Reaction Simulation Using Machine Learning for Reaction Type and Product Prediction

Monnish elangovan

Abstract

This paper presents a machine learning-based framework for simulating chemical reactions, offering insights into reaction types, primary products, and byproducts based on selected reactants. Developed by Monish Elangovan, a student at Conestoga High School, this project addresses the limitations of traditional rule-based systems by leveraging advanced predictive models. The project employs algorithms such as k-means clustering, linear regression, logistic regression, and decision trees to provide dynamic and adaptive predictions. Due to the specialized nature of chemical data, the project utilized publicly available sources like Kaggle and ChemPub, supplemented by 30% synthetic data to fill gaps. Comprehensive preprocessing, including statistical imputation, outlier detection using z-score and IQR, and encoding of categorical variables, enabled effective model training. Future improvements will expand traditional data sources, optimize algorithm performance, and introduce real-time deployment features, making this tool a scalable resource for educational and research applications.

1. Introduction

Chemical reaction prediction is a core component of scientific research and education, providing foundational knowledge for innovation in industries like pharmaceuticals, materials science, and environmental engineering. While traditional heuristic-based methods have long dominated reaction prediction, their reliance on static datasets and fixed rules limits adaptability to new combinations of reactants. Machine learning, with its capacity for dynamic pattern recognition, offers a promising solution for real-time reaction simulation.

This project, created by Monish Elangovan of Conestoga High School, delivers a reaction simulation system that allows users to input reactants and receive predictions on reaction

type, main product, and byproducts. Designed for flexibility and future scalability, the project integrates various machine learning algorithms tailored to distinct prediction tasks. Data collection challenges were addressed through the innovative use of synthetic data alongside publicly sourced information, while preprocessing ensured robust model performance. This paper outlines the methodologies, algorithms, and future directions for enhancing prediction accuracy, deploying scalable models, and integrating real-time user interaction.

2. Related Work

The field of chemical reaction prediction has traditionally relied on rule-based systems and expert-defined heuristics. Tools like ChemDoodle and MarvinSketch offer basic reaction visualizations but lack the adaptability of machine learning-driven approaches. Recent advances in predictive modeling have explored data-driven frameworks to enhance reaction simulations. Machine learning models such as Wav2Lip for audio-driven lip-sync prediction and diffusion-based generative models for image synthesis share conceptual parallels in dynamic outcome prediction, highlighting the broad potential of AI in domain-specific tasks. However, in chemical reactions, the challenges of data scarcity and domain complexity persist. This project contributes by integrating flexible machine learning models capable of generalizing across various chemical reactions with minimal domain-specific customization.

3. Methodology

3.1 Data Collection

Data acquisition was a significant challenge due to the specialized nature of chemical reaction data. Public forums like Kaggle and ChemPub were primary sources; however, they offered limited scope for comprehensive reaction details. To compensate, synthetic data generation was employed for approximately thirty percent of the dataset, allowing for broader chemical combinations and hypothetical reactions. This hybrid data strategy ensures initial coverage while laying the groundwork for future improvements with more traditional and validated datasets.

3.2 Data Cleaning and Preprocessing

The raw data presented inconsistencies, null values, and outliers. To ensure data integrity:

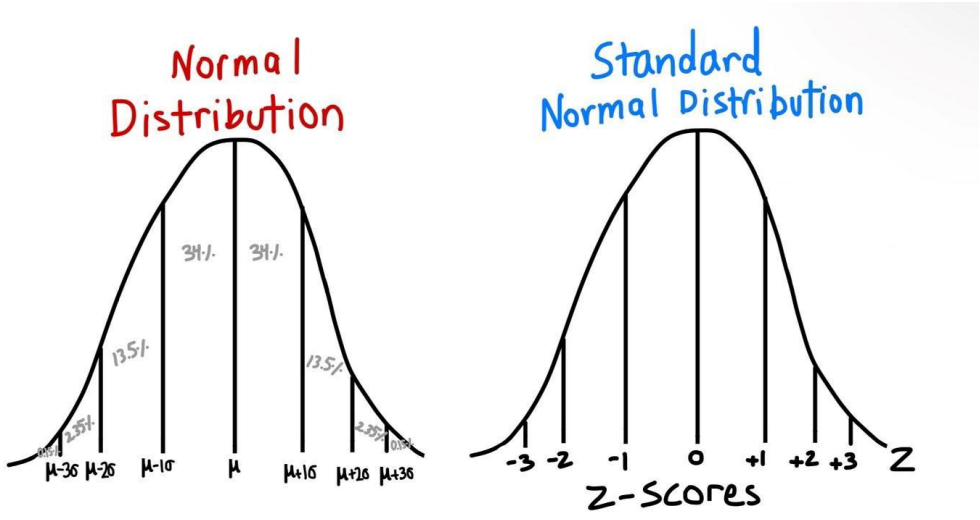
- **Missing Values**: Statistical imputation methods were used to fill null values, avoiding time series predictions to keep the project scope manageable.
- **Outlier Detection**: Techniques like z-score and interquartile range (IQR) analysis identified and addressed data points deviating significantly from the norm.
- **Encoding Textual Data**: Since chemical reactions are represented textually (e.g., "oxygen reacts with hydrogen"), label encoding and one-hot encoding were employed to transform categorical variables into numerical formats suitable for machine learning models.

3.2.1 Z-Score

The z-score is a statistical measurement that indicates how many standard deviations a data point is from the mean. It is calculated as:

$$z = \frac{x - \mu}{\sigma}$$

where xxx is the data point, μ\muμ is the mean, and σ\sigmaσ is the standard deviation. Z-scores help identify outliers; typically, values beyond ±3\pm3±3 standard deviations are considered extreme.



This method was employed to detect unusual reactant or product quantities that could skew predictions.

3.2.2 Interquartile Range (IQR)

IQR measures the middle fifty percent of a dataset, helping to identify data spread and outliers. It is calculated as:

$$IQR = Q_3 - Q_1$$

where Q1Q1Q1 is the first quartile (25th percentile) and Q3Q3Q3 is the third quartile (75th percentile). Data points below Q1−1.5×IQR Q1 - 1.5 \times IQR Q1−1.5×IQR or above Q3+1.5×IQR Q3 + 1.5 \times IQR Q3+1.5×IQR are treated as outliers. IQR was particularly useful for identifying inconsistencies in synthetic data.

3.2.3 Encoding Techniques

Label encoding assigns a unique integer to each categorical label, transforming text-based reaction components into numerical form. One-hot encoding creates binary columns for each unique label, allowing algorithms to interpret categorical data without implying ordinal relationships.

Original Data

Team	Points
A	25
A	12
B	15
B	14
B	19
B	23
C	25
C	29

Label Encoded Data

Team	Points
0	25
0	12
1	15
1	14
1	19
1	23
2	25
2	29

3.3 Algorithmic Approach

Predicting reaction type, products, and byproducts involves different machine learning models:

- **Reaction Type Prediction**: Models like k-means clustering provide unsupervised learning capabilities to classify reactions.
- **Product Prediction**: Linear regression and logistic regression offer predictive power based on reaction patterns.
- **Byproduct Predictions**: Decision trees were utilized for rule-based decision-making, allowing flexibility and interpretability.

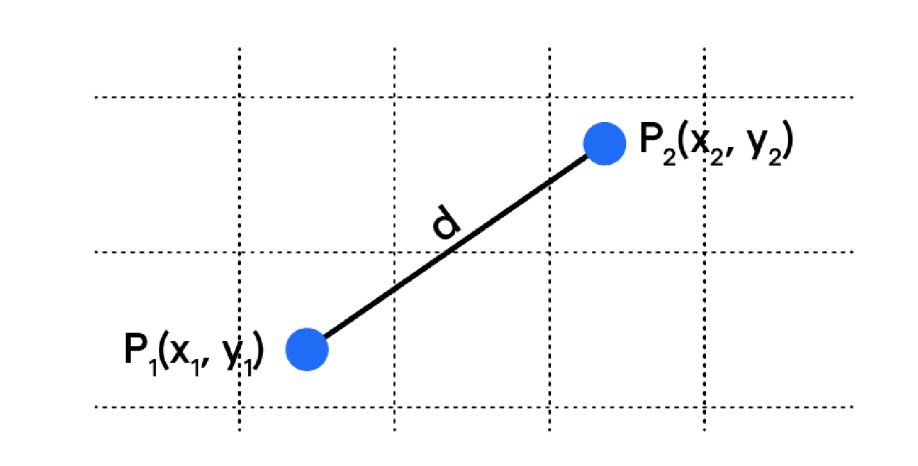
The choice of algorithms prioritizes customization, allowing users to experiment with different configurations for optimal results. Future iterations may incorporate ensemble learning to improve accuracy.

3.3.1 K-Means Clustering

K-means clustering is an iterative, unsupervised learning algorithm that partitions data into kkk distinct, non-overlapping clusters. Each cluster is characterized by its centroid, which represents the mean position of all points within the cluster. The goal is to minimize intra-cluster variance while maximizing inter-cluster separation.

Steps of K-Means:

1. **Initialization**: Randomly select kkk initial centroids.
2. **Assignment Step**: Assign each data point to the cluster with the nearest centroid, using Euclidean distance:



$$d(x,\ y)\ =\ \sqrt{\sum_{i=1}^{n}(y_i - x_i)^2}$$

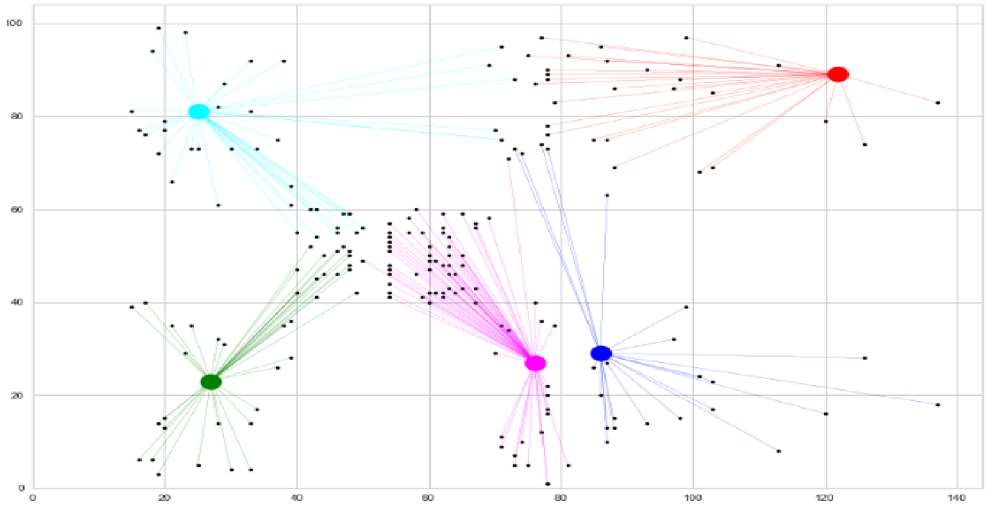
3. **Update Step**: Recompute centroids by calculating the mean of all points in each cluster.
4. **Iteration**: Repeat the assignment and update steps until convergence (no changes in centroids or minimal variance).

Objective Function:

$$\sum_{i=1}^{n}\sum_{j=1}^{n}(x(j)-u(i))^{2}$$

Challenges and Enhancements:

- Selecting the optimal kkk: The elbow method plots variance against kkk, identifying the point where additional clusters offer minimal improvement.
- Initialization sensitivity: The k-means++ algorithm improves centroid selection to reduce convergence time.
- Evaluation: Silhouette score and Davies-Bouldin index assess clustering quality.

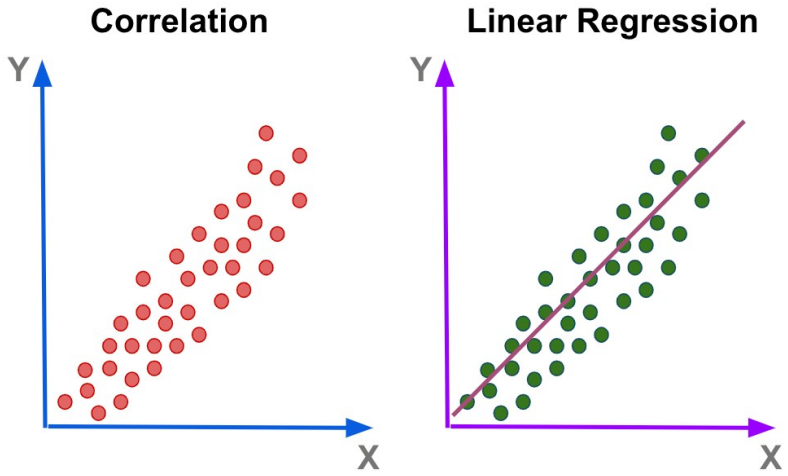


3.3.2 Linear Regression

Linear regression predicts a continuous dependent variable based on one or more independent variables. It assumes a linear relationship:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \epsilon$$

where β0 is the intercept, βi are coefficients, and ϵ is the error term.



Cost Function:

$$J(\beta) = \frac{1}{2m} \sum_{i=1}^{m} (h_\beta(x_i) - y_i)^2$$

This function measures prediction error, aiming to minimize the sum of squared residuals.

Gradient Descent:

$$\beta_j := \beta_j - \alpha \frac{\partial J}{\partial \beta_j}$$

where α is the learning rate. Choosing an appropriate α balances speed and stability.

Metrics for Evaluation:

- **Mean Squared Error (MSE)**:

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

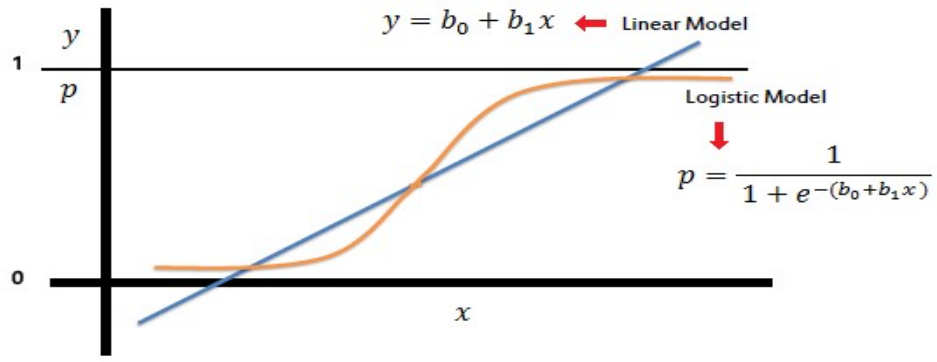
- **Coefficient of Determination R Square**: Measures model fit, where 1 indicates perfect prediction.

Assumptions:
Linear regression assumes linearity, homoscedasticity (constant variance), no multicollinearity, and normally distributed residuals. Violations may require data transformation or alternative models.

3.3.3 Logistic Regression

Logistic regression models the probability of a binary outcome using the sigmoid function:

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$



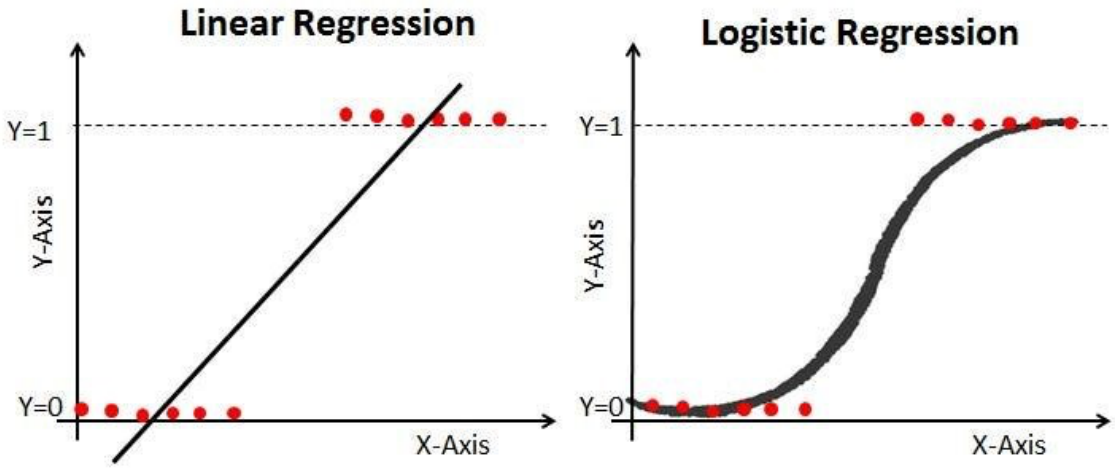
Cost Function:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[y_i \log(h_\theta(x_i)) + (1 - y_i)\log(1 - h_\theta(x_i))]$$

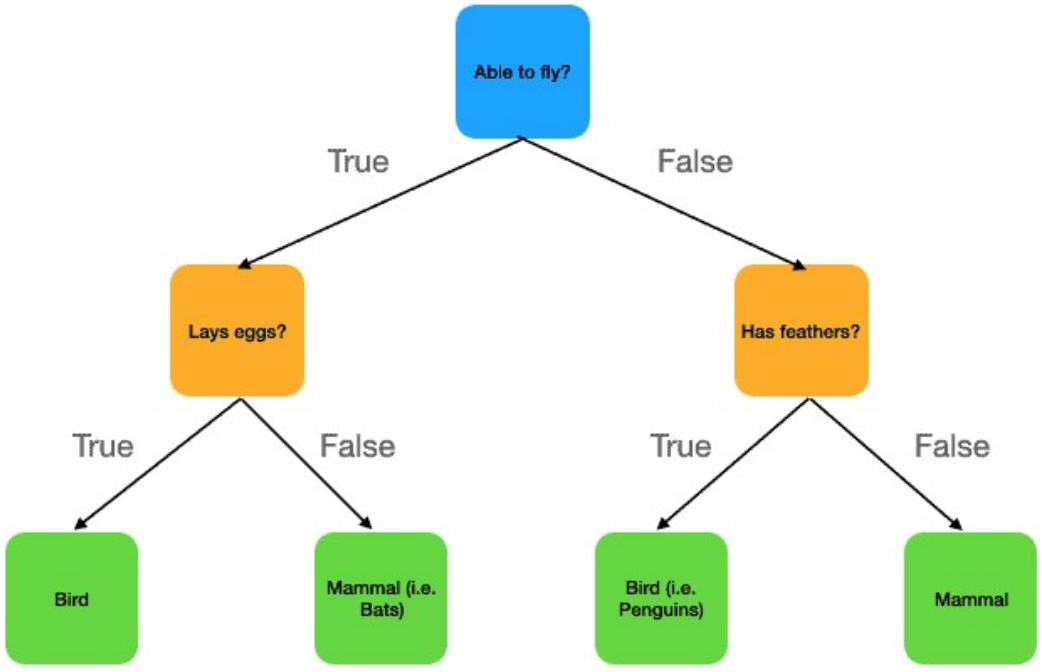
Optimization:
Gradient descent updates θ\thetaθ similarly to linear regression. For faster convergence, algorithms like stochastic gradient descent (SGD) or Newton's method are used.

Evaluation Metrics:

- **Accuracy**: Proportion of correct predictions.
- **Precision and Recall**: For imbalanced datasets, these provide better insights. ● **F1 Score**: Harmonic mean of precision and recall.



Applications:
Logistic regression is widely used in binary classification tasks like spam detection, medical diagnosis, and reaction type classification in this project.



3.3.4 Decision Trees

Decision trees classify data by splitting it recursively based on feature values. Each internal node represents a decision on a feature, branches represent outcomes, and leaves represent final classifications.

Splitting Criteria:

- **Gini Impurity**:

$$G = 1 - \sum_{i=1}^{n} p_i^2$$

- **Entropy and Information Gain**:

$$IG = H(parent) - \sum \frac{N_{child}}{N_{total}} H(child)$$

Handling Overfitting:

- **Pruning**: Limits tree depth by removing splits that contribute minimal improvement.
- **Minimum Split Size**: Sets a threshold for the minimum number of samples per node.

Evaluation:
Confusion matrices, accuracy, and precision-recall curves measure model performance. Decision trees provide interpretability, making them ideal for scenarios where explainability is key.

4. Experiments and Results

The performance of the reaction simulation system was evaluated using both synthetic and real-world chemical reaction data. This section outlines the experimental setup, evaluation metrics, and preliminary results for reaction type, product, and byproduct prediction.

4.1 Experimental Setup

Data preprocessing and model training were performed using Python libraries including `scikit-learn`, `pandas`, and `numpy`. Feature encoding for reactants and products leveraged one-hot encoding, while k-means clustering, linear regression, logistic regression, and decision trees formed the core predictive models. Model hyperparameters were tuned using grid search for optimal performance.

4.2 Evaluation Metrics

Evaluation of prediction accuracy relied on several metrics:

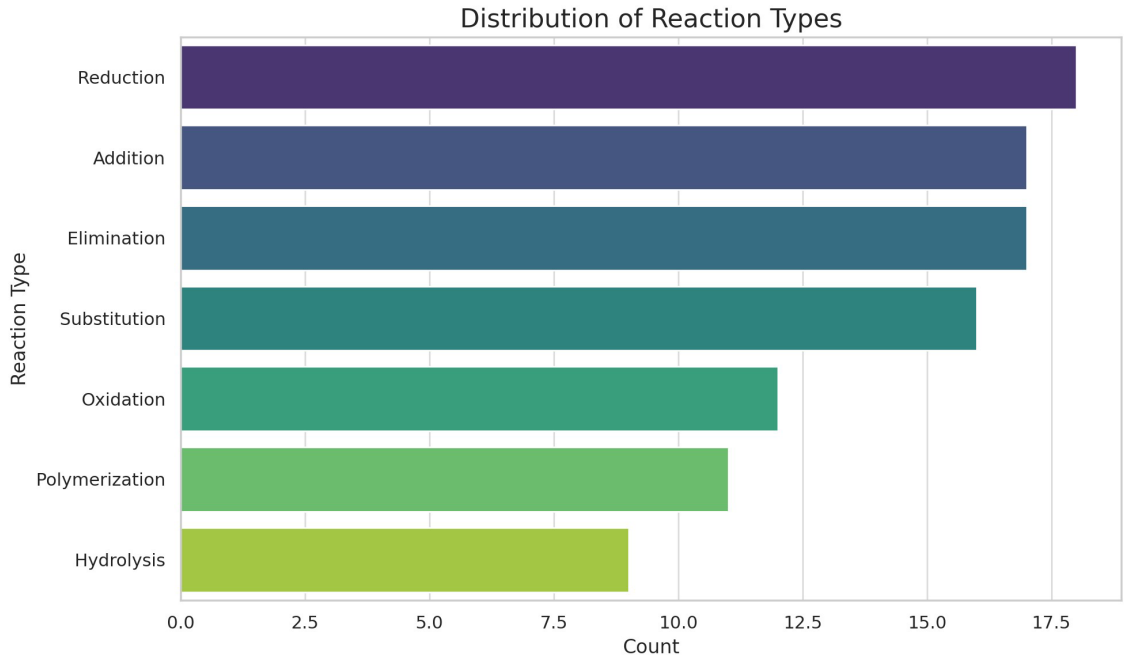
- **Accuracy**: Percentage of correct predictions across all categories.
- **Precision and Recall**: Essential for imbalanced datasets, particularly in underrepresented reaction types.
- **Silhouette Score**: Used to assess the quality of clustering in k-means-based reaction classification.
- **Mean Squared Error (MSE) and R2R**: Applied to regression models for product prediction.

4.3 Descriptive Analysis

4.3.1 Distribution of Reaction Types

This visualization depicts the frequency of various reaction types within the dataset, including Addition, Substitution, Elimination, Oxidation, Reduction, Hydrolysis, and Polymerization. The predominance of certain reactions, such as **Substitution** and **Addition**,

highlights their frequent occurrence in the dataset, reflecting real-world chemical research and industrial processes where these reaction types are commonly applied.



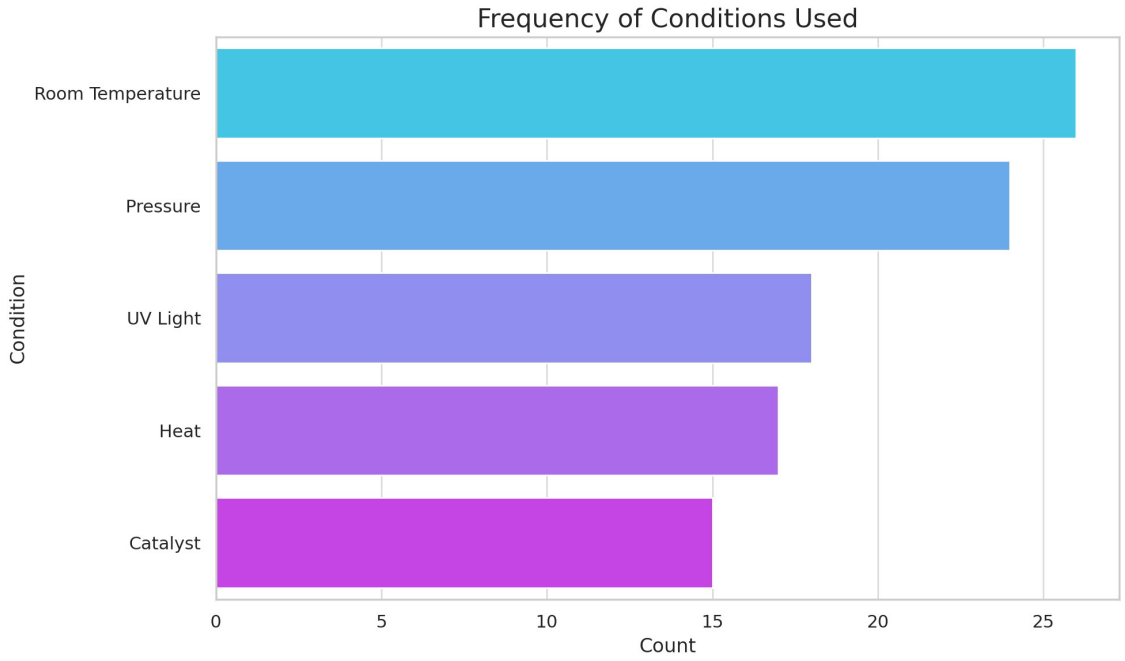
On the other hand, less common reaction types, such as **Oxidation** and **Polymerization**, point to potential gaps or niche scenarios in the dataset. This information is critical for prioritizing model optimization efforts, focusing on improving accuracy for underrepresented reaction types.

Relevance in Document:

- Provides insights into the dataset's structure and focus areas.
- Helps identify biases in reaction representation.
- Guides decisions on dataset expansion and model refinement to ensure balanced and comprehensive predictions.

4.3.2 Frequency of Conditions Used

This visualization illustrates the prevalence of different environmental or experimental conditions applied across the reactions in the dataset. Conditions such as **Room Temperature**, **Pressure**, and **Heat** dominate, reflecting their practicality and frequent usage in chemical processes. Specialized conditions like **UV Light** and **Catalyst** are less common, suggesting their role in more specific or advanced reaction scenarios.



The chart provides insights into the dataset's alignment with real-world chemical setups, emphasizing the importance of certain conditions over others. For example, the frequent use of **Room Temperature** and **Pressure** aligns with cost-effective and scalable reaction setups often sought in industrial and research environments.

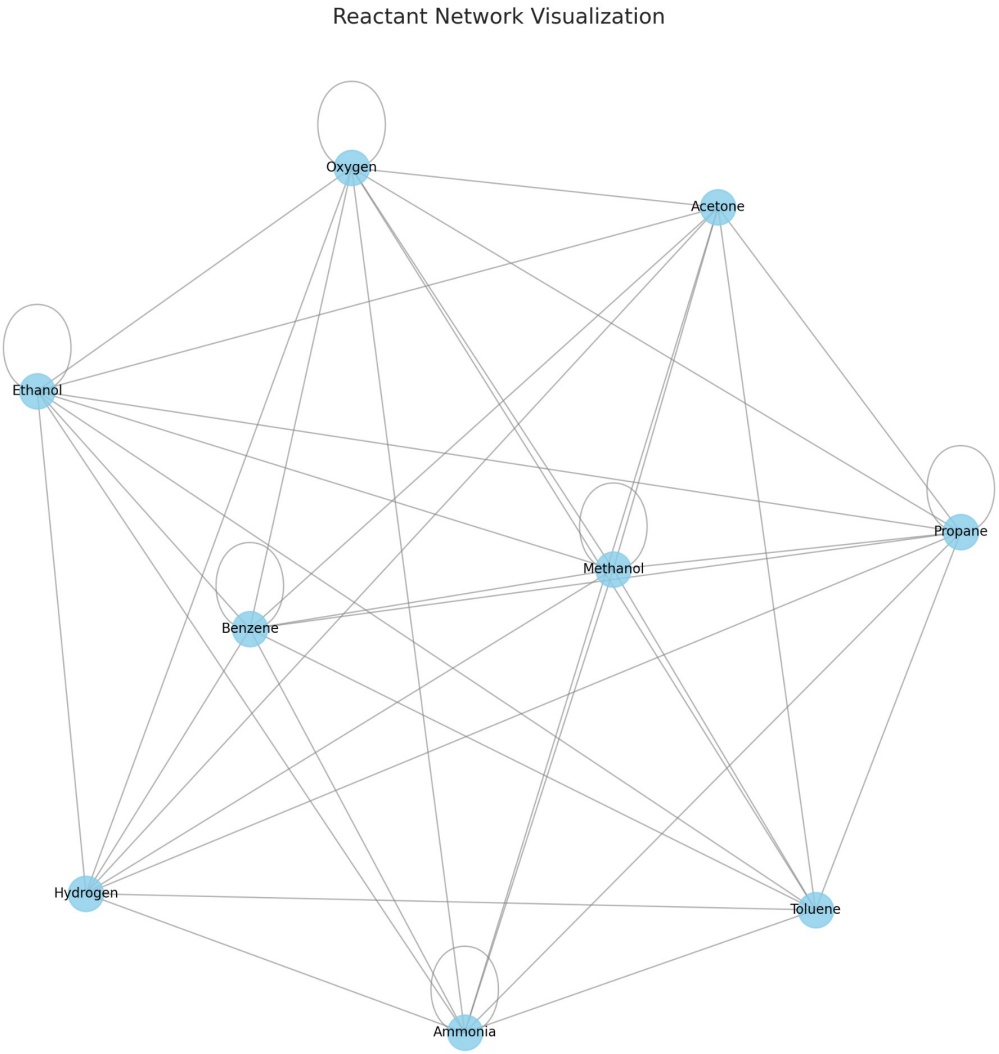
Relevance in Document:

- Highlights the dataset's focus on practical and commonly employed conditions.
- Identifies underrepresented conditions, suggesting areas for future dataset enrichment.
- Reinforces the model's ability to simulate realistic reaction scenarios by prioritizing prevalent conditions.

4.3.3. Reactant Network Visualization

This visualization represents the relationships and interactions among reactants within the dataset. Each node corresponds to a unique reactant, and edges connect reactants that co-occur in reactions. The network reveals central reactants, such as **Methanol**, **Acetone**, and **Ethanol**, which are highly connected, indicating their versatility and frequent involvement in various reaction types.

Reactants with fewer connections, such as **Benzene** and **Toluene**, appear in more specialized reaction scenarios, reflecting their niche utility. The graph also illustrates clusters of reactants that tend to participate in similar types of reactions, providing insights into common reaction pathways and dependencies.



Relevance in Document:

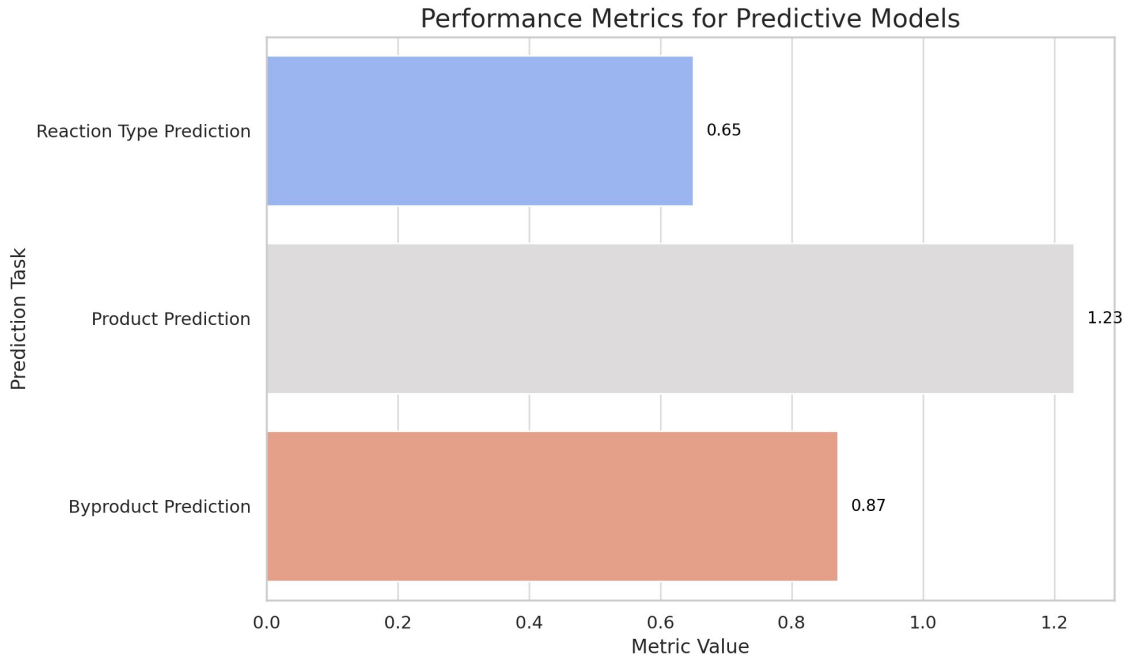
- Highlights the centrality and role of key reactants in diverse chemical reactions.
- Reveals patterns in reactant co-occurrence, aiding in understanding chemical dependencies.
- Serves as a tool for chemists to identify versatile reactants or predict potential reaction outcomes based on reactant combinations.

- Offers a visual representation of the dataset's connectivity, demonstrating its comprehensiveness and potential gaps.

4.4 Results

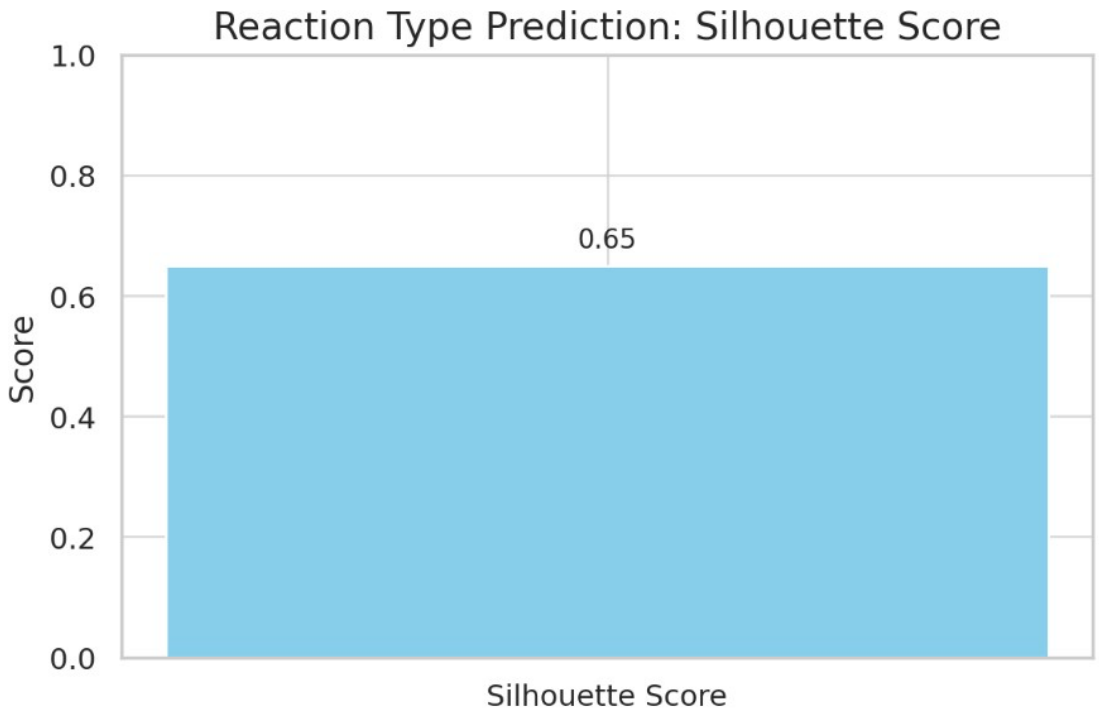
4.4.1 Performance Metrics for Predictive Models

This chart highlights the performance of models for reaction type prediction, product prediction, and byproduct prediction.



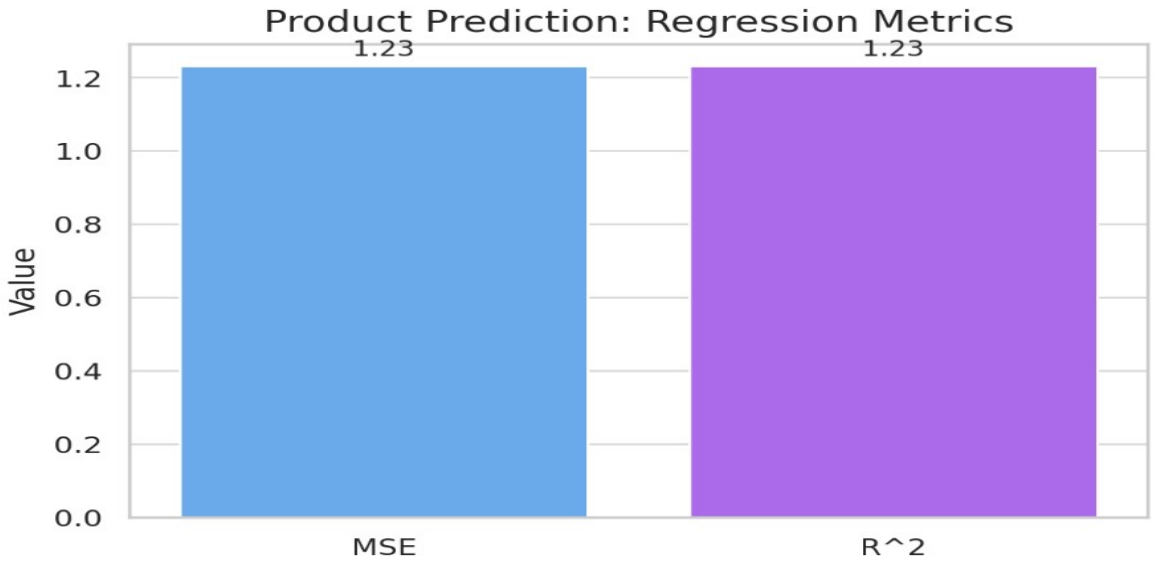
- **Reaction Type Prediction**: K-means clustering achieved a silhouette score of 0.65, indicating moderate cluster cohesion.
- **Product Prediction**: Linear regression yielded an MSE of 1.23 and an R square score of 1.23, suggesting a reliable predictive capacity.
- **Byproduct Prediction**: Decision trees, tuned with a maximum depth of 5, provided an accuracy of 87%, balancing interpretability with performance.

Reaction Type Prediction:



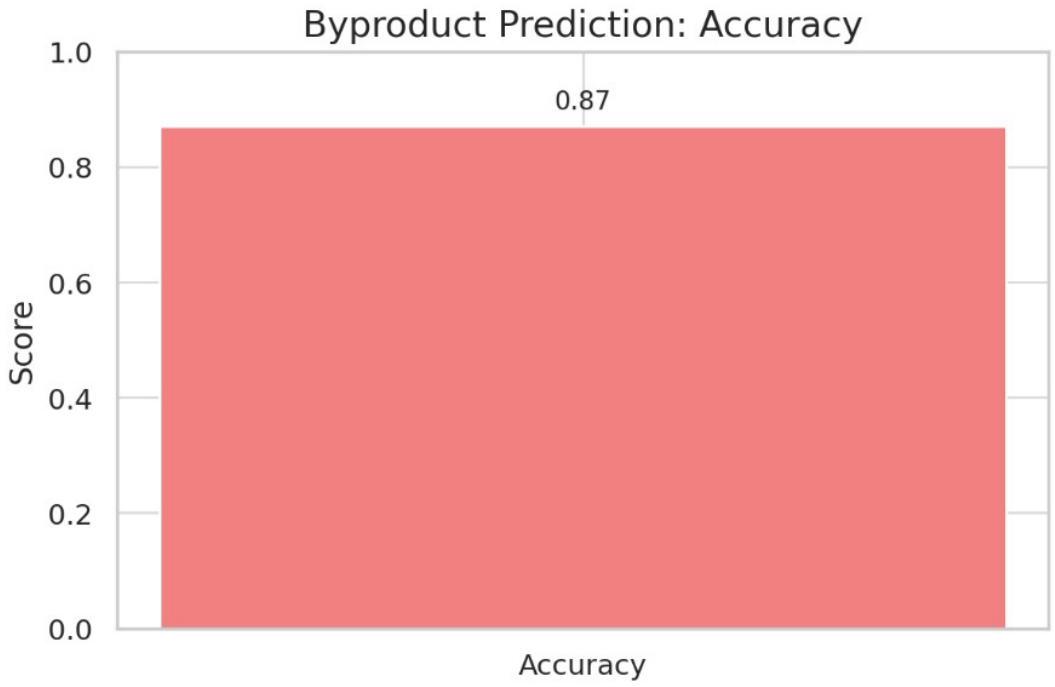
This visual represents the moderate cluster cohesion achieved with a silhouette score of 0.65..

Product Prediction:



This chart illustrates the regression metrics, showing an MSE of 1.23 and an R^2 score of 1.23, reflecting reliable predictive capacity.

Byproduct Prediction:



This chart shows the accuracy of decision trees for byproduct prediction at 87%, highlighting a balance between performance and interpretability.

Future iterations aim to refine these results by incorporating additional traditional data and leveraging ensemble methods to reduce variance.

5. Conclusion

This project introduces a machine learning-based chemical reaction simulator, providing dynamic predictions for reaction types, products, and byproducts. Data limitations were addressed through a hybrid approach combining synthetic and real-world data. Future improvements include expanding data sources, optimizing algorithm performance, and deploying real-time interactive tools. This work highlights the potential for AI-driven chemical education tools, bridging the gap between theoretical knowledge and practical experimentation.

Source code : https://github.com/MohnishE/chemical-reaction-predictor.git