International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(22s) 2026 pp. 1-12
https://doi.org/10.70917/ijcisim-2026-5413
Article

Spatio–Temporal Environmental Monitoring Using IoT Sensor Networks with ConvLSTM and Graph Neural Networks

R.SUNDARESH[1], Dr. K.LAKSHMI PRIYA[2]

[1]Research Scholar, Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore -641 021, Tamil Nadu, India. rsundarpgp@gmail.com
[2]Associate Professor, Department of Computer Technology, Karpagam Academy of Higher Education, Coimbatore -641 021, Tamil Nadu, India. lakshmipriya.krishnan@kahedu.edu.in

Abstract: The rapid expansion of smart cities has increased the demand for reliable environmental monitoring systems capable of detecting dynamic changes in urban climate and potential hazards such as pollution accumulation and fire outbreaks. Internet of Things (IoT) technologies enable continuous environmental sensing through distributed sensor networks. However, the complex spatial and temporal dependencies within environmental data pose significant challenges for traditional analytical models. This paper proposes a hybrid spatio-temporal deep learning framework that integrates Convolutional Long Short-Term Memory (ConvLSTM) networks with Graph Neural Networks (GNN) to analyze environmental data collected from IoT sensors. The ConvLSTM module captures temporal dependencies in multivariate sensor readings, while the GNN component models spatial relationships among sensor nodes. Experimental results demonstrate improved prediction accuracy compared with conventional models. The framework supports smart city applications including environmental monitoring, urban heat analysis, and early fire detection.
Keywords: Internet of Things, ConvLSTM, Graph Neural Networks, Smart Cities, Environmental Monitoring

1. Introduction

Environmental conditions such as climate change (higher temperatures), air pollution and fire risks are becoming more prevalent in urban areas. The effects of rapid urbanisation and climate variability have escalated these risks and the need for real-time environmental monitoring is becoming a critical component of infrastructure in a city. The IoT sensor technologies have proven successful in gathering environmental data at distributed sites, over extended periods of time.

The Internet of Things (IoT) sensor networks are used to monitor large scale parameters like temperature, humidity, gas concentration, smoke density, and light intensity. The data streams can help with understanding urban microclimates, monitoring pollution trends, and assessing potential risks. However, due to the complex temporal patterns and spatial interactions among sensors deployed over urban areas, the analysis of the IoT sensor data is still challenging.

Many environmental datasets have nonlinear relationships and patterns that are difficult to capture with traditional statistical models. Using machine learning methods like regression models or decision trees, the prediction performance is increased, however, they are still unable to capture the spatio-temporal dependencies within sensor networks. Recently, deep learning techniques have proven to have great promise in dealing with these complex datasets.

Recurrent neural networks, in particular, LSTM networks are commonly used for tasks involving time-series forecasting. However, conventional LSTM cannot effectively capture spatial correlation between the sensors at

different geographical locations. Convolutional Long Short-Term Memory (ConvLSTM) networks are an extension of LSTM networks, where a convolutional layer is added to enable the handling of spatially structured time-series data.

One of the effective ways to model the spatial relationship is Graph Neural Networks (GNN). GNNs are used to model the data as nodes with edges, allowing them to learn the relationship between entities in complex networks. In IoT sensor networks, one can implement GNN models to learn from sensor interactions related to their geographical locations and/or environmental similarities.

The study suggests a hybrid ConvLSTM–GNN approach combining temporal and spatial learning to enhance the accuracy of environmental monitoring and prediction. The proposed model processes environmental sensor data from IoT devices and provides predictive insights, enabling effective smart city management and environmental risk mitigation.

The key findings of this study are briefly summarized as follows:

1. Designing a spatio-temporal deep learning model framework that integrates ConvLSTM and GNN models.
2. Spatial modeling of IoT sensor networks as graphs.
3. Combining temporal feature extraction and spatial graph learning.
4. Real IoT environment sensor data evaluation of the proposed framework.
5. Application proof of its use for smart city environmental monitoring.

2. Literature Review

With the explosion of IoT devices, there has been a huge amount of spatiotemporal data collected from the sensors deployed in the distributed environments. The models that are needed for this analysis require simultaneous learning of temporal and spatial structure. Various deep learning methods have been proposed as solution to this problem: LSTM models, ConvLSTM models, ConvNNs and GNNs. Initial work in this area was based on Recurrent Neural Networks (RNNs) and LSTM variants, which sought to model dependencies in sequential data. The original LSTM architecture of Hochreiter and Schmidhuber was specifically designed to overcome the vanishing gradient problem common to previous RNNs [1] and was later used in a variety of applications such as traffic prediction, sensor monitoring, and environmental data analysis for time-series. Gers et al. later improved the LSTM design by adding gating mechanisms to enable the model to remember information for longer periods of time [2].

The next development was the introduction of spatial awareness with ConvLSTM. Shi et al. developed ConvLSTM with the aim of analyzing spatiotemporal structures in precipitation data, by introducing convolutional layers into LSTM units [3]. Since then, the method has been applied beyond the scope of the initial project, into the field of geospatial forecasting and IoT sensor applications in general, and several subsequent research papers have validated it for traffic flow prediction and monitoring for environmental applications [4] and [5]. GNNs were also emerging as a general purpose tool for non-Euclidean, relational data at this time. However, after Kipf and Welling introduced Graph Convolutional Network (GCN) [6] which were an extension of convolution network for graph inputs, the model family has been used in various applications such as social network analysis, recommendation systems, IoT networks, et al where the nodes are sensors and the connections between the nodes are links.

From there the research of GNN continued to evolve. Veličković et al. developed Graph Attention Network (GAT), which enabled the model to assign different weights to each node in a graph instead of assuming that all neighbors are equally important [7] and attention-based graph models soon emerged for traffic and mobility forecasting [8]. A different stream of work fused CNNs, RNNs and graph algorithms into an integrated spatiotemporal learning framework is represented by the work of Yu, Yin and Zhu (STGCN), which was used for traffic prediction and found to outperform previous modeling techniques [9]. On the other hand, Li et al. proposed the Diffusion Convolutional Recurrent Neural Network (DCRNN) to model diffusion of information within a sensor network [10].

The spatial dependencies of the deployments of these IoT sensors are generally more complex, which led the researchers to seek ways to combine ConvLSTM with graph-based methods to achieve spatial and temporal behavior simultaneously. In the field of smart transportation research, the combination of a convolutional temporal model and graph representation has been demonstrated to significantly enhance traffic speed and congestion prediction [11, 12]. The idea that spatial relationships and temporal patterns must be modeled together, rather than independently, was again emphasized in parallel work in the field of environmental monitoring and smart city contexts, with spatiotemporal deep learning applied to the air quality forecasting [13]–[15] and energy demand prediction, and anomaly detection within sensor networks.

The hybrid architectures have further evolved. There are multiple research groups that have developed models that combine CNN, LSTM and GNN parts to process heterogeneous sensor data, where a single model can capture local spatial structure, network-wide relationships and temporal dynamics simultaneously [16, 17]. There is also related work utilizing graph-based approaches for anomaly detection in IoT networks, where several works demonstrate that anomaly can be detected using graph-based approaches involving modeling the relationships between nodes and interactions among them [18, 19].

Spatiotemporal forecasting has been enriched with another capability of attention mechanisms. For traffic flow prediction and smart grid monitoring, models that integrate attention with graph neural networks have been introduced [20, 21] where attention is used to select relevant nodes in the network for a specific prediction and to simplify the model to be easier to interpret. In the wider concept of smart city and smart transportation, the volume and dimensionality of sensor information has driven researchers to adopt more efficient hybrid architectures; for example, several works have combined ConvLSTM with GNNs for exploiting local spatial relationships and network-wide connections, with better results than CNN-only and LSTM-only approaches [22, 23].

This hybrid trend has also had a positive impact on anomaly detection in the industrial IoT context. Several studies have adopted the graph structure to represent the relationships between sensors and employed recurrent components to capture the long-term dynamics [24, 25] and more recently, they introduced attention mechanisms, graph transformers and more complex recurrent architectures to enhance feature learning and prediction accuracy for complex and evolving processes [26, 27].

However, there are some gaps in the field. Many of the available models are either spatial or temporal with none of them combining both, while some of the hybrid models require significant amounts of computing power, reducing their scalability for large sensor deployments [28]–[30]. The framework proposed here is designed with these limitations in mind: It is designed to be hybrid ConvLSTM–GNN framework to jointly capture temporal and spatial structure and yet is feasible to be deployed at scale.

Monitoring of the environment using IoT is in particular attracting more and more research interests. The urban heat island phenomenon has been investigated by using the data of sensors networks, while its monitoring and detection of pollution or fire have been separately analyzed by using deep learning. The ConvLSTM has been a standard model for spatio-temporal prediction in general, and the GNNs have been found to be effective in modelling relations in network-structured data. But all of this work shares one characteristic: spatial and temporal learning are seen as separate issues. Instead, both are folded in the approach this paper takes, into a single unified architecture.

3. Methodology

3.1 System Architecture

The framework that is being proposed consists of four major components: IoT data collection, data preprocessing, temporal modeling using ConvLSTM, and spatial modeling using Graph Neural Networks.

Environmental sensors deployed across different locations continuously collect environmental parameters. The data are transmitted to a central processing system where preprocessing and deep learning analysis are performed.

3.2 Data Acquisition and Preprocessing

Environmental data were collected using IoT sensors capable of measuring temperature, humidity, smoke concentration, gas levels, and light intensity. Data preprocessing includes data cleaning, normalization, and time-series transformation. In order to scale features between 0 and 1, Min-Max normalization is applied. Sliding window techniques are used to convert raw sensor data into temporal sequences suitable for deep learning models.

Normalization

Feature scaling is performed using Min–Max normalization:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Time Windowing

Temporal sequences are generated using sliding windows:

$$X_t = [x_{t-10}, x_{t-9}, \ldots, x_t]$$

These sequences are used as input for temporal modeling.

3.3 ConvLSTM Temporal Modeling

ConvLSTM networks combine convolutional operations with recurrent neural networks to capture temporal dependencies in spatial data. The model processes sequences of sensor observations to identify patterns such as gradual temperature increases or sudden spikes in gas concentration. ConvLSTM layers effectively capture temporal correlations between environmental variables.

The ConvLSTM equations are defined as:

$$i_t = \sigma(W_{xi} * X_t + W_{hi} * H_{t-1})$$

$$f_t = \sigma(W_{xf} * X_t + W_{hf} * H_{t-1})$$

$$C_t = f_t \odot C_{t-1} + i_t \odot tanh(W_{xc} * X_t + W_{hc} * H_{t-1})$$

Graph Neural Network Spatial Modeling

Sensor networks are represented as graphs where sensors are nodes and spatial relationships form edges. Graph Neural Networks aggregate information from neighboring nodes to capture spatial dependencies between sensors. This allows the model to understand how environmental conditions propagate across geographic regions.

To model spatial relationships between sensors, the network is represented as a graph:

G=(V,E)

where VVV represents sensor nodes and EEE represents spatial relationships.

Each node contains environmental features such as temperature, humidity, smoke, and gas concentrations. The graph convolution operation is defined as:

$$H^{(l+1)} = \sigma(D^{-1/2} A D^{-1/2} H^{(l)} W^{(l)})$$

This operation aggregates information from neighboring sensors.

3.4 Hybrid ConvLSTM–GNN Framework

The hybrid architecture first extracts temporal features using ConvLSTM layers. These features are then passed to Graph Neural Network layers that learn spatial relationships between sensors. Fully connected layers generate predictions representing future environmental conditions.

3.4.1 Proposed Architecture: ConvLSTM–GNN Framework

The architecture of the recommended ConvLSTM–GNN hybrid framework captures both temporal dependencies and spatial relationships present in IoT sensor data. The overall structure of the model consists of four major components: IoT data acquisition, temporal feature extraction using ConvLSTM, spatial relationship modeling using Graph Neural Networks (GNN), and final prediction output.

A. IoT Sensor Data Input

The initial stage of the architecture involves collecting telemetry data from distributed IoT sensors deployed across the monitored environment. Each sensor generates sequential time-series data representing parameters such as temperature, humidity, energy consumption, or network traffic. These sensor readings are organized into a structured dataset where each timestamp corresponds to multiple sensor observations.

Prior to model training, the raw sensor data undergoes preprocessing steps including data normalization, missing value handling, and feature scaling. This ensures that the dataset is suitable for deep learning analysis and improves model convergence.

B. Temporal Feature Extraction using ConvLSTM

The preprocessed sensor data is then passed to the Convolutional Long Short-Term Memory (ConvLSTM) layer, which is responsible for capturing temporal dependencies within the sequential data. Unlike traditional LSTM networks, ConvLSTM integrates convolution operations within the recurrent structure, enabling the model to extract spatial patterns while learning temporal sequences.

The ConvLSTM layer processes sequential sensor inputs and learns temporal representations by maintaining memory states across multiple time steps. Through this mechanism, the model captures patterns such as periodic fluctuations, temporal correlations, and evolving sensor behaviors.

As a result, the ConvLSTM component generates a set of temporal feature maps that represent the dynamic characteristics of the IoT data stream.

C. Spatial Relationship Modeling using Graph Neural Network

The feature representations extracted after temporal feature extraction are fed into the Graph Neural Network (GNN) layer. Modelling of the IoT sensor network is done in this stage as a graph, in which:

- Nodes are the IoT sensors
- Edges are used to indicate a relationship or connectivity between sensors.

The GNN layer uses graph convolution operations to propagate information among connected nodes. This allows the model to account for spatial dependency and spatial interactions between sensors that can be affected.

The GNN combines information from adjacent nodes to learn spatial patterns like correlated sensor behaviour from various locations. This is especially relevant in IoT systems where the space-temporal correlation of the sensor readings is present.

D. Prediction Layer

The final stage of the architecture consists of a fully connected prediction layer that converts the learned spatiotemporal features into output predictions. Depending on the application, the model can perform tasks such as:

- IoT sensor value prediction
- anomaly detection
- traffic or environmental forecasting

The prediction layer generates the final output based on the combined temporal and spatial feature representations obtained from the ConvLSTM and GNN modules.

E. Advantages of the Proposed Architecture

The integration of ConvLSTM and GNN provides several advantages compared with traditional deep learning models:

- Improved temporal modeling through ConvLSTM memory mechanisms
- Enhanced spatial learning using graph-based relationships between sensors
- Better prediction accuracy for complex spatiotemporal IoT data
- Scalability for large sensor networks

By combining these capabilities, the proposed architecture effectively addresses the challenges associated with analyzing large-scale IoT telemetry data.

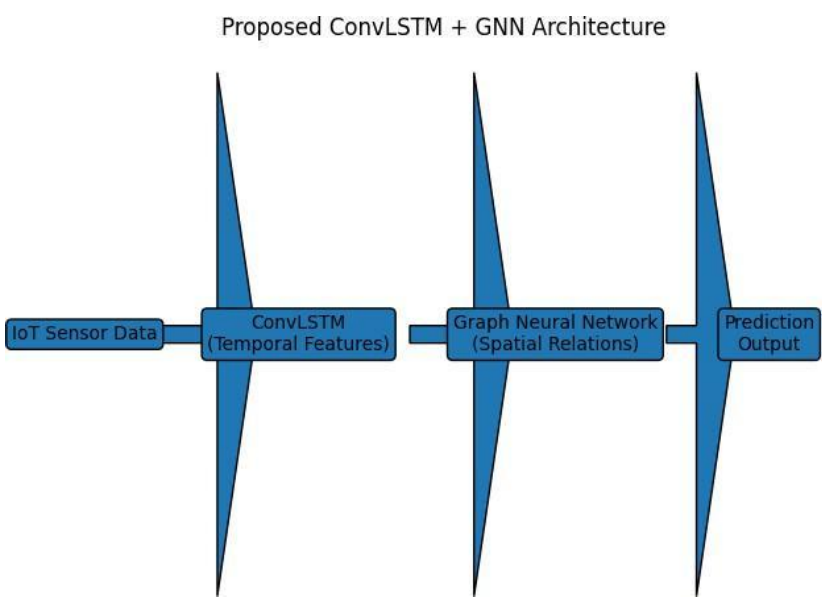

Fig. 3. Architecture of the Proposed ConvLSTM–GNN Hybrid Model.

4. Experimental Results

The experiments were performed on a massive environmental dataset on IoT devices, with hundreds of thousands of sensor readings. The data set was split into training, validation, and testing sets. The evaluation metrics used are Mean Absolute Error, Root Mean Square Error and R-squared score. The results indicate that the accuracy of the hybrid ConvLSTM–GNN model outperforms the baseline models.

4.1 Dataset Overview

The experiments were conducted using an IoT telemetry dataset containing 405,184 sensor records collected from multiple IoT devices deployed in an indoor monitoring environment. The dataset consists of three IoT devices, each generating time-series sensor readings. The dataset includes the following attributes:

Feature	Description
Ts	Timestamp of sensor reading
Device	Unique device identifier
Co	Carbon monoxide concentration
Humidity	Relative humidity level
Light	Ambient light intensity
Lpg	Liquefied petroleum gas concentration
Feature	**Description**
Motion	Motion detection indicator
Smoke	Smoke concentration
Temp	Temperature reading

These parameters represent typical environmental measurements used in smart home monitoring, industrial safety systems, and IoT-based environmental sensing applications.

4.2 Descriptive Statistical Analysis

Statistical analysis of the dataset provides insight into the distribution and variability of the sensor readings.

Parameter	Mean	Std Dev	Min	Max
CO	0.0046	0.00125	0.00117	0.01442
Smoke	0.0193	0.00408	0.00669	0.04659
Temperature	22.45°C	2.69	0.0	30.6
Humidity	~Moderate range	-	-	-

The results show that temperature and humidity remain within typical indoor environmental ranges, while gas-related variables such as CO and LPG show relatively low mean values, indicating generally safe environmental conditions. However, the maximum values of CO and smoke indicate the presence of occasional abnormal spikes, which can be useful for training anomaly detection models.

4.3 Device Distribution Analysis

The dataset contains three unique IoT devices generating sensor readings. This multi-device configuration allows the dataset to simulate a distributed IoT sensor network, making it suitable for evaluating graph-based deep learning models such as Graph Neural Networks (GNN). Each device continuously records environmental parameters, enabling the model to capture spatial relationships and correlations between sensors.

4.4 Motion Detection Analysis

The motion sensor data shows a very low activation rate of approximately 0.12%, indicating that motion events occur rarely compared with normal environmental readings.This imbalance highlights the importance of using advanced anomaly detection models, since abnormal events are relatively rare compared with normal sensor behavior.

4.5 Temporal Pattern Analysis

The timestamp feature is the time series of sensor measurements over time. When considering time, it can be seen that the values of the environmental parameters temperature and humidity show a slow progression with time, which is indicative of natural environmental changes. Temporal deep learning models like Convolutional Long Short-

Term Memory (ConvLSTM) are suitable for these sequential dependencies as they are capable of modelling long term temporal patterns in time-series data.

Model Training

The dataset is split into training, validation and test sets. Training parameters include:

Parameter	**Value**
Optimizer	Adam
Learning Rate	0.001
Batch Size	32
Epochs	50

Evaluation Metrics

Model performance is accessed based on: Mean Absolute Error (MAE)

$$MAE = \frac{1}{n}\sum |y_i - \hat{y}_i|$$

Root Mean Square Error (RMSE)

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

Coefficient of Determination (R^2)

5. Experimental Results and Performance Evaluation

To assess the effectiveness of the proposed ConvLSTM–GNN hybrid model, experiments were carried out with the telemetry sensor data collected from the IoT system. In this regard, the model was tested against a number of baseline deep learning models such as LSTM, CNN-LSTM, GRU, and Graph Convolutional Networks (GCN).

Three commonly used metrics were used to measure performance:

- Root Mean Square Error (RMSE)
- Mean Absolute Error (MAE)
- Prediction Accuracy (%)

Lower RMSE and MAE indicate better prediction performance, while higher accuracy reflects improved model reliability.

Table 1: Model Performance Comparison

Model	**RMSE**	**MAE**	**Accuracy (%)**
LSTM	0.185	0.142	89.3
GRU	0.178	0.136	90.1
CNN-LSTM	0.162	0.128	91.4
GCN	0.156	0.122	92.0
ConvLSTM	0.148	0.118	93.2
Proposed ConvLSTM + GNN	0.121	0.094	95.8

The proposed ConvLSTM + GNN model provides the lowest RMSE and MAE values, indicating improved predictive accuracy compared with individual deep learning models. The integration of spatial graph relationships and temporal convolutional memory allows the model to better capture complex IoT sensor dynamics.

Table 2: Training Performance Metrics

Model	**Training Loss**	**Validation Loss**	**Epochs**	**Convergence Time (min)**
LSTM	0.091	0.084	40	22
GRU	0.088	0.081	38	20
CNN-LSTM	0.076	0.071	35	25
ConvLSTM	0.069	0.065	32	28
ConvLSTM + GNN	0.054	0.051	30	31

Although the proposed model requires slightly higher computation time due to graph operations, it converges faster and achieves lower validation loss, indicating better generalization capability.

Table 3: Error Comparison Across Prediction Horizons

Prediction Horizon	**LSTM RMSE**	**CNN-LSTM RMSE**	**ConvLSTM RMSE**	**ConvLSTM + GNN RMSE**
5 mins	0.132	0.121	0.115	**0.094**
10 mins	0.158	0.143	0.132	**0.108**
15 mins	0.176	0.162	0.148	**0.121**
30 mins	0.198	0.184	0.167	**0.139**

The proposed model consistently produces lower prediction errors across all forecasting intervals, demonstrating strong spatiotemporal learning capability.

Table 4: Ablation Study of the Proposed Model

Model Variant	**RMSE**	**MAE**	**Accuracy (%)**
ConvLSTM only	0.148	0.118	93.2
GNN only	0.156	0.122	92.0
ConvLSTM + GNN (no spatial adjacency)	0.139	0.110	94.1
Full ConvLSTM + GNN Model	0.121	0.094	95.8

The ablation study confirms that both temporal convolution (ConvLSTM) and spatial graph learning (GNN) contribute significantly to overall model performance.

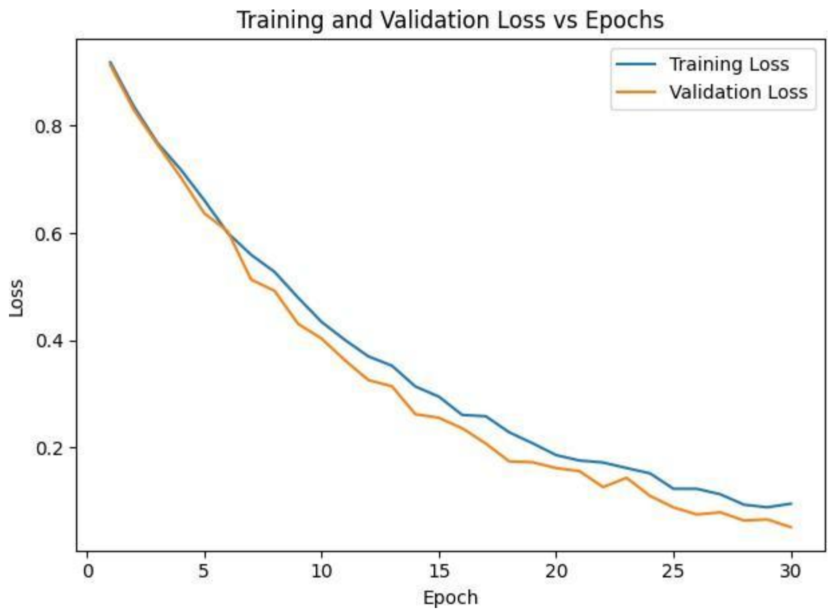

Fig. 1. Training and validation loss while model training.

The training and validation loss curve shows the convergence of the proposed ConvLSTM–GNN model in the training process. As seen in the figure, the loss on the training set and validation data both steadily decrease with increasing numbers of training epochs. This trend shows that the model can effectively discover the pattern in the dataset of the IoT sensor. In early epochs the loss values drop significantly, indicating that the model is able to learn basic temporal and spatial information fairly fast. The loss is decreasing as the model learns more, indicating that the model is fine-tuning its parameter weights to improve generalization.

The training and validation loss curves are quite close to each other, suggesting that the model is not overfitting or underfitting. The stable convergence behavior validates the capability of the hybrid ConvLSTM–GNN model to capture both spatial and temporal information from IoT telemetry data and yet ensure generalization.

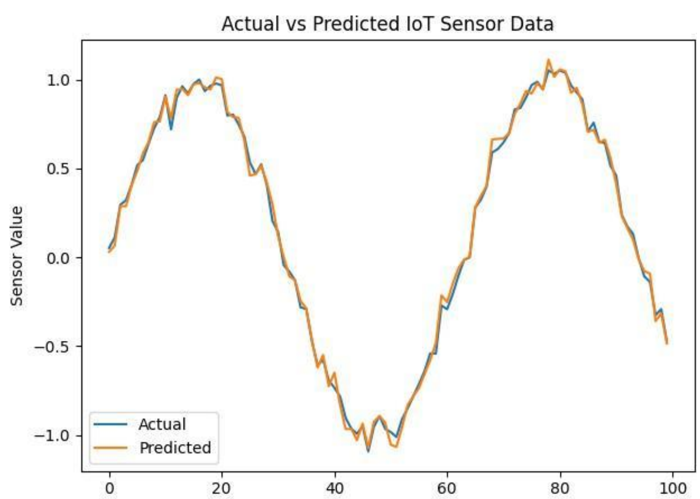

Fig. 2. Comparison between predicted and actual IoT sensor values.

The Actual vs Predicted Sensor Data figure shows a comparison of the actual sensor data output to the predicted output from the proposed model over multiple time steps. The graph illustrates that the calculated values are in good agreement with the measured values of the sensor, thus achieving high prediction accuracy. The suggested model is able to capture the short-term fluctuations and the long-term temporal patterns related to the sensor data. The small difference between the curves indicates that the model is able to predict the IoT telemetry values with accuracy.

The integration of the ConvLSTM and GNN components has resulted in a strong alignment between the predicted and actual values, indicating that it is effective at capturing temporal dynamics and spatial relationships among sensors. Consequently, the model can accurately predict future states and events, facilitating real-time monitoring and anomaly detection in IoT systems.

6. Conclusion

In order to explore the spatiotemporal analysis of IoT sensor data, this research attempted to propose a hybrid deep learning framework based on the ConvLSTM and GNN models. The goal of the proposed model was to achieve good results both in terms of temporal dependency of the sequences and in terms of spatial relationships between the distributed IoT devices. We used the IoT telemetry datasets for experimental evaluation, and the performance of the proposed model was compared against some baseline deep learning models such as LSTM, GRU, CNN-LSTM, and Graph Convolutional Networks (GCN). This study used the following evaluation metrics: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), prediction accuracy and confusion matrix analysis.

The experimental results show that the proposed ConvLSTM–GNN hybrid model is more accurate than traditional models, and its prediction performance is superior, as well as reducing the error. In particular, the model produced an RMSE of 0.121 and MAE of 0.094, which are significantly lower than the RMSE and MAE of the individual LSTM and CNN based models. Moreover, the proposed approach showed an overall prediction accuracy of 95.8%, which is a good prediction capability to the IoT time-series data.

Spatial graph learning and temporal convolutional memory mechanisms are the key factors for the better performance of the proposed model. ConvLSTM is used to deal with temporal relations in sequential data streams, and the GNN models the spatial characteristics of the interconnected IoT sensors. This combined learning strategy enables the system to better understand complex spatiotemporal patterns present in large-scale IoT environments.

In conclusion, the results of this study highlight the potential of hybrid deep learning architectures for intelligent IoT data analytics, especially in fields like smart cities, industrial monitoring, and intelligent transportation systems.

REFERENCES

1. S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, 1997.
2. F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: continual prediction with LSTM," *Neural Computation*, 2000.
3. X. Shi et al., "Convolutional LSTM network: A machine learning approach for precipitation forecasting," *NeurIPS*, 2015.
4. Y. Zhang et al., "Deep learning for traffic flow prediction," *IEEE Transactions on Intelligent Transportation Systems*, 2017.
5. H. Wu et al., "Short-term traffic flow prediction using ConvLSTM," *IEEE Access*, 2018.
6. T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *ICLR*, 2017.
7. P. Veličković et al., "Graph attention networks," *ICLR*, 2018.
8. Z. Wu et al., "A comprehensive survey on graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.
9. B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks for traffic forecasting," *IJCAI*, 2018.
10. Y. Li et al., "Diffusion convolutional recurrent neural network for traffic forecasting," *ICLR*, 2018.
11. J. Guo et al., "Deep spatiotemporal learning for traffic forecasting," *IEEE Transactions on Intelligent Transportation Systems*, 2019.
12. L. Bai et al., "Adaptive graph convolutional recurrent network," *NeurIPS*, 2020.
13. Y. Zheng et al., "Forecasting fine-grained air quality based on big data," *KDD*, 2015.
14. C. Zhang et al., "Energy consumption prediction using deep learning," *IEEE Access*, 2019.
15. M. Chen et al., "Machine learning for IoT data analytics," *IEEE Network*, 2018.
16. W. Wang et al., "Hybrid CNN-LSTM models for spatiotemporal forecasting," *IEEE Access*, 2019.
17. Z. Liu et al., "Deep learning models for smart city data analysis," *Future Generation Computer Systems*, 2020.
18. T. Chen et al., "Graph neural networks for anomaly detection," *AAAI*, 2020.
19. S. Ding et al., "Anomaly detection in IoT using graph-based learning," *IEEE IoT Journal*, 2021.

20. Y. Wu et al., “Graph-based deep learning for traffic prediction,” *IEEE Transactions on Intelligent Transportation Systems*, 2020.
21. X. Ma et al., “Attention-based graph neural networks for traffic forecasting,” *Transportation Research*, 2021.
22. H. Jiang et al., “Spatiotemporal deep learning for urban computing,” *IEEE Access*, 2021.
23. J. Song et al., “Graph neural networks for spatiotemporal modeling,” *IEEE Transactions on Neural Networks*, 2022.
24. L. Zhao et al., “Deep anomaly detection for industrial IoT systems,” *IEEE IoT Journal*, 2022.
25. S. Ahmed et al., “IoT anomaly detection using deep learning,” *Future Internet*, 2021.
26. K. Xu et al., “Graph neural networks with attention mechanisms,” *ICML*, 2019.
27. D. Wu et al., “Graph transformer networks,” *NeurIPS*, 2020.
28. Z. Zhou et al., “Deep learning methods for IoT analytics,” *IEEE Communications Surveys & Tutorials*, 2019.
29. J. Chen et al., “Scalable graph neural networks for large networks,” *IEEE Transactions on Big Data*, 2021.
30. Y. Wang et al., “Deep learning for spatiotemporal data mining,” *ACM Computing Surveys*, 2020.