Semester-Long Project Report (BTP-II)

On

Machine Learning Model for Predicting the Fluorescence Colour of As-Synthesized Carbon Dots

Submitted by

Swaroop Mahadeo Ratnaparkhi

20CY20035

Department of Chemistry, IIT Kharagpur

Under the guidance of

Prof. Amita (Pathak) Mahanty

Department of Chemistry

IIT Kharagpur

and mentor

Ms. Angana Bhattacharya

Mr. Rajarshi Basu

May 2, 2025

Abstract

Over the past 5–6 years, chemists have increasingly turned to machine learning to optimize the synthesis of carbon dots (CDs). However, designing CDs with tailored optical properties remains challenging due to their dependence on multiple variables, such as precursor concentration, reaction time, and solvent selection. This challenge is particularly critical for CDs emitting in the yellow-red fluorescence range. Herein, I report a hybrid machine-learning model combining extreme gradient boosting (XGBoost) with a neural network to help researchers filter out unsatisfactory synthesis conditions and predict outcomes, thereby reducing ineffective experiments and refining the design process.

Acknowledgement

I would like to express my sincere gratitude to everyone who contributed to the successful completion of my semester-long project at the Functional Nanomaterials Laboratory, Department of Chemistry, IIT Kharagpur.

I am immensely thankful to Mr. Rajarshi Basu and, especially, Ms. Angana Bhattacharya for her guidance, support, and assistance with sample preparation. Their expertise and willingness to share their knowledge have been helpful in my learning.

I would also like to recognize Prof. Amita (Pathak) Mahanty for her ongoing support and guidance. Her encouragement and advice have been invaluable throughout this journey.

Swaroop Ratnaparkhi
20CY20035
Department of Chemistry
IIT Kharagpur

Contents

1. Abstract
2. Acknowledgement
3. Introduction
4. Data Collection
5. Structure of the machine learning model
6. Evaluation metric
7. Results
8. Discussion
9. Experimental Section
10. Conclusion

Introduction

Carbon dots (CDs) are quasi-zero-dimensional carbon nanomaterials (<20 nm) with intrinsic fluorescence, exhibiting tunable emission, excellent water solubility, and ease of synthesis.[1,2] Their photoluminescence (PL) properties drive both fundamental research and practical applications.[1] However, a persistent challenge remains in controlling emission characteristics, with most synthesis methods yielding primarily blue/green-emitting CDs (400 - 550 nm) while red-emitting variants (> 600 nm) - needed for biomedical applications - remain difficult to produce.[1,3]

Current CD synthesis predominantly relies on empirical trial-and-error approaches, where reaction outcomes are influenced by multiple interdependent factors including precursor chemistry, solvent environment, temperature, and reaction duration. This empirical paradigm suffers from low efficiency and randomness.[2]

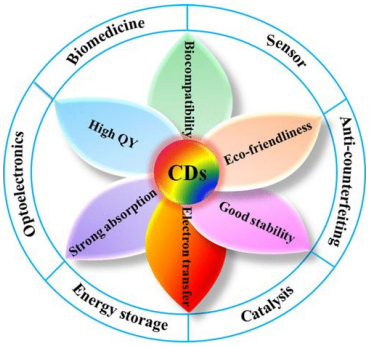

Figure 1: Carbon dots – properties and applications[1]

Machine learning (ML) has garnered significant attention among chemists in recent years as a powerful tool for optimizing the synthesis of carbon dots (CDs) with tailored emission colours. Qian *et al.* developed a deep learning model for predicting the colour of CDs under ultraviolet irradiation.[5] They also utilized the XGBoost model to prepare CDs with high fluorescence intensity and tunable emission from *p*-benzoquinone (PBQ) and ethylenediamine (EDA) in different solvents at room temperature.[6] Zhou et al. developed a machine learning (ML)-assisted strategy for the targeted synthesis of red fluorescent carbon dots (CDs).[4]

Their approach implements a four-stage computational pipeline: (1) feature extraction using an Extreme Gradient Boosting (XGBoost) model, (2) categorical data encoding via One-hot transformation, (3) dimensionality reduction employing Principal Component Analysis (PCA), and (4) classification using one of six machine learning algorithms—Decision Tree (DT), k-Nearest Neighbors (KNN), Random Forest (RF), Support Vector Classifier (SVC), XGBoost, or Logistic Regression (LR). This data-driven framework systematically predicts optimal synthesis conditions for obtaining red-emitting CDs, demonstrating how ML can overcome the inefficiencies of conventional empirical methods in nanomaterial development.

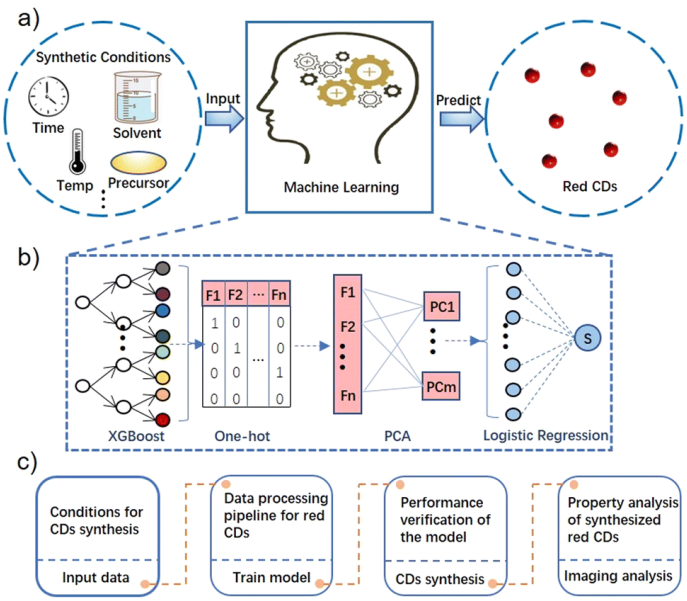

Figure 2: The ML model as proposed by Zhou *et al*.[4]

In this work, I developed a hybrid machine-learning model for predicting the maximum intensity emission colour of carbon dots (CDs) for a given set of synthesis conditions. This approach integrates the first three stages of Zhou et al.'s data processing pipeline - comprising (1) feature extraction using XGBoost, (2) One-hot encoding, and (3) dimensionality reduction via Principal Component Analysis (PCA) – with (4) a neural network-based classifier.

Data Collection

A dataset of 169 carbon dot (CD) synthesis protocols was compiled from published literature (Supporting Data File 1), adhering to the following selection criteria:

1. Synthesis Method: Only hydrothermal or solvothermal synthesis routes were included.
2. Precursor Limitations:
    i. Maximum of two precursors per synthesis
    ii. Exclusion of polymeric precursors
    iii. Restricted to C, H, N, and O-containing compounds only
3. Chemical Environment:
    i. Prohibited use of strong acids/bases
    ii. Single-component solvents (no mixtures)
4. Fluorescence Measurement:
    i. Solvent consistency between synthesis and characterization
    ii. Default assumption of water (water-soluble CDs) or ethanol for unspecified cases

Rationale for Selection Criteria: To minimize input feature dimensionality, enabling effective model training on our limited dataset (n=169).

The dataset includes precursor identities and molar amounts, solvent type, reaction temperature, heating duration, and maximum emission wavelength for each carbon dot synthesis, as detailed in Supporting Data File 1. I structured this data into a '$pandas\ DataFrame$', a tabular data format for Python analysis. After performing data processing and cleaning (see Supporting Data File 2 for the curated dataset), the resulting refined dataset was used for model development. The complete data processing code is available in Supporting Python File 1.

The CD emission colour serves as the model's output label, categorized by wavelength ranges:

- 0 for blue (380–500 nm)
- 1 for green (500–565 nm)
- 2 for yellow-to-red (565–750 nm)

The yellow, orange, and red regions were combined into a single class (2) due to limited available data for these longer-wavelength emissions in the literature.

To encode solvent features for computational analysis, I adopted the methodology of Qian et al.[6]. Key solvent properties - including permittivity, viscosity, density, melting point, boiling point, and refractive index - were compiled for all solvents in a dataset. Principal component analysis (PCA) was then applied to these physicochemical parameters, reducing the dimensionality to two principal components that collectively capture ~ 94% of the information in the original data. PCA simplifies complex data (like multiple solvent properties) by identifying key patterns and combining correlated features into new variables called *principal components* (PCs). These components are ranked by how much variance they explain, allowing us to reduce dimensions while preserving ~ 95% of the original information. For solvents, PCA converted the properties into just two numerical values, making the data easier for models to process without losing critical trends. The code using which PCA was performed is given in the file Supporting Python File 2.

In Supporting Python File 1 I systematically characterized all precursors using SMILES (Simplified Molecular Input Line Entry System) notations and converted them into MACCS (Molecular Access System) fingerprints. Each MACCS fingerprint is a 167-bit binary vector that provides a complete structural signature of a molecule, where:

- Each bit position corresponds to a specific, predefined chemical substructure (e.g., bit 29 = P, bit 42 = F)
- A value of 1 indicates the presence of that substructure in the molecule
- A value of 0 denotes its absence

For example:

- Bit 42 = 1 (F present)
- Bit 29 = 0 (P absent)

Given the high dimensionality of the 167-bit MACCS fingerprints, I performed principal component analysis (PCA) on the complete set of precursor fingerprints (Supporting Python File 1) to reduce each vector to 7 principal components. This dimensionality reduction facilitates more efficient model training while preserving the essential chemical information encoded in the original fingerprints.

Structure of the machine learning model

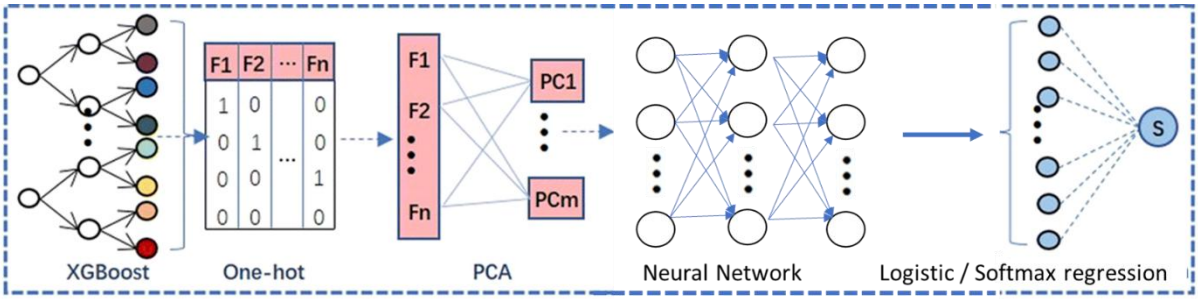

Figure 3: The structure of the machine learning model that was used in this project.

The machine learning model I have proposed is largely based on the model that was prepared by Zhou *et al.*[4]. Here is a description of the pipeline of this model:

1. **XGBoost**

A **decision tree (DT)** is a flowchart-like structure that helps predict outcomes based on given inputs (such as reaction conditions). It begins at the **root node** (the top of the tree/the top most node the starting point) and follows branches (**internal nodes**) based on specific conditions until reaching a **leaf node** (the final nodes that make the predictions). For example:

First question (say): "Is the reaction temperature > 150°C?"
If yes, the tree might next ask (say): "Is the fraction of precursor A > 0.5 (mol/mol)?"
If no, it might check (say): "Does the precursor contain nitrogen?"
This continues until a leaf node (final prediction) is reached, like (say): "80% chance of red emission".

There can exist different decision trees for the same task. Some perform better than the others. The goal of a decision tree algorithm is to find a tree that performs well. Figure 4 gives an illustrative example of decision trees.

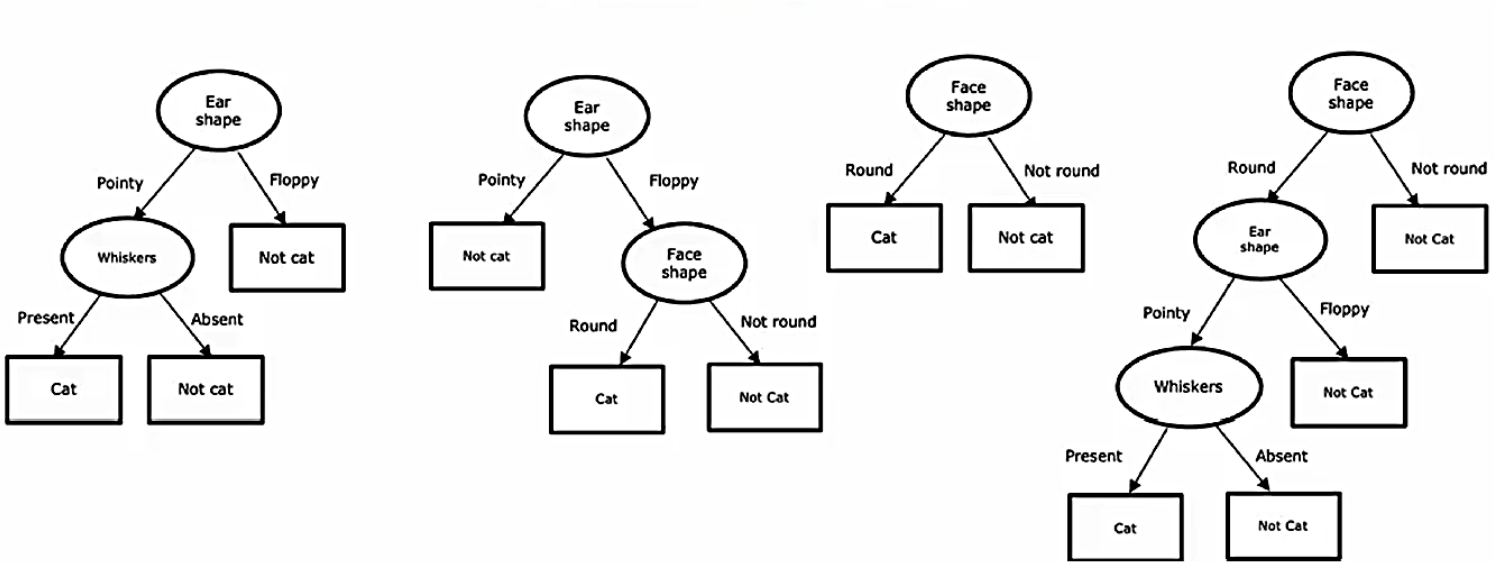

Figure 4: Decision trees for predicting if an animal is a cat

XGBoost builds trees **sequentially**, where each new tree fixes errors of the previous ones (during training):

- **Step 1**: Train a simple tree (e.g., based only on temperature). It makes some wrong predictions.
- **Step 2**: The next tree focuses on the *residual errors* (e.g., samples misclassified in Step 1, it might now consider solvent type.)
- **Repeat**: Over hundreds of trees, the model becomes highly accurate.

Final XGBoost predictions are made using the outputs from all trees collectively, where each tree sequentially corrects residual errors from previous trees. Early trees learn broad patterns, while later trees focus on correcting residual errors, with all trees' predictions being combined for the final output.

2. **Leaf Node Encoding:**

Each decision tree in XGBoost splits data based on features (e.g., temperature, solvent polarity), creating distinct paths from root to leaf. For a given CD synthesis, its conditions determine which leaf it reaches in each tree. An illustrative example:

- **Tree 1**: Splits first on temperature → then fraction of precursor A → stops at **Leaf 5**
  - Path: *Temp >150°C* → fraction of precursor A *< 0.5 (mol/mol)* → *Leaf 5*
- **Tree 2**: Splits first on precursor type → then time → stops at **Leaf 3**
  - Path: Precursor = N-rich → Time < 2h → Leaf 3

These leaf positions encode *nonlinear interactions* between features.

Each tree's leaf is converted into a binary vector where:

- **1** = Sample reached this leaf
- **0** = Otherwise

For 10 trees with 8 leaves each:

- Original output: [Leaf 5, Leaf 3, ..., Leaf 7]
- One-hot encoding:
  - Tree 1: [0,0,0,0,1,0,0,0] (Leaf 5 = 5th position)
  - Tree 2: [0,0,1,0,0,0,0,0] (Leaf 3 = 3rd position)
  - ...

Performing this "One-Hot Encoding" turns categorical leaf IDs into machine-readable numerical data.

The binary fingerprint represents:

- **Key splits**: Each "1" marks a critical threshold (e.g., "Temp >150°C + N-rich precursor").
- **Feature interactions**: Co-occurring "1"s reveal synergies (e.g., *high temp + low precursor fraction* only works with *short reaction times*).

Example:
A fingerprint with [...,1,0,…1,...] might mean:

- Bit 42: *"Low precursor fraction+ long time"* → Usually green emission
- Bit 89: *"N-rich precursor"* → Overrides to red when combined with Bit 42

- Leaf encoding transforms XGBoost from a "black box" into a traceable feature generator, capturing the hierarchical, threshold-dependent chemistry of CD formation. This is why we process leaves rather than using raw outputs.

3. **Performing Principal Component Analysis:**

After leaf-node encoding, each CD synthesis is represented by a long binary vector (e.g., 1,000+ bits if using 100 trees with 10 leaves each). To reduce the dimensionality while keeping as much information as desired, I performed PCA on the one-hot vectors to transform the data.

4. **Feeding to a Neural Network:**

   The dimensionality-reduced principal components (PCs) are input into a neural network, which employs a final logistic/softmax regression layer to classify the emission colour of the carbon dots. The neural network architecture features:

   - Four hidden layers with 256, 128, 64, and 32 neurons respectively
   - A final output layer utilizing:
     - 1 neuron with sigmoid activation for binary classification, or
     - 3 neurons with softmax activation for multi-class colour prediction (blue/green/yellow-red)

Evaluation Metric

The dataset was randomly split into training (70%) and test (30%) sets, with stratified sampling to maintain proportional representation of all emission colour categories. The training set was used for model development, while the test set served for both hyperparameter tuning and final evaluation. The trained model was then applied to predict the emission colours of synthesized carbon dots.

Model performance was assessed using:

F1-score: A harmonic mean of precision and recall that provides a balanced metric, particularly valuable for handling potential class imbalances. Scores approaching 1 indicate optimal classification performance.

Results

I systematically trained the model on multiple variants of the dataset by:

1. Varying the number of input features (feature ablation/progressive addition)
2. Adjusting the size of the training set (subsampling/expansion)

The model consistently incorporated three key features: (1) reaction temperature, (2) solvent identity, and (3) precursor mole fraction. Initial training and testing were performed using 169 synthesis conditions (Supporting Data File 2). The model was first trained on 169 conditions (Supporting Data File 2), then tested with modified feature sets (Table 1).

Table 1: Model Performance with Varied Feature Combinations

Sr no.	Input Features	Output Features	precision	recall	f1-score	Macro average f1-score	Weighted average f1-score	Code File
1	MACCS, atomic content, time	B(0)	0.90	0.82	0.86	0.71	0.73	Link_1
		G(1)	0.65	0.69	0.67			
		YOR(2)	0.57	0.62	0.59			
2	MACCS, atomic content, time	B or G (0)	0.84	1.00	0.92	0.77	0.84	Link_2
		YOR(1)	1.00	0.46	0.63			
		-	-	-	-			
3	Atomic content, time	B(0)	0.79	0.86	0.83	0.64	0.68	Link_3
		G(1)	0.61	0.69	0.65			
		YOR(2)	0.56	0.38	0.45			
4	Atomic content, time	B or G (0)	0.88	0.97	0.93	0.83	0.87	Link_4
		YOR(1)	0.89	0.62	0.73			
		-	-	-	-			
5	Atomic content	B(0)	0.86	0.86	0.86	0.64	0.68	Link_5
		G(1)	0.58	0.69	0.63			
		YOR(2)	0.50	0.38	0.43			
6	Atomic content	B or G (0)	0.88	0.97	0.93	0.83	0.87	Link_6
		YOR(1)	0.89	0.62	0.73			
		-	-	-	-			
7	MACCS, time	B(0)	0.85	0.77	0.81	0.69	0.71	Link_7
		G(1)	0.61	0.69	0.65			
		YOR(2)	0.62	0.62	0.62			
8	MACCS, time	B or G (0)	0.86	1.00	0.93	0.81	0.87	Link 8
		YOR(1)	1.00	0.54	0.70			
		-	-	-	-			
9	MACCS	B(0)	0.85	0.77	0.81	0.69	0.71	Link_9
		G(1)	0.61	0.69	0.65			
		YOR(2)	0.62	0.62	0.62			
10	MACCS	B or G (0)	0.84	1.00	0.92	0.77	0.84	Link_10
		YOR(1)	1.00	0.46	0.63			
		-	-	-	-			
11	MACCS, Atomic content	B(0)	0.90	0.82	0.86	0.71	0.73	Link_11
		G(1)	0.65	0.69	0.67			
		YOR(2)	0.57	0.62	0.59			
12	MACCS, Atomic Content	B or G (0)	0.90	0.95	0.92	0.84	0.88	Link_12
		YOR(1)	0.82	0.69	0.75			
		-	-	-	-			

Analysis of Table 1 suggests two key observations:

1. The model demonstrates better performance in binary classification tasks (distinguishing shorter-wavelength blue-green emissions from longer-wavelength yellow-red emissions) compared to finer spectral categorization.
2. Heating duration appears to exhibit not have any correlation with emission colour outcomes, though this preliminary finding requires further experimental validation before definitive conclusions can be drawn.

Given the predominance of citric acid and urea as precursors in carbon dot synthesis literature, the dataset was expanded to include additional synthesis conditions using these two precursors. This increased the total number of experimental conditions to 208, as documented in Supporting Data File 3. The expanded dataset was pre-processed using identical methods to those applied to Supporting Data File 1, resulting in Supporting Data File 4. Using the same methodology as for Table 1, the model was retrained on the expanded dataset, yielding the results in Table 2.

Table 2: Model Performance with Varied Feature Combinations after additional synthesis data

Sr no.	Input Features	Output Features	precision	recall	f1-score	Macro average f1-score	Weighted average f1-score	Code File
1	MACCS, atomic content, time	B(0)	0.72	0.87	0.79	0.73	0.74	Link 1
		G(1)	0.83	0.62	0.71			
		YOR(2)	0.73	0.65	0.69			
2	MACCS, atomic content, time	B or G (0)	0.90	0.80	0.85	0.76	0.80	Link 2
		YOR(1)	0.59	0.76	0.67			
		-	-	-	-			
3	Atomic content, time	B(0)	0.72	0.77	0.74	0.66	0.68	Link 3
		G(1)	0.60	0.56	0.58			
		YOR(2)	0.69	0.65	0.67			
4	Atomic content, time	B or G (0)	0.90	0.83	0.86	0.77	0.82	Link 4
		YOR(1)	0.62	0.76	0.68			
		-	-	-	-			
5	Atomic content	B(0)	0.72	0.77	0.74	0.56	0.6	Link 5
		G(1)	0.47	0.44	0.45			
		YOR(2)	0.50	0.47	0.48			
6	Atomic content	B or G (0)	0.86	0.91	0.88	0.76	0.82	Link 6
		YOR(1)	0.71	0.59	0.65			
		-	-	-	-			
7	MACCS, time	B(0)	0.70	0.87	0.78	0.68	0.70	Link 7
		G(1)	0.78	0.44	0.56			
		YOR(2)	0.71	0.71	0.71			
8	MACCS, time	B or G (0)	0.90	0.76	0.82	0.73	0.77	Link 8
		YOR(1)	0.54	0.76	0.63			
		-	-	-	-			
9	MACCS	B(0)	0.70	0.87	0.78	0.71	0.73	Link 9
		G(1)	0.83	0.62	0.71			
		YOR(2)	0.71	0.59	0.65			
10	MACCS	B or G (0)	0.84	0.93	0.89	0.75	0.81	Link 10
		YOR(1)	0.75	0.53	0.62			

		-	-	-	-			
11	MACCS, Atomic content	B(0)	0.75	0.80	0.77	0.72	0.73	Link 11
		G(1)	0.69	0.69	0.69			
		YOR(2)	0.73	0.65	0.69			
12	MACCS, Atomic Content	B or G (0)	0.86	0.93	0.90	0.78	0.83	Link 12
		YOR(1)	0.77	0.59	0.67			
		-	-	-	-			

Analysis of Table 2 reveals that incorporating additional synthesis data for citric acid- and urea-derived carbon dots did not yield measurable improvements in model performance compared to the original dataset Supporting Data File 1. The results demonstrate that large-scale addition of synthetically similar data (citric acid/urea precursors under comparable conditions) does not significantly improve model performance.

Given the predominance of citric acid/urea precursor systems in the literature, I created a dedicated dataset (n=80 conditions) using Supporting Data File 3. This new dataset could be accessed through Supporting Python File 3. This precursor-specific dataset comprises exclusively citric acid/urea-derived carbon dots, with the following input features: Temperature (°C), solvent identity (categorical), time (hours) and molar ratio of citric acid:urea.

Table 3: Model performance with varying hyperparameters for carbon dots prepared only using citric acid and urea.

	Neural Layers											
Sr no.	n_1	n_2	n_3	n_4	Final Layer	Output Features	precision	recall	f1-score	Macro average f1-score	Weighted average f1-score	Code File
1	256	128	64	32	3	B(0)	0.67	0.22	0.33	0.62	0.62	Link
						G(1)	0.82	1.00	0.90			
						YOR(2)	0.46	0.75	0.57			
2	40	20	10	5	3	B(0)	0.67	0.22	0.33	0.58	0.58	
						G(1)	0.82	1.00	0.90			
						YOR(2)	0.42	0.62	0.5			
3	20	10	5	3	3	B(0)	0.60	0.33	0.43	0.61	0.61	
						G(1)	0.89	0.89	0.89			
						YOR(2)	0.42	0.62	0.50			
4	20	20	10	5	3	B(0)	0.50	0.22	0.31	0.58	0.58	
						G(1)	0.82	1.00	0.90			
						YOR(2)	0.45	0.62	0.53			
5	20	10	5	5	3	B(0)	0.67	0.22	0.33	0.62	0.62	
						G(1)	0.90	1.00	0.95			
						YOR(2)	0.46	0.75	0.57			

The model exhibits significantly poorer performance when trained exclusively on citric acid (CA) and urea-derived carbon dots (CDs) with three output classes: blue (0), green (1), and yellow-orange-red (YOR, 2).

The model was similarly trained for binary classification tasks to evaluate its ability to distinguish between Blue-green emission and yellow-orange-red emission carbon dots synthesized from CA and urea. The results are shown in table 4.

Table 4: Model performance with varying hyperparameters for carbon dots prepared only using citric acid and urea with binary output features.

	Neural Layers											
Sr no.	n_1	n_2	n_3	n_4	Final Layer	Output Features	precision	recall	f1-score	Macro average f1-score	Weighted average f1-score	Code File
1	256	128	64	32	1	B or G (0)	0.85	0.94	0.89	0.8	0.84	Link
						YOR(1)	0.83	0.62	0.71			
						-	-	-	-			
2	40	20	10	5	1	B or G (0)	0.89	0.94	0.92	0.86	0.88	
						YOR(1)	0.86	0.75	0.80			
						-	-	-	-			
3	20	10	5	3	1	B or G (0)	0.92	0.67	0.77	0.72	0.74	
						YOR(1)	0.54	0.88	0.67			
						-	-	-	-			
4	20	20	10	5	1	B or G (0)	0.89	0.89	0.89	0.82	0.85	
						YOR(1)	0.75	0.75	0.75			
						-	-	-	-			
5	20	10	5	5	1	B or G (0)	0.85	0.94	0.89	0.8	0.84	
						YOR(1)	0.83	0.62	0.71			
						-	-	-	-			

As evidenced by Table 4, the model achieves significantly better performance in binary classification (distinguishing blue-green [BG, 400–565 nm] from yellow-orange-red [YOR, 565–750 nm] emissions) for CA/urea-derived carbon dots.

Discussion:

1. **Binary vs. Multiclass Performance:**

Comparative analysis of Tables 1–4 demonstrates that the model achieves superior performance in binary classification (distinguishing blue-green [BG] from yellow-orange-red [YOR] emissions) compared to three-class classification. While the binary F1-score remains suboptimal, it represents a significant improvement over the multiclass system. This enhancement likely stems from:

- Increased effective sample size per class
- Reduced spectral overlap between broader emission ranges

2. **Feature Importance Analysis:**
    a. Time redundancy: Table 2 reveals less correlation between heating duration and emission colour, suggesting that reaction time may not exerts any influence under hydrothermal/solvothermal conditions. This likely occurs because high-temperature hydrothermal reactions are kinetically dominated by precursor chemistry rather than time.
    b. Critical features: Atomic composition and MACCS keys together achieved good F1-score, indicating they sufficiently encode the chemical determinants of emission colour.
3. **Experimental Validation:**

Model predictions were tested through controlled synthesis using citric acid/urea precursors (see Experimental Section for details).

4. A significant limitation in neural network implementation is the inherent variability of optimal hyperparameters across training sessions. This problem is more prevalent for smaller datasets.

Experimental Section:

To evaluate the model's robustness and gain practical insights, we synthesized carbon dots using citric acid monohydrate (1.6891 g, 8.0 mmol) and urea (960.96 mg, 16.0 mmol) in 20 mL ethanol. The mixture was sealed in a Teflon-lined stainless-steel autoclave and heated at 180°C for 6 hours in a muffle furnace, followed by gradual cooling to room temperature over 12 hours. The reaction yielded a pure powdered product, eliminating the need for centrifugation or additional purification. Fluorescence characterization of the obtained CDs will be used to validate the model's predictions.

Conclusion:

This study demonstrates the application of machine learning for predicting the carbon dot emission colours from synthesis conditions, significant challenges emerge when working with limited chemical datasets. The proposed model achieves reasonable accuracy for binary classification (blue-green vs. yellow-orange-red) but struggles with finer colour distinctions due to data scarcity, particularly for rare emissions. Neural networks prove especially sensitive to hyperparameter variations in small datasets (<200 samples). Crucially, the work underscores that effective ML applications in chemistry require both domain expertise to design chemically meaningful features and critical awareness of dataset limitations—highlighting the need for expanded, standardized synthesis data collection to unlock the full potential of data-driven materials discovery.

References:

1. Carbon Dots: A New Type of Carbon-Based Nanomaterial with Wide Applications (https://doi.org/10.1021/acscentsci.0c01306)
2. Controlled Synthesis of Multicolor Carbon Dots Assisted by Machine Learning (https://doi.org/10.1002/adfm.202210095)
3. Red carbon dots: Optical property regulations and applications (https://doi.org/10.1016/j.mattod.2019.05.003)
4. High-efficiency synthesis of red carbon dots using machine learning (https://doi.org/10.1039/D2CC03473E)
5. Exploiting deep learning for predictable carbon dot design (https://doi.org/10.1039/D0CC07882D)
6. Customized Carbon Dots with Predictable Optical Properties Synthesized at Room Temperature Guided by Machine Learning (https://doi.org/10.1021/acs.chemmater.1c03220)