



**AI-BASED CONTENT ANALYTICS FOR
IOT-DRIVEN SMART CITIES**

FARIA FATIMA

09-244242-001

A thesis submitted to fulfill the requisites for awarding
the degree of Master of Science (Electrical Engineering)

Department of Electrical Engineering

BAHRIA UNIVERSITY ISLAMABAD (H 11- CAMPUS)

MS-13

Thesis Completion Certificate

Name: Faria Fatima

Registration Number: 09-244242-001

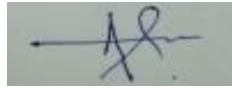
Program of Study: MS in Electrical Engineering

Thesis Title: ***AI-Based Content Analytics for IoT-Driven Smart Cities***

Hereby I certify that the said scholar's thesis is completed and has met the standards.

prerequisites for thesis submission. Meanwhile, the thesis is submitted for examination in accordance with the requirements of the MS program.

Principal Supervisor Signature:



Date: __6/09/2026_____

Name: Dr. Adil Ali Raja

MS-14 A

Author`s Declaration

I Faria Fatima hereby state that my MS thesis "**AI-Based Content Analytics for IoT-Driven Smart Cities**" is my own work and has not been submitted previously by me for taking any degree from "**Bahria University, Islamabad**" or anywhere else in the country / world.

At any time if my statement is found to be incorrect even after my graduation, the university has the right to withdraw / cancel my MS degree.

FARIA FATIMA

Registration No.: 09-244242-001__

Sep, 2026

Author's Sign: ____**FARIA FATIMA**____

Name of Scholar: Faria Fatima

MS-14 B

Plagiarism Undertaking

I solemnly declare that the research work presented in this thesis titled:

“AI-Based Content Analytics for IoT-Driven Smart Cities”

is purely my own research work. All sources and previous work used in framing the baseline of this thesis are cited, and the whole thesis is written by me. I duly understand and acknowledge the policy of zero tolerance towards plagiarism of HEC and **Bahria University**. I declare here that no single line/portion of this thesis is plagiarized or goes without citation. I hereby declare that if I am found guilty of plagiarism in this thesis, even after the award of the MS degree, the University has the right to revoke my MS degree and HEC can publish my name among the names of scholars who submitted plagiarized thesis.

Author's Sign: ____FARIA FATIMA_____

Name of Scholar: Faria Fatima

Acknowledgements

I am highly grateful to my supervisor, **Dr. Adil Ali Raja**, for his continuous support, guidance and for showing full confidence in my capabilities throughout my thesis. I am also thankful to my department and all those who supported me during the completion of this research.

also wish to convey my sincere thanks to my parents and especially my father,

Dr. Muhammad Touseef Tahir who constantly encourage me in my study. He's always been there for me and his counsel has made me who I am today. I am equally thankful to my mother, **Sana Touseef** for her unconditional love, prayers, encouragement and consistent support which have been a source of strength during my educational path. Their sacrifices, devotion and unending belief in me were key factors in my achieving this accomplishment.

I am very thankful to my mentor **Dr. Maryam Iqbal** for her guidance that motivated me to do a thesis instead of a course and paved the way for my adventure. She has been my consistent support and encouragement since my very first semester, freely sharing her guidance and support. Without her, this feat would not have been possible, and I will always be grateful to her for believing in me.

I am very thankful to my husband **Engr. Muhammad Sami Shoukat** whose unconditioned support, patience and understanding have been the base of my achievement. Through every moment of doubt and difficulty that this journey presented, his steadfast belief in me never faltered and his encouragement gave me the strength to persevere when the path forward seemed uncertain. Words are insufficient to express my appreciation for his constant presence by my side and I dedicate this accomplishment, in no small part, to him.

Abstract

A vast amount of diversified data from sensors, service platforms, and environmental monitoring systems has resulted from the quick development of smart cities and Internet of Things (IoT) infrastructure. IoT devices provide real-time data collection. However, the majority of structured data is processed by current analytics techniques, which restricts the identification of valuable insights required for real-time urban decision-making. This gap drives the creation of an AI-based content analytics platform to enable informed city administration by analyzing a range of smart city data streams. The thesis focuses on three key areas of the general idea of AI-based content analytics for IoT-driven smart cities: air quality monitoring and prediction, traffic congestion and detection, and citizen complaint analytics. The METR-LA traffic dataset is used to detect congested traffic using Random Forest, Decision Tree, K-Nearest Neighbours, LSTM, and CNN-LSTM models. Accuracy, precision, recall, F1-score, MAE, and RMSE are assessed. Support Vector Machine, Random Forest, and Decision Tree classifiers are applied to the Chicago 311 service request dataset, and citizen complaint analytics are evaluated using accuracy, precision, recall, F1-score, and confusion matrix analysis. Machine learning and deep learning regression models for monitoring and forecasting air quality are assessed using the Beijing PM2.5 data set. We utilize machine learning and deep learning regression models for monitoring and forecasting air quality using the Beijing PM2.5 data set. The models are assessed using R, RMSE, and MAE. The models chosen depend on the task and modality. Sequence models may identify temporal patterns in traffic, tree-based models work well for classification on tabular data, and the granularity and feature representation of the data determine how well deep learning models perform for regression.

TABLE OF CONTENTS

CHAPTER 1 – INTRODUCTION

- 1.1 Background
- 1.2 Problem Statement
- 1.3 Objectives
- 1.4 Scope of the Study
- 1.5 Significance of the Study

CHAPTER 2 – LITERATURE REVIEW

- 2.1 Introduction
- 2.2 Smart Cities and Internet of Things (IoT)
- 2.3 Artificial Intelligence in Smart Cities
- 2.4 Traffic Forecasting and Congestion Analysis
- 2.5 Citizen Complaint Analytics
- 2.6 Air-Quality Monitoring and Prediction
- 2.7 LSTM and CNN-LSTM Models
 - 2.7.1 Long Short-Term Memory
 - 2.7.2 CNN-LSTM
- 2.8 Evaluation of AI Models
- 2.9 Research Gap

CHAPTER 3 – METHODOLOGY

- 3.1 Overview
- 3.2 Proposed Architecture
 - 3.2.1 Smart-City Data Sources
 - 3.2.2 Dataset Layer
 - 3.2.3 Data Preprocessing Layer
 - 3.2.4 AI Content Analytics Layer
 - 3.2.5 Evaluation Layer
- 3.3 Dataset Layer
 - 3.3.1 METR-LA Traffic Dataset
 - 3.3.2 Citizen Complaint Dataset and Experimental Analysis
 - 3.3.3 Beijing PM2.5 Dataset
- 3.4 Data Preprocessing
 - 3.4.1 Traffic Data Preprocessing
 - 3.4.2 Air-Quality Data Preprocessing

- 3.4.3 Citizen Complaint Data Preprocessing
- 3.5 AI Content Analytics
 - 3.5.1 Traffic Content Analytics
 - 3.5.2 Citizen`s Complaint Content Analytics
 - 3.5.3 Content Analytics for Air Quality
- 3.6 Evaluation Metrics
 - 3.6.1 Mean Absolute Error
 - 3.6.2 Root Mean Square Error
 - 3.6.3 Mean Absolute Percentage Error
 - 3.6.4 Weighted Absolute Percentage Error
 - 3.6.5 Coefficient of Determination

CHAPTER 4 – RESULTS AND DISCUSSION

- 4.1 Traffic Forecasting and Congestion Analysis
 - 4.1.1 Introduction
 - 4.1.2 Experimental Dataset and Evaluation Configuration
 - 4.1.3 Data Preprocessing and Sequence Preparation
 - 4.1.4 LSTM Baseline Results
 - 4.1.5 CNN-LSTM Results
 - 4.1.6 Overall Comparison with the Persistence Baseline
 - 4.1.7 Improvement over the Persistence Baseline
 - 4.1.8 Horizon-Wise Forecasting Performance
 - 4.1.9 CNN-LSTM versus LSTM Across Forecasting Horizons
 - 4.1.10 Error and Residual Analysis
 - 4.1.11 Error Variation with Forecasting Horizon
 - 4.1.12 Error Analysis by Traffic-Speed Level
 - 4.1.13 Statistical Significance and Paired Error Analysis
 - 4.1.14 Multi-Seed Robustness Analysis
 - 4.1.15 Consolidated Model Comparison
 - 4.1.16 Discussion of Main Findings
 - 4.1.16.1 Deep Learning versus Persistence
 - 4.1.16.2 LSTM versus CNN-LSTM
 - 4.1.16.3 Effect of Forecasting Horizon
 - 4.1.16.4 Traffic-Speed Conditions
 - 4.1.16.5 Statistical versus Practical Significance
 - 4.1.17 Key Experimental Findings
 - 4.1.18 Chapter Summary
- 4.2 Air Quality Monitoring and Prediction
 - 4.2.1 Experimental Objective
 - 4.2.2 Dataset Description

- 4.2.3 Exploratory Data Analysis
- 4.2.4 Feature Engineering
- 4.2.5 Chronological Data Splitting
- 4.2.6 Feature Scaling and Sequence Generation
- 4.2.7 LSTM Model
- 4.2.8 CNN-LSTM Model
- 4.2.9 Overall Prediction Performance
- 4.2.10 Comparative Error Analysis
- 4.2.11 Statistical Significance Analysis
- 4.2.12 Robustness and Paired Win Analysis
- 4.2.13 Bootstrap Confidence-Interval Analysis
- 4.2.14 Discussion of Results
- 4.2.15 Implications for IoT-Driven Smart Cities
- 4.2.16 Summary of Air Quality Experiment
- 4.3 Citizen Complaint Analytics
 - 4.3.1 Overview
 - 4.3.2 Model Development
 - 4.3.3 Model Performance
 - 4.3.4 Error Analysis
 - 4.3.5 Discussion
 - 4.3.6 Summary
- 4.4 Cross-Experiment Interpretation

CHAPTER 5 – DISCUSSION

- 5.1 Overview
- 5.2 Traffic Forecasting
- 5.3 Air-Quality Prediction
- 5.4 Citizen Complaint Analytics
- 5.5 Overall Discussion

CHAPTER 6 – CONCLUSION AND FUTURE WORK

- 6.1 Conclusion
- 6.2 Key Findings
- 6.3 Limitations
- 6.4 Future Work
- 6.5 Final Statement

REFERENCES

CHAPTER 1

INTRODUCTION

1.1 Background

Urban governance has become more complex due to rapid urbanization. Cities must deal with issues like air pollution, traffic jams, infrastructure constraints, public service delivery, and resident communication. Smart cities use technologies such as the Internet of Things (IoT), sensors, digital platforms and data analysis to deal with such difficulties. IoT devices continuously acquire huge data from many sources but data collection alone is not sufficient. The major difficulty is to turn these data into meaningful information, to assist the city officials to take better and faster decisions. Artificial intelligence can help to recognize patterns, find anomalies, classify information and forecast outcomes. In this work, we focus on three main directions of AI-based content analytics for IoT driven smart cities; traffic analytics, citizen complaint analytics and air-quality prediction. These are essential areas because transport, public services and environmental factors directly affect the quality of urban living. The traffic analysis is carried out on the METR-LA dataset containing data from 207 traffic sensors with a sampling rate of 5 minutes. LSTM and CNN-LSTM models are trained to predict traffic speed for the next 60 minutes based on the last 60 minutes. Furthermore, the research forecasts the subsequent hour's PM2.5 levels utilizing LSTM and CNN-LSTM models grounded in prior meteorological and environmental information.

A third direction is the analytics of citizens' complaints. With AI, complaints can be organized, categorized and prioritized, repeated complaints can be detected, and trends or problem areas can be identified. In this study, the Chicago 311 service request dataset is experimentally evaluated using AI-based classification techniques. The examination centers on structuring and categorising citizen service enquiries to discern trends that can aid in public service oversight and decision-making. The primary aim of the study is to illustrate how conventional artificial intelligence methodologies may convert diverse smart-city data streams into insightful information for

forecasting

and

decision-making.



Figure 1.1. Role of IoT-based sensing and connectivity in enabling a smart city.

1.2 Problem Statement

Intelligent urban areas produce vast amounts of data from Internet of Things sensors, ecological systems, transportation networks and community platforms. This data is typically complicated, continuously changing and available in numerous formats making it challenging to analyze using standard approaches. Traffic condition is dynamic and the air pollution is altered with elements such as time and weather. Citizen complaints can also have unstructured information which is difficult to analyze manually. Traditional monitoring systems are mainly reactive in nature, as they only provide information after a problem has already occurred. Deep learning models and conventional artificial intelligence techniques offer better opportunities for prediction and analysis, but a more complex model does not always yield better results. There is therefore a need to assess various AI models and develop an integrated approach that can transform smart-city data into useful and timely information for decision-making.

1.3 Objectives

1. To develop a unified AI-based framework for the analysis of traffic, citizen complaints and air-quality data in smart city environments.
2. To evaluate and compare selected AI and deep learning techniques for traffic forecast and congestion analysis, air quality prediction and citizen complaint classification

3. To analyze results obtained from traffic, citizen complaint and air-quality datasets and derive actionable information to support smart-city monitoring and decision making.

1.4 Scope of the Study

This research employs artificial intelligence and machine learning methodologies to tackle three specific smart city analytical challenges: traffic prediction and congestion assessment, classification of citizen complaints, and forecasting air quality. Traffic analysis uses historical traffic data to study traffic conditions and to forecast future traffic patterns. Citizen Complaint Analytics In order to classify and detect trends in citizen complaints, Chicago 311 service request data are analyzed. Tradition air quality data is utilised to create and assess predictive models for air quality evaluation. The study analyzes chosen machine-learning and deep-learning techniques along with their performance indicators for each of the analytical tasks. The work is confined to the experimental analysis based on datasets and does not entail actual deployment of IoT sensors or a real-time smart-city monitoring system.



Figure 1.2. Conceptual flow of the proposed AI-based content analytics

The scope is limited to the datasets, analytical techniques and experimental tasks selected and described in the methodology. The results are expected to show how AI-based analytics can support smart-city monitoring and data-driven decision making.

1.5 Significance of the Study

This research is significant as it illustrates the utilization of AI to examine diverse categories of smart-city data inside a unified framework. It delivers AI-driven analytics that integrate traffic management, environmental monitoring and citizen services. Such a framework can help improve decision-making by city authorities by providing traffic forecasts, air-quality predictions and useful information derived from citizen complaints. The study is technical in nature, comparing LSTM and CNN-LSTM models to assess whether a more complex model actually results in better performance. The results show that LSTM is a robust and stable model, while CNN-LSTM can achieve similar performance in some cases.

AI-Based Content Analytics for IoT-Driven Smart Cities

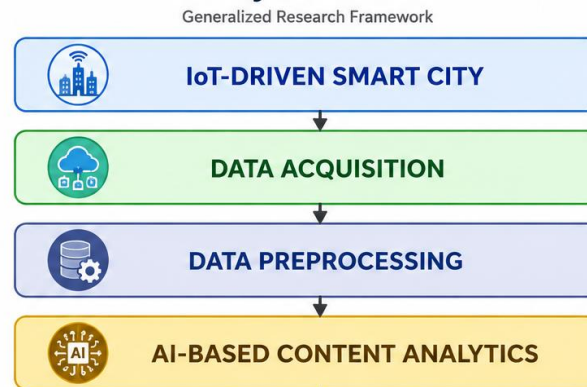


Figure 1.3. Generalized research framework showing the IoT-driven smart city.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The combined use of Artificial Intelligence (AI) with an Internet of Things (IoT) has emerged as a pivotal strategy for cultivating intelligent, data-centric smart cities. As the proliferation of IoT devices continues, the volume of urban data has expanded significantly, and this data can be examined by AI methodologies for forecasting, categorization, and identification. This chapter reviews the literature on AI-based analytics for smart cities, focusing on traffic congestion, citizen complaints and air-quality forecasting. It additionally encompasses the deep-learning frameworks and assessment standards employed in this research., with particular focus on AI-based content analytics for IoT-driven smart cities. The review includes smart-city principles, IoT-based urban data generation, AI and ML approaches, data analytics and the three application domains studied in this thesis: traffic analytics, citizen complaint analytics, and air-quality monitoring and prediction. Finally, existing limitations are discussed and the research gap addressed by the proposed work is outlined.

2.2 Smart Cities and Internet of Things

A smart city uses digital technology, connected infrastructure and data-driven systems to improve urban services, sustainability and decision-making. Through the Internet of Things (IoT), sensors and connected equipment continuously collect urban data. Smart-city applications enabled by IoT can monitor traffic conditions, environmental parameters, energy consumption, public services and infrastructures. But data collection is not enough – smart analytical procedures are needed to turn raw data into meaningful information. The combination of IoT and AI thus provides the framework to develop smart urban monitoring and decision-support systems.

2.3 Artificial Intelligence in Smart Cities

Artificial intelligence is useful in smart-city systems to find patterns, predict, organize information and identify anomalies. Decision Tree, Random Forest and K-Nearest Neighbors (KNN) algorithms are among the most popular for structured urban data. Deep learning models are extremely useful for complex and sequential data since they are able to understand nonlinear and temporal correlations. The selection of an AI technique is based on the data type, analysis and time parameters of the application.

2.4 Traffic Forecasting and Congestion Analysis

In contemporary cities, traffic congestion is a major issue that increases travel times, fuel consumption and pollution. Traffic sensors based on IoT enable continuous measurements of the traffic flow, which can be used by artificial intelligence algorithms to identify traffic congestion patterns and anomalous traffic conditions. Traditional machine-learning models can detect static traffic conditions, while sequence-based deep-learning models can learn the temporal relationships present in traffic data. In this work, traffic analytics is studied using the METR-LA dataset, which contains real-world observations from traffic sensors for AI-based analysis.

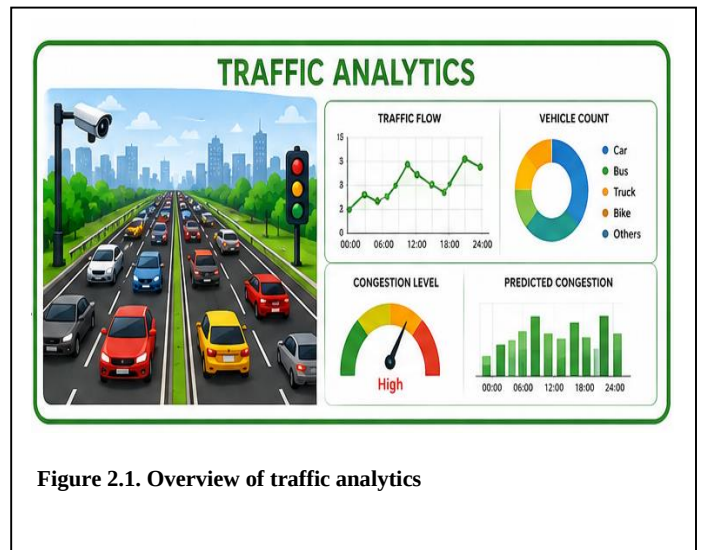


Figure 2.1. Overview of traffic analytics

2.5 Citizen`s Complaint Analytics

Citizen service requests yield useful data on urban problems and the demand for public services. Systems such as Chicago 311 provide massive datasets containing complaint types, locations, responsible agencies and time information. AI-based classification can automatically sort citizen complaints and recognize trends in service requests. This research examines the Random Forest and Decision Tree models for the categorization of citizen grievances, assessing their efficacy through conventional classification metrics and confusion matrices.

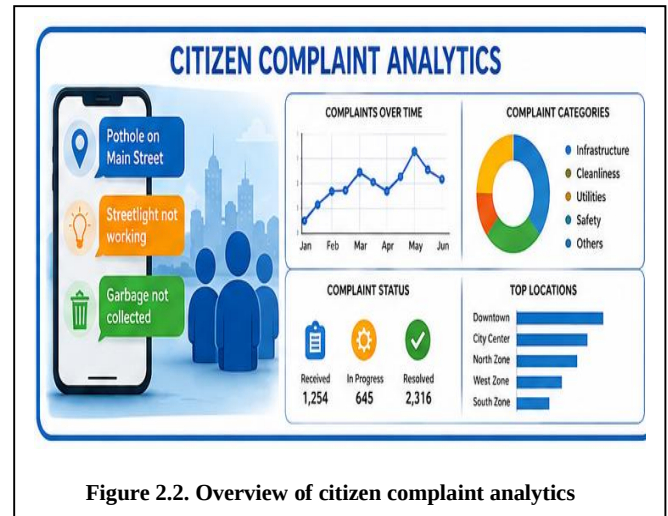


Figure 2.2. Overview of citizen complaint analytics

2.6 Air-Quality Monitoring and Prediction

One major environmental risk in smart cities is air pollution. Real-time data on contaminants and climate variables can be gathered via IoT-based monitoring systems, allowing AI models to predict future pollution levels. PM_{2.5} is particularly important, as it varies over time and is influenced by meteorological and temporal conditions. The Beijing PM_{2.5} data collection is used in this study to investigate air quality prediction using deep learning algorithms. The performance of the LSTM and CNN-LSTM models is compared using prediction-error and goodness-of-fit metrics.

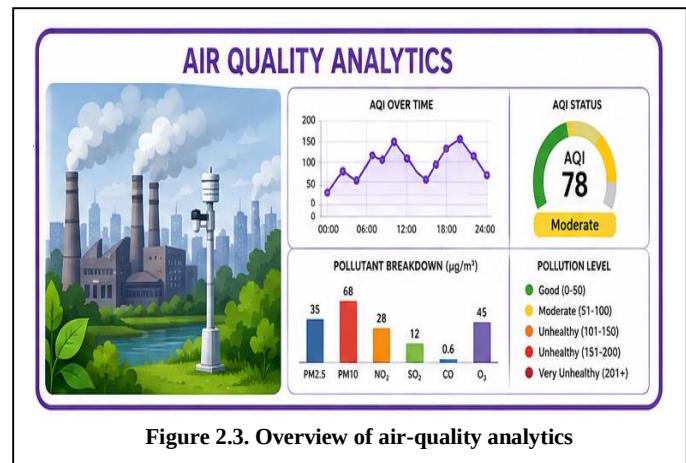


Figure 2.3. Overview of air-quality analytics

2.7 LSTM and CNN LSTM Models

Different types of urban data are better suited for different AI algorithms. While deep learning models are particularly effective at understanding intricate temporal connections from sequential data, traditional machine learning methods work well for many structured classification applications.

2.7.1 Long Short Term Memory

LSTM is a variant of recurrent neural networks that acquires dependencies within sequential data. Its memory system enables knowledge from prior time intervals to influence subsequent forecasts. This solves the problem of ordinary recurrent networks. LSTM is particularly adept at projecting time-series data, as traffic and air-quality metrics are contingent on time. In this research, LSTM is utilized as a deep-learning framework for forecasting air quality and as a temporal modeling method for traffic assessment when relevant.

2.7.2 CNN LSTM

A Convolutional Neural Network (CNN) and an LSTM are combined to create CNN-LSTM. The CNN component that captures local patterns or features, while the LSTM component learns their temporal dependencies. This design can be effective when the sequential data contains relevant local patterns. In this research, CNN-LSTM is compared with LSTM for PM2.5 prediction level to determine whether convolutional feature extraction provides additional predictive advantage.

2.8 Evaluation of AI Models

AI models ought to be assessed utilising appropriate performance metrics. For regression and forecasting assignments, often employed metrics consist of Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), Weighted Absolute Percentage Error (WAPE), and the coefficient of determination (R^2).

2.9 Research Gap

Prior research has demonstrated the efficacy of AI and IoT in numerous smart-city initiatives, including transportation, environmental surveillance and public services. Firstly, most of the studies focus on a single application or data modality, while smart cities collect a diverse set of data from sensors, residents and environmental monitoring systems. Second, model performance is often evaluated with limited metrics, without in-depth error analysis or comparative assessment. model performance is often evaluated using limited metrics, without in depth error analysis or comparative assessment. Third, the performance of an AI model can depend heavily on data

characteristics and the specific analytical task. This research addresses these gaps by offering a comparative, AI-based analytics strategy across three distinct smart-city applications:

- Traffic Forecasting and Congestion Analysis
- Citizen`s Complaint Analytics
- Monitoring and Forecasting of Air Quality
-

The study analyzes several AI algorithms in terms of task-specific metrics and evaluates their performance on heterogeneous urban datasets.

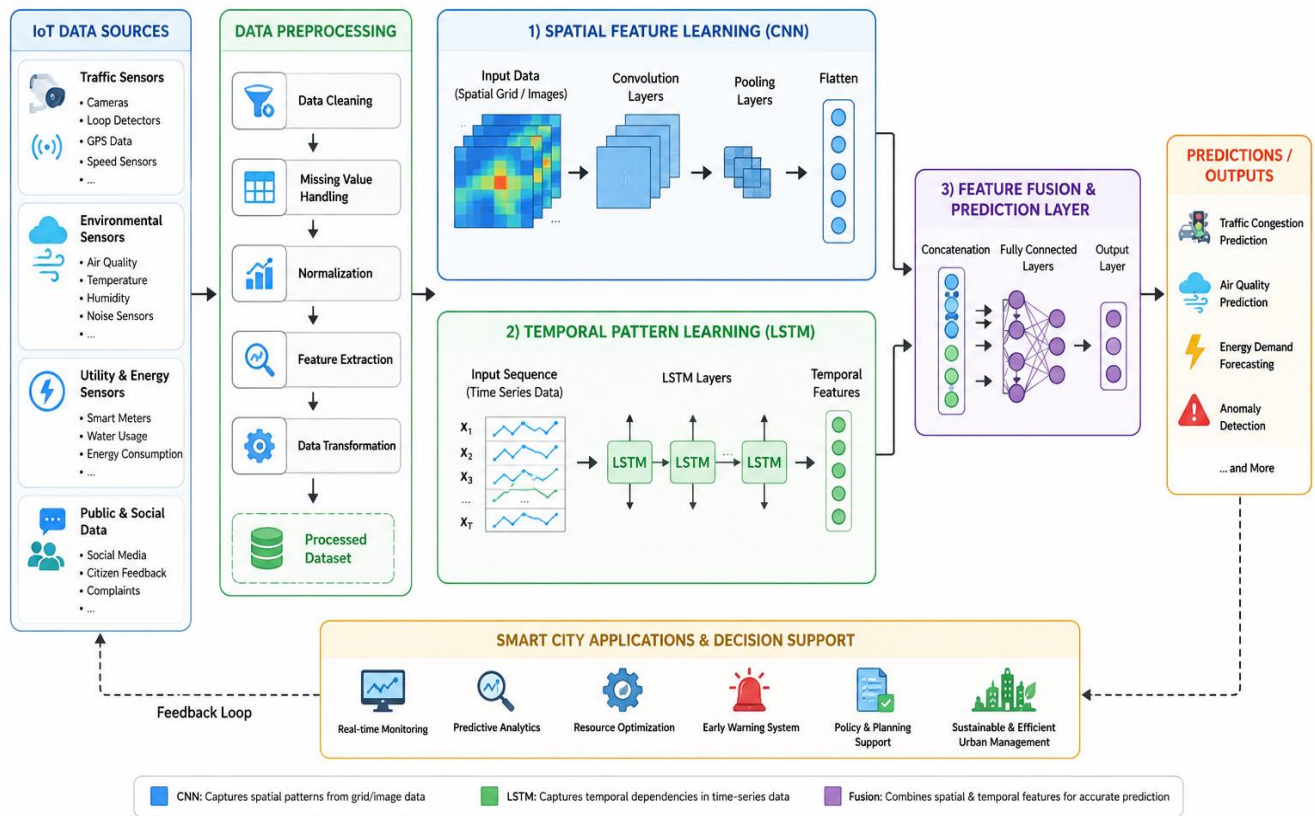


Figure 2.4. Generalized LSTM–CNN hybrid deep-learning framework for IoT-driven smart city.

CHAPTER 3

METHODOLOGY

3.1 Overview

This section outlines the methodology employed in the development and assessment of an AI-centric content analytics platform tailored for IoT-enabled smart cities. The suggested methodology is aimed to process different types of smart-city data and to transform raw data into meaningful information for monitoring, prediction and decision making. The research covers three primary application areas: traffic analytics, citizen complaint analytics and air-quality monitoring and prediction. The technique comprises of five basic stages, including the suggested architecture, dataset layer, preprocessing of data, AI-based content analytics and assessment of the produced models. Two application domains, traffic and air pollution are experimentally examined using real-world datasets. Citizen complaint analytics is experimentally evaluated using the Chicago 311 service request dataset. Machine learning classification models are applied to the dataset and evaluated using standard classification metrics. The historical traffic observations for the last 60 minutes are utilized to predict the traffic speed for the next 60 minutes. A persistent baseline is utilized as a reference model and LSTM and CNN-LSTM are used as deep-learning techniques. The Beijing PM2.5 data set is utilized for air-quality analytics. We forecast PM2.5 level one hour ahead in the future with a historical window of 24 hours. We constructed LSTM and CNN-LSTM models and compared them to see whether the addition of convolutional feature extraction enhanced the performance of the predictions. The citizen complaint component provides a methodology for the use of AI on citizen-generated service request information. The proposed analytics include: complaint classification, urgency identification, duplicate complaint detection, trend analysis and location based analysis.

3.2 Proposed Architecture

The suggested architecture provides a unified framework for applying AI-based content analytics to heterogeneous smart-city data. This architecture is meant to support data coming from IoT sensors, environmental monitoring systems and citizen service platforms.

The framework consists of the following major layers:

Proposed Smart-City Analytics Architecture

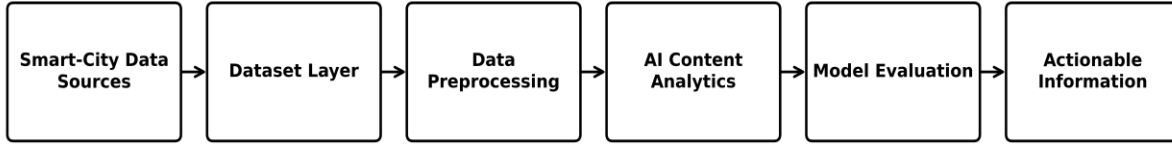


Figure 3.1. Proposed smart-city analytics architecture.

3.2.1 Smart-City Data Sources

The first layer is the sources of information generation for smart-city. Traffic sensors constantly record information about transportation, environmental monitoring systems record pollutant and meteorological measurements, and citizen-service platforms record details about complaints and service requests from the public. In this research, the representative datasets for these sources include the METR-LA dataset for traffic analytics, the Beijing PM2.5 data set for air-quality forecast and the suggested citizen complaint analytics framework.

3.2.2 Dataset Layer

The data needed for the analytical activities is stored and arranged in the dataset layer. Each dataset is processed based on its features because the three application domains contain distinct variables and data structures. While the Beijing dataset includes climatic and pollution variables recorded over time, the traffic dataset is made up of time-series observations from several traffic sensors. Chicago 311 citizen complaint data would include details about service requests, including the type of complaint, the location, the time and the accountable department.

3.2.3 Data Preprocessing Layer

AI models cannot be directly fed raw smart-city data. Preprocessing is performed to improve data quality and prepare datasets for model creation. Data inspection, handling missing values, temporal processing, feature preparation, normalization and input output sequence generation are some of these procedures.

3.2.4 AI Content Analytics Layer

The AI layer uses appropriate analytical methods based on the features of each application. CNN-LSTM, LSTM, and Persistence models are taken into consideration for traffic forecasting. LSTM and CNN-LSTM models are used to predict air quality. The suggested framework outlines location-based analysis, trend analysis, duplicate detection, AI-based classification and classification by priority for citizen complaints.

3.2.5 Evaluation Layer

Finally, the last layer measures the performance of the experimental models using task-specific indicators. The metrics used to evaluate the performance of traffic forecasting are MAE, RMSE, MAPE and WAPE; for air quality prediction, the metrics are MAE, RMSE and R2. Thus the architecture offers a common methodological framework while allowing the use of different AI techniques depending on the requirements of each smart-city application.

3.3 Dataset Layer

The suggested AI-based smart-city analytics framework is built on the dataset layer. This study examines three smart-city application areas: traffic, citizen complaints and air quality.

3.3.1 METR-LA Traffic Dataset

The METR-LA dataset is for short-term traffic-speed forecasting, which incorporates the observations of 207 traffic sensors every five minutes. In this study, the previous 60 minutes 12 observations are used as input to forecast traffic speed for the following 60 minutes. A persistence baseline, LSTM and CNN-LSTM are compared to assess their short-term forecasting performance. The dataset is used to compare a persistence baseline with LSTM and CNN-LSTM models. The purpose is to determine whether deep-learning models can provide improved short-term traffic forecasts compared with a simple baseline.

3.3.2 Citizen Complaint Dataset and Experimental Analysis

The Chicago 311 dataset is used for citizen complaint analytics. It contains data on complaint types, locations, timestamps and service details. The information serves as a foundation for using AI methods that include KNN, RF and confusion matrix to categorize complaints, determine priority, find duplicate requests, examine trends and carry out location based analysis. This element facilitates the incorporation of data created by citizens into the suggested framework for smart-city analytics.

3.3.3 Beijing PM2.5 Dataset

The data set of Beijing PM2.5 is used to monitor and anticipate the air quality. The dataset contains meteorological and environmental factors and the historical PM2.5 data. Since the PM2.5 concentration level is an important indicator of air pollution and it is dependent on environmental and temporal factors, we choose it as the main prediction target. The PM2.5 is predicted one hour ahead of time with a 24-hour history frame. The prepared time-series data are used to train and evaluate LSTM and CNN-LSTM models.

3.4 Data Preprocessing

As the datasets are obtained from different smart-city sources and have different formats, data preparation is an important step in the proposed methodology. Chicago 311 citizen complaint dataset were cleaned and transformed according to their respective data characteristics and experimental objectives.

3.4.1 Traffic Data Preprocessing

Traffic sensor readings are studied and sorted chronologically for prediction from the METR-LA dataset. The data is transformed into sequential input-output samples and missing or invalid values are handled accordingly. The next 60 minutes are predicted using a 60-minute historical window (12 observations). Training, validation, and testing sets of data are separated temporally, and only training data is used for scaling.

3.4.2 Air-Quality Data Preprocessing

Air quality analytics is mainly dedicated to PM2.5 forecasting using CNN-LSTM and LSTM models. The PM2.5 level is predicted one hour in advance using a 24-hour history frame. The data is split sequentially into training, and testing sets to prevent data leaks. Training data is used for scaling.

3.4.3 Citizen Complaint Data Preprocessing

In order to compile the Chicago 311 dataset, duplicate and missing entries are handled, timestamps and locations are processed and complaint categories and service details are arranged. For AI-based analytics, such as complaint classification, priority identification, duplication detection, trend analysis, and location-based analysis, important category, temporal, geographical and textual elements are prepared.

3.5 AI Content Analytics

This section describes the AI techniques used to transform preprocessed smart-city data into predictions and actionable analytical information.

3.5.1 Traffic Content Analytics

Short-term traffic-speed forecasting employing CNN-LSTM, LSTM and persistent models is the main emphasis of traffic analytics. The next 60 minutes are predicted using the traffic data from the last 60 minutes. LSTM learns temporal dependencies, CNN-LSTM integrates temporal modelling and local feature extraction, and persistence offers a baseline. The chosen forecasting metrics are used to compare the models.

3.5.2 Citizen`s Complaint Content Analytics

The Chicago 311 dataset is used by citizen complaint analytics to enable AI-based analysis of requests for public services. Trend analysis, location-based analysis, duplication detection, priority identification and complaint classification are among the suggested analytics. The goal of these

studies is to turn data created by citizens into information that can be used for public service monitoring and decision-making.

3.5.3 Content Analytics for Air Quality

The main contribution of air quality analytics is the PM2.5 forecasting using the CNN-LSTM and LSTM models. PM2.5 level is predicted one hour in advance using a 24-hour history frame. To find out if CNN-based feature extraction outperforms LSTM in terms of prediction performance, the two models are compared.

3.6 Evaluation Metrics

Different evaluation measures are selected according to the nature of the prediction task. Since traffic and air-quality experiments are forecasting problems, regression based metrics are used.

3.6.1 Mean Absolute Error

MAE (Mean Absolute Error) The average of the absolute error that exists between the expected and true values.

$$MAE = (1/n) \sum |y_i - \hat{y}_i| \quad (3.1)$$

Lower MAE means better forecast accuracy.

3.6.2 Root Mean Square Error

The Root Mean Square Error (RMSE) is the square root of the mean of the squared prediction errors.

$$RMSE = \sqrt{(1/n) \sum (y_i - \hat{y}_i)^2} \quad (3.2)$$

RMSE is effective for detecting models with large predicting mistakes.

3.6.3 Mean Absolute Percentage Error

The Mean Absolute Percentage Error (MAPE) is a forecast error as a percentage of the actual number.

$$MAPE = (100/n) \sum |(y_i - \hat{y}_i) / y_i| \quad (3.3)$$

A lower MAPE means a better forecasting performance.

3.6.4 Weighted Absolute Percentage Error

Weighted Absolute Percentage Error (WAPE) is the sum of absolute error of the prediction divided by the sum magnitude of the actual observation.

$$WAPE = [\sum |y_i - \hat{y}_i| / \sum |y_i|] \times 100 \quad (3.4)$$

The traffic forecast experiment uses WAPE. It is used to provide an additional scale-aware measure of the prediction performance.

3.6.5 Coefficient of Determination

R^2 is a measure of the goodness of fit of the projected values to explain the variation of the actual observations.

$$R^2 = 1 - [\sum (y_i - \hat{y}_i)^2 / \sum (y_i - \bar{y})^2] \quad (3.5)$$

where $\bar{y}$ is the mean of the actual observations.

A higher R^2 generally indicates better agreement.

Chapter 3 Methodology Flow

The complete methodology can therefore be represented as:

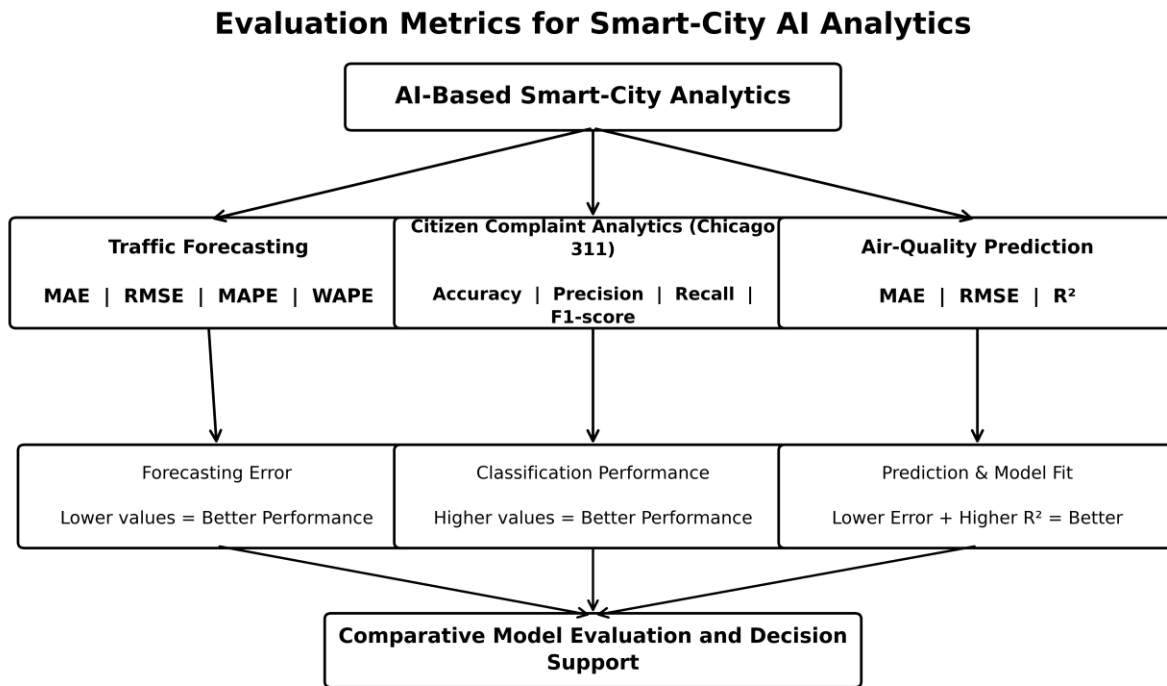


Figure 3.2. Conceptual evaluation-metrics framework for the three smart-city AI analytics.

CHAPTER 4

Results and Discussion

4.1 Traffic Forecasting and Congestion Analysis

4.1.1 Introduction

In this chapter, we report experimental results on multi-step traffic-speed forecasting on the METR-LA dataset. The project aimed at investigating the performance of deep-learning architectures in predicting the traffic speed for the next 60 minutes based on the last 60 minutes of traffic observations.

Three forecasting approaches were evaluated:

1. Persistence baseline,
2. Long Short-Term Memory (LSTM), and
3. Convolutional Neural Network and Long Short Term Memory (CNN-LSTM).

The models were assessed utilizing Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE) and Weighted Absolute Percentage Error (WAPE). Alongside overall performance, the analysis included horizon-specific performance, error traits, robustness across random seeds, bootstrap confidence intervals and paired effect sizes.

The purpose of this analysis is not only to identify the model with the lowest error but also to determine whether the observed differences are consistent, practically meaningful, and robust to changes in model initialization.

4.1.2 Experimental Dataset and Evaluation Configuration

The investigations were carried out utilizing the METR-LA traffic-speed dataset. The collection comprises traffic-speed measurements gathered from 207 sensors. The forecasting formulation used a 60-minute historical input window to predict traffic speed for the following 60 minutes. The sequence-generation procedure produced 4,461 training sequences, 706 validation sequences, and 286 test sequences. Each input sequence contained 12 time steps, corresponding to 60 minutes because observations were considered at five-minute intervals. Similarly, each target sequence contained 12 future time steps.

Table 4.1**Experimental data**

Parameter	Value
Dataset	METR-LA
No of sensors	207
Time interval	5 minutes
Input history	60 minutes
Prediction horizon	60 minutes
Input time steps	12
Output time steps	12
Training sequences	4,461
Validation sequences	706
Testing sequences	286
Models	Persistence, LSTM, CNN-LSTM

The experimental design therefore represents a multi-sensor, multi-step traffic forecasting problem in which the models receive one hour of historical traffic information and generate predictions for the subsequent hour.

4.1.3 Data Preprocessing and Sequence Preparation

The preprocessing analysis showed that the METR-LA dataset contains missing and zero traffic-speed observations, including extended gaps affecting multiple sensors. Consequently, preprocessing was treated as an important component of the experimental methodology rather than merely a preliminary technical operation.

The final forecasting experiment used valid and properly processed forecasting sequences. Long missing periods were not artificially reconstructed without sufficient evidence. This approach reduced the risk of introducing unsupported information into the forecasting problem. After preprocessing, the data were converted into supervised learning sequences. Each sequence

contained 12 historical observations from 207 sensors, while the corresponding target consisted of the next 12 observations from the same 207 sensors. The data was divided chronologically into training, validation, and testing subsets. This sequential organization was employed to maintain the temporal aspect of traffic prediction and to minimize the risk of information seepage from future data into the training phase. The resulting arrays were:

- $(X_{\text{train}} = (4461, 12, 207))$
- $(y_{\text{train}} = (4461, 12, 207))$
- $(X_{\text{val}} = (706, 12, 207))$
- $(y_{\text{val}} = (706, 12, 207))$
- $(X_{\text{test}} = (286, 12, 207))$
- $(y_{\text{test}} = (286, 12, 207))$

4.1.4 LSTM Baseline Results

Initially, the main deep-learning baseline was the LSTM model. This model was intended to serve as a benchmark for assessing the CNN-LSTM architecture. On the initial traffic-speed scale, the LSTM obtained an overall MAE of 3.6912 and an RMSE of 6.5029. Its WAPE was 6.2543% and its MAPE was 10.3332%. These findings show that a significant amount of the temporal structure in the traffic-speed data was captured by the LSTM. In particular, its superior performance under the squared-error criterion is illustrated by its relatively low RMSE in comparison to the other evaluated models.

4.1.5 CNN-LSTM Results

Then we experimented with the CNN-LSTM architecture to see if the convolutional feature extraction before the temporal processing can improve the traffic speed forecasts. The CNN-LSTM was composed of convolutional layers followed by an LSTM layer and fully connected output layers. The model held 413,940 trainable parameters. The model achieved the best validation loss at epoch 9 with a validation loss of roughly 0.3805 and validation MAE of around 0.3546 during training. On the original test scale, the CNN-LSTM achieved:

- **MAE = 3.6846**
- **RMSE = 6.6701**
- **MAPE = 10.2443%**

- WAPE = **6.2430%**

The CNN-LSTM therefore achieved the lowest overall MAE and WAPE among the three evaluated approaches.

4.1.6 Overall Comparison with the Persistence Baseline

A persistence forecasting approach was used as a non-learning baseline. In this technique, the forecast was the most recently observed traffic-speed value. The persistence baseline achieved MAE of 3.7973, a RMSE of 7.0914, a MAPE of 9.3137%, and a WAPE of 6.4340%. The full comparison is shown in Table 4.2.

Table 4.2 Overall Performance of Persistence, LSTM, and CNN-LSTM

Model	MAE	RMSE	MAPE (%)	WAPE (%)
Persistence	3.797314	7.091432	9.313734	6.434046
LSTM	3.691220	6.502861	10.333219	6.254285
CNN-LSTM	3.684600	6.670100	10.244300	6.243000

The lowest MAE was attained by the CNN-LSTM with a value of 3.6846, whereas the LSTM had an MAE of 3.6912. The LSTM achieved lowest RMSE of 6.5029. The persistence baseline performed best with a MAPE of 9.3137%. Thus, no single model performed best across all evaluation metrics. The CNN-LSTM showed an improvement over the persistence baseline of around 2.97% for MAE, while the LSTM showed an improvement of around 2.79%. For RMSE, the LSTM improved by around 8.30% compared to persistence, and the CNN-LSTM by about 5.94%. The MAPE results need to be interpreted more cautiously. Persistence produced the lowest MAPE; however, MAPE can be heavily influenced by observations with relatively low actual speed values. Thus, MAE, RMSE, and WAPE provide vital supplementary evidence of the overall quality of the forecast.

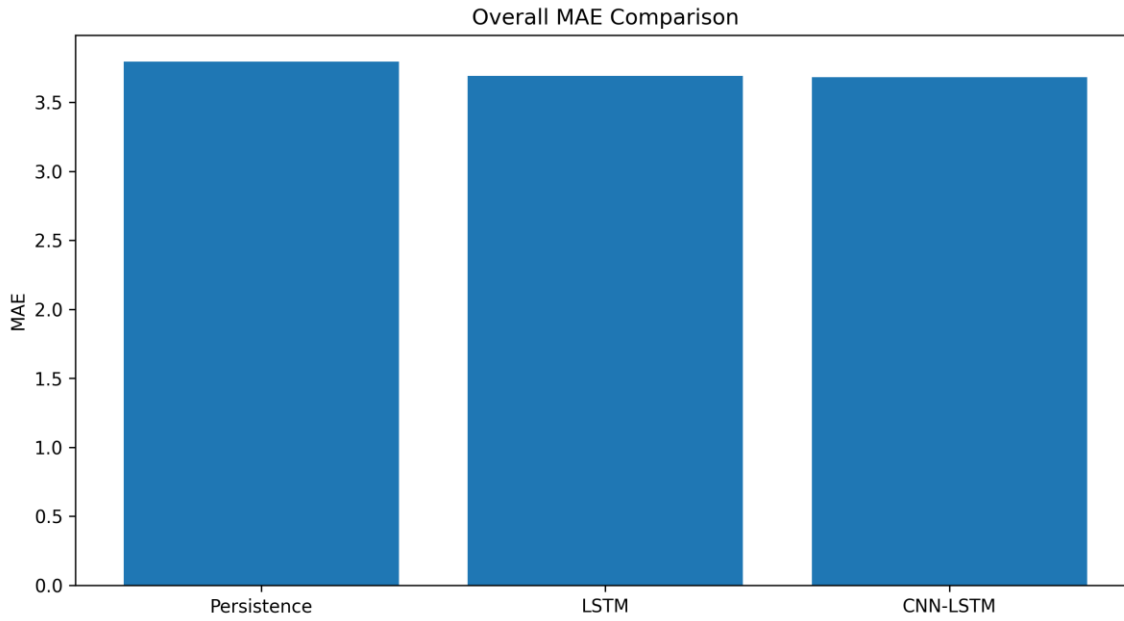


Figure 4.1. Overall MAE comparison among Persistence, LSTM and CNN-LSTM.

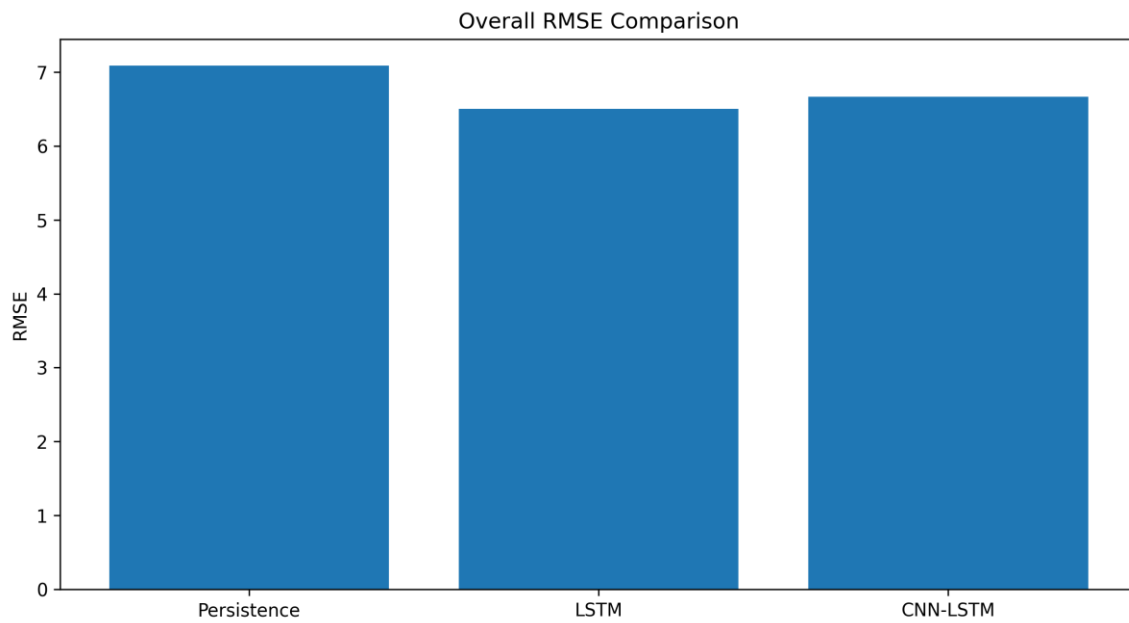


Figure 4.2. Overall RMSE comparison among Persistence, LSTM and CNN-LSTM.

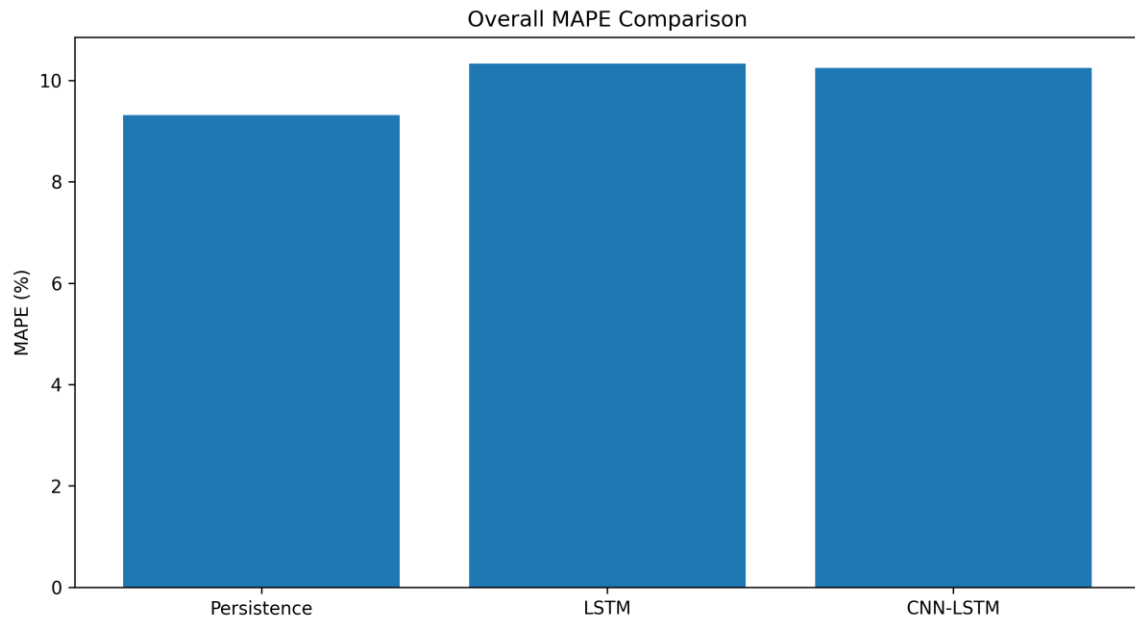


Figure 4.3. Overall MAPE comparison among Persistence, LSTM and CNN-LSTM.

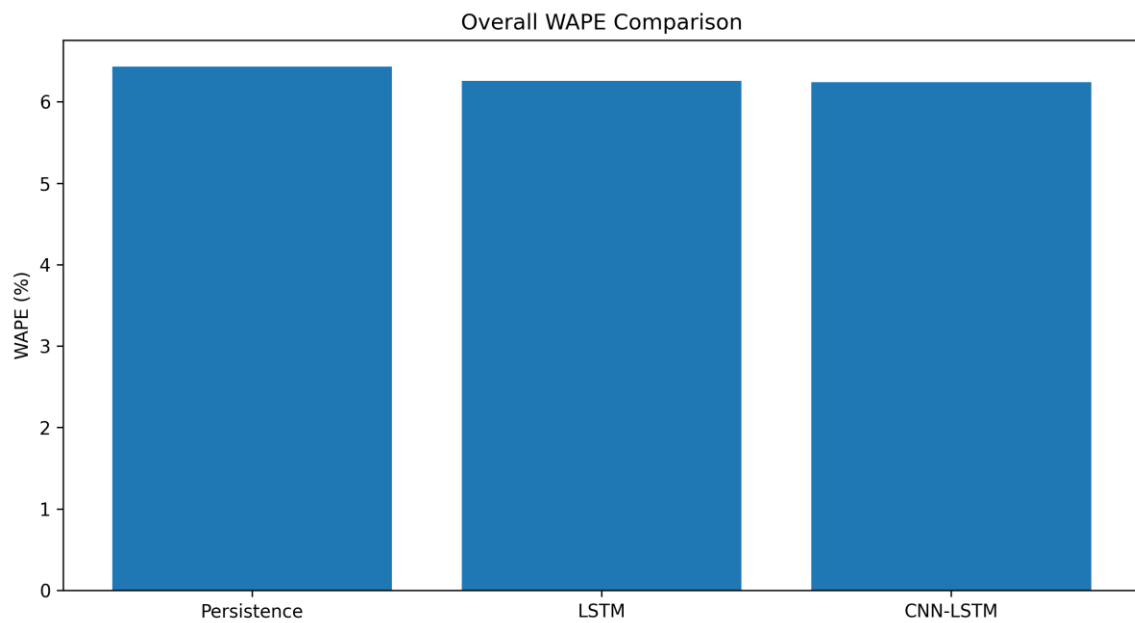


Figure 4.4. Overall WAPE comparison among Persistence, LSTM and CNN-LSTM.

4.1.7 Improvement over the Persistence Baseline

Table 4.3 summarizes the percentage improvement of the learning-based models relative to the persistence baseline.

Table 4.3

Improvement over Persistence Baseline

Model	MAE Improvement (%)	RMSE Improvement (%)	MAPE Improvement (%)	WAPE Improvement (%)
Persistence	0.000000	0.000000	0.000000	0.000000
LSTM	2.793922	8.299748	-10.946039	2.793903
CNN-LSTM	2.968256	5.941423	-9.991331	2.969298

Positive values indicate improvement over the persistence baseline, whereas the negative MAPE values indicate that the learning-based models produced higher MAPE than persistence. The results show that both learning-based models improved upon persistence in MAE, RMSE, and WAPE. CNN-LSTM achieved the highest MAE and WAPE improvements, while LSTM achieved the largest RMSE improvement. However, the improvement was relatively modest in terms of absolute MAE, indicating that short-term traffic-speed persistence remains a strong benchmark for this dataset.

4.1.8 Horizon-Wise Forecasting Performance

Because the experiment involved forecasting 12 future time steps, performance was also evaluated separately at each forecasting horizon.

Table 4.4**Horizon-Wise MAE, RMSE and MAPE Results**

Horizon	Minutes Ahead	Persistence MAE	LSTM MAE	CNN-LSTM MAE	Persistence RMSE	LSTM RMSE	CNN-LSTM RMSE	Persistence MAPE	LSTM MAPE	CNN-LSTM MAPE
1	5	2.718892	3.371302	3.416762	4.315944	5.892008	6.143479	5.623274	9.593363	9.649815
2	10	3.019232	3.410344	3.451976	5.126944	5.980339	6.230892	6.551089	9.605048	9.663661
3	15	3.225719	3.450124	3.489497	5.683478	6.078616	6.333680	7.213750	9.683542	9.712894
4	20	3.417892	3.514728	3.542583	6.152943	6.188146	6.438095	7.879617	9.773520	9.808271
5	25	3.572750	3.572101	3.583852	6.530742	6.314229	6.516298	8.429654	9.930370	9.897171
6	30	3.734620	3.638206	3.629869	6.911327	6.401088	6.591505	9.018400	10.081143	10.009706
7	35	3.924180	3.702319	3.703912	7.316801	6.506178	6.688224	9.647884	10.230441	10.148920
8	40	4.079447	3.767508	3.744596	7.647839	6.602147	6.764255	10.233901	10.395715	10.267240
9	45	4.246342	3.836479	3.803102	7.982349	6.735180	6.857803	10.864928	10.666964	10.476204
10	50	4.376852	3.920531	3.873957	8.271153	6.877107	6.983370	11.429374	10.970678	10.756461
11	55	4.549599	4.017969	3.955016	8.610520	7.072503	7.135641	12.092129	11.354253	11.101566
12	60	4.702243	4.093033	4.019954	8.926050	7.230750	7.255535	12.780805	11.713587	11.439809

The forecasting inaccuracy clearly increases with an increase in the prediction horizon.

E.g., LSTM MAE went from 3.3713 at 5 minutes to 4.0930 at 60 minutes. Similarly, CNN-LSTM MAE has grown from 3.4168 to 4.0200. The persistence model has a better short-term performance with an MAE of 2.7189 at the 5-minute horizon. However, its inaccuracy increased rapidly with the forecasting horizon to 4.7022 at 60 minutes. This pattern is essential since persistence is naturally strong for extremely short forecasting horizons, when the most recent observation is still a fair estimate of the near-future traffic status. The use of the learnt temporal patterns increases

when the horizon is lengthened. At the 60-minute horizon, CNN-LSTM achieved an MAE of 4.0200, compared to 4.0930 for LSTM and 4.7022 for persistence. Therefore, CNN-LSTM outperformed persistence by an average of 14.51% MAE at the longest forecast horizon.

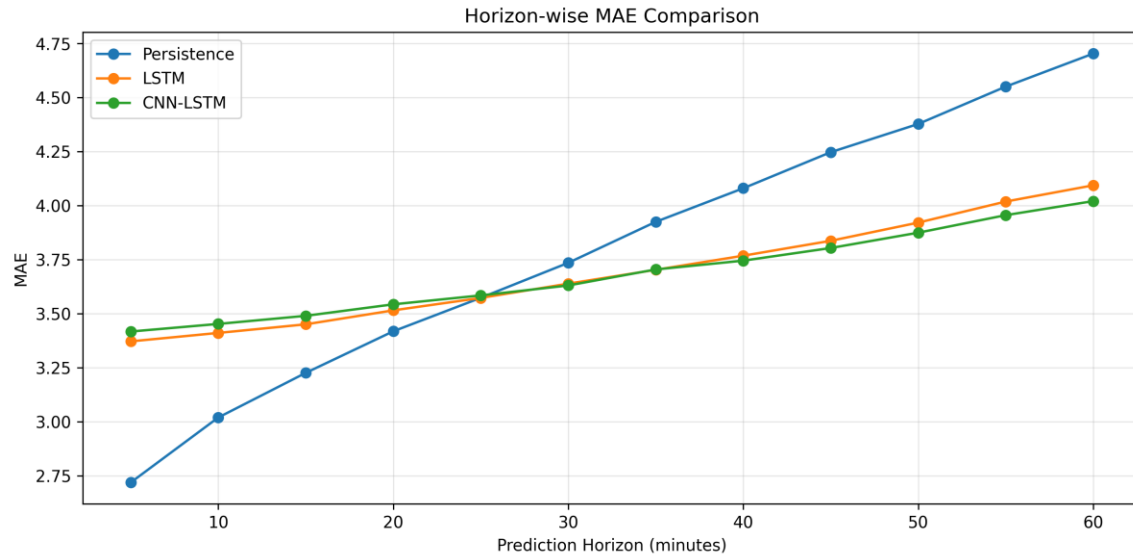


Figure 4.5. Horizon-wise MAE comparison for Persistence, LSTM and CNN-LSTM.

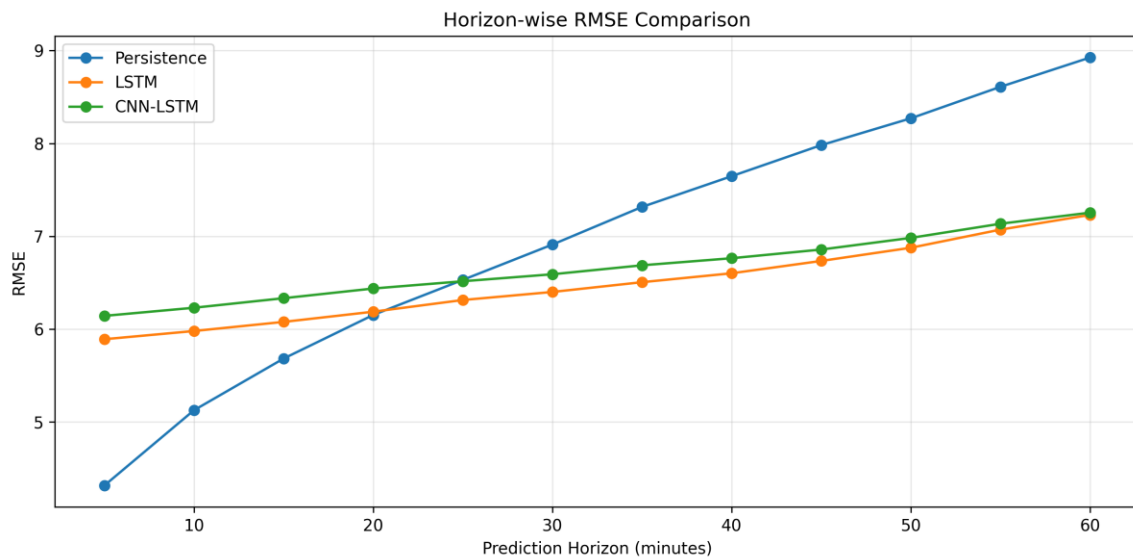


Figure 4.6. Horizon-wise RMSE comparison for Persistence, LSTM and CNN-LSTM.

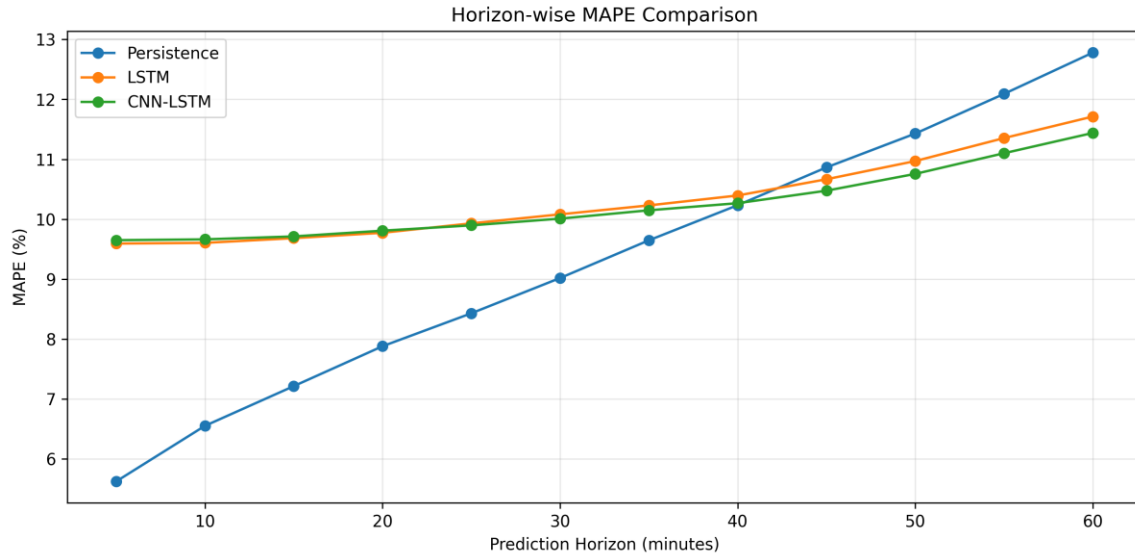


Figure 4.7. Horizon-wise MAPE comparison for Persistence, LSTM and CNN-LSTM.

4.1.9 CNN-LSTM versus LSTM Across Forecasting Horizons

The comparison between the two deep-learning models provides a more specific assessment of the contribution of the convolutional component.

Table 4.5

CNN-LSTM Percentage Change Relative to LSTM by Forecasting Horizon

Horizon	Minutes Ahead	MAE Change (%)	RMSE Change (%)	MAPE Change (%)
1	5	-1.348440	-4.268002	-0.588448
2	10	-1.220757	-4.189612	-0.610231
3	15	-1.141205	-4.196087	-0.303112
4	20	-0.792522	-4.039158	-0.355563
5	25	-0.328966	-3.200217	0.334318
6	30	0.229151	-2.974760	0.708620
7	35	-0.043027	-2.798048	0.796847
8	40	0.608147	-2.455383	1.235846
9	45	0.869990	-1.820634	1.788325
10	50	1.187951	-1.545170	1.952632
11	55	1.566787	-0.892725	2.225483
12	60	1.785449	-0.342772	2.337269

The negative values in this table imply that CNN-LSTM performed worse than LSTM on the relevant metric, while the positive values show improvement. The results show that CNN-LSTM does not always outperform LSTM at any forecast horizon. LSTM often has lower MAE at short horizons. However, CNN-LSTM was more competitive with longer forecasting horizon. For MAE, CNN-LSTM performed better than LSTM at 30 minutes ahead and at the 40, 45, 50, 55 and 60 minute horizons. At 60 minutes CNN-LSTM had an MAE improvement of 1.7854% vs to LSTM. LSTM continued to outperform CNN-LSTM on all 12 horizons for RMSE, however the gap narrowed at the longest horizon. At 60 minutes, CNN-LSTM RMSE was only 0.3428% higher than that of LSTM. These results indicate that the convolutional part may be useful for feature extraction for longer-range forecasts, but the gain is quite minor and not enough to claim that CNN-LSTM is generally better.

4.1.10 Error and Residual Analysis

In addition to the average forecasting performance was carried out to investigate the distribution and direction of prediction mistakes. The original-scale prediction ranges were:

- Actual speed: 2.0 to 70.0
- LSTM prediction: -12.08 to 79.77
- CNN-LSTM prediction: -1.21 to 75.03

The LSTM therefore produced a wider range of predictions and included more substantial values outside the observed physical range. CNN-LSTM reduced this range substantially, although some predictions still fell below zero or above the observed maximum.

Table 4.6
Residual Statistics

Statistic	LSTM	CNN-LSTM
Mean residual	-0.2789	-0.4246
Mean absolute residual	3.6912	3.6846
Residual standard deviation	6.4969	6.6566
Under predictions (%)	57.32	55.73

Statistic	LSTM	CNN-LSTM
Over predictions (%)	42.68	44.27

Both models exhibited a small negative mean residual, indicating a slight tendency toward under prediction. The proportion of under predictions was also higher than over predictions for both models. CNN-LSTM produced a slightly lower mean absolute residual than LSTM. However, its residual standard deviation was higher, indicating greater variability in the residuals. This finding is consistent with the overall RMSE result, where LSTM outperformed CNN-LSTM.

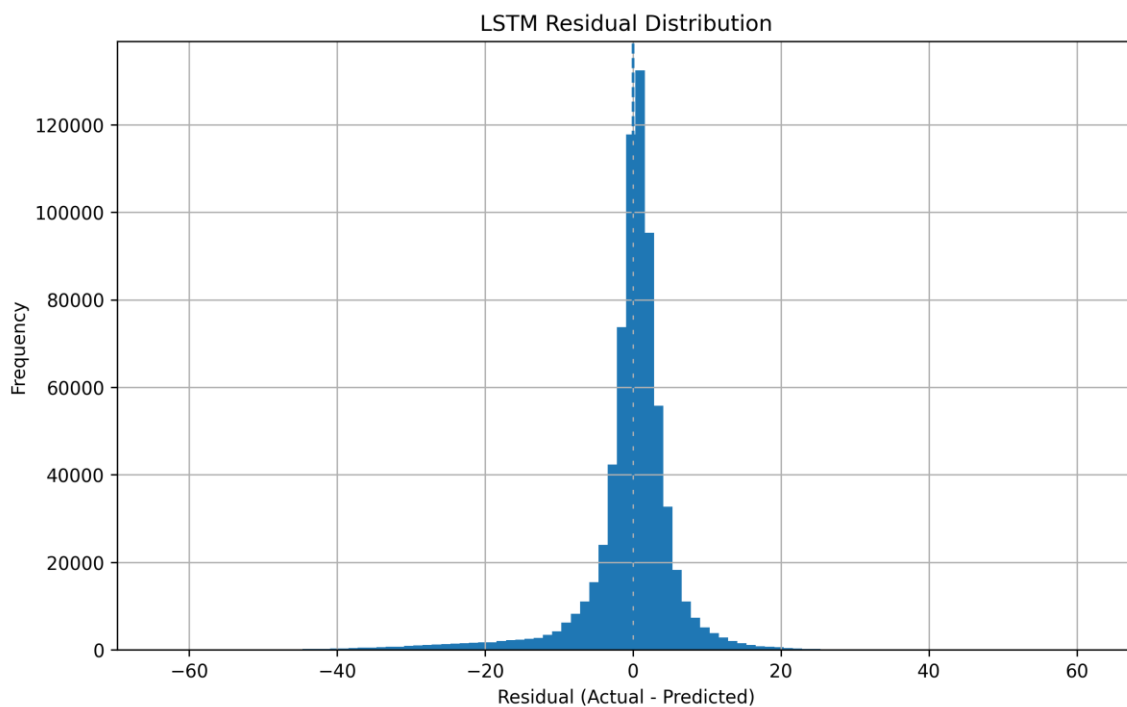


Figure 4.8. Residual distribution of the LSTM model.

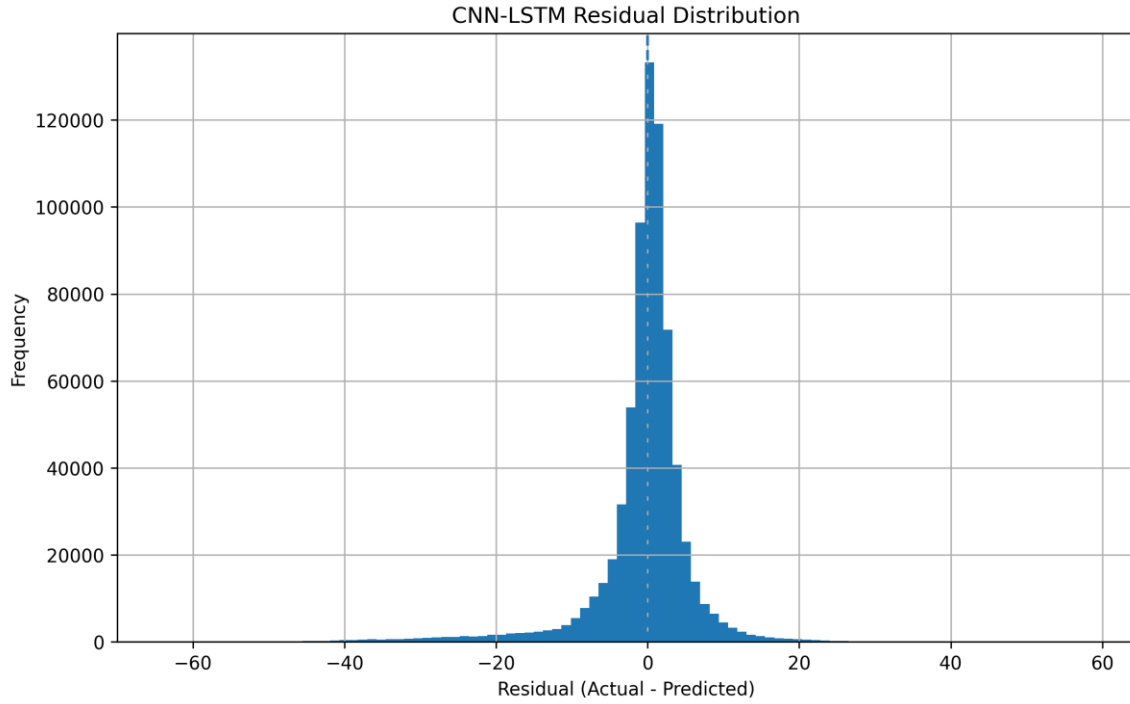


Figure 4.9. Residual distribution of the CNN-LSTM model.

4.1.11 Error Variation with Forecasting Horizon

The horizon-specific error analysis revealed an increasing MAE for both deep-learning models. LSTM MAE increased from 3.3713 at 5 min to 4.0930 at 60 min. The CNN-LSTM MAE increased from 3.4168 to 4.0200 in the same period. Maximum prediction errors also appeared to grow with longer horizons. It implies that uncertainty builds up as the forecasting horizon increases away from the historical observation window. At the shorter horizons, the LSTM often showed lower MAE. At longer horizons CNN-LSTM becomes more and more competitive and eventually achieves lower MAE than LSTM. This supports the finding that the CNN-LSTM architecture may be more beneficial when forecasting longer into the future, albeit the scale of improvement is small.

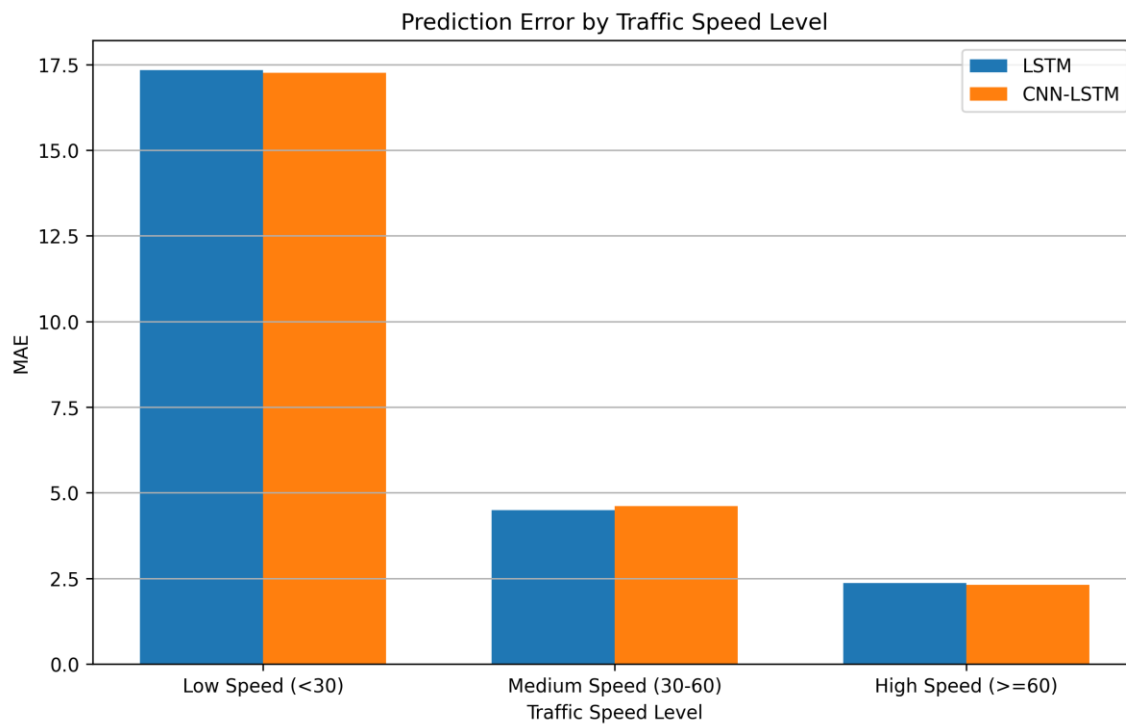
4.1.12 Error Analysis by Traffic-Speed Level

To investigate whether model performance varied according to traffic conditions, errors were additionally grouped into low-, medium-, and high-speed ranges.

Table 4.7**Model Error by Traffic-Speed Level**

Traffic Level	Samples	LSTM MAE	CNN-LSTM MAE
Low Speed (<30)	31,461	17.334608	17.262629
Medium Speed (30–60)	220,002	4.493998	4.605956
High Speed (≥ 60)	458,961	2.371179	2.312183

The largest inaccuracies were in the low-speed group. LSTM achieved an MAE of 17.3346, while CNN-LSTM achieved 17.2626. Thus, both models proved harder to apply in low-speed traffic conditions. For the medium-speed observations, LSTM outperformed CNN-LSTM with an MAE of 4.4940 vs. 4.6060. For high-speed observations, CNN-LSTM produced the lower MAE of 2.3122 compared with 2.3712 for LSTM. These results show that model performance is traffic condition dependent. The larger error in the low-speed category could be related to the more unpredictable and non-linear character of congested traffic states.

**Figure 4.10. MAE of LSTM and CNN-LSTM across traffic-speed levels.**

4.1.13 Statistical Significance and Paired Error Analysis

Since the two models produced predictions for the same test observations, we performed a paired comparison. So, the comparison looked at how much each model missed the mark on every test prediction. In total, 710,064 individual predictions were analyzed, corresponding to 286 test samples $\times$ 12 forecasting steps $\times$ 207 traffic sensors. The mean absolute errors were:

- LSTM = **3.6912**
- CNN-LSTM = **3.6846**

The observed mean difference in original-scale absolute error was around 0.00663 using CNN-LSTM. The analysis was performed that resulted in a very small p-value ($p < 0.05$), which indicates a statistically detectable difference between the paired error distributions. But statistical significance is not the same thing as evidence of a big practical improvement. A more meaningful interpretation is provided by the effect-size analysis.

Table 4.8

Original-Scale Bootstrap Confidence Intervals and Paired Effect Size

Metric	Value
LSTM MAE	3.691220
CNN-LSTM MAE	3.684590
Observed error difference	0.006631
Bootstrap difference CI lower 95%	0.001795
Bootstrap difference CI upper 95%	0.011552
Standard deviation of paired difference	2.100699
Cohen's (d _ z)	0.003156
CNN-LSTM win rate	51.299506%
LSTM win rate	48.700494%

The 95% bootstrap confidence intervals for the individual MAEs were:

- LSTM: **3.679124–3.703635**
- CNN-LSTM: **3.672036–3.697513**

The observed difference is in support of CNN-LSTM but amount of difference is quite minor. A negligible practical effect is indicated by the Cohen's (d _ z) value of 0.0032. This is a crucial

finding for the present investigation. The CNN-LSTM architecture obtains a slightly lower overall MAE than LSTM, but the difference is too small to claim that CNN-LSTM provides a substantial practical improvement over LSTM. The CNN-LSTM architecture achieves a slightly lower overall MAE than LSTM (3.6846 vs. 3.6912), with an observed difference of 0.00663. However, the Cohen's (d_z) of 0.0032 indicates a negligible practical effect. Therefore, the results do not support a substantial practical improvement of CNN-LSTM over LSTM.

4.1.14 Multi-Seed Robustness Analysis

To avoid a biased comparison of models due to a specific random initialization, the LSTM and CNN-LSTM were trained with three random seeds (42, 123 and 2024). The revised multi-seed experiment was assessed in the original traffic-speed scale.

Table 4.9
Corrected Multi-Seed Robustness Results

Seed	Model	MAE	RMSE	MAPE (%)	WAPE (%)
42	LSTM	3.864958	6.394640	10.027534	6.548660
42	CNN-LSTM	4.116386	7.138557	11.640476	6.974673
123	LSTM	3.806446	6.353373	10.076518	6.449520
123	CNN-LSTM	4.013168	7.054326	11.522772	6.799782
2024	LSTM	3.750855	6.334684	9.954768	6.355327
2024	CNN-LSTM	4.124086	7.047031	11.428738	6.987719

Table 4.10
Multi-Seed Robustness Summary (Mean $\pm$ Standard Deviation)

Model	MAE	RMSE	MAPE (%)	WAPE (%)
LSTM	3.807419 $\pm$ 0.057058	6.360899 $\pm$ 0.030678	10.019607 $\pm$ 0.061261	6.451169 $\pm$ 0.096677
CNN-LSTM	4.084547 $\pm$ 0.061936	7.079971 $\pm$ 0.050868	11.530662 $\pm$ 0.106089	6.920725 $\pm$ 0.104942

The robustness experiment qualifies importantly the results from a single run. In the original single-run evaluation, CNN-LSTM demonstrated a modest overall MAE improvement over

LSTM, however the multi-seed trial showed that LSTM achieved lower mean MAE, RMSE, MAPE, and WAPE across all three tested random seeds.

These findings suggest that the apparent CNN-LSTM advantage in the solitary final run was not consistently replicated across various random initializations. Therefore, in the current experimental setup, LSTM should be regarded as the more stable model, although CNN-LSTM maintains a competitive architecture with particular advantages at specified forecasting horizons.

4.1.15 Consolidated Model Comparison

The final experimental comparison is summarized in Table 4.11.

Table 4.11

Final Comparison of Forecasting Models

Model	MAE	RMSE	MAPE (%)	WAPE (%)
Persistence	3.797314	7.091432	9.313734	6.434046
LSTM	3.691220	6.502861	10.333219	6.254285
CNN-LSTM	3.684600	6.670100	10.244300	6.243000

The results demonstrate three different aspects of forecasting performance. First, CNN-LSTM obtained the lowest MAE and WAPE in the primary final evaluation. Second, LSTM obtained the lowest RMSE, indicating better control of larger prediction errors. Third, persistence obtained the lowest MAPE. The difference between the two deep-learning models was very small in overall MAE. CNN-LSTM improved MAE relative to LSTM by approximately 0.1793%, while its RMSE was approximately 2.5718% higher. Therefore, the overall results do not support the conclusion that adding convolutional layers automatically produces a substantial forecasting improvement.

4.1.16 Discussion of Main Findings

Several important findings emerge from the experimental results.

4.1.16.1 Deep Learning versus Persistence

The LSTM and CNN LSTM models performed better than the persistence baseline on MAE, RMSE and WAPE. This implies that the learning-based techniques learned something more than just keeping track of the last seen traffic speed. But at low forecasting horizons persistence was still very competitive. The MAE of persistence, LSTM, and CNN-LSTM are 2.7189, 3.3713, and 3.4168 respectively, for 5 minutes ahead. This is fair, as the pace of traffic generally has high short-term temporal continuity. Consequently, a basic persistence technique is hard to beat at very short horizons.

4.1.16.2 LSTM versus CNN-LSTM

The comparison between LSTM and CNN-LSTM demonstrates that the two architectures are generally similar in terms of performance. CNN-LSTM had a little bit lower total MAE but LSTM had lower RMSE. For each horizon, the results provide better evidence for the relative operation of the architectures. CNN-LSTM was more competitive at longer horizons and achieved lower MAE than LSTM at several horizons after 30 minutes. But the multi-seed robustness experiment with the correction showed a reduced mean error for LSTM. CNN-LSTM is not always better.

4.1.16.3 Effect of Forecasting Horizon

Forecasting performance deteriorated with increasing forecast horizon. The MAE for LSTM increased from 3.3713 at 5 minutes to 4.0930 at 60 minutes. For CNN-LSTM MAE increased from 3.4168 to 4.0200. The error increases with the horizon showing the increasing uncertainty of predicting traffic in the long run. However, the relative loss of persistence was more severe, which allowed the deep-learning models to be more beneficial at longer horizons.

4.1.16.4 Traffic-Speed Conditions

Both models gave the highest MAE in low-speed traffic situations. Therefore, traffic states with congestion are more difficult to predict properly than high-speed states. The CNN-LSTM was marginally better than LSTM in low and high speed cases and LSTM was better in medium speed case. It implies that the underlying traffic regime can influence the performance of a forecasting system.

4.1.16.5 Statistical versus Practical Significance

Statistical analysis of the paired mistakes revealed a substantial difference between the paired errors. But, the Cohen's (d z) value was just 0.0032, which is a minimal practical effect. This distinction is significant. If you have a very high number of individual predictions, then even a very tiny change may lead to a statistically significant test result. The effect size shows that the actual size of the difference between LSTM and CNN-LSTM is very tiny. Thus, the practical conclusion is that these two systems achieve relatively similar forecasting accuracy on the main experiment.

4.1.17 Key Experimental Findings

The results of the study are as follows:

1. The preprocessing approach generated valid forecasting sequences without any measurements of (N a N) in the final train, validation and test arrays.
2. The experiment was implemented with 207 traffic sensors, 60-minutes historical input window and 60-minutes forecast horizon.
3. The persistence baseline performed especially well at short forecast horizons.
4. The LSTM showed the best overall RMSE of 6.5029.
5. CNN-LSTM had the lowest MAE (3.6846) and WAPE (6.2430%) in the main final examination.
6. Persistence had the lowest overall MAPE (9.3137%).
7. CNN-LSTM had a marginal improvement of 0.1793% in overall MAE over LSTM.
8. CNN-LSTM outperformed LSTM in many longer predicting horizons, for example, 40–60 min with an improvement of 1.7854% at 60 min.
9. LSTM outperformed CNN-LSTM in RMSE at all 12 forecasting horizons.
10. Low-speed traffic circumstances resulted in far larger predicting errors compared to medium- and high-speed conditions.
11. Corrected multi-seed robustness experiment: LSTM had an advantage in mean MAE, RMSE, MAPE and WAPE.
12. The effect-size analysis for the paired comparison yielded Cohen's (d_z) of ~0.0032, which indicates a small practical difference between LSTM and CNN-LSTM.

CNN-LSTM demonstrates similar performance with some benefits at longer horizons, but overall improvement is too tiny and not robust across random seeds to claim universal supremacy.

4.1.18 Chapter Summary

In this chapter the experimental evaluation of the Persistence, LSTM and CNN-LSTM models for multi-step traffic-speed forecasting using the METR-LA dataset was provided.

In the main experiment, it was found that CNN-LSTM had the lowest MAE and WAPE overall, LSTM had the lowest RMSE, and persistence had the lowest MAPE. The analysis by horizon showed that the persistence was particularly good for short-term prediction, and the deep learning models became more competitive with longer prediction horizon.

The error study revealed that low speed traffic conditions were far harder to predict than high speed conditions. The CNN-LSTM fared marginally better for the low and high speed categories whereas the LSTM performed better for middle speed observations.

The statistical analysis and the robustness assessments gave a significant qualifier to the primary results. The paired statistical test showed a statistically significant difference but the paired effect size was small. In addition, it was observed that LSTM had less mean error for all the metrics tested in the corrected multi-seed experiment.

Thus the final conclusion from the experimental results is not that CNN-LSTM is always better than LSTM. Instead, the evidence shows that LSTM provides a solid and consistent forecasting baseline, and CNN-LSTM achieves similar results with small benefits at some extended forecasting horizons and measures.

4.2 Air Quality Monitoring and Prediction

The third core topic of the proposed AI-based content analytics framework focuses on air quality monitoring and prediction within an IoT driven smart-city environment.

4.2.1 Experimental Objective

The project goal was to create and test deep learning models that can guess future levels of particulate matter (PM_{2.5}) by looking at data from past weather and air quality observations. PM_{2.5} was picked as the goal variable because it shows how polluted the air is in cities. A good prediction of the PM_{2.5} level can help with proactive monitoring of the environment and give

smart-city decision-making tools useful data. Long Short Term Memory (LSTM) and CNN LSTM are two deep learning architectures that we built and compared. We used the same dataset, preprocessing method, temporal split and evaluation criteria for both. There are three common ways to test regression models: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and coefficient of determination (R^2). A smaller MAE and RMSE value means that the forecast is more accurate and a higher R^2 value means that the model explains a larger portion of the changes in the measured PM2.5 levels.

4.2.2 Dataset Description

This test was done on the Beijing PM2.5 data set, which has hourly records of Beijing's air quality from January 1, 2010, to December 31, 2014. There are 43824 hourly observations and 13 factors in the original dataset. The variables are the number of observations, the date and time, the PM2.5 level, the temperature, the dew point temperature, the atmospheric pressure, the direction of the wind, the cumulative wind speed and factors connected to precipitation. The variable being studied in this experiment was PM2.5, and the variables related to weather and time were used as predictors. There were 2,067 missing PM2.5 observations in the dataset, which is roughly 4.717% of the full dataset. A suitable date time index was created and then missing PM2.5 data were interpolated according to time. After the preprocessing, 24 observations were unobserved temporarily at the edges of the interpolation process and were handled in such a way that the final dataset contained no missing target values. The cleaned dataset contained all 43,824 observations. The obtained PM2.5 levels varied from 0 to 994 $\mu\text{g}/\text{m}^3$. The mean was about 98.26 $\mu\text{g}/\text{m}^3$ and the median was 73 $\mu\text{g}/\text{m}^3$. The distribution was positively skewed (skewness = 1.82) indicating that high pollution observations were less common than observations of moderate concentration but contributed strongly to the upper tail of the distribution.

4.2.3 Exploratory Data Analysis

The descriptive analysis showed a high variation in PM2.5 level. The first was 30 $\mu\text{g}/\text{m}^3$, the median was 73 $\mu\text{g}/\text{m}^3$ and the third was 136 $\mu\text{g}/\text{m}^3$. Maximum recorded value after cleaning was 994 $\mu\text{g}/\text{m}^3$. The large range reflects the very changeable nature of urban particle pollution. The correlation analysis was also performed for investigating the linear association between PM2.5 and the numerical environmental factors. Of the studied variables, the cumulative wind speed (I_{ws}) was the most negatively correlated with PM2.5, with a value of about -0.246 . The dew point

(DEWP) was positively correlated with about 0.169 while the temperature (TEMP) was negatively correlated with about -0.089 . PRES (atmospheric pressure), Is (precipitation of snow), and Ir (precipitation of rain) showed rather minor linear associations with PM2.5. The wind direction was also considered because the atmospheric dispersion may vary depending on the wind conditions prevailing. The data collection had four classes of wind direction: SE, NW, cv, and NE.

The cv category had the highest average PM2.5 concentration (approx. $125.36 \mu\text{g}/\text{m}^3$), followed by SE (approx. $110.23 \mu\text{g}/\text{m}^3$), NE (approx. $89.90 \mu\text{g}/\text{m}^3$) and NW (approx. $70.31 \mu\text{g}/\text{m}^3$).

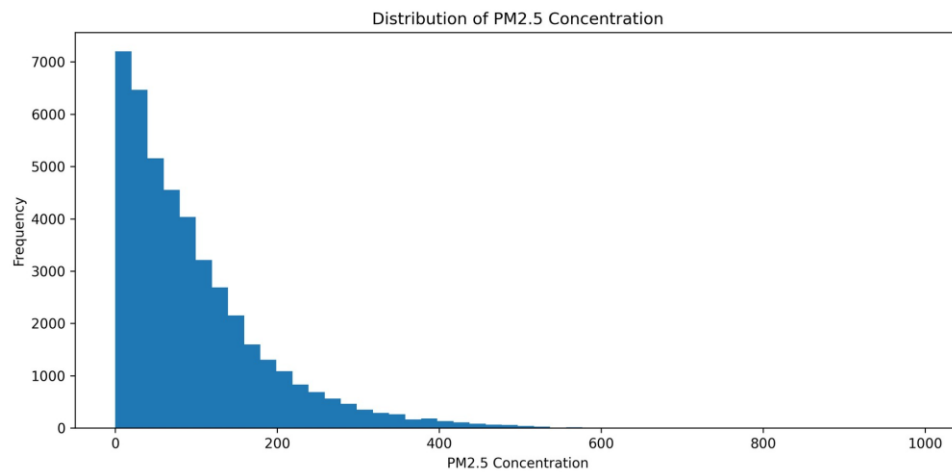


Figure 4.2.1. Distribution of PM2.5 concentrations.

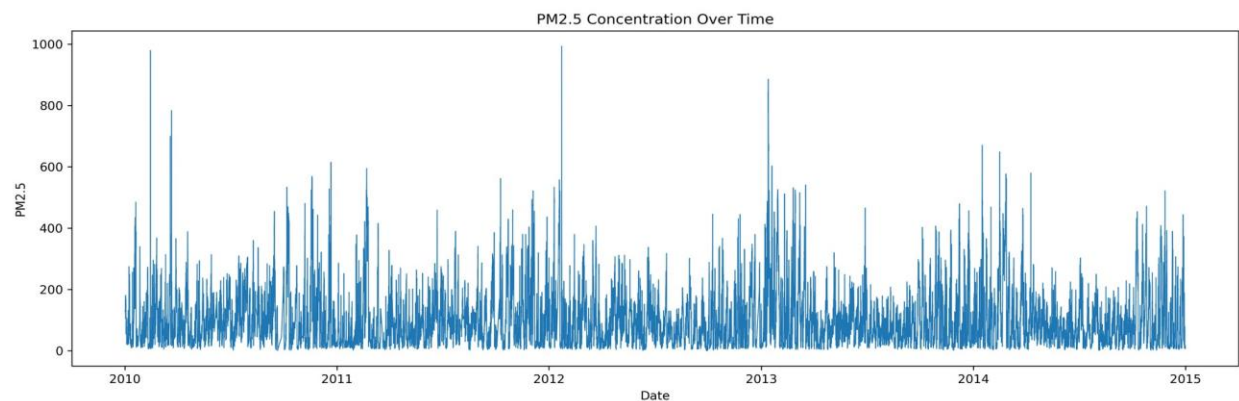


Figure 4.2.2. PM2.5 level over time.

These observations demonstrate that PM2.5 level is influenced by temporal and meteorological conditions and therefore provides a suitable time-series prediction problem for evaluating deep learning models.

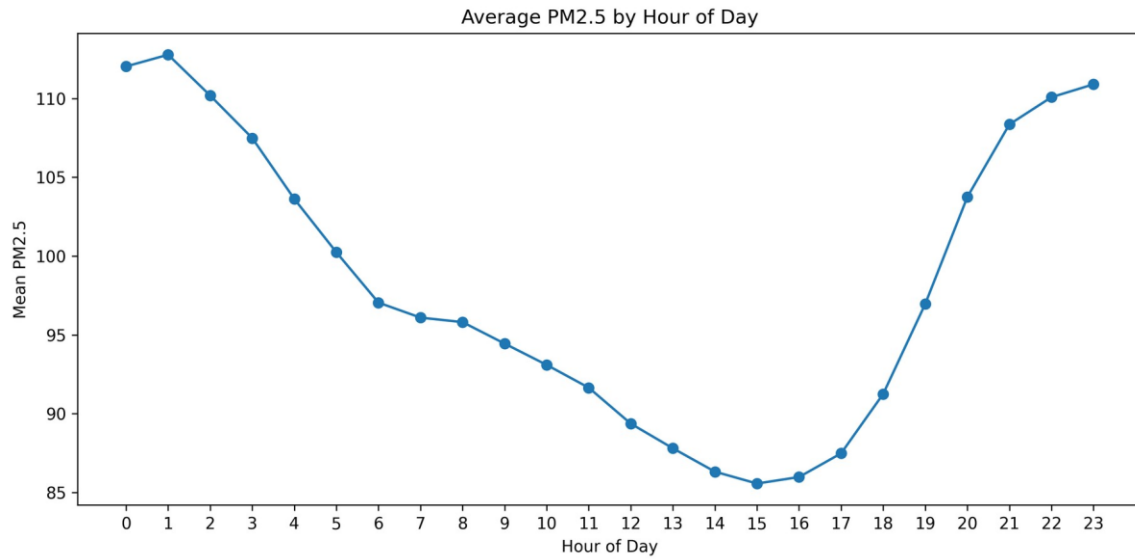


Figure 4.2.3. Average PM2.5 concentration level by hour of day.

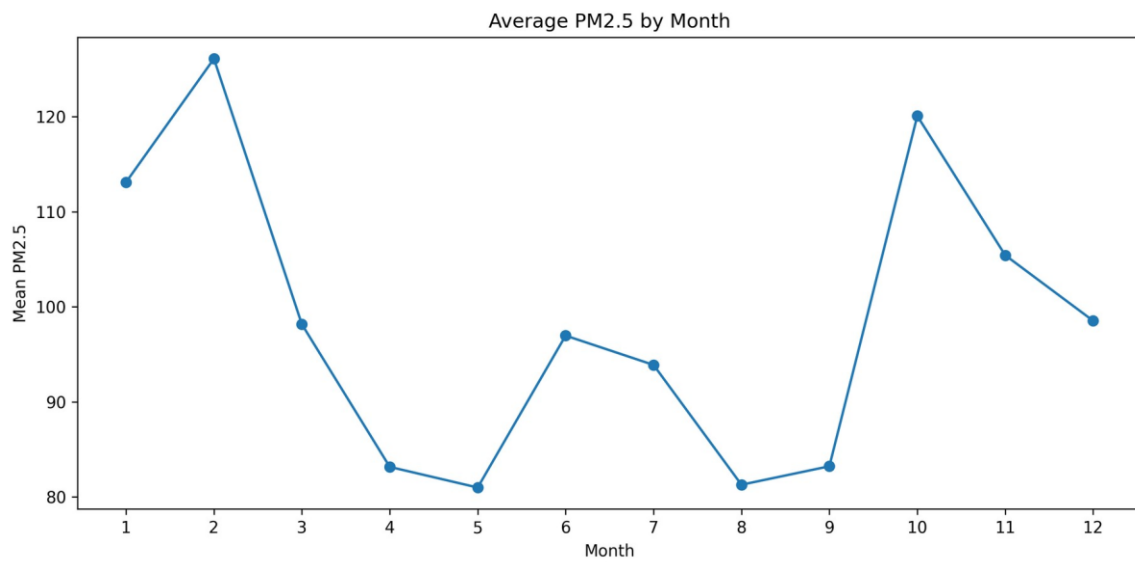


Figure 4.2.4. Average PM2.5 concentration level by month.

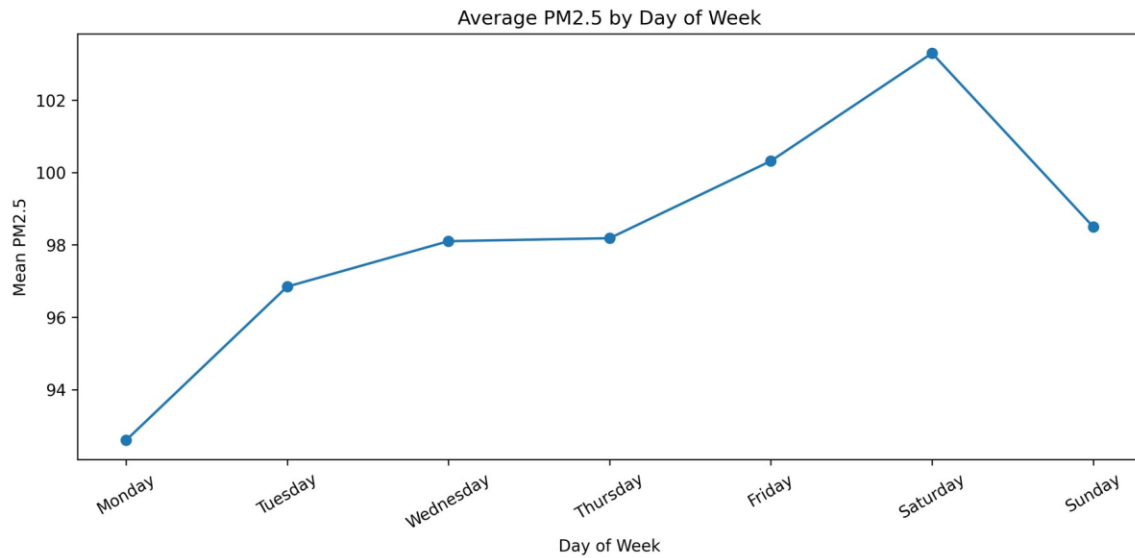


Figure 4.2.5. Average PM2.5 level by day of week.

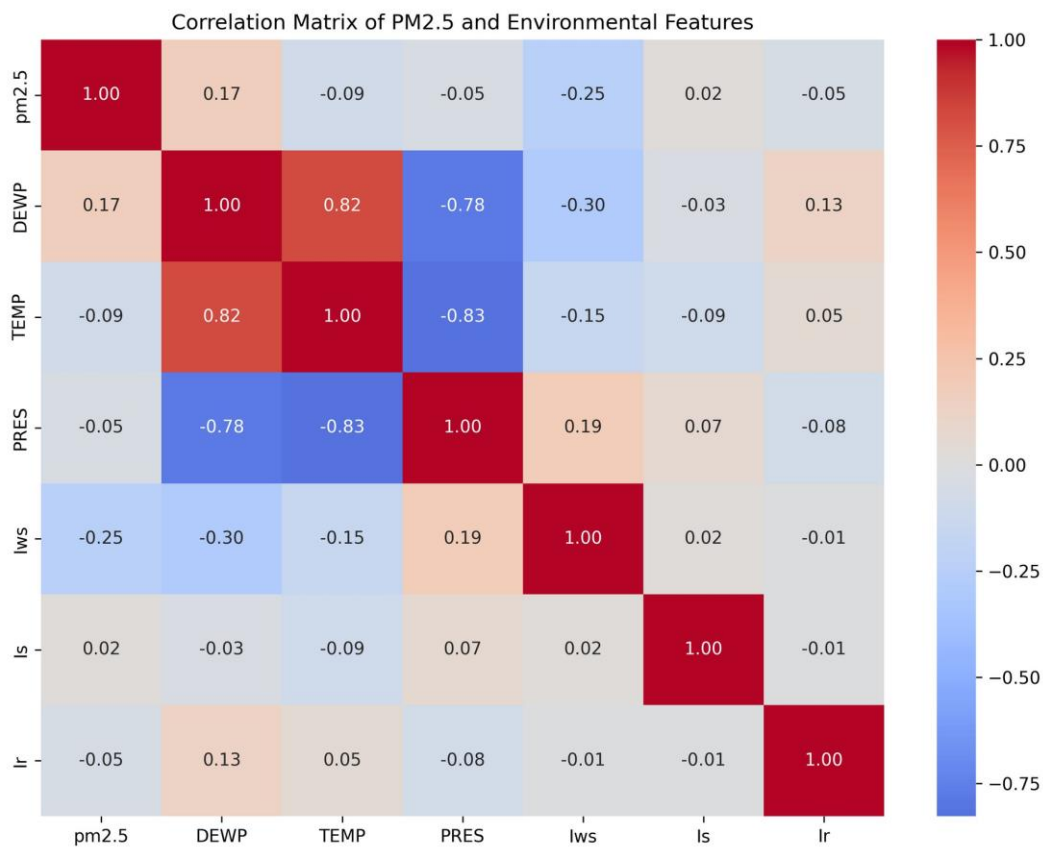


Figure 4.2.6. Correlation heat map of air-quality and meteorological variables.

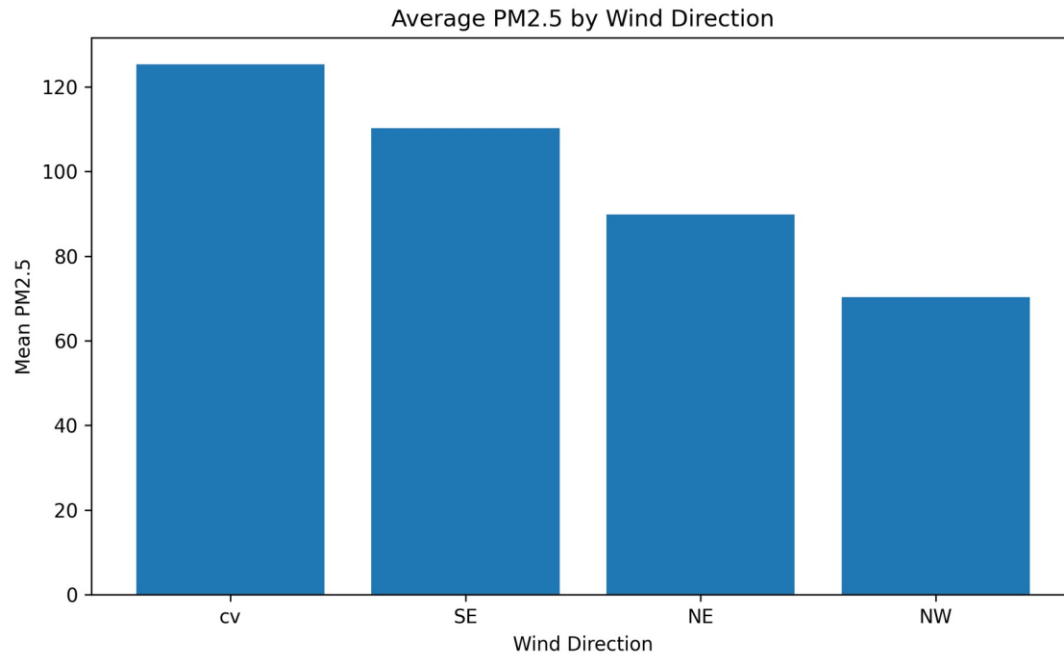


Figure 4.2.7. PM2.5 concentration, by wind-direction category.

4.2.4 Feature Engineering

Fourteen predictive features were developed for the air-quality forecasting models. The environmental variables were:

DEWP - dew point temperature

TEMP Temperature

PRES — Atmospheric pressure

I w s — Cumulative wind speed

Is — Snow-related precipitation indicator

I r — Rain-related precipitation indicator

The hour and month variables were cyclical encoded to capture temporal patterns. Which led to:

Hour _ sin

hour _ cos

month _ sin

month _ cos

Wind direction was converted into four binary one-hot encoded variables:

Cbwd _ NE

cbwd_ NW

cbwd_ SE

cbwd_ cv

This model allows neural networks to handle temporal and category information quantitatively, but maintaining the cyclical structure of time-related variables.

4.2.5 Chronological Data Splitting

The dataset was separated into training and testing subsets in chronological order to prevent the model from training on future information. The data were split into 70% training and 15% testing observations. The training set consists of 30,676 observations, the validation set consists of 6,574 observations, and the testing set consists of 6,574 observations. The training period was 1 January 2010 to 2 July 2013, the validation period was 2 July 2013 to 2 April 2014, and the test period was 2 April 2014 to 31 December 2014. No temporal overlap was found between the three groupings. Thus, the chronological order check confirmed that the validation and testing data strictly followed the respective training periods. This technique minimizes the likelihood of temporal leakage and gives a more realistic evaluation of the forecasting ability.

4.2.6 Feature Scaling and Sequence Generation

Feature scaling was conducted using standardization. Note that the feature scaler and target scaler were fitted on the training data and then applied to the validation and test sets. This avoids the information of the validation or test periods from impacting the scaling parameters. Temporal learning was done during a historical window of 24 hours. Therefore, the input sequence consisted of observations in the past 24 hourly time steps, and the objective was the next hour's PM2.5 level.

The datasets generated after sequence generation included:

Table 4.20

Dataset	Sequences	Time Steps	Features
Training	30,652	24	14
Validation	6,550	24	14
Testing	6,550	24	14

Therefore, each prediction was generated using one complete day of historical information. This configuration was selected to provide the models with sufficient temporal context for learning daily pollution patterns.

4.2.7 LSTM Model

The first deep learning model was an LSTM network. The LSTM networks are built for sequential data and may learn time dependencies from past observations because to its internal memory architecture. For predicting PM2.5, we built an LSTM architecture with a 64-unit LSTM layer, a dropout layer, a dense layer with 32 units and an output layer with a single unit. There are 22,337 factors that the model learned. As part of its training, the model went through 50 rounds of early termination and learning-rate reduction. Training ended after 11 epochs and the best weights were back to where they were in epoch 3. This means that the validation loss didn't get better after the early training stage and stopping training early kept the method from becoming too good.

4.2.8 CNN-LSTM Model

The second model was based on convolutional feature extraction and LSTM-based temporal modelling. To get local temporal patterns from the 24-hour input sequence, the CNN-LSTM design began with a 1D convolutional layer. The temporal model was then sped up using a max-pooling layer and the sequence that was made was sent to an LSTM layer. One-dimensional convolutional layer with 64 filters, max pooling, 64-unit LSTM layer, dropout, 32-unit dense layer and single-unit output layer made up the design. The entire architecture had 37889 trainable parameters. The CNN-LSTM model was trained with the same general technique as the LSTM model. Training halted at 12 epochs, best model weights restored from epoch 4.

4.2.9 Overall Prediction Performance

The finished models were tested on a dataset that had never been seen before, using the original scale PM2.5 values.

Table 4.2.1

Overall performance of the air-quality prediction models

Model	MAE	RMSE	R ²
LSTM	36.7535	50.2094	0.6008
CNN-LSTM	38.2610	51.9629	0.5725

The LSTM model resulted in an MAE of $36.7535 \mu\text{g}/\text{m}^3$, an RMSE of $50.2094 \mu\text{g}/\text{m}^3$ and an R^2 value of 0.6008. CNN-LSTM model attained an MAE, RMSE and R^2 value of $38.2610 \mu\text{g}/\text{m}^3$, $51.9629 \mu\text{g}/\text{m}^3$, and 0.5725, respectively. Based on all three evaluation metrics, LSTM gave the better overall prediction performance.

The MAE of the proposed model was 4.10% less than that of CNN-LSTM, and the RMSE was 3.49% less. The LSTM also achieved a higher R^2 value, higher by roughly 0.0284.

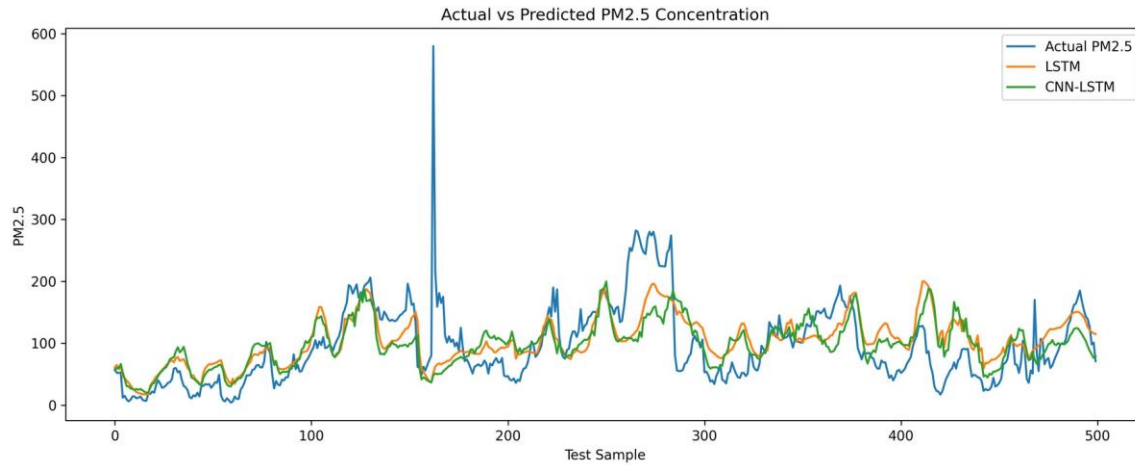


Figure 4.2.8. Actual versus predicted PM2.5 concentrations for LSTM and CNN-LSTM.

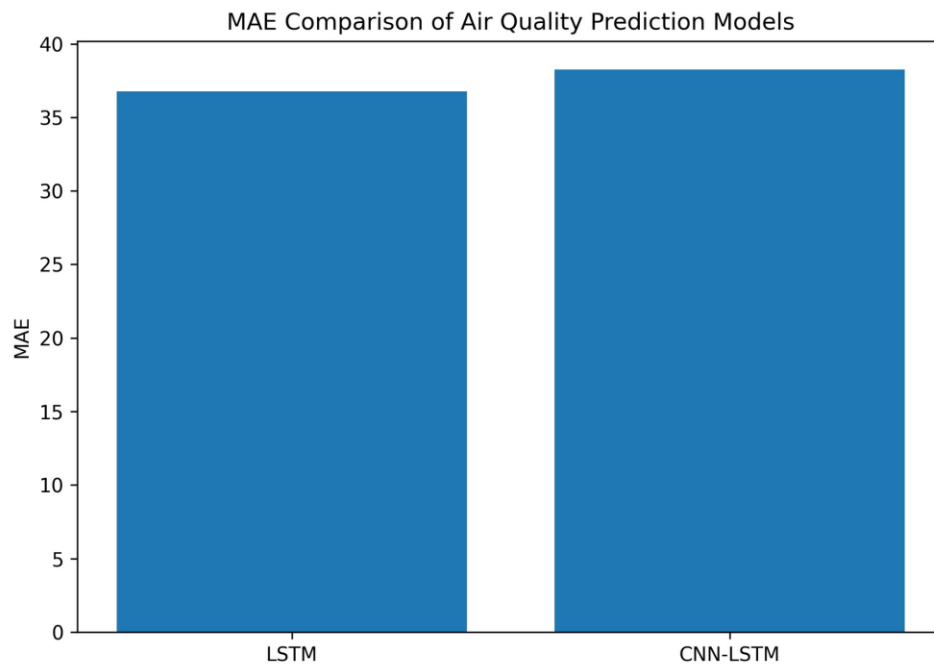


Figure 4.2.9. MAE of LSTM and CNN-LSTM.

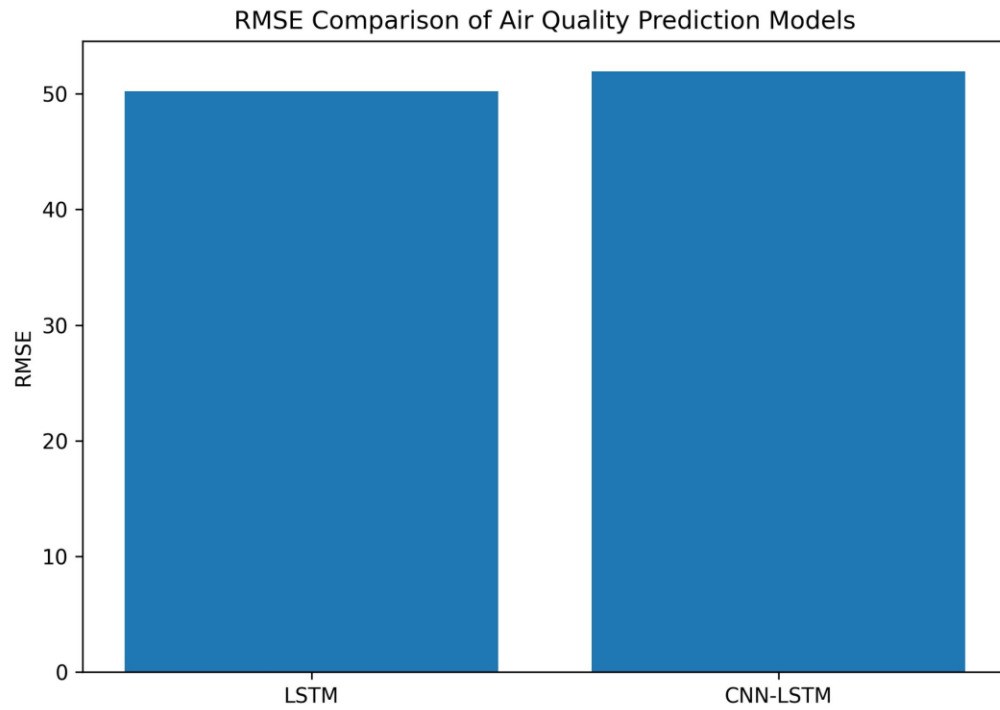


Figure 4.2.10. RMSE of LSTM and CNN-LSTM.

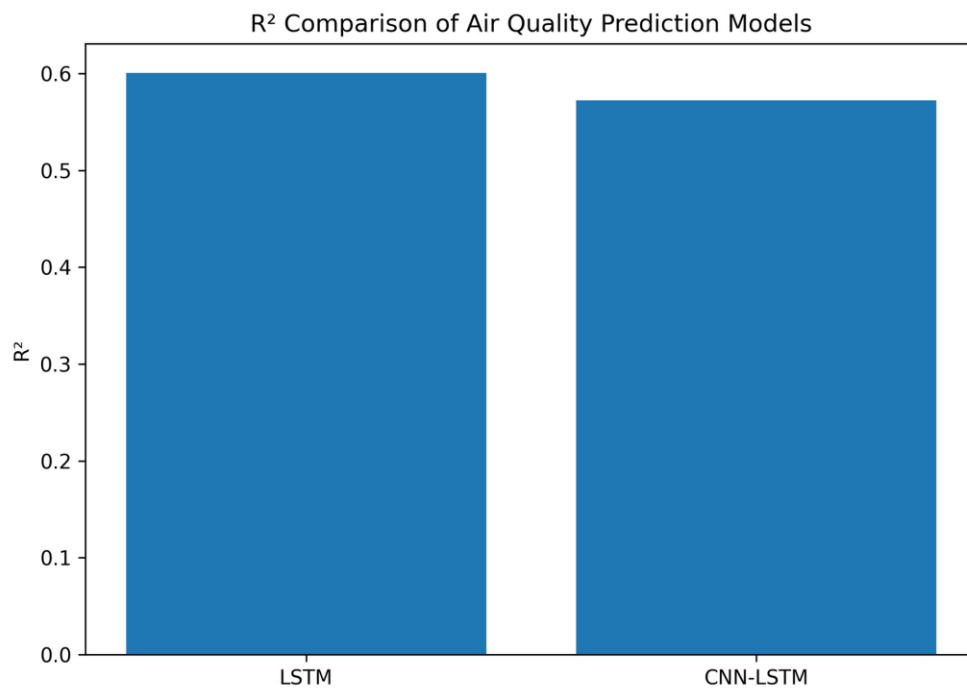


Figure 4.2.11. R² of LSTM and CNN-LSTM.

4.2.10 Comparative Error Analysis

The analysis was performed using the absolute prediction errors of the two models. The results are presented in Table 4.2.2.

Table 4.2.2

Error analysis of LSTM and CNN-LSTM

Error Measure	LSTM	CNN-LSTM
Mean Absolute Error	36.7535	38.2610
Median Absolute Error	27.5923	28.7989
Maximum Absolute Error	527.9629	531.6663

The LSTM has a lower mean absolute error and lower median absolute error. The median absolute error is 27.5923 $\mu\text{g}/\text{m}^3$ which suggests that the prediction error was smaller for LSTM than for CNN-LSTM in the average. Also the maximum absolute error was slightly lower in LSTM (527.9629 $\mu\text{g}/\text{m}^3$) than CNN-LSTM (531.6663 $\mu\text{g}/\text{m}^3$). The substantial maximum errors suggest that both models had difficulty in accurately predicting some extreme PM_{2.5} measurements. The distinction between residual distributions were analyzed.

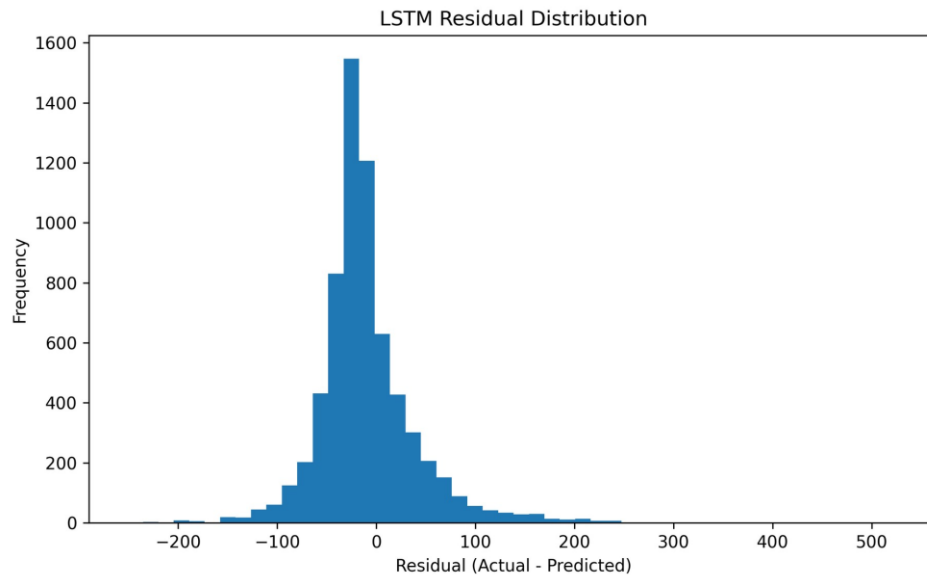


Figure 4.2.12. Residual distribution of the LSTM model.

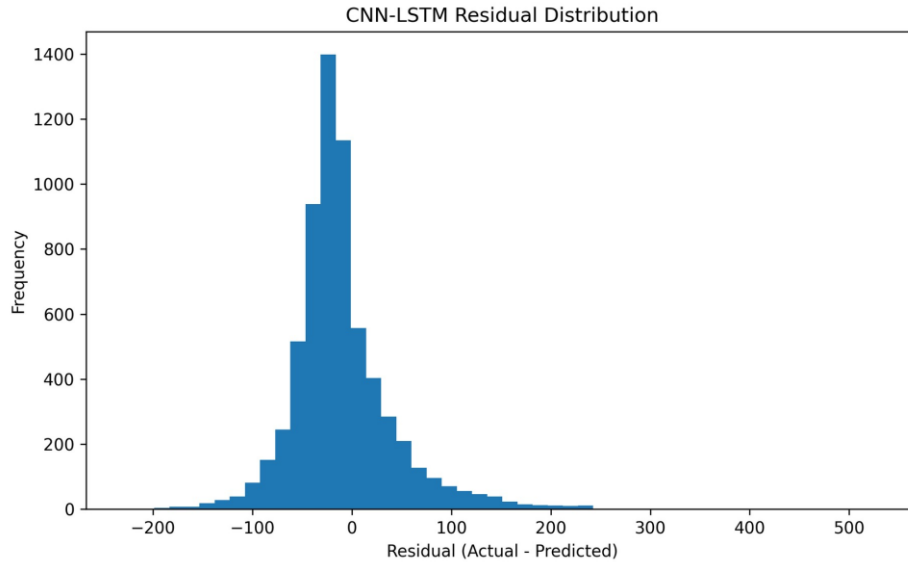


Figure 4.2.13. Residual distribution of the CNN-LSTM model.

4.2.11 Statistical Significance Analysis

Since both models predicted on the same set of 6550 test data, a paired analysis was performed. The Wilcoxon signed-rank test was used for comparing the absolute errors caused by LSTM and CNN-LSTM. The average paired error difference (CNN-LSTM error minus LSTM error) was $1.5075 \mu\text{g}/\text{m}^3$ and the median difference was $1.1267 \mu\text{g}/\text{m}^3$. The standard deviation of paired differences was $17.8278 \mu\text{g}/\text{m}^3$. The Wilcoxon signed rank test returned a statistic of 9398854.0 and a p-value of $3.960765343979055\text{e-}18$. Hence, the difference of the paired error distributions was significant at the significance level of $\alpha = 0.05$. But statistical significance is not necessarily proof of a big practical improvement. The calculated Cohen's d was 0.0846, which is a minimal to extremely minor influence. Hence, the difference in paired mistakes is statistically detectable, but is restricted in practical magnitude.

4.2.12 Robustness and Paired Win Analysis

We also performed a paired win-rate analysis to evaluate how often each model with the smallest absolute error for each test observation. CNN-LSTM had a lower error for 3006 observations (45.89%) while LSTM had a lower error for 3544 observations (54.11%).

Table 4.2.3

Paired prediction-error comparison

Model	Observations with Lower Error	Percentage
LSTM	3,544	54.11%
CNN-LSTM	3,006	45.89%

The paired win-rate data further confirm that the LSTM was more consistent than CNN-LSTM on the test observations. However, this difference was not significant enough to say that CNN-LSTM is not a good design. However, the results show that for this particular dataset and forecasting setup the additional neural feature-extraction stage did not consistently improve.

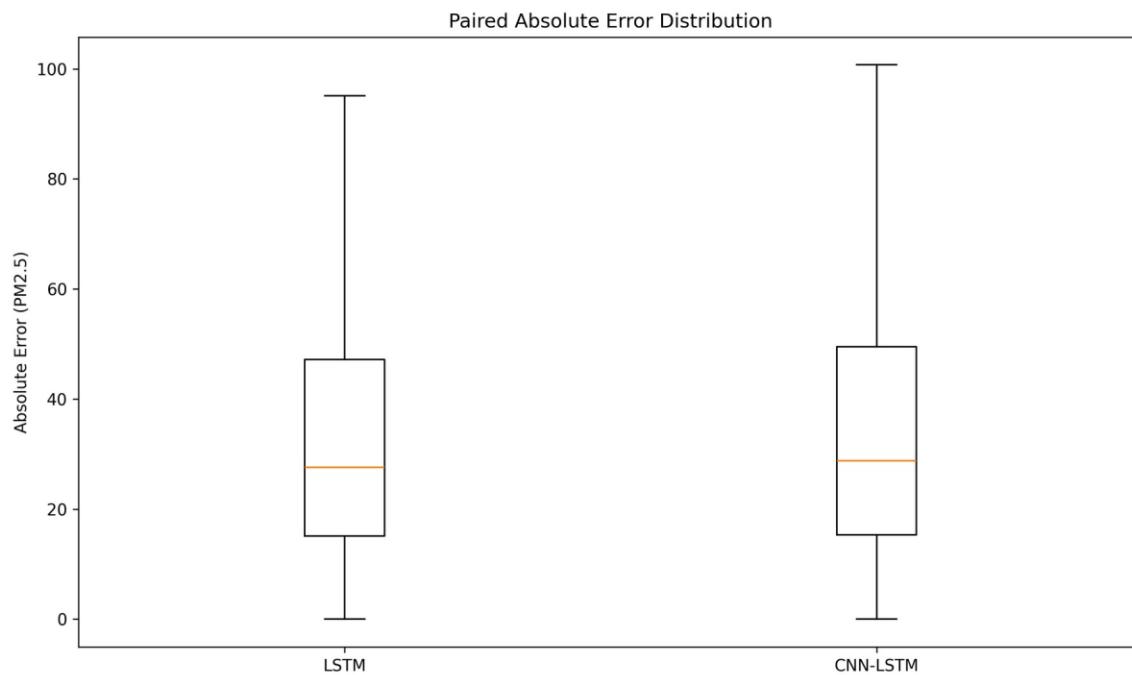


Figure 4.2.15. Distribution of paired absolute-error

4.2.13 Bootstrap Confidence-Interval Analysis

The uncertainty of the observed difference in MAE was estimated via bootstrap resampling ($n = 5,000$ iterations). The difference in MAE observed (CNN-LSTM MAE - LSTM MAE) was $1.5075 \mu\text{g}/\text{m}^3$ with 95% bootstrap confidence interval $[1.0830, 1.9226] \mu\text{g}/\text{m}^3$.

The bootstrap analysis indicates that CNN-LSTM resulted in a greater MAE than LSTM for the examined test sample, because the confidence interval is above zero.

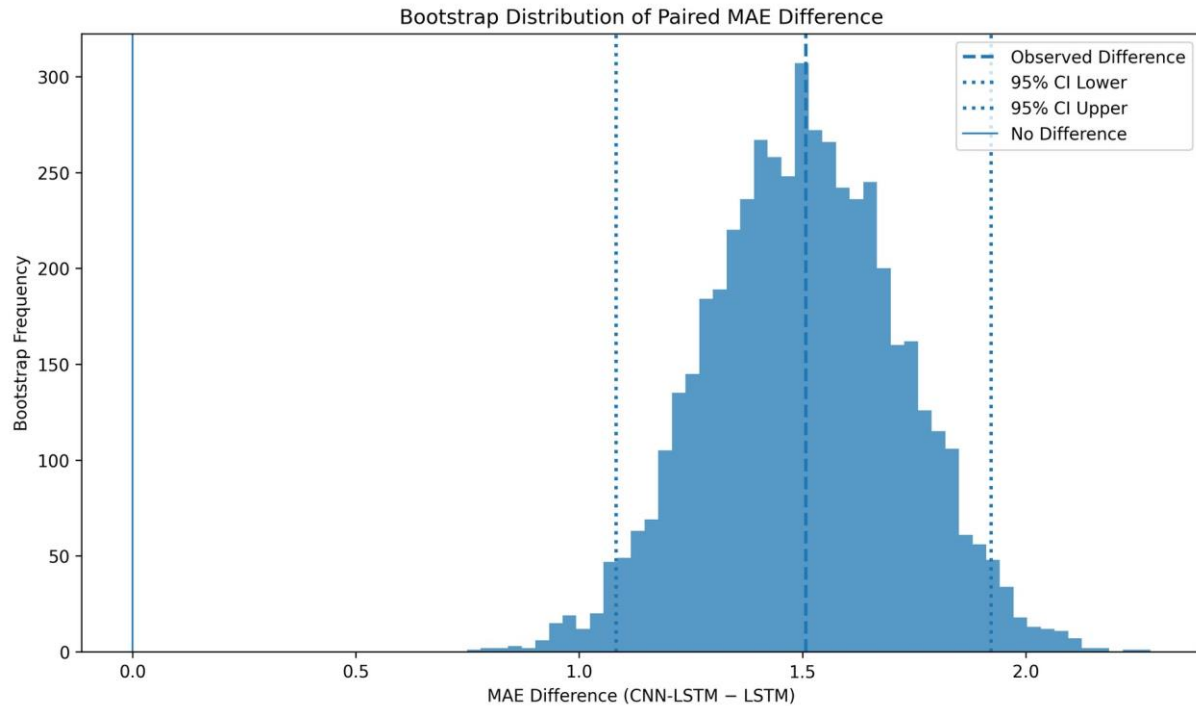


Figure 4.2.14. Bootstrap distribution of the MAE difference between CNN-LSTM and LSTM.

The individual model MAE confidence intervals were:

Table 4.24

Model	MAE	95% Bootstrap CI
LSTM	36.7535	[35.9433, 37.5611]
CNN-LSTM	38.2610	[37.4352, 39.0968]

4.2.14 Discussion of Results

The results show that the LSTM model has a better performance in the overall PM2.5 prediction level, than the CNN-LSTM model under the experimental conditions employed in this study. The LSTM achieved reduced MAE and RMSE values and greater R². The result indicates that a relatively simple recurrent architecture that capture the temporal dependencies in the Beijing PM2.5 data well. The CNN-LSTM architecture added an additional convolutional feature-extraction stage, but this added complexity did not improve test performance. The absolute difference between the models was somewhat minor. The LSTM improved MAE by about 4.10% and RMSE by approximately 3.49% compared to CNN-LSTM. The effect size was also 0.0846, which is relatively small in terms of practical significance of the change. The R² values also give additional evidence for the predictive capabilities of both models. The LSTM model produced a R² of 0.6008, which explains that the model accounted for about 60% of the variation in the test PM2.5 observations. CNN-LSTM model attained R² of 0.5725. However, large prediction errors were obtained by both models for several of the observations, particularly the severe PM2.5 values. This is seen from the largest absolute errors of more than 527 µg/m³ in both models. Such mistakes show that it is not yet easy to predict high pollution events well with the current feature set and one-hour prediction configuration. Therefore, the results do not support the statement that CNN-LSTM is always better than LSTM for air-quality prediction. Rather, they show that model choice needs to be informed by empirical assessment for the unique IoT sensing environment, target variable, temporal resolution and forecasting objective.

4.2.15 Implications for IoT-Driven Smart Cities

The air quality experiment complements the suggested AI-based content analytics framework in terms of illustrating how the environmental observations, in IoT style, might be translated into predictive information. Instead of presenting the present PM2.5 level, the forecasting part delivers a prediction of the next hour pollution level based on recent environmental observations. Such predictive information can enable smart-city applications such as environmental monitoring dashboards, pollution alarms, urban planning, and proactive public-health decision support. The results also illustrate the value of considering alternative AI designs, rather than assuming a more complicated architecture will always yield better predictions. In this experiment, the LSTM model outperformed even with less trainable parameters than the CNN-LSTM model.

4.2.16 Summary of Air Quality Experiment

The air-quality prediction experiment utilized 43,824 hourly observations in the Beijing PM2.5 data set. Missing PM2.5 measurements were imputed with time-based interpolation. Temporal features were encoded with cyclical encoding and wind direction was one-hot encoded. The 24-h history window was utilized to predict the PM2.5, level one hour ahead. Two AI models that are LSTM and CNN-LSTM, were trained and evaluated using MAE, RMSE and R2.

The LSTM achieved the best overall performance with MAE, RMSE and R² values of 36.7535, 50.2094 and 0.6008 correspondingly compared to 38.2610, 51.9629 and 0.5725 for CNN-LSTM.

Statistical analysis demonstrated a substantial pairwise difference between the models, but the matching Cohen's d of 0.0846 indicated a low to very tiny practical effect. The paired win-rate analysis similarly supported LSTM, with LSTM providing reduced errors for 54.11% of the test observations compared to 45.89% for CNN-LSTM.

The experiment shows that the LSTM architecture provides a robust and somewhat efficient solution for short-term PM2.5 prediction level in the analyzed smart-city air-quality scenario. The results further emphasize that the architectural complexity should be justified by measurable gains in accuracy rather than anticipated performance gains.

4.3 Citizen Complaint Analytics

4.3.1 Overview

Citizen complaint analytics is a significant part of smart-city analytics since the municipal service requests might reflect the public-service demands and responses. This work developed a machine-learning classification algorithm to predict complaint response types based on the Chicago 311 Sanitation Code Complaints dataset.

After data cleaning and removal of duplicate and incorrect records, 152398 records were used for this investigation. Complaints were categorized into three response times: Fast (≤ 3 days), Moderate (4–9 days), and Slow (≥ 10 days) to complete the complaint. The total data set included 51.28% Fast, 24.91% Moderate and 23.81% Slow complaints.

4.3.2 Model Development

We tried out four different classifiers: KNN, Decision Tree, Random Forest and Support Vector Machine (SVM). The input features were temporal, geographical and categorical complaint characteristics. In an effort to limit data leaking, we omitted variables directly relevant to the end response time (e.g. completion date and completion duration). The data was divided chronologically into 70 % training, 15 % validation and 15 % testing sets. The final test set consisted of 22,860 complaints. Measures of model performance included Accuracy, Macro Precision, Macro Recall, Macro F1-score and Weighted F1-score.

4.3.3 Model Performance

Table 4.3 presents the performance of the evaluated models.

Table 4.3: Performance of Citizen Complaint Classification Models

Model	Accuracy	Macro Precision	Macro Recall	Macro F1	Weighted F1
Random Forest	59.65%	56.51%	58.17%	56.08%	59.72%
KNN	54.04%	49.54%	50.76%	49.62%	53.63%
Decision Tree	50.57%	50.25%	50.34%	49.03%	51.86%
SVM	51.01%	47.73%	47.05%	47.00%	51.15%

Random Forest obtained the greatest overall performance with accuracy 59.65% and Macro F1-score 56.08%. KNN was second, followed by SVM and Decision Tree in third and fourth. The majority class baseline is 51.28%, corresponding to the Fast category. Thus, the Random Forest accuracy represents an improvement of approximately 8.37 percentage points over the majority class-baseline.

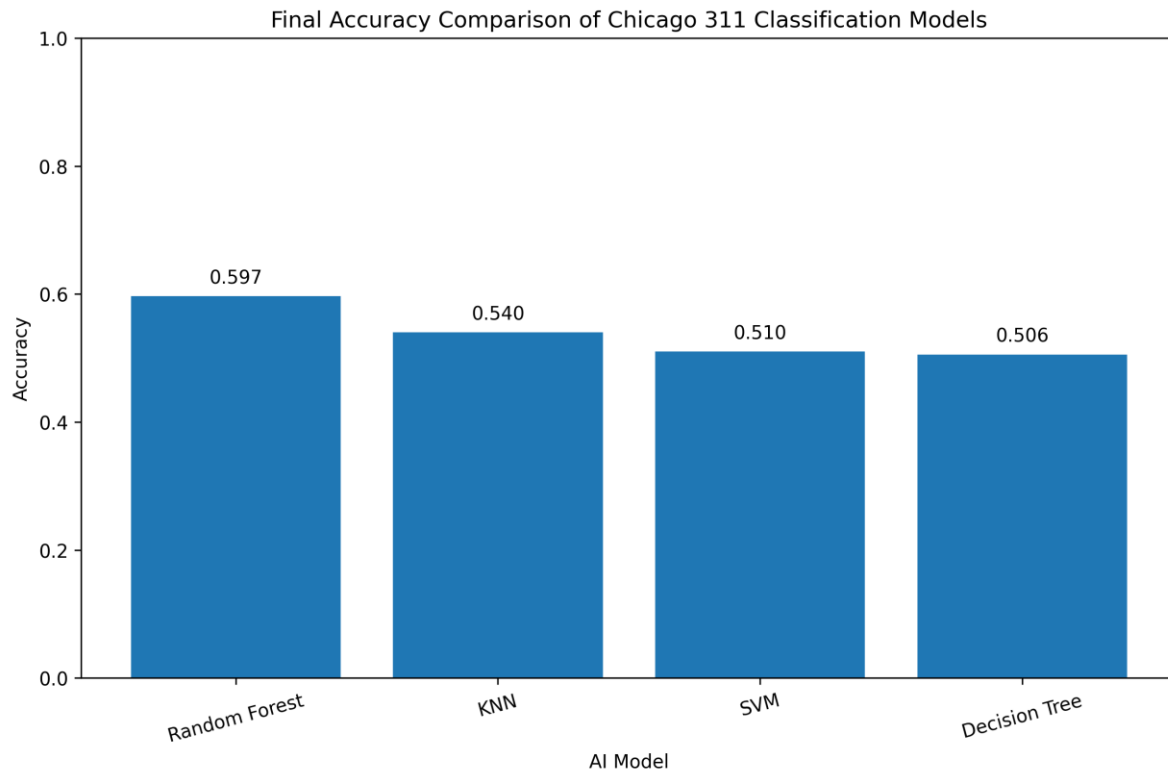


Figure 4.3: Accuracy Comparison of the Four Citizen Complaint Classification Models

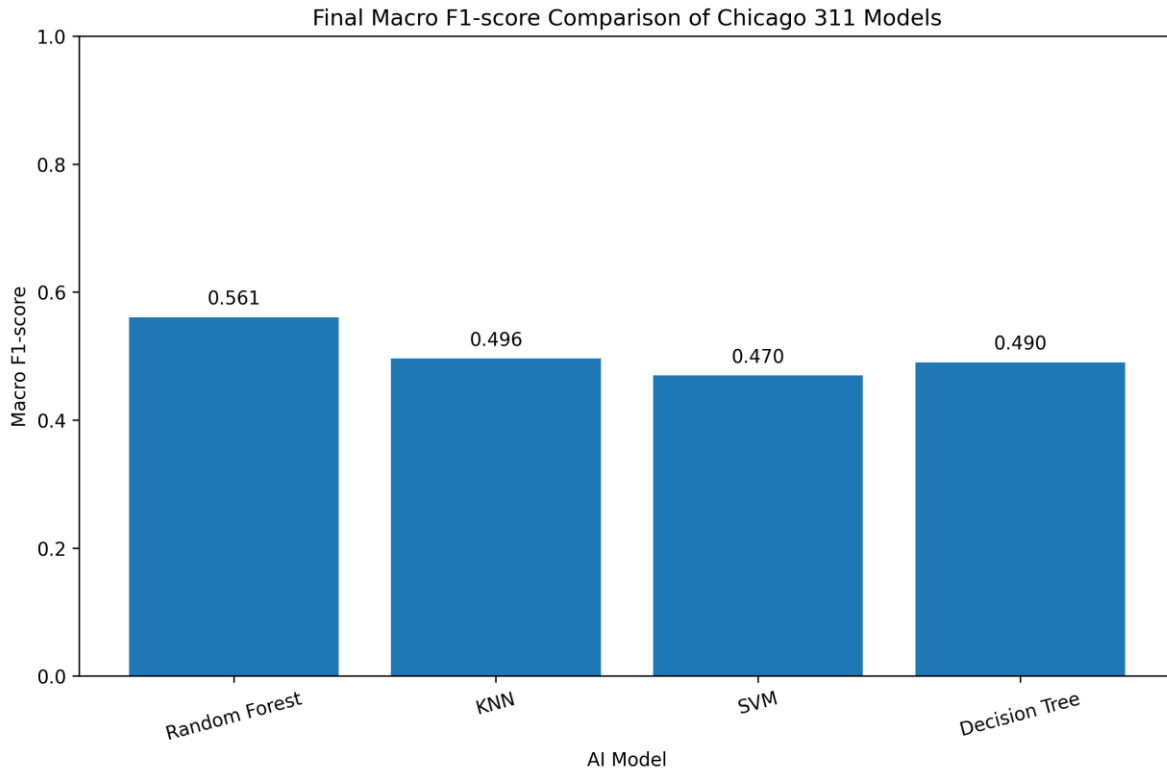


Figure 4.4: Macro F1-Score of the Four Citizen Complaint Classification Models

4.3.4 Error Analysis

We employed confusion matrices to explore the class-wise prediction performance. The investigation found that the category Moderate was the most hard to classify. Figure 4.5 shows the normalized KNN confusion matrix. The class-wise recall for KNN was 66.27% (Fast), 27.36% (Moderate), 58.65% (Slow).

The Moderate class had the highest amount of classification errors, demonstrating a significant overlap between intermediate response times and the adjacent Fast and Slow classes. Figure 4.6.

Error rates by class.

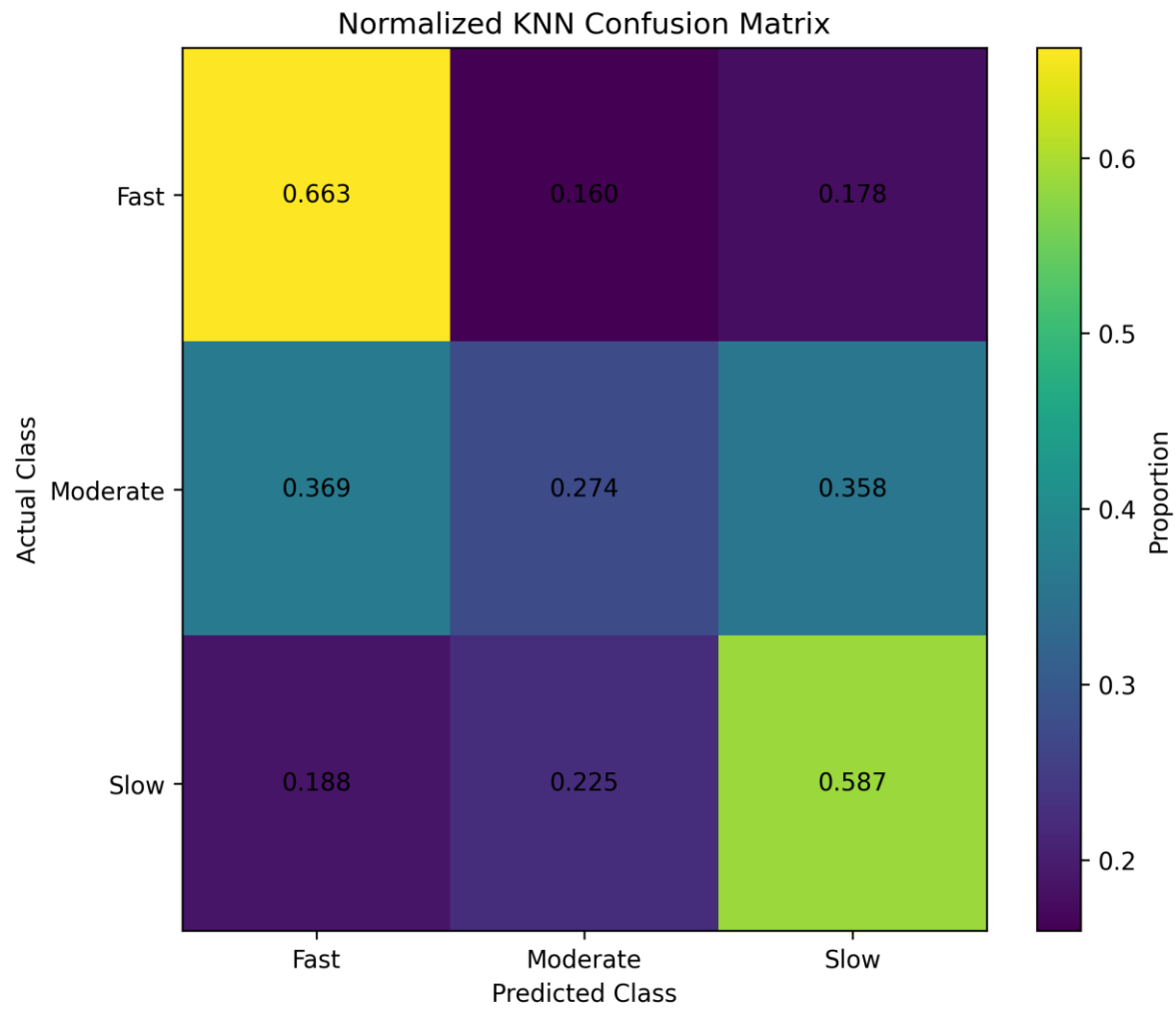


Figure 4.5: Normalized Confusion Matrix for K-Nearest Neighbors (KNN)

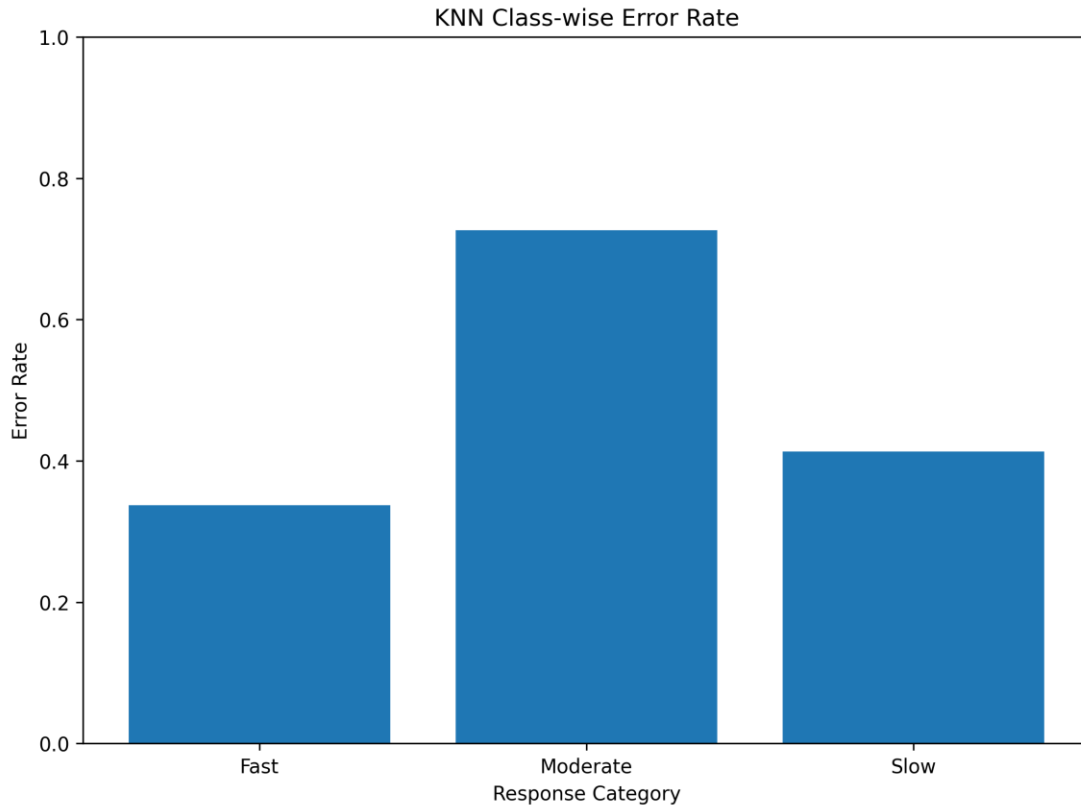


Figure 4.6: Class-Wise Error Rate for K-Nearest Neighbors (KNN)

The overall error rates were **40.35% for Random Forest, 45.96% for KNN, 48.99% for SVM, and 49.43% for Decision Tree**. Random Forest consequently produced the lowest error rate among the evaluated models.

4.3.5 Discussion

The experimental results suggest that relevant patterns can be extracted from citizen complaint records using machine-learning methods. Random Forest performed the best across the major evaluation measures and hence was picked as the optimal model for the citizen complaint analytics component. However, the prediction task is still tough as evidenced by the accuracy of about 59.65%. The difficulty in identifying Moderate concerns indicates that reaction time is influenced by elements not entirely captured in the existing complaint records such as workload, resource availability, inspection needs and administrative processes. Therefore, the model produced should be viewed as a decision support device and not as a substitute for municipal decision making. It

could help authorities to detect concerns that may require further monitoring, and enable data-driven service planning.

4.3.6 Summary

The Chicago 311 project tested four machine learning classifiers to predict citizen complaint response-category. Random Forest had the best performance with 59.65% accuracy, 58.17% Macro Recall and 56.08% Macro F1-score. The data also revealed that the Moderate class was the hardest class to forecast. The results suggest that AI-based citizen complaint analytics can be used to support smart-city service management.

4.4 Cross-Experiment Interpretation

To conclude, the results reveal that model performance depends on the task and the data modalities. For traffic forecasting, LSTM and CNN-LSTM models efficiently incorporated temporal relationships in multivariate traffic-speed sequences. The persistence model offered a robust short-horizon baseline. For air-quality prediction the LSTM model has shown a better overall performance than CNN-LSTM, suggesting that a higher architectural complexity does not necessarily imply a better forecasting accuracy. The results provide evidence for the usage of task-specific AI architectures, where sequence-based deep-learning models are selected for IoT data depending on the temporal features and prediction requirements of each application.

CHAPTER 5

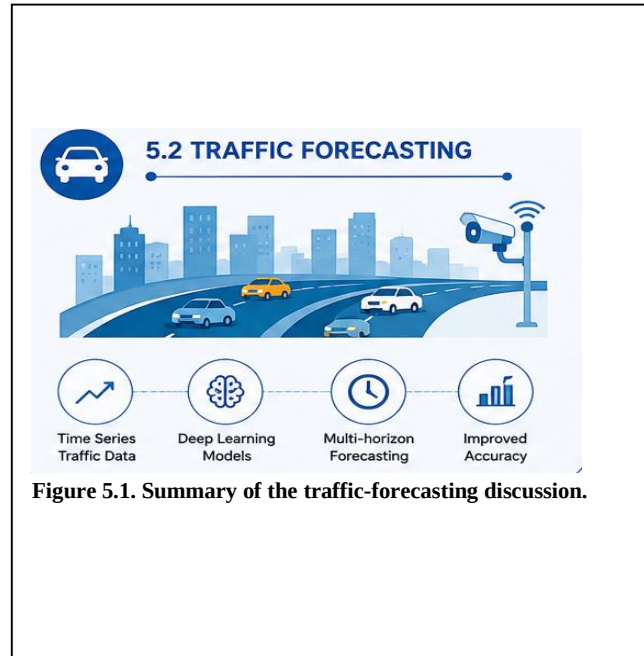
Discussion

5.1 Overview

This the experimental results derived from traffic forecasting, air-quality prediction and citizen complaint analytics are discussed. The results show that the performance of AI depends on the properties of the dataset, the prediction job and the model architecture that is chosen.

5.2 Traffic Forecasting

LSTM and CNN-LSTM improved important error measures compared with the persistence baseline. CNN-LSTM achieved the lowest overall MAE and WAPE, while LSTM achieved the lowest RMSE. However, the difference between the two deep-learning models was small, and LSTM demonstrated greater stability across multiple random seeds. The horizon-wise analysis showed that persistence performed particularly well at short horizons, whereas deep-learning models became more competitive at longer horizons. Low-speed traffic conditions produced higher errors, indicating that congested traffic states are more difficult to predict.



5.3 Citizen Complaint Analytics

The citizen complaint experiment for the application of machine learning algorithms for classification of service demands. The investigation revealed applicability of AI-based classification to large-scale urban service-request data. The investigated classification models demonstrated varying performance levels. The confusion-matrix study also showed disparities in the models' ability to correctly identify each complaint category. These results illustrate how citizen complaint analytics may convert unstructured or categorized service-request data into

useful insights. Such classification can help in automatic categorization of complaints, prioritization of services and better allocation of municipal resources. Forecasting traffic and air quality are mainly time series regression problems, whereas citizen complaint analytics is a classification challenge. Hence, its evaluation needs classification-specific measures and models rather than forecasting metrics.

5.4 Air-Quality Prediction

The LSTM model outperforms CNN-LSTM for PM2.5 prediction level in terms of MAE, RMSE and R^2 . Therefore, the additional neural feature extraction stage did not increase the generalization on the examined air-quality dataset. The results indicate that more sophisticated models do not always imply higher prediction ability. The simplified LSTM model was shown to be a better solution for the data set considered and also somewhat efficient.



Figure 5.2. Summary of the citizen-complaint-analytics

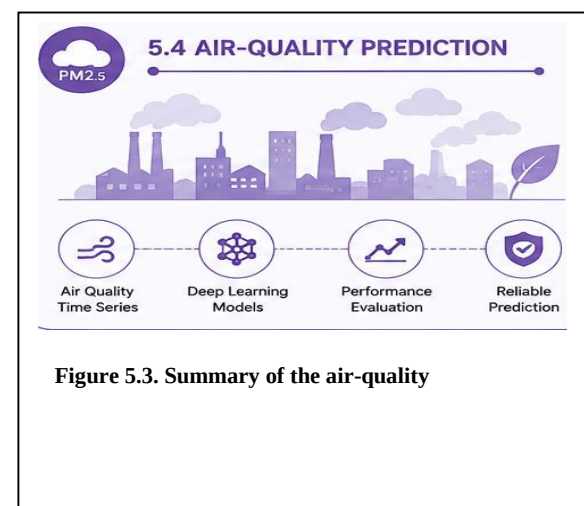


Figure 5.3. Summary of the air-quality

5.5 Overall Discussion

The combined findings justify the use of task-specific AI architectures for content analytics of IoT projects in smart-city. LSTM performed well and stably for all the forecasting tasks. CNN-LSTM had selected advantages in traffic forecasting but not in air-quality prediction. Citizen complaint analytics also illustrated the utility of machine-learning classification for urban service administration. The overall experimental results proved that there is no one best approach for all the cases. Thus, model selection should include data modalities, task requirements, predictive performance, interpretability and robustness rather than depending on model complexity alone.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

The explored traffic, air quality and citizen complaint data for AI-based content analytics for IoT-driven smart city applications. Persistence baseline was used for traffic forecasting with LSTM and CNN LSTM models. LSTM and CNN LSTM models were used for PM2.5 prediction level. Furthermore, the application of machine-learning classification algorithms was used to explore citizen complaint analytics. Both deep-learning models were helpful for traffic forecasting in terms of predictive performance compared with the persistence baseline. LSTM showed very consistent forecasting performance whereas CNN-LSTM got a little improved overall MAE and WAPE. LSTM beat CNN-LSTM for air-quality prediction over the investigated performance measures. The citizen complaint analysis also revealed the potential of machine-learning methods for finding meaningful patterns from urban service-request data. The results overall show the ability of AI-based temporal and machine-learning models to support predictive and analytical activities in smart-city contexts. However, the results also show that the performance of the model is dependent on the properties of the dataset and the prediction job. A more sophisticated model does not always generalize better than a simpler architecture.

6.2 Key Findings

The findings of research are summarized as follows:

- The LSTM model demonstrated good and consistent predicting performance for the assessed time-series tasks.
- CNN-LSTM demonstrated marginal gains on some traffic forecasting metrics.
- The persistence baseline still competed well for short-term traffic forecast.
- The LSTM was superior to CNN-LSTM in PM2.5 prediction level.
- The longer the traffic forecasting horizons the more difficult it was to forecast accurately.
- Relatively larger prediction errors were related with low-speed traffic circumstances.
- More sophisticated models did not always improve generalization.
- Machine-learning categorization revealed the feasibility of evaluating citizen complaints and service-request data.

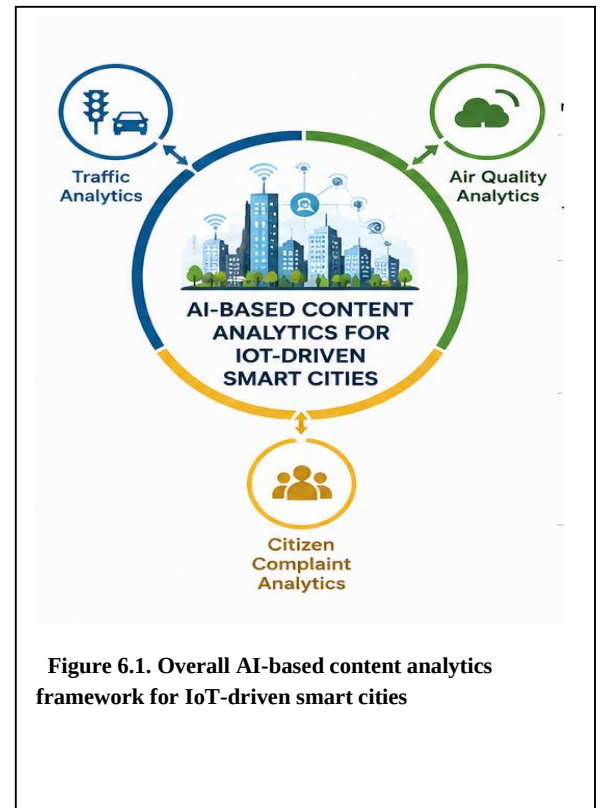


Figure 6.1. Overall AI-based content analytics framework for IoT-driven smart cities

6.3 Limitations

The study has some limitations. The experiments were conducted using specific historical datasets and therefore may not fully represent the variability of real-time IoT environments or other cities. The traffic and air-quality experiments also focused on a limited number of deep-learning architectures. In addition, the citizen complaint analysis was limited by the available historical dataset characteristics and the scope of the classification experiments. This study analyzes historical IoT representative datasets rather than live and streaming sensor data.



Figure 6.2. Overall impact of the proposed framework, from smarter cities and efficient services to citizen satisfaction, sustainable environment and data-driven decision-making.

6.4 Future Work

Future work can extend the proposed smart-city analytics framework in several directions:

- Integration of real-time IoT sensor streams for continuous prediction;
- Evaluation of Transformer and attention-based architectures;
- Incorporation of explainable AI techniques to improve model interpretability;
- Fusion of traffic and air-quality data for multimodal urban analysis;
- Validation of the proposed models across multiple cities and datasets;
- Development of real-time prediction and alert systems;
- Expansion of citizen complaint analytics using larger datasets and comprehensive numerical model evaluation; and
- Investigation of multimodal AI approaches for jointly analyzing heterogeneous smart-city data.

6.5 Final Statement

This study presents the viability of AI-based content analytics for extracting predictive and analytical insights from smart-city data. The experimental results reveal that AI models are useful for traffic forecasting, air-quality prediction and citizen-service analytics and suggest that the choice of model should depend on the features of the task. The future combination of real-time IoT streams, multimodal data, explainable AI and state-of-the-art deep-learning architectures can significantly improve the practical usefulness of AI-based analytics for intelligent and data-driven city administration.

REFERENCES

- [1] H. Nguyen, D. Nawara, and R. Kashef “Connecting the indispensable roles of IoT and artificial intelligence in smart cities: A survey,” *Journal of Information and Intelligence*, vol. 2, no. 3, pp. 261–285, 2024.
- [2] H. Omrany, K. M. Al-Obaidi, M. Hossain, N. A. M. Alduais, H. S. Al-Duais, and A. Ghaffarianhoseini, “IoT-enabled smart cities: A hybrid systematic analysis of key research areas, challenges, and recommendations for future direction,” *Discover Cities*, vol. 1, art. no. 2, 2024.
- [3] V. Bhardwaj, A. Anooja, L. S. Vermani, Sunita, and B. K. Dhaliwal, “Smart cities and the IoT: An in-depth analysis of global research trends and future directions,” *Discover Internet of Things*, vol. 4, art. no. 19, 2024.
- [4] [X] A. R. Sattarzadeh, P. N. Pathirana, R. Kutadinata, and V. T. Huynh, “Extracting long-term spatiotemporal characteristics of traffic flow using attention-based convolutional transformer,” *IET Intelligent Transport Systems*, vol. 18, no. 10, pp. 1797–1814, 2024.
- [5] S. K. Jagatheesaperumal, S. E. Bibri, J. Huang, J. Rajapandian, and B. Parthiban, “Artificial intelligence of things for smart cities: Advanced solutions for enhancing transportation safety,” *Computational Urban Science*, vol. 4, art. no. 10, 2024.
- [6] A. Binbusayyis, M. A. Khan, M. M. A. Ahmed, and W. R. Sam Emmanuel, “A deep learning approach for prediction of air quality index in smart city,” *Discover Sustainability*, vol. 5, art. no. 89, 2024.
- [7] I. Rakshit, “AI-driven cloud solutions for smart city data analytics,” *Systemic Analytics*, vol. 3, no. 1, pp. 27–34, 2025.
- [8] “AI-driven urban analytics using IoT-enabled mobile sensor networks for environmental monitoring,” *Procedia Computer Science*, vol. 272, pp. 619–624, 2025.
- [9] S. Dey, “Urban air quality index forecasting using multivariate convolutional neural network based customized stacked long short-term memory model,” *Process Safety and Environmental Protection*, vol. 191, 2024.
- [10] F. Shama, A. Aziz, and L. B. M. Deya, “CitySolution: A complaining task distributive mobile application for smart city corporation using deep learning,” *SoftwareX*, vol. 27, art. no. 101829, 2024.

THE END

MS Thesis

ORIGINALITY REPORT

14%
SIMILARITY INDEX

10%
INTERNET SOURCES

10%
PUBLICATIONS

6%
STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Bahria University, Islamabad Student Paper	1 %
2	mdpi-res.com Internet Source	1 %
3	doctorpenguin.com Internet Source	<1 %
4	arxiv.org Internet Source	<1 %
5	Bowen Dong, Xinyu Zhang, Weiyan Zhu, Lingmin Hou, Chaoya Yan, Yifan Feng, Lixing Lin. "Sensor-Driven Short-Term Forecasting on the Metropolitan LA Traffic Dataset: A Comparative Study for Multi-Step Prediction", Sensors, 2026 Publication	<1 %
6	peninsula-press.ae Internet Source	<1 %
7	discol.umk.edu.my Internet Source	<1 %
8	www.mdpi.com Internet Source	<1 %
9	Mega Orina Fitri, Muzaffar Hamzah, Adian Fatchur Rochim. "Emerging Trends in Statistical, Machine Learning, and Deep Learning Models for Air Quality Prediction: a Bibliometric Analysis", 2025 International Conference on Converging Technology in Electrical and Information Engineering (ICCTEIE), 2025 Publication	<1 %
10	Bhabani Prasanna Pattanaik, Dhirendra Kumar Shukla, Sambit Satpathy, Kamalakanta Muduli. "Sustainable Developments in Computer Engineering, Green Technology and Smart Systems", CRC Press, 2026 Publication	<1 %
11	ijcrt.org Internet Source	<1 %
12	Submitted to Macquarie University Student Paper	<1 %

13	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	<1 %
14	Submitted to Liverpool John Moores University Student Paper	<1 %
15	Poonam Nandal, Tapas Kumar, Meeta Singh, Mamta Dahiya. "Progressive Computational Intelligence, Information Technology, and Networking - Volume 2", CRC Press, 2026 Publication	<1 %
16	Sabereh Taghdisi Rastkar, Saeid Jamili, Enrico De Santis, Antonello Rizzi. "Graph-Aug LSTM with Weighted Loss for Enhanced Energy Forecasting", 2025 International Joint Conference on Neural Networks (IJCNN), 2025 Publication	<1 %
17	ijmscs.org Internet Source	<1 %
18	Franklin Oliveira, Laercio Pioli, Douglas D. J. De Macedo, Daniel G. Costa. "AI-Driven Low-Cost Sensors for Wildfire Detection: Performance Issues and Energy Efficiency", 2025 IEEE Symposium on Computers and Communications (ISCC), 2025 Publication	<1 %
19	Sakirin Tam, Rachid Ben Said, Omer Ozgur Tanriover. "A ConvBiLSTM Deep Learning Model Based Approach for Twitter Sentiment Classification", IEEE Access, 2021 Publication	<1 %
20	Submitted to Amrita Vishwa Vidyapeetham Student Paper	<1 %
21	R.K. Naresh, Rajan Bhatt, Krishan K. Verma, Hari Om Katiyar, Arun Kumar, Mauro Wagner de Oliveira. "Artificial Intelligence in Regenerative Agriculture - Transforming Soil Health and Crop Productivity", Elite Publishing House, 2026 Publication	<1 %
22	Submitted to Sri Krishna College of Engineering and Technology Student Paper	<1 %
23	9thinternationalcongresssoni.sched.com Internet Source	<1 %
24	thesis.cust.edu.pk Internet Source	<1 %
25	ijtm.my.id Internet Source	<1 %

26	Danie Zhang, Yang Zhang, Qi Li, Nathan Vance, Dong Wang. "Robust State Prediction with Incomplete and Noisy Measurements in Collaborative Sensing", 2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), 2018 Publication	<1 %
27	Mojtaba A. Farahani, Md Irfan Khan, Thorsten Wuest. "Hybrid agentic AI and multi-agent systems in smart manufacturing", Journal of Manufacturing Systems, 2026 Publication	<1 %
28	Submitted to Sunway Education Group Student Paper	<1 %
29	fastercapital.com Internet Source	<1 %
30	pmc.ncbi.nlm.nih.gov Internet Source	<1 %
31	bilselkongreleri.com Internet Source	<1 %
32	elmi.hbku.edu.qa Internet Source	<1 %
33	Submitted to Macau University of Science and Technology Student Paper	<1 %
34	www.diva-portal.org Internet Source	<1 %
35	upcommons.upc.edu Internet Source	<1 %
36	Fawad Ahmad, Khurram S. Khattak, Sheheryar Ahmad, Zawar Hussain Khan. "A LightGBM-Transformer Hybrid Model for Traffic Flow Prediction Using Temporal Features", 2025 20th International Conference on Emerging Technologies (ICET), 2025 Publication	<1 %
37	Submitted to Higher Education Commission Pakistan Student Paper	<1 %
38	Sreenivasulu T, Mokesh Rayalu G. "Accurate Hourly AQI Prediction Using Temporal CNN-LSTM-MHA+GRU: A Case Study of Seasonal Variations and Pollution Extremes in Visakhapatnam, India", Results in Engineering, 2025 Publication	<1 %
39	conferences.computer.org	

	Internet Source	<1 %
40	dspace.nwu.ac.za Internet Source	<1 %
41	iarjset.com Internet Source	<1 %
42	scholar.ppu.edu Internet Source	<1 %
43	Submitted to Rajagiri School Of Engineering & Technology Student Paper	<1 %
44	Submitted to The Hong Kong Polytechnic University Student Paper	<1 %
45	Submitted to University of Essex Student Paper	<1 %
46	"Intelligence Science and Big Data Engineering. Big Data and Machine Learning", Springer Science and Business Media LLC, 2019 Publication	<1 %
47	"Internet of Vehicles and Computer Vision Solutions for Smart City Transformations", Springer Science and Business Media LLC, 2025 Publication	<1 %
48	Submitted to George Mason University Student Paper	<1 %
49	Sukhpal Singh Gill. "Modern Computing - Principles and Paradigms", CRC Press, 2026 Publication	<1 %
50	Submitted to University of Liverpool Student Paper	<1 %
51	Submitted to Westcliff University Student Paper	<1 %
52	algoml-industry.netlify.app Internet Source	<1 %
53	d.lib.msu.edu Internet Source	<1 %
54	digitalcommons.liu.edu Internet Source	<1 %
55	ir-library.ku.ac.ke	

	Internet Source	<1 %
56	www.frontiersin.org Internet Source	<1 %
57	Moreira, Guilherme Valler. "Predicting Bike Stations Occupation Using Time-Series Models: the Case Study of Lisbon", Universidade NOVA de Lisboa (Portugal), 2025 Publication	<1 %
58	api.sciforum.net Internet Source	<1 %
59	elibrary-dev.nusamandiri.ac.id Internet Source	<1 %
60	learning-gate.com Internet Source	<1 %
61	research.ict.ru.ac.za Internet Source	<1 %
62	www2.mdpi.com Internet Source	<1 %
63	Submitted to Middle East Technical University Student Paper	<1 %
64	Suneeta Satpathy, Álvaro Rocha, Sachi Nandan Mohanty, Tanupriya Choudhury. "Intelligent Data-Driven Systems with Innovations in Artificial Intelligence", CRC Press, 2025 Publication	<1 %
65	Submitted to Vrije Universiteit Amsterdam Student Paper	<1 %
66	azdok.org Internet Source	<1 %
67	gupea.ub.gu.se Internet Source	<1 %
68	ieomsociety.org Internet Source	<1 %
69	mde.maryland.gov Internet Source	<1 %
70	www.egekongresi.org Internet Source	<1 %

71	Bokor, O.. "Improving Labour Productivity in Construction. A Hybrid Machine Learning Approach", University of Northumbria at Newcastle (United Kingdom), 2024 Publication	<1 %
72	cspub-ijcism.org Internet Source	<1 %
73	elar.rsvpu.ru Internet Source	<1 %
74	jmmf.atu.ac.ir Internet Source	<1 %
75	odr.chalmers.se Internet Source	<1 %
76	repositorio.unesp.br Internet Source	<1 %
77	theseus.fi Internet Source	<1 %
78	vbook.pub Internet Source	<1 %
79	www.aishe.info Internet Source	<1 %
80	www.ijprems.com Internet Source	<1 %
81	Berna Cengiz, Iliyasa Yahya Adam, Mehmet Ozdem, Resul Das. "A survey on data fusion approaches in IoT-based smart cities: Smart applications, taxonomies, challenges, and future research directions", Information Fusion, 2025 Publication	<1 %
82	Daniel S. Berman. "DGA CapsNet: 1D Application of Capsule Networks to DGA Detection", Information, 2019 Publication	<1 %
83	Kolaghassi, Rania. "Deep Learning for Gait Prediction: An Application to Exoskeletons for Children with Neurological Disorders", University of Kent (United Kingdom), 2024 Publication	<1 %
84	Xiaolong Liang, Xing Qiao, Yongtao Li. "Obstructive Sleep Apnea Detection Using Combination of CNN and LSTM Techniques", 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), 2019 Publication	<1 %

85	amsdottorato.unibo.it Internet Source	<1 %
86	assets-eu.researchsquare.com Internet Source	<1 %
87	dataaspirant.com Internet Source	<1 %
88	files.hawaii.gov Internet Source	<1 %
89	impa.usc.edu Internet Source	<1 %
90	pubs.sciepub.com Internet Source	<1 %
91	repository.lcu.edu.ng Internet Source	<1 %
92	rucforsk.ruc.dk Internet Source	<1 %
93	smartcity.efri.uniri.hr Internet Source	<1 %
94	theses.hal.science Internet Source	<1 %
95	www.inderscience.com Internet Source	<1 %
96	www.open-access.bcu.ac.uk Internet Source	<1 %
97	Ladeira, Edgar Filipe Loureiro. "A Machine Learning-Based Decision Support System for IoT Monitoring Solutions.", Universidade da Beira Interior (Portugal) Publication	<1 %
98	Abejide Ade-Ibijola, Solomon Oyelere, Jude Osakwe. "Innovation in Africa for Industry 4.0 - AI, Data Analytics and Digital Transformation", Chapman and Hall/CRC, 2026 Publication	<1 %
99	Bahman Zadeh, Zeinab. "Modeling Spatial Distribution of Particles in Transportation Systems Using Computational Fluid Dynamics and Machine Learning Approaches", Drexel University Publication	<1 %

100	Gaurav Aggarwal, Rishabh Dev Shukla, Innocent Ewean Davidson, Ramesh Bansal. "Integrated Technologies in Electrical, Electronics and Bio-technology Engineering", CRC Press, 2026 Publication	<1 %
101	Likcani, Kevin. "Use of Machine Learning to Predict Drug Court Success.", University of Kansas Publication	<1 %
102	Makkar, Gaurav. "Learning Based Model Correction and Data Imputation Algorithms for Rotorcraft Systems", Rensselaer Polytechnic Institute, 2023 Publication	<1 %
103	Rui Yan, Jiaqiang Liao, Jie Yang, Wei Sun, Mingyue Nong, Feipeng Li. "Multi-Hour and Multi-Site Air Quality Index Forecasting in Beijing Using CNN, LSTM, CNN-LSTM, and Spatiotemporal Clustering", Expert Systems with Applications, 2020 Publication	<1 %
104	Sarkar, Md. Rasel. "Deep Learning Techniques Improvement for Time Series Forecasting.", University of New South Wales (Australia) Publication	<1 %
105	Simon Elias Bibri, Jeffrey Huang, Osama Omar, Inji Kenawy. "Synergistic integration of digital twins and zero energy buildings for climate change mitigation in sustainable smart cities: A systematic review and novel framework", Energy and Buildings, 2025 Publication	<1 %

Exclude quotes
Exclude bibliography

Off
Off

Exclude matches

Off