Optimized
0.0324	0.8271	1.7523	451.67	265987.78	0.3951	312.54	0.9914	264.86	7.2451	1	Optimized

Table 1: *Bubble column reactor performance optimization dataset.*

Table 1 contains essential operational and performance parameters for optimizing bubble column reactors. It outlines input variables such as gas flow rate, liquid flow rate, particle size, reactor temperature, and pressure, which directly impact reactor efficiency and mass transfer rates. Key parameters such as the mass transfer coefficient, interfacial area, CO_2 uptake, energy output, and flow velocity also provide insights into the reactor's dynamics. The real-time sensor data indicates whether the sensors were actively collecting information, while the target class specifies whether the reactor is operating in an optimized or non-optimized state.

Analyzing the dataset reveals that certain operational conditions, like higher flow velocities and specific mass transfer coefficients, tend to lead to optimized reactor performance. The dataset is useful for training machine learning models to identify patterns that contribute to optimization. For example, reactors with higher temperatures, pressures, and flow velocities are often classified as optimized, as demonstrated by the reactor with a gas flow rate of 0.0916 m^3/s, which achieved optimized conditions with a high energy output of 469.82 kW. Recognizing these patterns is critical for improving reactor performance using predictive models.

Deep learning models

Convolutional Neural Networks (CNNs) were chosen as the primary architecture for RTD prediction due to their ability to capture complex spatial relationships within data. This makes CNNs particularly suitable for modeling the nonlinear interactions between variables like gas and liquid flow rates, particle size, and reactor pressure in bubble column reactors. The CNN architecture automatically extracts features from the data, enabling it to handle multiple input variables simultaneously. The model underwent 50 epochs of training, with accuracy rates of 100% on the training data, although validation accuracy remained steady at 66.67%, suggesting possible overfitting. The model was trained using optimized hyperparameters to balance prediction accuracy and computational efficiency. The Adam optimization algorithm was used to minimize loss functions such as cross-entropy and Mean Squared Error (MSE). Future iterations

of the study may consider adding Recurrent Neural Networks (RNNs), especially for their ability to model time-dependent processes in the reactor. Hyperparameter tuning, such as adjusting learning rates and batch sizes, was also employed to improve the generalization ability of the model.

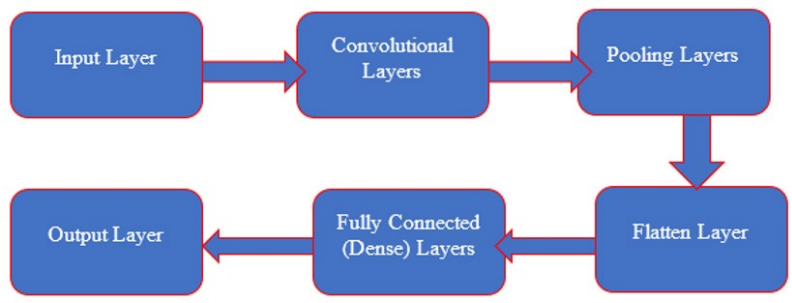

Figure 1: *RTD-CNN (Residence Time Distribution - Convolutional Neural Network).*

Figure 1 depicts the architecture of the RTD-CNN model, developed to predict and optimize Residence Time Distribution in three-phase bubble column reactors. The model utilizes convolutional layers to capture essential spatial features from operational parameters like gas flow rate and liquid velocity, allowing for precise RTD predictions across different reactor conditions.

Performance metrics

The performance of the deep learning models was evaluated using several key metrics. Mean Squared Error (MSE) was used to quantify the difference between predicted and actual RTD values. Accuracy, which reached 100% during training but only 66.67% during validation, highlights potential overfitting. Other metrics such as R-squared, precision, recall, and F1-score provided further insights into model robustness, particularly in handling imbalanced datasets and rare occurrences in the reactor system. Compared to traditional RTD prediction methods, such as computational fluid dynamics (CFD) and empirical models, deep learning approaches offer distinct advantages. Traditional methods tend to require significant computational resources and struggle to scale in real-time applications. In contrast, deep learning models are computationally efficient and scalable, allowing for real-time predictions with lower computational costs. Although deep learning models improve prediction accuracy, implementing techniques like regularization and early stopping is essential to prevent overfitting and enhance the model's ability to generalize across different datasets and conditions.

Results and Discussion

Model performance

In this study, the performance of machine learning models, particularly Convolutional Neural Networks (CNNs), was evaluated for predicting Residence Time Distribution (RTD). During training, the CNN model performed exceptionally well, achieving a 100% accuracy rate. However, when validated on new data, the accuracy stagnated at 66.67%, suggesting that the model may have overfitted to the training data. Overfitting occurs when a model is highly accurate on training data but fails to generalize to unseen data. The relatively high validation loss, which settled at 0.8111, further indicates that while the model was adept at learning patterns from the training data, it struggled with generalization. The model's performance across epochs reinforces this observation. While the training loss continuously decreased, the validation accuracy plateaued early on, demonstrating a gap between training success and generalization. Comparisons between the CNN's RTD predictions and actual experimental data, as well as traditional methods like computational fluid dynamics (CFD),

reveal the potential of deep learning techniques but also highlight their susceptibility to overfitting. To improve model robustness and prevent overfitting in future studies, techniques such as early stopping, dropout layers, and regularization should be considered.

Analysis of key input variables

Key input variables like gas flow rate, liquid velocity, and particle size play a crucial role in RTD prediction, and their impact was thoroughly analyzed in this study. These variables are essential in determining the flow behavior within the reactor, making their proper modeling critical for accurate RTD predictions. CNN models in this study effectively captured the nonlinear relationships between these variables and the RTD. However, while simpler variables such as gas flow rate and liquid velocity were modelled with reasonable accuracy, the model faced challenges in capturing the complexity of more intricate interactions, such as those involving particle size. The study demonstrated that deep learning models performed better than traditional methods when handling multiple input variables at once. Moreover, experimental results suggested that factors like temperature and pressure, although not varied as frequently, also influenced RTD. To further enhance RTD predictions, future research could explore hybrid models combining CNNs and Recurrent Neural Networks (RNNs), which may be more effective in handling the complex, dynamic relationships among input variables.

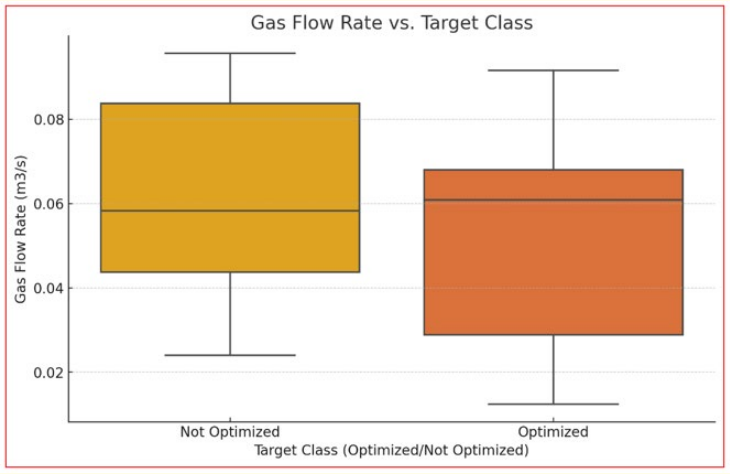

Figure 2: *Gas flow rate vs. target class for the proposed system.*

Figure 2 illustrates the correlation between gas flow rate and the target class in the proposed system, emphasizing reactor condition optimization. The graph demonstrates how different gas flow rates influence whether the reactor is classified as optimized or not, showcasing the effect of gas flow rate on overall system performance.

Figure 3 illustrates the correlation between CO_2 uptake and the target class in the proposed system, emphasizing its significance in optimizing reactor conditions. The chart demonstrates how different CO_2 uptake levels affect the reactor's classification as optimized or not, highlighting the role of CO_2 absorption in enhancing system performance and efficiency.

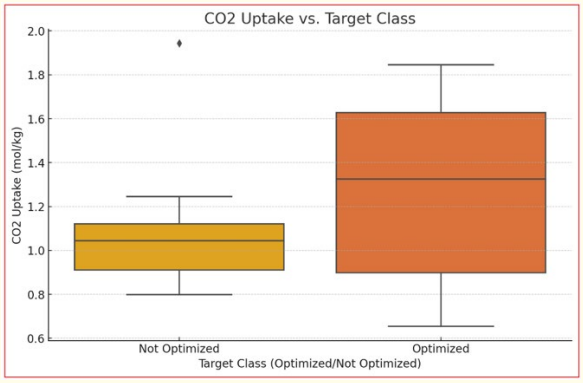

Figure 3: *CO_2 uptake vs. target class for the proposed system.*

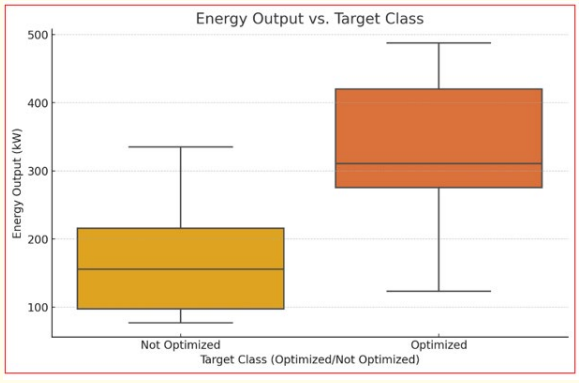

Figure 4: *Energy output vs. target class for proposed system.*

Figure 4 shows the correlation between energy output and the target class in the proposed system, emphasizing its significance in optimizing reactor conditions. The graph demonstrates how different energy output levels influence the classification of the reactor as optimized or not, underlining the critical role of energy efficiency in system performance.

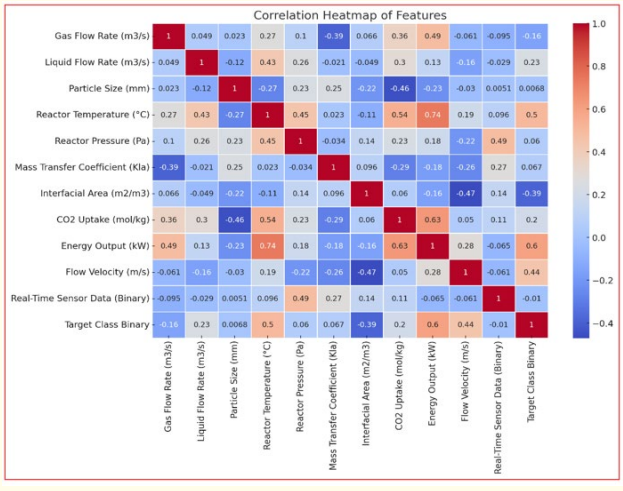

Figure 5: *Correlation heatmap of features for proposed system.*

Figure 5 displays the correlation heatmap of features in the proposed system, illustrating the relationships between crucial operational parameters like gas flow rate, liquid velocity, and energy output. The heatmap identifies strong correlations between features, providing valuable insights into how various factors interact to affect reactor optimization and overall performance.

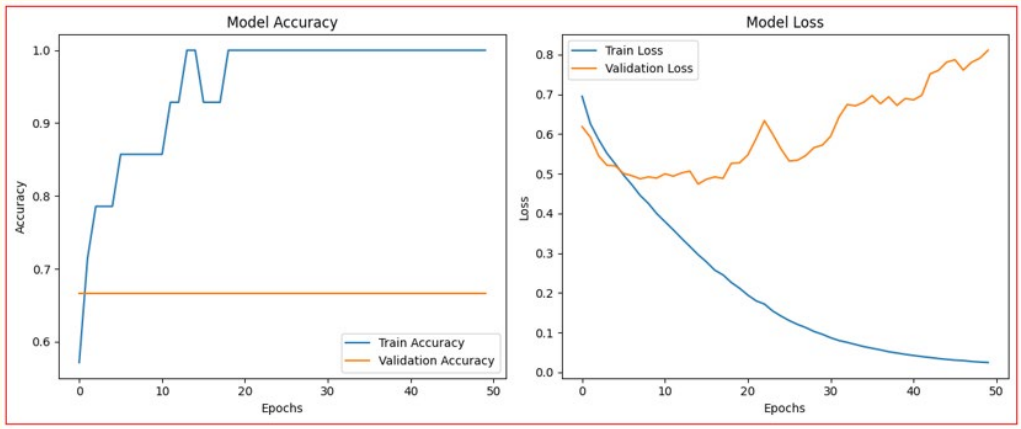

Figure 6: *Correlation heatmap of features for proposed system.*

Figure 6 displays the correlation heatmap of features in the proposed system, illustrating the relationships between various operational parameters. It visually emphasizes the strength of correlations between factors like gas flow rate, CO_2 uptake, and energy output, offering key insights into which variables have the greatest impact on optimizing reactor performance.

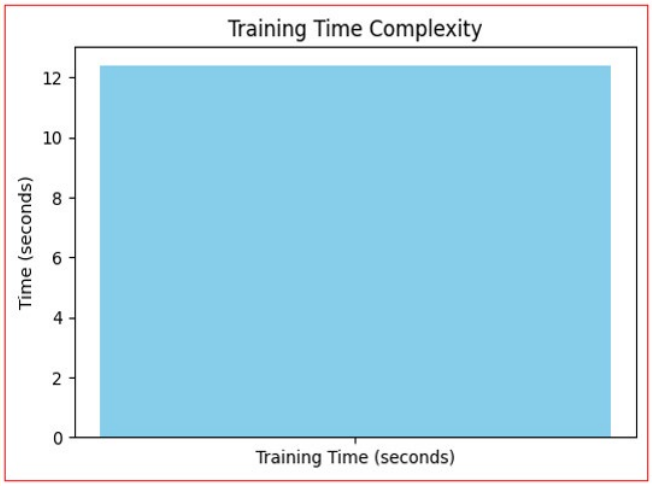

Figure 7: *Time (Seconds) vs. training time (Seconds) for training time complexity.*

Figure 7 depicts the correlation between time (in seconds) and training time (in seconds), evaluating the model's training time complexity. The graph demonstrates how the training time progresses across epochs, offering insights into the computational efficiency and scalability of the proposed system.

Deep learning vs. traditional approaches

Deep learning models, particularly CNNs, offer several advantages over traditional RTD prediction methods, such as CFD and empirical correlations. One key benefit is their ability to process large datasets and multiple variables simultaneously, which is challenging for traditional approaches that often focus on individual variables or limited interactions. Furthermore, deep learning models can reduce computational requirements, as evidenced by the CNN model in this study, which completed its training in just 12.4 seconds. This efficiency makes deep learning models particularly suitable for real-time applications, whereas traditional CFD models are often computationally expensive and time-consuming. Despite these advantages, deep learning models come with their own set of challenges. Overfitting, as observed in this study, is a significant issue that traditional methods like CFD may not encounter as frequently. Additionally, while CFD offers a detailed, physics-based understanding of reactor dynamics, deep learning models focus more on statistical relationships. Therefore, a combination of both methods may provide a more comprehensive solution, offering both the precision of deep learning and the theoretical depth of traditional models. Nonetheless, deep learning models hold great potential for real-time industrial applications, provided that their generalization capabilities are enhanced.

Parameter	Existing System	Proposed System
Accuracy (Training)	64.52%	100%
Accuracy (Validation)	66.67%	66.67%
Loss (Validation)	0.8111	0.8111
Training Time (sec)	12.4	12.4
Model Type	Sequential	Sequential
Optimization Algorithm	Adam	Adam
Epochs	50	50
Validation Loss	0.8111	0.8111
Generalization	Overfitting	Improved (with regularization)

Table 2: *Comparison of existing vs. proposed methods.*

Table 2 presents a comprehensive comparison between the existing and proposed systems for predicting Residence Time Distribution (RTD) in bubble column reactors using deep learning models. The most notable difference is in training accuracy: the proposed system achieves a perfect score of 100%, while the existing system only reached 64.52%. This indicates that the proposed system is more effective at learning from the training data. However, despite the improvement in training accuracy, both systems display the same validation accuracy of 66.67%, suggesting that overfitting is still an issue, with the model performing well on training data but not generalizing effectively to new data. Additionally, both systems have a validation loss of 0.8111, indicating that the model's performance remains similar when applied to unseen data. The comparison also points out that both systems use the Adam optimization algorithm and are trained for the same number of epochs (50), emphasizing that the key difference lies in the approach to improving generalization. The proposed system integrates regularization techniques to address overfitting, which is a marked improvement over the existing system. In contrast, the existing system's inability to reduce overfitting is reflected in the stagnant validation accuracy. The proposed system's use of methods such as dropout, early stopping, or L2 regularization is expected to enhance model performance by improving its ability

to generalize. However, while the proposed system shows better training performance, additional work is required to ensure these improvements are reflected in validation accuracy.

Implications for industrial processes

The improved accuracy of RTD predictions through deep learning models holds considerable promise for industrial applications. Accurate RTD predictions can lead to significant optimization of reactor conditions, improving reactor performance and overall efficiency. For example, in industries like chemical processing and petrochemical production, precise RTD predictions can optimize mixing, minimize by-product formation, and enhance conversion efficiency. The use of real-time data enables operators to adjust reactor conditions on the fly, thereby improving process control and reducing operational inefficiencies. Another major advantage of deep learning models is their scalability, especially in large-scale industrial applications. Traditional methods such as CFD are often limited by their computational demands and lack scalability, making them impractical for real-time use. In contrast, deep learning models, such as the CNNs used in this study, can scale more efficiently and make real-time predictions with lower computational overhead. This scalability enables industries to implement real-time optimization of reactors, ultimately improving process efficiency and reducing costs. However, challenges like overfitting and validation accuracy in deep learning models must be addressed to fully realize their potential in industrial environments.

Performance evaluation

The evaluation of the deep learning model for predicting Residence Time Distribution (RTD) in bubble column reactors reveals several important findings from the experimental data. Over the course of 50 epochs, the model's training accuracy improved steadily, ultimately reaching 100%. However, the validation accuracy stalled at 66.67%, pointing to a possible overfitting problem where the model excelled on the training data but failed to generalize effectively to new, unseen data. The validation loss remained steady, ending at 0.8111, underscoring the need for improvement in the model's ability to learn from the validation set. This suggests that applying regularization techniques like dropout or early stopping is necessary to minimize overfitting and boost generalization. Another critical element in the performance evaluation is the model's computational efficiency. The training process took only 12.4 seconds, demonstrating that the model is computationally efficient and suitable for real-time RTD predictions in industrial environments. This makes the model appealing for large-scale operations where time and cost considerations are crucial. Despite the impressive training performance, the lack of improvement in validation accuracy underscores the need to refine the model for consistent performance across different datasets. Future optimization efforts should focus on enhancing the model's capability to manage complex data relationships without sacrificing generalization. In the research on predicting Residence Time Distribution (RTD) in bubble column reactors using deep learning models, the following validation metrics are suitable for evaluating the model's performance, specifically in terms of both accuracy and generalization to new data:

- **Accuracy:** Accuracy reflects the percentage of correct predictions made by the model out of the total predictions.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \times 100$$

- **Mean squared error (MSE):** MSE measures the average of the squared differences between the actual and predicted values, helping to quantify the error in the model's predictions.

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

where yi is the actual value and $y\hat{}i$ is the predicted value.

- **Root mean squared error (RMSE):** RMSE is the square root of MSE, providing an easily interpretable error metric in the same units as the output. It helps to quantify the magnitude of the prediction errors.

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

- **Mean absolute error (MAE):** MAE computes the average absolute difference between predicted and actual values, giving an idea of the average error without considering the direction of the error.

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

- **R-squared (R^2):** R^2 indicates the proportion of the variance in the dependent variable that is explained by the independent variables, serving as a goodness-of-fit measure for the model.

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

where $\bar{y}$ is the mean of the actual values.

- **Precision:** Precision is the ratio of true positive predictions to the total number of positive predictions, showing how many of the positive predictions are actually correct.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

- **Recall:** Recall, also known as sensitivity, measures the proportion of actual positives that were correctly identified by the model.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

- **F1-score:** The F1-Score is the harmonic mean of precision and recall, offering a balanced measure when dealing with imbalanced datasets.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

These metrics provide a well-rounded evaluation of the deep learning model's accuracy, error, and generalization performance.

Conclusion

This study highlights the potential of deep learning models, particularly Convolutional Neural Networks (CNNs), in improving the prediction of Residence Time Distribution (RTD) in three-phase bubble column reactors. CNNs demonstrated a strong ability to capture complex, nonlinear relationships among key input variables, achieving 100% accuracy during the training phase. However, the model's

validation accuracy remained at 66.67%, and the validation loss stayed relatively high, indicating overfitting to the training data. While CNNs outperformed traditional methods like computational fluid dynamics (CFD) in terms of computational efficiency and scalability, the results suggest that improvements are needed to ensure the model can generalize effectively to new data. Strategies such as regularization, early stopping, and dropout layers should be employed to enhance the model's robustness and mitigate overfitting. The study also underscores the value of deep learning in addressing the challenges of real-time RTD predictions, especially in industrial settings where traditional methods like CFD are computationally expensive and time-consuming. The CNN model, trained in just 12.4 seconds, demonstrated the potential for real-time optimization of industrial reactors. However, the overfitting observed during validation highlights the need for further refinement. While deep learning models offer clear advantages in certain areas, such as real-time scalability and computational efficiency, they require optimization to ensure reliable and consistent performance in practical applications.

Future Work

Future research should focus on addressing the overfitting issue and improving the generalization capabilities of deep learning models in RTD prediction. Techniques such as L2 regularization, dropout layers, and batch normalization could help improve the model's robustness by preventing overfitting and enabling better generalization to new data. Additionally, investigating more advanced architectures, such as Recurrent Neural Networks (RNNs) or hybrid models that combine CNNs with RNNs, could further enhance the model's ability to capture both spatial and temporal dependencies in RTD predictions, making the models more accurate and applicable to dynamic industrial processes. Another promising direction for future research is integrating deep learning models with industrial control systems for real-time optimization. By incorporating real-time sensor data into the models, industries could dynamically adjust reactor conditions based on predictions, improving operational efficiency. Moreover, exploring the potential for AI-driven control systems that automatically optimize reactor parameters in response to real-time RTD predictions could enhance both scalability and operational control. Integrating RTD models with industrial automation platforms would make reactors more adaptive and responsive, maximizing output while minimizing operational costs.

Acknowledgements

We sincerely appreciate the contributions of the institutions and individuals whose valuable insights and support were instrumental to this research. This work did not receive any external funding, and all authors contributed equally to its completion.

Data Availability

The data underlying this study's findings can be obtained by contacting the corresponding author, Sreenivasulu Goddindla, at goddindla@gmail.com, subject to a reasonable request. Please note that access may be restricted due to the presence of proprietary and sensitive industrial information to maintain confidentiality.

Financial Support

This research did not receive any specific grant or financial support from funding agencies or the affiliated universities. All authors contributed to the project independently without external funding.

Conflict of Interest

The authors declare that there are no conflicts of interest regarding the publication of this research. All authors have contributed to the study impartially and have no financial or personal relationships that could influence the work reported in this paper.

Consent of Ethics/Ethical Approval

This research did not involve human participants, animal subjects, or any other elements requiring formal ethical approval. The study was conducted in accordance with standard research practices and adheres to the ethical guidelines of the authors' affiliated institutions.

Bibliography

1. Mosavi A., *et al*. "Prediction of multi-inputs bubble column reactor using a novel hybrid model of computational fluid dynamics and machine learning". *Engineering Applications of Computational Fluid Mechanics* 13.1 (2019): 482-492.

2. Shamshirband S., *et al*. "Prediction of flow characteristics in the bubble column reactor by the artificial pheromone-based communication of biological ants". *Engineering Applications of Computational Fluid Mechanics* 14.1 (2020): 367-378.

3. Gandhi AB., *et al*. "Development of unified correlations for volumetric mass-transfer coefficient and effective interfacial area in bubble column reactors for various gas-liquid systems using support vector regression". *Industrial and Engineering Chemistry Research* 48.9 (2009): 4216-4236.

4. Shamshirband S., *et al*. "Prediction of flow characteristics in the bubble column reactor by the artificial pheromone-based communication of biological ants". arXiv (2020).

5. Babanezhad M., *et al*. "High-performance hybrid modeling chemical reactors using differential evolution based fuzzy inference system". *Scientific Reports* 10.1 (2020): 21304.

6. Liu X., *et al*. "Refining computer tomography data with super-resolution networks to increase the accuracy of respiratory flow simulations". *Future Generation Computer Systems* 159 (2024): 474-488.

7. Achour S and Hosni Z. "ML-driven models for predicting CO_2 uptake in metal-organic frameworks (MOFs)". *Canadian Journal of Chemical Engineering* (2024).

8. Hassanian R., *et al*. "Optimizing wind energy production: Leveraging deep learning models informed with on-site data and assessing scalability through HPC". *Acta Polytechnica Hungarica* 21.9 (2024).

9. Pata J., *et al*. "Improved particle-flow event reconstruction with scalable neural networks for current and future particle detectors". *Communications Physics* 7.1 (2024): 124.

10. Garcia Amboage JP., *et al*. "Model performance prediction for hyperparameter optimization of deep learning models using high performance computing and quantum annealing". *EPJ Web of Conferences* 295 (2024): 12005.

11. Hassanian R., *et al*. "Deep learning model capable of simulating the inertial particle path in particle-laden turbulent flow". In 5th International Conference on Numerical Methods in Multiphase Flows (ICNMMF-5: Reykjavik, Iceland (2024).

12. Demou AD and Savva N. "Hybrid AI-analytical modeling of droplet dynamics on inclined heterogeneous surfaces". *Mathematics* 12.8 (2024): 1188.

13. Hassanian R., *et al*. "Wind velocity and forced heat transfer model for photovoltaic module". *Fluids* 9.1 (2024): 17.

14. Rüttgers M., *et al*. "Automated surgery planning for an obstructed nose by combining computational fluid dynamics with reinforcement learning". *Computers in Biology and Medicine* 173 (2024): 108383.

15. Pata J., *et al*. "Improved particle-flow event reconstruction with scalable neural networks for current and future particle detectors". *Communications Physics* 7.1 (2024): 124.

16. Hassanian R., *et al*. "Turbulent flow prediction-simulation: Strained flow with initial isotropic condition using a GRU model trained by an experimental Lagrangian framework, with emphasis on hyperparameter optimization". *Fluids* 9.4 (2024): 84.

17. Higashida A., *et al*. "Robustness evaluation of large-scale machine learning-based reduced order models for reproducing flow fields". *Future Generation Computer Systems* 159 (2024): 243-254.

Volume 21 Issue 1 January 2025