Delhi AQI Forecasting Using Deep Learning

UML501 Machine Learning Project Report
End-Semester Evaluation

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

BE Third Year
Members:
Nilay Singh , COE , nsingh4_be23@thapar.edu
Gaurang Mangla , COE , gmangla_be23@thapar.edu

Submitted By:
Nilay Singh 102313043
Gaurang Mangla 102303907

Submitted to:
Dr. Nitin Arora
Assistant Prof.
Dept. of CSE
TIET, Patiala

Computer Science and Engineering Department
TIET, Patiala

TABLE OF CONTENTS

1. INTRODUCTION

A. Technical Background of Project

Air Quality Index (AQI) is a numerical measure used to describe the quality of air and its impact on human health. Forecasting AQI is a time-series prediction problem because pollutant levels depend on past values, meteorological conditions, and seasonal patterns. Rapid urbanization and industrialization in Delhi have resulted in frequent air quality deterioration, making accurate forecasting critically important.

Machine Learning and Deep Learning provide robust tools for modeling temporal dependencies and non-linear relationships. In particular, recurrent neural networks (RNNs) such as LSTM and GRU, along with hybrid architectures like CNN+LSTM, are widely used for sequential forecasting tasks. These models can learn pollutant trends, short-term fluctuations, and the effects of weather variables.

B. Technical Concepts Used

This project utilizes several key technical concepts:

- **Time Series Forecasting**
  Predicting future values using historical data.
- **Feature Engineering**
  Lag features (AQI_lag_1, lag_3, lag_7), rolling averages, exponential weighted moving averages (EWMA), cyclical encoding (sin/cos for month, day).
- **Recurrent Neural Networks**
  - **LSTM**: Long Short-Term Memory networks
  - **Stacked LSTM**: Multi-layer LSTMs
  - **CNN+LSTM Hybrid**: Convolutional layers to extract patterns + LSTM to model temporal sequence
  - **GRU**: Gated Recurrent Units
- **Normalization (MinMaxScaler)**
  Scales features to improve training stability.
- **Training Techniques**
  Early stopping, learning rate scheduling, dropout for regularization.
- **Evaluation Metrics**
  RMSE, R^2 score, residual analysis.

C. Motivation

Delhi frequently ranks among the most polluted cities in the world. Poor air quality leads to:

- Respiratory illnesses
- Reduced visibility
- School closures
- Traffic restrictions
- Increased hospitalization

Accurate AQI forecasting empowers:

- Citizens to take precautionary measures
- Government agencies to plan interventions
- Researchers to understand pollution patterns
- Health organizations to issue advisories

The motivation of this project is to build a **reliable, data-driven AQI forecasting model** using deep learning to help predict pollution in advance.

D. Problem Statement

To develop an efficient predictive model that can **forecast next-day AQI levels in Delhi** using historical pollutant data combined with meteorological and engineered features, and to identify the deep learning architecture that provides the best forecasting performance.

E. Area of Application

- Environmental Monitoring
- Smart City Planning
- Health & Safety Alerts
- Real-time Pollution Tracking Systems
- Weather & Climate Analytics
- Governmental Environmental Agencies (CPCB, DPCC)
- Public Mobile Apps (e.g., AQI apps, pollution dashboards)

F. Dataset and Input Format

Dataset Source: Delhi AQI Dataset (2017–2025), merged from pollutant readings and meteorological data.

Important Columns:

- AQI (target variable)
- Meteorological features: temp, humidity, windspeed, precip, pressure
- Lag features: AQI_lag_1, AQI_lag_3, AQI_lag_7
- Rolling & EWMA features: AQI_roll_mean_7, AQI_ewma_7
- Cyclical encodings: Month_sin, Month_cos, DayOfYear_sin, DayOfYear_cos
- Weather condition one-hot flags

Input Format After Preprocessing:
Each training sample is a **7-day sliding window** of feature vectors

2. OBJECTIVE

A. Main Objective

To accurately forecast next-day AQI for Delhi using deep learning sequence models and determine the best-performing architecture.

B. Sub Objectives

- Perform feature engineering to enhance forecasting accuracy
- Train and compare LSTM, Stacked LSTM, CNN+LSTM, and GRU models
- Evaluate models using RMSE, R^2, residual plots, and error distribution
- Analyze the impact of temporal features on prediction performance
- Provide a final model recommendation based on evidence

3. METHODOLOGY

A. Steps

A structured approach was followed:

1. **Data Collection**
   Merge AQI data with meteorological variables.
2. **Data Cleaning & Preprocessing**
   Handle missing values, scale features, remove outliers.
3. **Feature Engineering**
   Add lag features, rolling averages, EWMA, cyclical encodings.
4. **Time-Series Windowing**
   Convert data to supervised sequences (lookback = 7 days).
5. **Model Building**
   Construct LSTM, Stacked LSTM, CNN+LSTM, and GRU networks.
6. **Model Training**
   EarlyStopping, ReduceLROnPlateau, dropout-based regularization.
7. **Model Evaluation**
   RMSE, R^2 score, actual vs predicted graphs, residual analysis.
8. **Comparison & Selection**
   Identify the best model (GRU model).

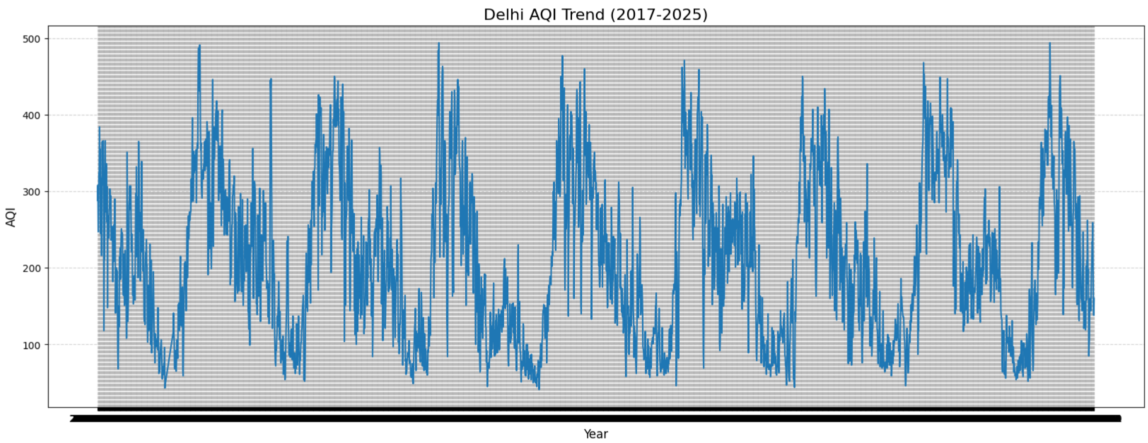
B. Deliverables of Each Step / Phase

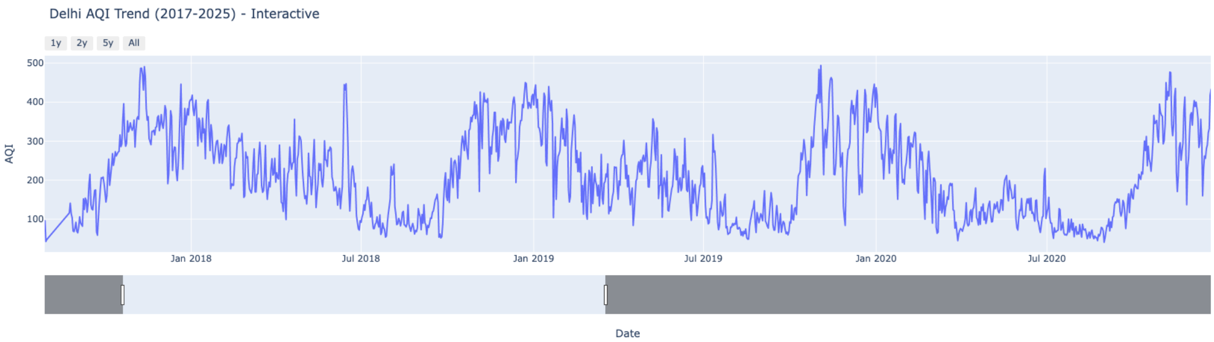
Phase	Deliverable
Data Preprocessing	Cleaned dataset, scaled features
Feature Engineering	30+ enhanced features
Modeling	4 deep learning architectures
Training	Trained models with loss curves
Evaluation	RMSE, R^2, prediction graphs
Comparison	Best model selection (GRU)
Final Output	AQI forecast system + report

4. Exploratory Data Analysis

A detailed Exploratory Data Analysis was conducted to understand the temporal behavior, seasonal variations, and feature relationships in Delhi's AQI dataset.
The key visualizations and insights are summarized below.

1. Delhi AQI Trend (2017–2025)





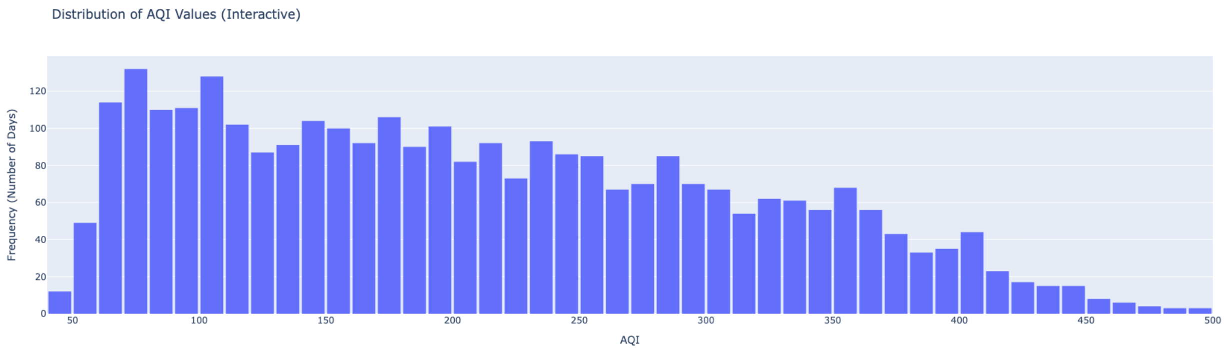
Key Insights

- **Strong seasonality:** AQI consistently peaks in **winter (Nov–Jan)** due to inversion, stubble burning, and calm winds.
- **Monsoon dip:** AQI falls sharply during **July–September**, with rain and wind dispersing pollutants.
- **High volatility:** AQI fluctuates heavily day-to-day, confirming a **short-memory dynamic system**.
- **COVID-19 dip:** A significant drop in pollution is visible during **March–June 2020**, reflecting reduced transport & industrial activity during lockdown.

Modeling Implication

Short-term patterns dominate → justifies using **7-day window** and **GRU/LSTM-based models**.

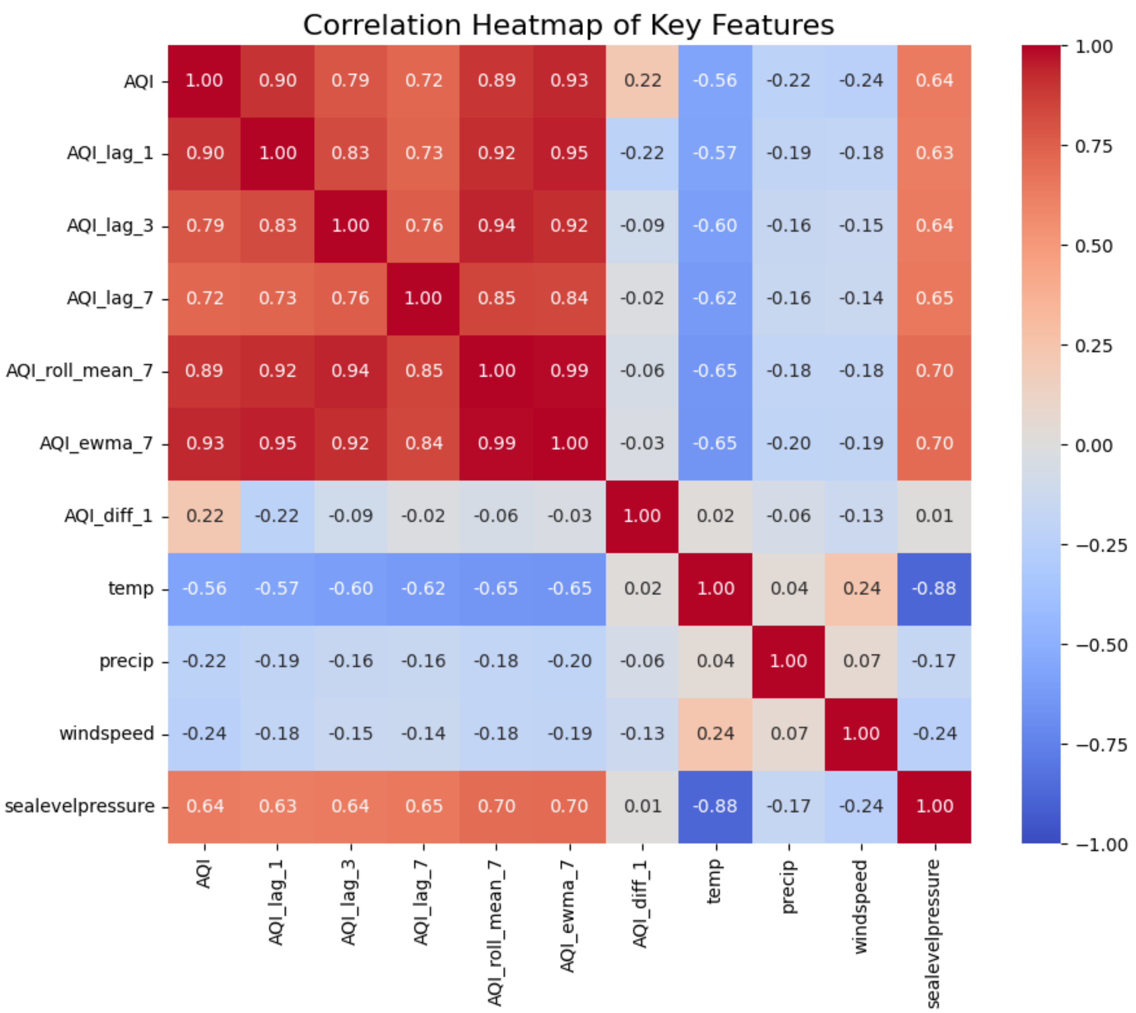
2. Distribution of AQI Values



Key Insights

- Most days fall between **100–250 AQI** → chronic moderate-to-poor air quality.
- Right tail shows frequent high-pollution episodes (**AQI > 350**), especially in winter.
- COVID period shows noticeably more clean-air days (<100 AQI).

3. Correlation Heatmap



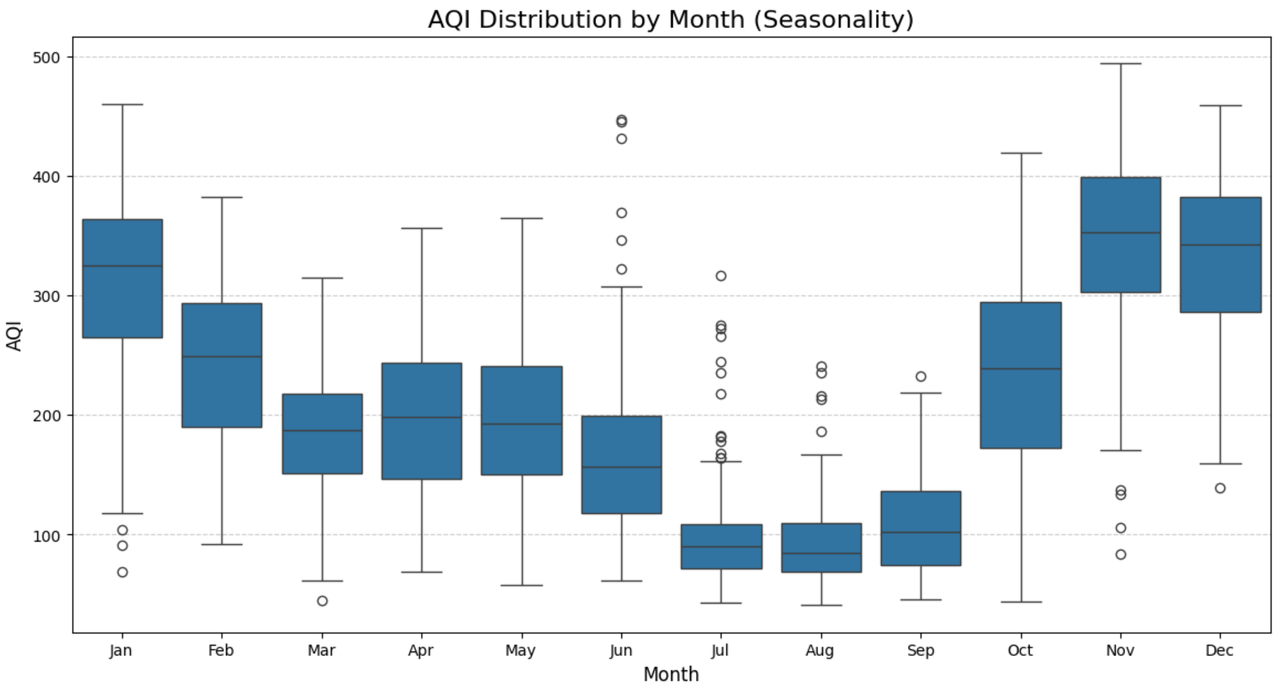
Insights

- **EWMA_7** has the strongest correlation with AQI (**≈0.93**).
- Lag features (AQI_lag_1, _lag_3, _lag_7) show very high correlation (0.72–0.90).
- Meteorological factors show moderate influence:
    - Windspeed & humidity → negative correlation
    - Pressure → positive correlation

Implication

Pollutant-history features are the most important for prediction; weather variables provide secondary improvements.

4. AQI Distribution by Month

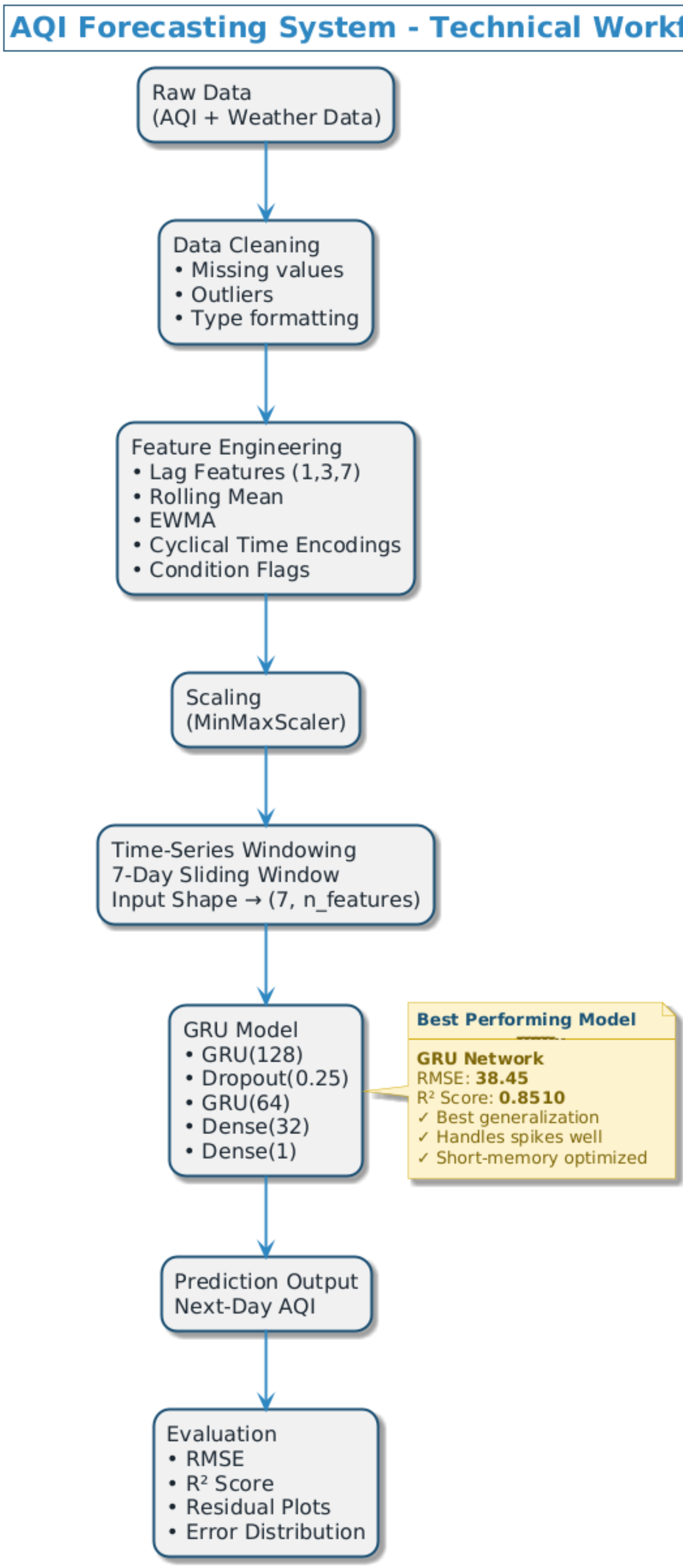


Insights

- **Winter (Nov–Jan)** shows highest and most variable AQI values.
- **Monsoon (Jul–Sep)** consistently has the cleanest air.
- Clear seasonal cycle validates the use of **cyclical time encodings**.

5. WORKING MODEL

A. Technical Diagram



B. Working Module

The system is organized into a series of interconnected modules that together perform end-to-end AQI forecasting:

1. Data Acquisition Module

Collects daily AQI values and meteorological parameters such as temperature, humidity, pressure, windspeed, and weather conditions.

2. Data Preprocessing Module

Handles missing values, removes inconsistencies, formats dates, and prepares the dataset for modeling.

3. Feature Engineering Module

Generates predictive features including lag values, rolling averages, EWMA, cyclical time encodings, and weather flags.

4. Scaling & Windowing Module

Applies MinMax scaling and converts the dataset into 7-day time-series sequences for supervised learning.

5. Model Development Module

Implements and trains multiple deep learning models:
LSTM, Stacked LSTM, CNN+LSTM, and GRU.

6. Evaluation Module

Produces RMSE, R^2 scores, actual vs predicted plots, and residual/error analysis to compare model performance.

7. Model Selection Module

Selects the **GRU model** as the final forecasting architecture based on highest accuracy and stability.

C. Attained Deliverable

The project successfully delivered the following outcomes:

1. Fully Preprocessed & Engineered Dataset

With lag features, rolling statistics, EWMA, cyclical encodings, and weather-based indicators.

2. Multiple Deep Learning Models Developed & Trained

Including LSTM, Stacked LSTM, CNN+LSTM, GRU, and GRU with Huber loss.

3. Final Model Selection

Identification of the **GRU model** as the best performer with:

- **RMSE = 38.45**
- **R^2 = 0.851**

4. Comprehensive Visualizations

Generated training curves, prediction plots, residual charts, and model comparison graphs.

5. End-to-End AQI Forecasting Pipeline

A complete workflow from raw data → preprocessing → modeling → evaluation → prediction.

6. RESULTS

A. Quantitative Performance Metrics

Models Used in This Study

Four deep learning models were implemented and evaluated:

- **Base LSTM** – Single-layer LSTM used as the baseline sequence model.
- **Stacked LSTM** – Deeper LSTM architecture to capture complex temporal patterns.
- **CNN + LSTM** – Convolution layer for feature extraction followed by LSTM for sequence modeling.
- **GRU (Best Model)** – Gated Recurrent Unit offering faster training and strong performance with fewer parameters.

All models were assessed using **RMSE**, **R^2 score**, **Actual vs Predicted plots**, **Residual analysis**, and **Error distributions**.

Model Performance Summary

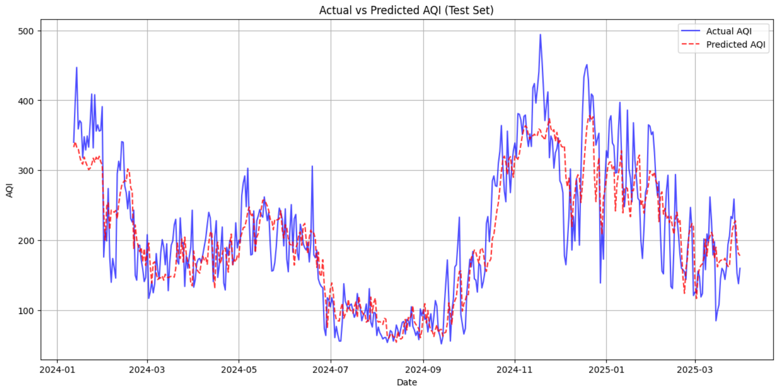
Model	RMSE ↓	R^2 Score ↑
Base LSTM	45.42	0.7920
Stacked LSTM	45.22	0.7938
CNN + LSTM	54.45	0.7012
GRU (Best Model)	**38.45**	**0.8510**

Interpretation

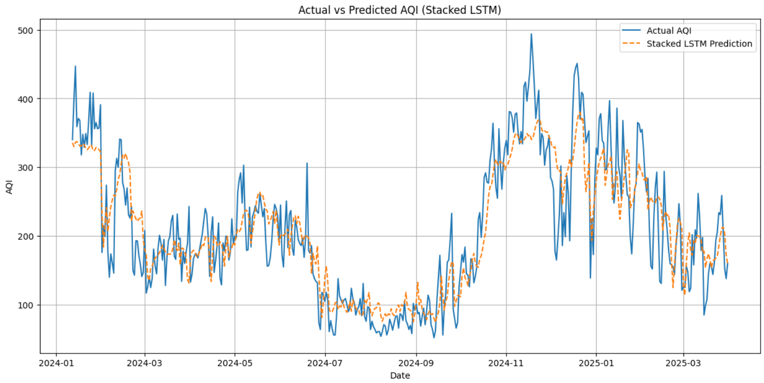
- **GRU achieved the best results**, with the lowest RMSE and highest R^2 score.
- Base LSTM and Stacked LSTM performed moderately well but frequently underpredicted sharp pollution spikes.
- CNN+LSTM had the weakest performance due to heavy smoothing and inability to capture fast fluctuations in AQI.

B. Outcome Graphs

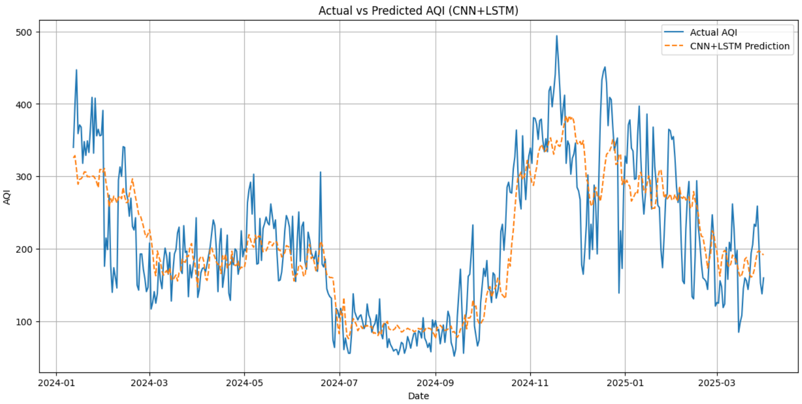
1. Actual vs Predicted AQI Trends



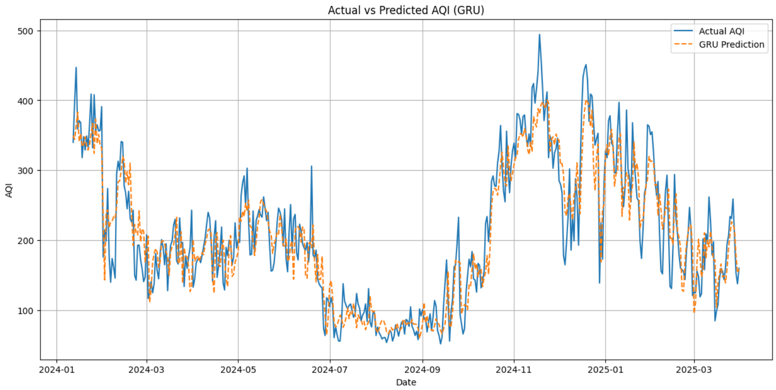
Base LSTM



Stacked LSTM



CNN+LSTM



GRU

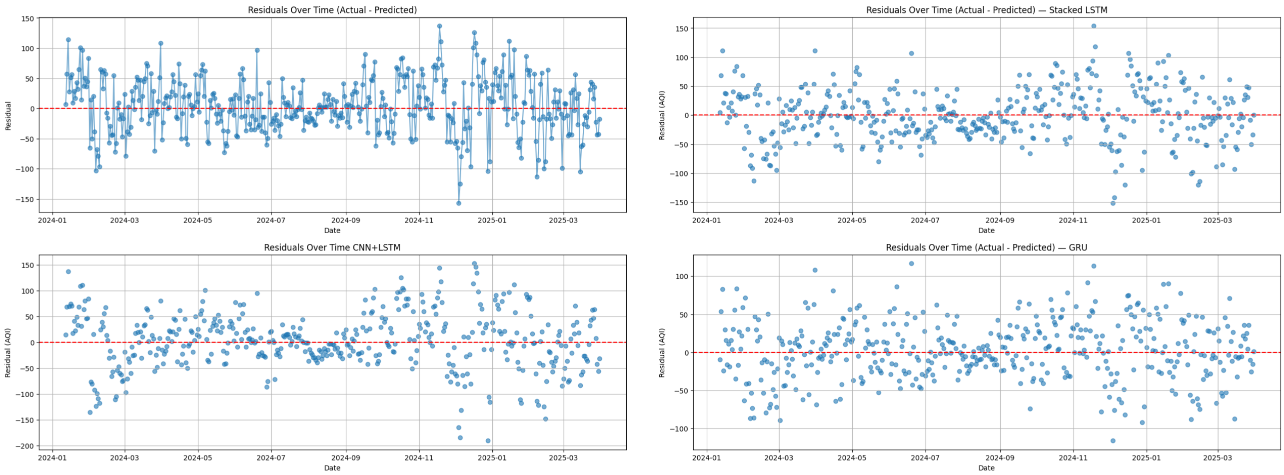
Key Observations

- **GRU** tracks seasonal peaks and daily fluctuations **closest** to the true values.
- **LSTM and Stacked LSTM** correctly capture the general trend but smooth out several sharp peaks.
- **CNN+LSTM** significantly underestimates rapid AQI rises and produces over-smoothed predictions.

Conclusion from Trend Analysis

GRU's predictions best match the *shape*, *timing*, and *magnitude* of AQI variations.

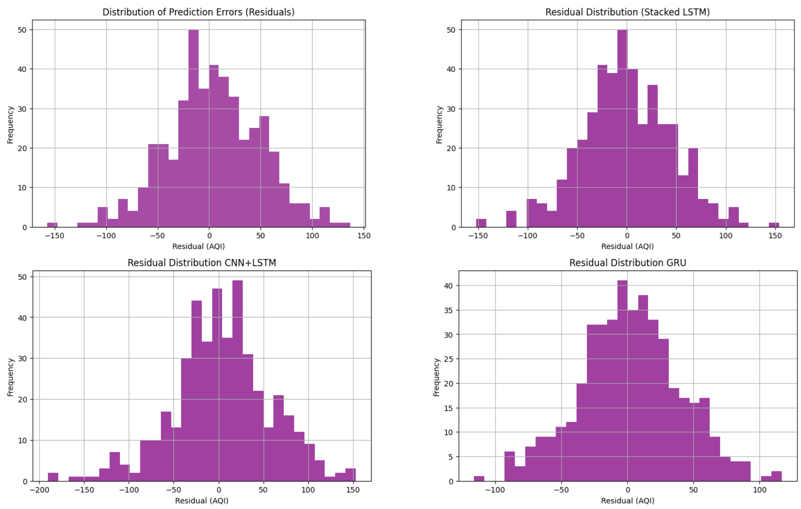
2. Residual Analysis



Insights

- GRU residuals are tightly centered around zero → low bias & low error variance.
- LSTM residuals have moderate spread.
- CNN+LSTM residuals show large deviations, indicating unstable predictions.

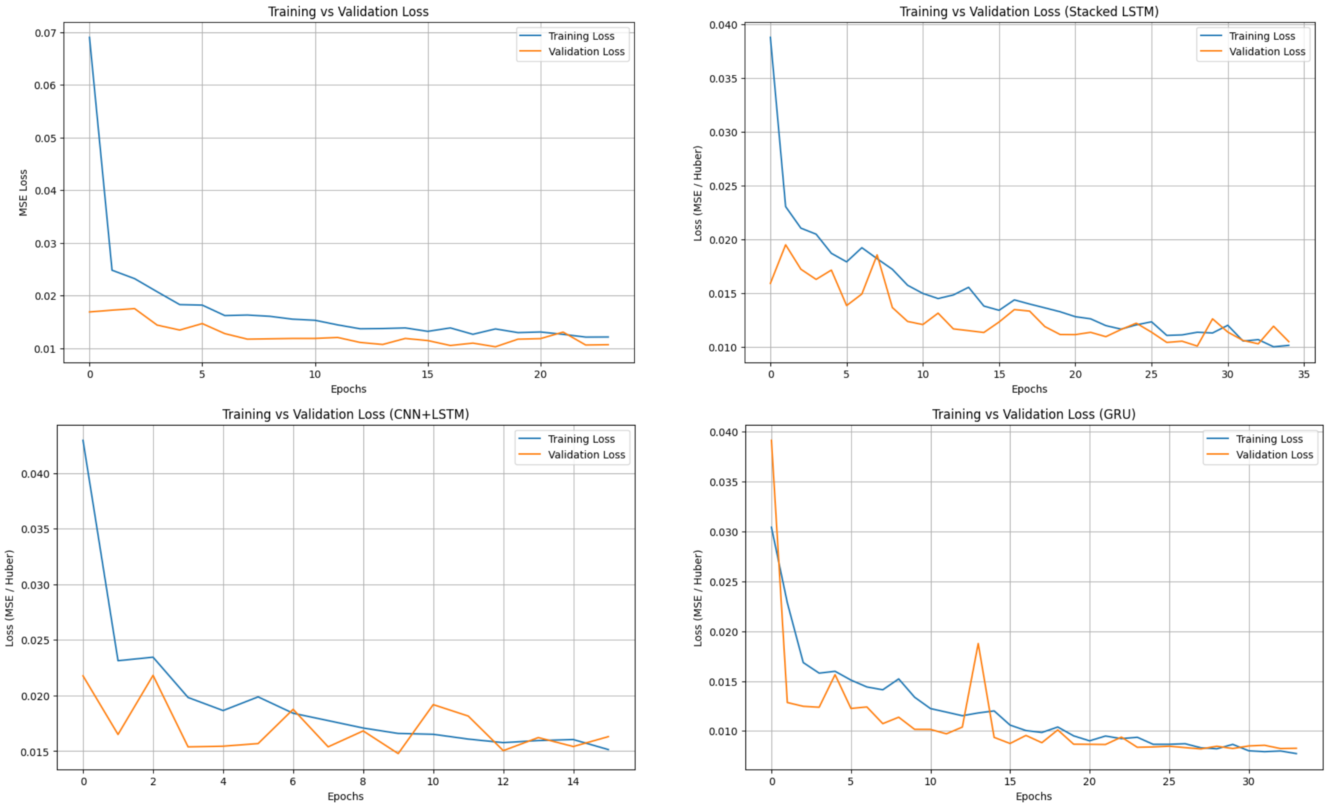
3. Error Distribution



Summary

- GRU errors fall mostly within **±40 AQI points**, much tighter than the other models.
- The distribution is symmetric, indicating consistent performance across conditions.

4.Training vs Validation Loss Curves



Insights

- GRU showed the most stable and smooth convergence, with training and validation loss decreasing consistently.
- LSTM and Stacked LSTM converged well but showed slight fluctuations due to model complexity.
- CNN+LSTM showed higher validation loss and clear signs of overfitting.

C. Comparative Studies

A comparative analysis was conducted to identify the best model based on performance metrics, visualization outputs, and temporal pattern learning.

1. Quantitative Comparison

Model	RMSE ↓	R^2 ↑
Base LSTM	45.42	0.7920
Stacked LSTM	45.22	0.7938
CNN + LSTM	54.45	0.7012
GRU	**38.45**	**0.8510**

Interpretation

- GRU performs the best across both metrics.
- Stacked LSTM provides only marginal improvement over base LSTM.
- CNN+LSTM performs the worst, struggling with rapid AQI fluctuations.

2. Visual Prediction Comparison

Actual vs Predicted Trends

- **GRU** tracks day-to-day AQI trends accurately and adapts well to sudden spikes.
- **LSTM/Stacked LSTM** follow the overall shape but miss sharp peaks.
- **CNN+LSTM** produces overly smooth outputs.

3. Model Complexity & Training Stability

Model	Training Stability	Overfitting Risk	Speed
LSTM	Stable	Moderate	Medium
Stacked LSTM	Less stable	Higher	Slow
CNN+LSTM	Unstable	High	Medium
GRU	**Most stable**	**Low**	**Fastest**

4. Overall Comparison & Final Verdict

Why GRU is the Best Model

- Lowest RMSE
- Highest R^2 score
- Best ability to capture sudden AQI changes
- Least overfitting
- Fastest training
- Handles noisy & volatile time series exceptionally well

Final Selection

GRU is chosen as the final forecasting model due to superior accuracy, robustness, and computational efficiency.

7. CONCLUSION

This project successfully developed a deep learning–based AQI forecasting system for Delhi using multiple neural architectures, including LSTM, Stacked LSTM, CNN+LSTM, and GRU. Through extensive preprocessing, feature engineering, time-series windowing, and model evaluation, the **GRU model** emerged as the most accurate and reliable forecasting approach.

GRU achieved the **lowest RMSE (38.45)** and **highest R^2 score (0.851)** on the test set, outperforming deeper LSTM models and hybrid CNN-based models. The analysis revealed that AQI is a highly volatile but strongly short-memory time series, making GRU particularly effective due to its simplified gating mechanism and strong ability to capture short-term dependencies.

The project also demonstrated the importance of engineered features such as lag values, rolling means, and EWMA, as well as cyclical encodings for capturing seasonality. Visual evaluations—including actual vs predicted curves, residual plots, and error distributions—further confirmed the superiority of the GRU model.

Overall, the system provides accurate next-day AQI predictions and serves as a foundation for scalable environmental forecasting systems.

A. Justification of Objectives

The objectives defined at the beginning of the project were fully met, as demonstrated below:

Objective 1: Build a predictive model for next-day AQI

- Achieved using deep learning models (LSTM, Stacked LSTM, CNN+LSTM, GRU).
- Final GRU model provides strong predictive accuracy.

Objective 2: Perform feature engineering to improve predictions

- Completed with lag features, rolling averages, EWMA, and cyclical encodings.
- Heatmap analysis shows these engineered features strongly correlate with AQI.

Objective 3: Compare multiple model architectures

- Four architectures implemented and evaluated

- Comprehensive comparison using RMSE, R^2, trend alignment, residuals, and errors

Objective 4: Identify the best model for AQI forecasting

- GRU model selected based on highest accuracy and stability.

Objective 5: Provide detailed visualizations and performance analysis

- Generated loss curves, trend plots, residual plots, error histograms, EDA graphs, and comparative charts.

All objectives were fulfilled, and the project outcomes align with the expected goals.

B. Future Scope

The AQI forecasting system can be further improved and expanded in several ways:

1. Multi-day Forecasting

Extend the model to predict AQI for the next **3–7 days** instead of only the next day.

2. Integration of Additional Data

Incorporate satellite aerosol data, traffic density, and crop-burning hotspots to improve accuracy.

3. Hyperparameter Optimization

Use automated tuning methods (Optuna, Bayesian Optimization) to further improve model performance.

4. Advanced Deep Learning Models

Explore Transformers, Temporal Fusion Transformers (TFT), or dilated CNNs for more powerful forecasting.

5. Deployment as a Live System

Build a real-time AQI prediction dashboard using FastAPI or Streamlit for public or institutional use.

8. REFERENCES

1. TensorFlow Official Documentation – TensorFlow Developers, Google.
2. Scikit-learn Documentation – Scikit-learn Machine Learning Library.
3. “Understanding GRU Networks” – Technical article on gated recurrent units and their applications in sequence modeling.
4. “A Beginner’s Guide to Recurrent Neural Networks (RNNs)” – Online article explaining RNN fundamentals and sequence learning behavior.
5. “Time-Series Forecasting with Deep Learning Models” – Industry article discussing LSTM, GRU, and hybrid models for forecasting.
6. Kaggle & public AQI datasets used for environmental trend analysis.