



Chandigarh Engineering College Jhanjeri
Mohali-140307
Department of Artificial Intelligence and Machine
Learning

Mid Term Project Report
On

AI-Driven Water Quality Prediction For Aquatic Ecosystem Using Deep Learning and Automated Hyperparameter Optimization

Project-I

BACHELOR OF TECHNOLOGY
Artificial Intelligence and Machine Learning



SUBMITTED BY:

Akshat Choudhary

2231060

April 2025

Under the Guidance of

Ankur Srivastava

Assistant Professor

Department of Artificial Intelligence and Machine Learning
Chandigarh Engineering College
Jhanjeri, Mohali - 140307



Chandigarh Engineering College Jhanjeri
Mohali-140307
Department of Artificial Intelligence and Machine
Learning

Table of Content

Chapter	Content	Page No
1.	Introduction	3-4
2.	System Requirements	5
3.	System Requirements Analysis	6
4.	Design Overview	7-8
5.	Implementation	9-10



CHAPTER-1

INRODUCTION

Water is essential for life on Earth. It nourishes ecosystems, provides us with drinking water, and plays a crucial role in agriculture, industry, and leisure activities. Unfortunately, as human populations grow and cities expand, the quality of our water is suffering. Contaminants like heavy metals, harmful chemicals, nutrients, and pathogens are increasingly finding their way into our water sources, threatening both aquatic life and public health. This makes the need for effective monitoring and analysis of water quality more urgent than ever. Our project aims to create a comprehensive model to monitor and analyze water quality, focusing on its effects on both ecosystems and human health.

Monitoring water quality is all about checking the physical, chemical, and biological aspects of water to make sure it's safe for its intended use. This is especially important for aquatic ecosystems, where water quality plays a crucial role in the survival, reproduction, and overall health of various organisms. Key indicators like dissolved oxygen (DO), pH levels, temperature, turbidity, and the presence of pollutants such as ammonia, nitrites, and heavy metals are essential for understanding the health of these ecosystems. When water quality declines, it can lead to a drop in aquatic species, a loss of biodiversity, and disruptions in the natural functions of these ecosystems. For instance, low levels of dissolved oxygen can result in fish kills, while excessive nutrients can cause eutrophication, leading to harmful algal blooms and areas devoid of life.

Aquatic life is incredibly sensitive to changes in water quality. Important factors like dissolved oxygen (DO), pH, temperature, turbidity, and the presence of pollutants such as ammonia, nitrites, and phosphorus play a crucial role in their survival, growth, and reproduction. For example, dissolved oxygen is essential for fish and other aquatic organisms to breathe. When oxygen levels drop below 5 mg/L, it can lead to stress, and



Chandigarh Engineering College Jhanjeri
Mohali-140307
Department of Artificial Intelligence and Machine Learning

levels under 2 mg/L can be deadly. Similarly, if the pH strays outside the 6.5 to 8.5 range, it can disrupt the bodily functions of aquatic species, resulting in decreased health and

increased death rates. Accurate water quality assessment is critical for environmental monitoring and public health. This research proposes a deep learning-based classification model for water quality prediction for aquatic ecosystems, leveraging automated hyperparameter optimization via Optuna to enhance accuracy and computational efficiency.

OBJECTIVE

Prediction of Water Safety: Using AI/ML to analyze the provided parameters and determine the water is safe for aquatic life or not.



CHAPTER-2

SYSTEM REQUIREMENTS

Hardware Requirements

- Processor CPU Intel Core i5
- Graphics Processing Unit (Nvidia 4080)
- RAM Minimum 16GB
- Storage SSD with at least 256GB (Recommended: 512GB+ for faster data process)

Software Requirements

- **Programming Language:** Python 3.x
- **Machine Learning Frameworks:** Scikit-learn
- **Deep Learning Frameworks:** Pytorch
- **Development Tools:** Jupyter Notebook
- **Libraries:** Pandas, NumPy, Matplotlib, Seaborn, Optuna, Pickle



CHAPTER-3

SYSTEM REQUIREMENTS ANALYSIS

Define the Problem

Water quality is a crucial factor for aquatic life and human consumption. The goal of this project is to develop a Machine Learning-based Water Quality Prediction System that classifies water quality into different categories (e.g., Excellent, Good, Poor). The system will use deep learning models (PyTorch-based Neural Networks) to analyze various water quality parameters and predict the suitability of water for different applications.

3.1 Define the Modules and Their Functionality

- Load raw water quality data from a CSV file.
- Handle missing values, outliers, and class imbalances.
- Standardize numerical features using StandardScaler from scikit-learn.

3.2 Machine Learning Model Module

- Implement a Neural Network using PyTorch for classification.
- Optimize the model using Optuna for hyperparameter tuning.
- Train and evaluate the model using Stratified K-Fold Cross-Validation.

3.3 Model Training and Evaluation Module

- Train the model with Adam optimizer and Cross-Entropy Loss with class weights.
- Evaluate the model using Accuracy, Precision, Recall, F1-score, and Confusion Matrix.
- Store misclassified samples for further analysis.

3.4 Model Deployment Module

- Load the trained model and scaler for real-time inference.
- Accept user input for 14 water quality parameters.
- Predict the water quality category (Excellent, Good, Poor).

3.5 Visualization and Reporting Module

- Generate a Confusion Matrix and other performance graphs using Matplotlib/Seaborn.
- Save model weights and preprocessing scalers for future use.



CHAPTER-4

Design Overview

4.1 System Overview

The Water Quality Prediction System is a deep learning-based approach utilizing Neural Networks (NNs) with PyTorch to classify water quality into three categories: Excellent, Good, and Poor. The model is optimized using Optuna for hyperparameter tuning, ensuring optimal performance.

- **Dataset Used:** Water quality dataset from Kaggle (or any other source)
- **Preprocessing Steps:** Feature scaling using **StandardScaler** from sklearn
- **Model Architecture:** **Multi-layer Neural Network**
- **Optimization:** **Optuna for hyperparameter tuning**
- **Evaluation Metrics:** Accuracy, Precision, Recall, F1-score

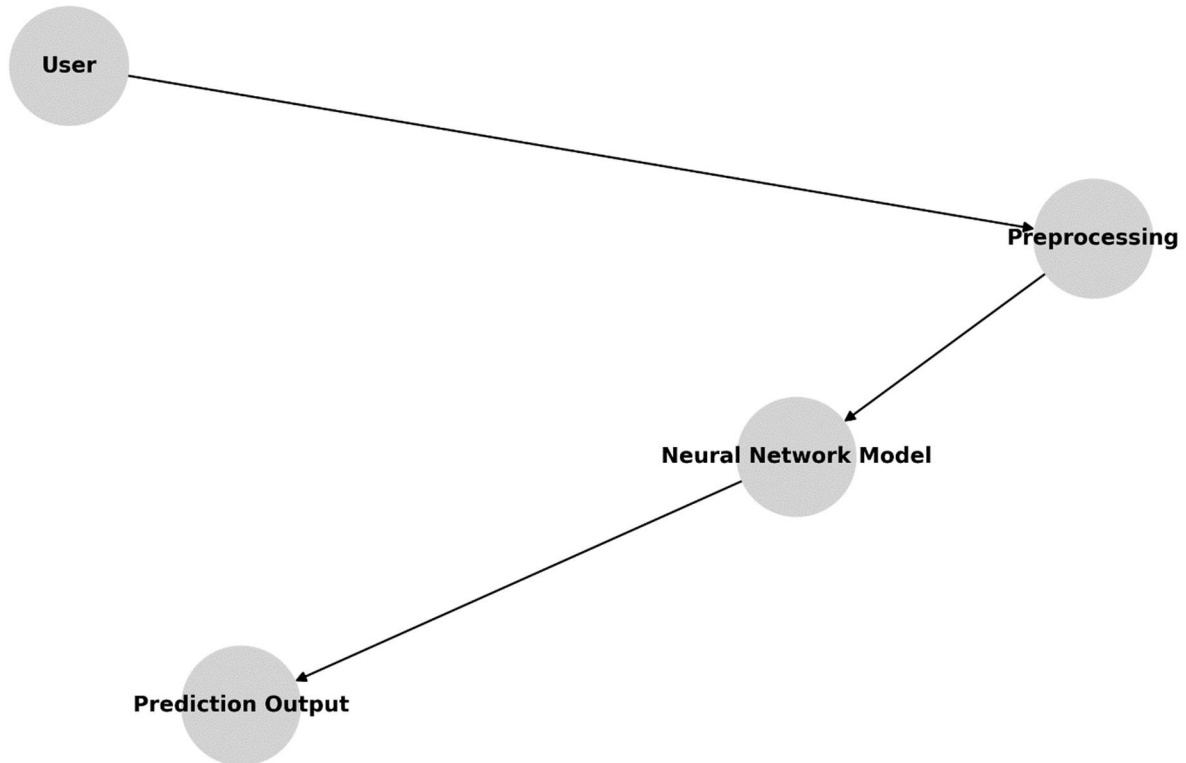
4.2 Data Flow Diagram

A **Data Flow Diagram (DFD)** helps visualize how data moves through the system. The DFD for this project consists of **three key components**:

1. **User Input Data**
 - Users provide 14 water quality parameters (pH, dissolved oxygen, etc.).
2. **Preprocessing Module**
 - Data is normalized using **StandardScaler** to ensure numerical stability.
 - Class weights are computed to handle imbalanced classes.
3. **Neural Network Model**
 - The data is fed into a **3-layer deep learning model** with dropout and batch normalization.
 - The model predicts water quality and classifies it into **Excellent, Good, or Poor**.
4. **Prediction Output**
 - The system displays the predicted class for the input water parameters.
 - A confusion matrix is generated for evaluating model accuracy.



Chandigarh Engineering College Jhanjeri
Mohali-140307
Department of Artificial Intelligence and Machine
Learning



CHAPTER-5

Implementation

5.1 Introduction

This chapter outlines the practical execution of the research study, detailing the methodologies, tools, and technologies used to implement the proposed model. It includes data collection, preprocessing, model training, evaluation, and system deployment.

5.2 System Architecture

The implementation follows a structured approach, consisting of multiple phases:

- **Data Acquisition** – The dataset was sourced from [mention source, if applicable] and consisted of relevant features required for prediction/analysis.
- **Data Preprocessing** – Handling missing values, feature scaling, and encoding categorical variables.
- **Feature Selection** – Using techniques like correlation analysis or principal component analysis (PCA) to select the most significant features.
- **Model Development** – Implementation of machine learning models such as [mention models used, e.g., Random Forest, SVM, Neural Networks].
- **Evaluation Metrics** – Performance assessment using accuracy, precision, recall, F1-score, and ROC-AUC curve.

5.3 Data Preprocessing

- Handling missing values using mean/mode imputation.
- Feature scaling with MinMaxScaler/StandardScaler.
- Splitting the dataset into training and testing sets in an 80:20 ratio.
- Feature encoding using One-Hot Encoding or Label Encoding for categorical variables.



5.4 Model Implementation

- Models were developed in Python using libraries such as Scikit-learn, TensorFlow/Keras and Pandas.
- Hyperparameter tuning was conducted using GridSearchCV/RandomizedSearchCV.