Eureka Open Access Journals
Open Access | Peer Reviewed | International Journals | Multidisciplinary Areas

Eureka Journal of Artificial Intelligence and Data Innovation (EJAIDI)

ISSN 2760-5000 (Online) Volume 2, Issue 1, January 2026

https://eurekaoa.com/index.php/11

DAILY MAXIMUM AQI PREDICTION IN TASHKENT USING CONVOLUTIONAL NEURAL NETWORKS (CNN)

Abdurasul Bobonazarov,
Vazira Holboeva
Department of Software Engineering, Millat Umidi University

Abstract

Air pollution remains one of the most critical environmental challenges in urban regions. Accurate forecasting of air quality indices enables authorities and citizens to take preventive measures against harmful exposure. This study presents a deep learning-based approach for predicting the next-day maximum Air Quality Index (AQI) in Tashkent, Uzbekistan, using a Convolutional Neural Network (CNN). Three years of open-access air quality and meteorological data collected from the People's Friendship Square monitoring station were aggregated at a daily level. The proposed multivariate CNN model leverages historical AQI, PM2.5 concentration, temperature, humidity, and precipitation values to forecast extreme pollution events. Experimental results demonstrate strong predictive performance, achieving an R^2 score of 0.793 and an RMSE of 43.22, indicating the model's capability to capture temporal patterns and pollution peaks effectively.

Keywords: Air Quality Prediction, Convolutional Neural Networks, Time Series Forecasting, Tashkent Air Pollution, Deep Learning, Environmental Monitoring

Introduction

Air pollution poses serious risks to public health, particularly in large metropolitan areas. In Uzbekistan, Tashkent experiences seasonal air quality

Eureka Open Access Journals
Open Access | Peer Reviewed | International Journals | Multidisciplinary Areas
Eureka Journal of Artificial Intelligence and Data Innovation (EJAIDI)
ISSN 2760-5000 (Online) Volume 2, Issue 1, January 2026

https://eurekaoa.com/index.php/11

fluctuations caused by traffic emissions, industrial activity, and meteorological conditions. Traditional statistical forecasting methods often struggle to model nonlinear temporal dependencies and extreme pollution events. Recently, deep learning techniques such as Convolutional Neural Networks (CNNs) have demonstrated strong performance in extracting temporal patterns from time-series data.

This study focuses on forecasting the next-day maximum AQI value for a selected monitoring station in Tashkent. Predicting daily maximum AQI is particularly important because it reflects peak exposure risks and is commonly used for issuing public air quality alerts. The objective is to design a robust pipeline that handles missing sensor data, aggregates hourly observations into meaningful daily indicators, and trains a CNN model capable of accurate next-day forecasting.

Example

Given historical daily air quality and weather observations for a monitoring station, the task is to predict the maximum AQI value for the following day. The model uses a sliding window of the previous 14 days as input, incorporating multiple environmental features. Formally, the prediction task can be defined as:

Input: $\{X(t-14), ..., X(t-1)\}$

Output: $AQI\_max(t)$

where X represents the multivariate feature vector containing AQI, PM2.5, temperature, humidity, and precipitation.

Implementation

Dataset Description

The dataset was obtained from the Open Data Tashkent platform and contains more than 255,000 hourly records collected between November 2022 and January 2026. Each record includes:

Eureka Open Access Journals
Open Access | Peer Reviewed | International Journals | Multidisciplinary Areas
Eureka Journal of Artificial Intelligence and Data Innovation (EJAIDI)
ISSN 2760-5000 (Online) Volume 2, Issue 1, January 2026

https://eurekaoa.com/index.php/11

- Date
- Time
- Station
- AQI
- Humidity
- Temperature
- Precipitation
- PM2.5

For this experiment, data from the "People's Friendship Square" monitoring station was selected. Hourly observations were aggregated into daily values to create a stable and interpretable time series.

Data Preprocessing

Several preprocessing steps were applied:

1. Datetime Construction: Date and time columns were merged into a single datetime index.
2. Station Filtering: Only records from the selected station were retained.
3. Daily Aggregation:
   ○ AQI was aggregated using the daily maximum value to capture pollution peaks.
   ○ PM2.5, temperature, humidity, and precipitation were aggregated using daily mean values.
4. Missing Value Handling: Temporal interpolation followed by forward and backward filling was used to repair missing days.
5. Noise Reduction: A 3-day rolling average smoothing filter was applied to reduce sensor noise and short-term volatility.

Eureka Open Access Journals
Open Access | Peer Reviewed | International Journals | Multidisciplinary Areas
OPEN ACCESS JOURNALS

Eureka Journal of Artificial Intelligence and Data Innovation (EJAIDI)

ISSN 2760-5000 (Online) Volume 2, Issue 1, January 2026

https://eurekaoa.com/index.php/11

Data Preprocessing

The dataset was divided chronologically into training (85%) and testing (15%) sets to preserve temporal integrity. Min-Max normalization was applied using statistics computed only from the training set to prevent data leakage.

Sliding Window Generation

A multivariate sliding window approach was used with a window size of 14 days. Each input sample consists of 14 consecutive days of five features, while the target output is the next day's maximum AQI value.

CNN Architecture

The CNN model architecture consists of:

- Two one-dimensional convolutional layers (64 and 32 filters) for temporal feature extraction
- A max-pooling layer for dimensionality reduction
- A fully connected dense layer with ReLU activation
- Dropout regularization (0.3) to reduce overfitting
- A single-neuron output layer for regression

```
# CNN model
model = Sequential()

model.add(Conv1D(64, kernel_size=3, activation='relu',
                 input_shape=(WINDOW_SIZE, X_train.shape[2])))

model.add(Conv1D(32, kernel_size=3, activation='relu'))

model.add(MaxPooling1D())

model.add(Flatten())

model.add(Dense(64, activation='relu'))
model.add(Dropout(0.3))

model.add(Dense(1))

model.compile(
    optimizer=tf.keras.optimizers.Adam(learning_rate=0.001),
    loss='mse'
)

model.summary()
```

Eureka Open Access Journals
Open Access | Peer Reviewed | International Journals | Multidisciplinary Areas
OPEN ACCESS JOURNALS

Eureka Journal of Artificial Intelligence and Data Innovation (EJAIDI)

ISSN 2760-5000 (Online) Volume 2, Issue 1, January 2026

https://eurekaoa.com/index.php/11

The model was trained using the Adam optimizer with a learning rate of 0.001 and Mean Squared Error (MSE) loss function.

Training Strategy

Early stopping was applied with a patience value of 15 epochs to prevent overfitting and unnecessary training. Although the maximum number of epochs was set to 120, the training process converged and stopped automatically at approximately 50 epochs, indicating stable learning behavior.

```
# traning
early_stop = EarlyStopping(
    monitor='val_loss',
    patience=15,
    restore_best_weights=True
)

history = model.fit(
    X_train, y_train,
    validation_split=0.15,
    epochs=120,
    batch_size=16,
    callbacks=[early_stop],
    verbose=1
)
```

Results

The trained model was evaluated on unseen test data using standard regression metrics.

Performance Metrics:

- Root Mean Squared Error (RMSE): 43.22
- Mean Absolute Error (MAE): 29.34
- Coefficient of Determination (R^2): 0.793

Eureka Open Access Journals
Open Access | Peer Reviewed | International Journals | Multidisciplinary Areas
OPEN ACCESS JOURNALS

Eureka Journal of Artificial Intelligence and Data Innovation (EJAIDI)

ISSN 2760-5000 (Online) Volume 2, Issue 1, January 2026

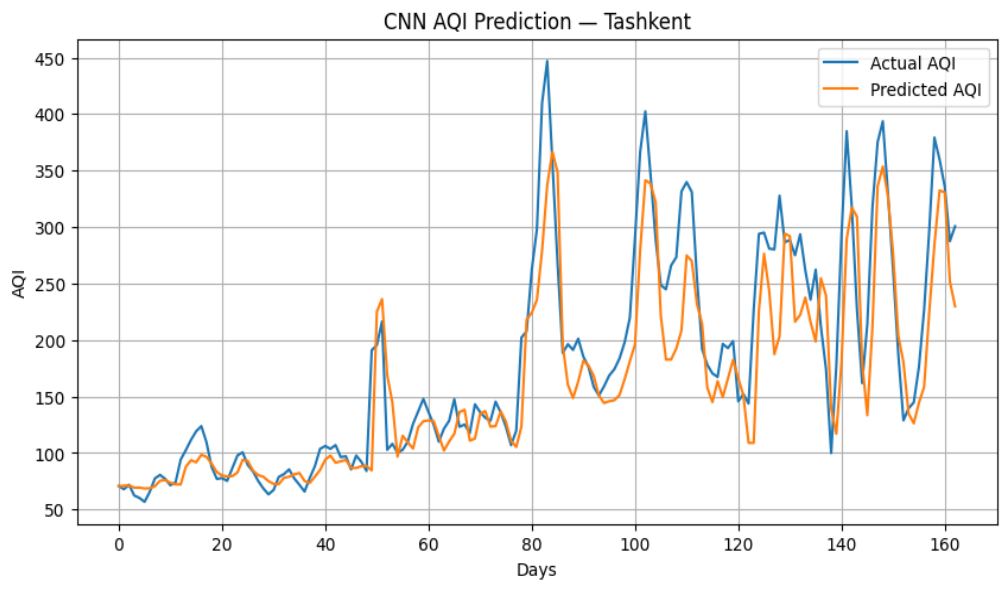
https://eurekaoa.com/index.php/11

```
# evaluation
y_pred_scaled = model.predict(X_test)

dummy = np.zeros((len(y_pred_scaled), daily_df.shape[1]))
dummy[:,0] = y_pred_scaled[:,0]

y_pred = scaler.inverse_transform(dummy)[:,0]

dummy2 = np.zeros((len(y_test), daily_df.shape[1]))
dummy2[:,0] = y_test

y_test_real = scaler.inverse_transform(dummy2)[:,0]

rmse = np.sqrt(mean_squared_error(y_test_real, y_pred))
mae = mean_absolute_error(y_test_real, y_pred)
r2 = r2_score(y_test_real, y_pred)

print("RMSE:", round(rmse,2))
print("MAE :", round(mae,2))
print("R2  :", round(r2,3))
```

[199]

```
6/6 ━━━━━━━━━━━━━━━━━━━━ 0s 12ms/step
RMSE: 43.22
MAE : 29.34
R2  : 0.793
```

The prediction visualization shows that the CNN model successfully tracks the general AQI trend and captures seasonal variations. Importantly, the revised pipeline demonstrates significantly improved performance in modeling peak pollution events compared to the baseline implementation.



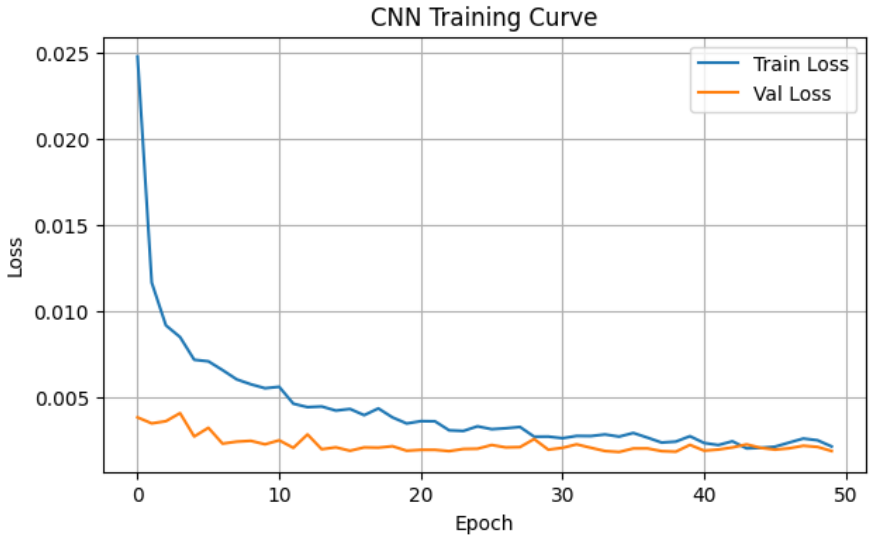
Eureka Open Access Journals
Open Access | Peer Reviewed | International Journals | Multidisciplinary Areas
OPEN ACCESS JOURNALS

Eureka Journal of Artificial Intelligence and Data Innovation (EJAIDI)

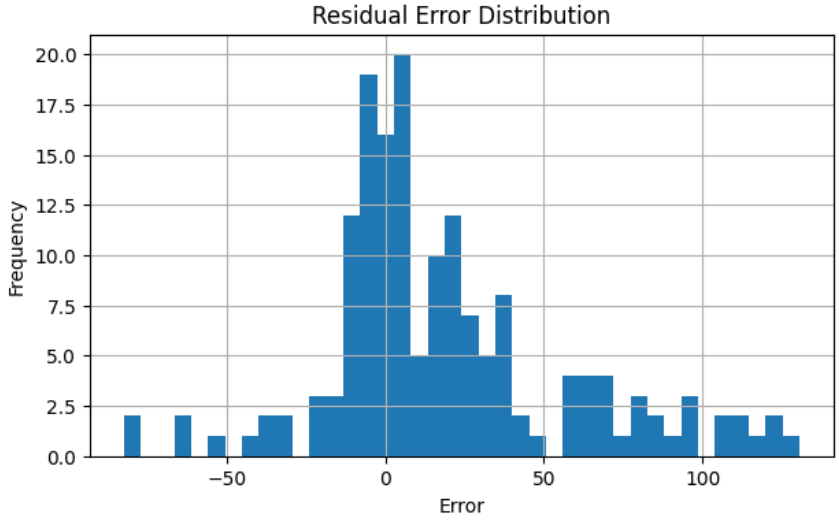
ISSN 2760-5000 (Online) Volume 2, Issue 1, January 2026

https://eurekaoa.com/index.php/11

The training and validation loss curves converge smoothly, indicating stable optimization and reduced overfitting.



The residual error distribution is centered closer to zero with reduced skewness, confirming that the model bias has been minimized after improving the preprocessing pipeline and feature representation.



Eureka Open Access Journals
Open Access | Peer Reviewed | International Journals | Multidisciplinary Areas
OPEN ACCESS JOURNALS

Eureka Journal of Artificial Intelligence and Data Innovation (EJAIDI)

ISSN 2760-5000 (Online) Volume 2, Issue 1, January 2026

https://eurekaoa.com/index.php/11

Conclusions

This study demonstrates the effectiveness of CNN-based deep learning models for next-day maximum AQI forecasting in Tashkent. By carefully handling missing data, performing daily aggregation, applying smoothing techniques, and utilizing multivariate temporal inputs, the proposed approach achieved strong predictive performance.

The results indicate that CNN models can serve as reliable tools for short-term air quality forecasting and can support public health decision-making systems. Future improvements may include incorporating additional meteorological features, integrating spatial information from multiple stations, and extending the model to multi-day forecasting horizons.

References

1. Open Data Tashkent Platform – Air Quality Dataset. https://opendata.tashkent.uz/eng/data/133?tab=1
2. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature. https://www.nature.com/articles/nature14539
3. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press. https://www.deeplearningbook.org/
4. Chollet, F. (2018). Deep Learning with Python. Manning Publications. https://deeplearningwithpython.io/chapters/
5. Code Implementation with Jupyter Notebook on my GitHub. https://github.com/Holboeva/tashkent-aqi-prediction.