IoT-Based Environmental Monitoring Systems for Air Quality Assessment Using Graph-Aware Transformer Neural Control

D.S. Vijayan,
SRM Institute of Science and Technology, Vadapalani Campus,
Chennai, India.

Email : Vijayan.has.siva@gmail.com

Corresponding Author

D.S. Vijayan[1*]
Email : Vijayan.has.siva@gmail.com

Received: *15 April 2026* ***Revised:*** *18 June 2026* ***Accepted:*** *10 July 2026*
Published: *10 September 2026*

Abstract

Air pollution has become a serious public health and environmental issue, driven by quick urbanization, industrialization and a growing number of cars on the road, which necessitates smart city air quality monitoring systems with real-time intelligent capabilities. The current air quality forecasting techniques both rely on spatial learning or temporal analysis and do not provide an adaptive environmental control over distributed IoT sensor networks. Addressed these limitations, this research aims at proposing a novel Graph-Aware Transformer Neural Control (GAT-NC) framework for IoT-based environmental monitoring and air quality assessment. The proposed GAT-NC algorithm combines the spatial dependency learning capability of Graph Attention Networks (GAT), long-range pollution pattern analysis using Transformer-based temporal modeling, spatiotemporal feature fusion, adaptive nonlinear control generation and attention-based learning mechanisms for intelligent environmental regulation. The experimental results show that the proposed GAT-NC model can outperform the baseline models in terms of accuracy (98.1%) and minimum RMSE (0.072). The framework is robust, stable and reliable through statistical validation using t-test and ANOVA. Based on the research, the proposed GAT-NC framework is efficient, scalable and intelligent solution for real-time air quality prediction and adaptive air quality control in IoT-based smart city infrastructures.

Keywords: Adaptive Environmental Control, Air Quality Monitoring, Graph Attention Network, Internet of Things, Smart City, Spatiotemporal Feature Fusion.

1. Introduction

Rapid urbanization, industrialization and growing vehicle traffic produce one of the most severe public health and environmental problems: air pollution, in which deep learning techniques can use in air pollution monitoring by automatically extracting features from the data [1]. Attention-based modal decomposition also contributes to the stability and robustness of AQI forecasting [2] and IoT-based machine learning techniques boost AQI prediction in smart cities [3]. Long-range temporal dependence is well learned by self-attention mechanisms and Transformer-BiLSTM models, which are effective for accurate pollution prediction [4]. Conventional monitoring systems are costly and not suitable for large-scale real-time deployment in smart cities [5] while, the IoT-based deep learning systems can be used for efficient real-time monitoring of pollutants like CO, NOx and NO_2 [6]. The temporal dependence of air quality time-series is well captured by LSTM models [7].The spatio-temporal Graph Convolutional Networks (ST-GCN) is a technique that is well suited to learn the spatial interaction of the distributed sensors and thus accurately predict the air quality [8]. In distributed fusion networks, it adds some spatial and temporal information [9] and in hybrid networks, it use Transformer Encoder, CNN and BiLSTM to boost the temporal local and global information [10]. The type of monitoring systems used in the continuous prediction of air quality, i.e. IoT type, is associated with the smart city context. As indicated by the research [12] there are spatial gradients both within the city and between urban, suburban and rural areas with respect to pollution. The GNNs can be used to model the spatial relationship between distributed sensors well in the city-wide air quality forecasting [13]. Three dimensional (3D) CNN models can be used to accurately predict multi-step air quality forecasts [14] and may have great potential in smart cities where air quality monitoring and (short-term) forecasts need to be accurate, reliable and resilient. Such prediction models, which use powerful deep learning techniques, have great potential to improve the performance of these smart air quality monitoring and forecasting system, especially for the smart cities in which high accuracy, reliability and resilience are vital requirements [15]. To address these issues, a novel Graph-Aware Transformer Neural Control (GAT-NC) paradigm is proposed in this research, which uses the graph attention mechanism and transformer modules to capture the temporal information and predict the air quality more efficiently and precisely in smart cities to adapt environment.

- An innovative GAT-NC paradigm, using the combination of graph attention and transformer is proposed.
- To gain an understanding of how to efficiently extract features of air quality from the spatiotemporal domain via attention techniques and transformer model.
- In smart cities, an adaptive nonlinear control generation module use in dynamic environmental regulation.
- The system will be integrated into the integrated system along with the inclusion of air quality and spatiotemporal characteristics and adaptive control in air quality forecasting.

The rest of the research is organized as follows: In Section 2 provide the background study. In Section 3 present the proposed GAT-NC. The results of the experiments are given in section 4. A discussion is included in Section 5 and the key findings and directions for future research are summarized in Section 6.

2. Background study

Fang et al. (2023) [16] to enhance the accuracy of pollution mapping, introduced a dual-stage satellite-based PM2.5 estimation framework using machine learning. While boosting prediction accuracy, the framework did not incorporate real-time integration with IoT and didn't provide dynamism to its use in changing conditions.

Kang et al. (2023) [17] has proposed an LWS-based real-time air quality monitoring system with lower cost sensors, which utilizes the edge computing and IoT. Although it provided rapid local processing, sensor drift and reduced accuracy compared to industrial-grade sensors were still a problem.

Devarakonda et al. (2023) [18] proposed a mobile crowd sensing system for urban air quality monitoring system based on mobile devices and cloud computing. The data collection was inconsistent influencing data quality and deep learning models were not included in the framework.

Yücel and Talu (2025) [19] proposed an IoT based real-time air quality prediction system with sensor networks, with the use of machine learning methods. The accuracy of predictive performance was enhanced, but the model failed to adequately reflect the spatiotemporal relationships or the adaptive control of environment.

Liu et al. (2023) [20] proposed an adaptive attention graph convolution network for city-wide air quality prediction. While the model performed well in prediction, it was still cost intensive in terms of computing resources and was therefore not suitable for real-time application in IoT.

He et al. (2022) [21] has proposed an integrated spatiotemporal forecasting model based on graph neural network (GNN), which used multiple environmental factors. Although the prediction performance was improved, the model was very complex, hard to interpret.

Liu et al. (2023) [22] proposed a model of spatial-temporal data called the MSDR model to improve the long-term prediction and multi-step forecasting of spatial-temporal data. It demanded however, a serious tuning of parameters and big training sizes, which lowered the efficiency in calculation.

Xu et al. (2025) [23] proposed the Spatiotemporal Graph Attention Network (STGATN) to forecast air pollutant concentrations at multiple monitoring stations. The model became more accurate, but it was also rather complex and would not run on resource-limited IoT devices.

Wu et al. (2023) [24] developed a hybrid deep learning framework for AQI prediction by extracting spatial-temporal patterns. Although the model had better prediction accuracy, it had high computational cost and did not take into consideration real time responsiveness of IoT.

Table 1: Comparative Literature Review of Existing Environmental Monitoring Systems for Air Quality Assessment

Author (Year)	Method Used	Limitation	Research Gap	Result
Kim et al. (2024) [25]	Spatiotemporal Graph Convolutional	Limited to PM10 only; no multi-pollutant	No adaptive control or multi-pollutant	Improved allergy patient prediction using PM10

	Network (ST-GCN)	integration	spatiotemporal fusion	spatial dependencies
Liang et al. (2024) [26]	Graph Convolutional Network + Multi-Head Attention (GCN-MHAM)	High complexity; not scalable for real-time IoT deployment	No dynamic sensor graph construction or adaptive control mechanism	Enhanced forecasting accuracy over standalone GCN and LSTM models
Chen & Guestrin (2022) [27]	eXtreme Gradient Boosting (XGBoost)	Cannot capture spatiotemporal dependencies between sensors	Lacks deep spatiotemporal learning and real-time IoT control response	Competitive AQI prediction with fast training on tabular environmental data
Liang et al. (2023) [28]	AirFormer – Transformer with stochastic spatiotemporal attention	High computational cost; unsuitable for lightweight IoT edge devices	No adaptive control output or real-time feedback for smart city monitoring	Reduced prediction error by 5–8% on 72-hour air quality forecasts
Liang et al. (2022) [29]	Geo-sensory Multi-level Attention Network (GeoMAN)	Static spatial relationships; limited to fixed sensor graphs	No graph attention-based node weighting or IoT pollution control	Outperformed RNN baselines on multi-scale geo-sensory prediction tasks
Pazhanivel et al. (2024) [30]	Fog-enabled Air Quality Monitoring and Prediction (FAQMP)	No graph attention or transformer-based temporal fusion	Lacks inter-sensor relational learning and adaptive environmental control	Efficient real-time monitoring with reduced latency via fog computing

Table 1 compares six air quality prediction models, revealing that ST-GCN and GCN-MHAM are limited to single-pollutant prediction, XGBoost lacks spatiotemporal modelling capability and AirFormer and GeoMAN are unsuitable for lightweight IoT deployment without adaptive control. The proposed GAT-NC framework addresses these gaps by integrating graph attention, temporal fusion and adaptive control for accurate real-time IoT-based air quality monitoring.

2.1 Research Gap

Present air quality forecast models only consider the single-pollutant prediction task or feature engineering approach in a table-based structure without considering the dynamics spatiotemporal dependency between IoT sensor networks. The transformer and attention-based approaches can capture the spatiotemporal dependency well but have computationally high complexity and cannot apply to lightweight IoT architectures where adaptive control does not exist. Presently used methods do not consider spatial dependency, temporal dependency and adaptive control generation in the model. This limitation can be solved by our proposed model known as GAT-NC framework based on the graph attention network approach for spatial dependency, temporal dependency based on the transformer model and adaptive control layer with the non-linear approach.

3. Proposed Methodology

The section explains that the spatial relationship between the IoT sensor data is learned in the GAT-NC framework with the GAT and the temporal relationship between pollution measurements is learned in the Transformer component. These spatiotemporal characteristics are then used to generate adaptive control commands according to AQI after undergoing non-linear transformations.

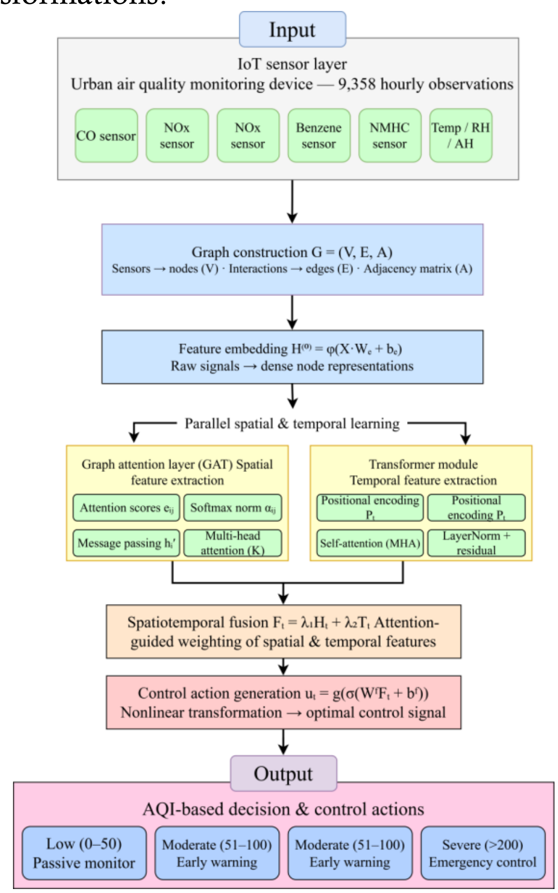


Figure 1: IoT-Based Environmental Monitoring Systems

Figure 1 shows the proposed GAT-NC framework. The graph is a representation of data from the sensor of IoT, CO, NOx, NO_2 and relations. A Graph Attention Network is used to capture the spatial features and a Transformer network to learn the temporal patterns. The features are combined to create control actions based on the AQI that will trigger the related monitoring or emergency intervention: those features are combined.

3.1 Dataset Description

Air Quality Data Set (9358 rows, 10 columns) https://www.kaggle.com/datasets/fedesoriano/air-quality-data-set from consists of 9,358 observations for an hour time scale collected by an air quality monitoring sensor deployed as an Internet of Things (IoT) application in a polluted urban environment. It incorporates information from a number of metal oxide sensors for key pollutants such as CO, NOx, NO_2 , Benzene and NMHC. Some pollutant data, as well as some environmental data such as temperature, relative humidity and absolute humidity are also recorded. Also the data set includes the ground truth values from a reference analyzer for the accurate evaluation and validation of the model. It is sufficiently long to accommodate real world alterations such as weather, traffic flow and sensor drift.

3.2 Graphs-Aware Transformer Neural Control (GAT-NC)

The GAT-NC is a hybrid intelligent control paradigm based on the Graph Attention Network (GAT) to model a dynamically changing graph with different characteristics by learning this graph representation and using the Transformer Networks for temporal modeling in the hybrid paradigm. The framework is a representation of the physical or cyber-physical system as a graph, where the sensors, actuators and subsystems are modeled as nodes on the graph. It consists of two parts: feature embedding and graph attention based message passing, where attention coefficients are adaptive and only select the relevant nodes for information aggregation to include their associated spatial features. The self-attention mechanism is then used to extract long-range temporal dependencies and context information from sequential data with a Transformer Network. The extracted spatial and temporal features are fused by the attention-guided fusion strategy and then fed into the nonlinear transformation layers to get optimal and adaptive control actions. The system state is obtained, a graph is constructed, hierarchical feature learning is implemented, spatiotemporal feature fusion is performed and a decision to make the control operation is generated in the system of Graph Attention Network-Neural Controller (GAT-NC). The advanced techniques, such as multi-head attention, residual connections, layer normalization and iterative parameter updates, improve the convergence, stability and generalization of the model and the model delivers strong performance by minimizing tracking error or maximizing cumulative reward. Furthermore, it supports real-time streaming data processing with fast inference and enables real-time and adaptive control decision in the dynamically changing operating condition.

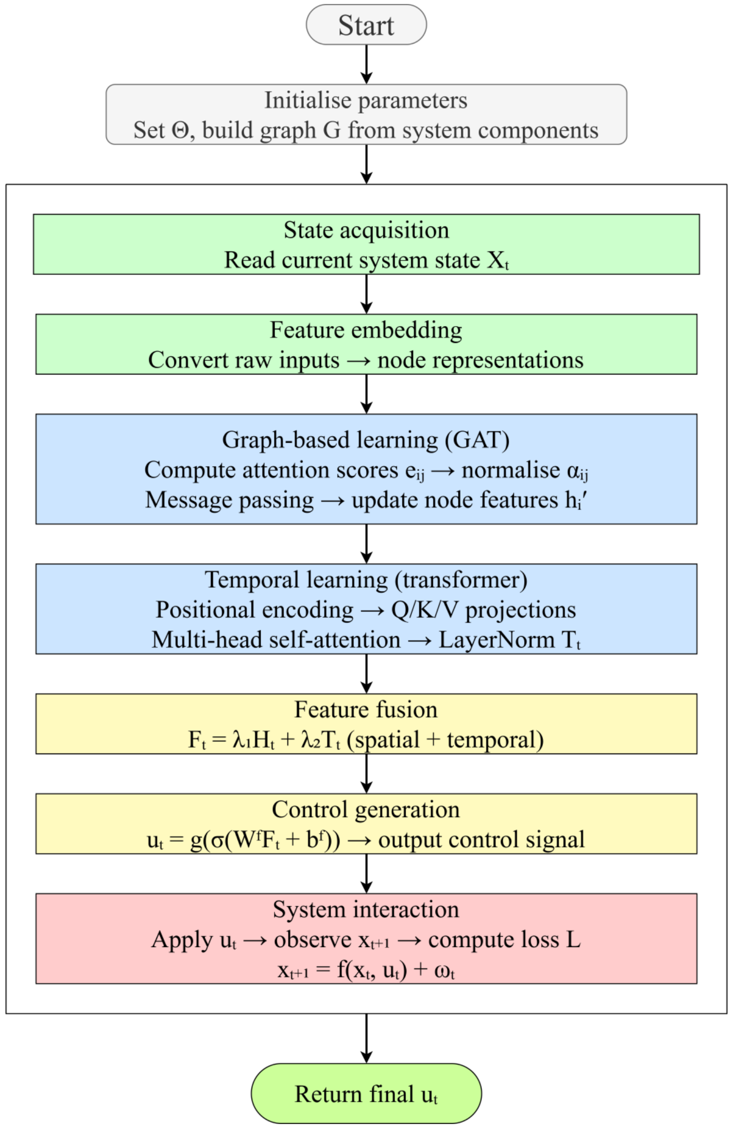


Figure 2: GAT-NC Workflow

The figure 2 illustrates the iterative process of the GAT-NC with 7 steps (which are repeated at each time step t). The blocks of Teal are responsible for input acquisition and embedding, purple blocks are for parallel spatial GAT learning and temporal (Transformer) learning, Coral blocks are for spatiotemporal fusion and control generation and Amber blocks are for system interaction and loss computation. The dashed arrow means that the loop has to be repeated for the next time step until all T steps are completed.

3.21 Graph-Based Representation and Feature Embedding

The system is represented graphically, with nodes representing components of the system and the lines representing interactions between the components. The raw data that comes from the sensors and subsystems are passed to the system where converted to dense feature representations of the system. This embedding allows the model to locate information on a local level at each node as well as the structural relationship on a global level. It is a simple representation, which is needed for the next graph-based learning and attention mechanism.

$$H^{(0)} = \emptyset(XW_e + b_e) \tag{1}$$

The first embedding step can be modelled mentioned in the above equation (1) , where the nonlinear activation function $\emptyset(\cdot)$ is applied to the feature matrix X and the learnable embedding weight matrix W_e and its bias vector b_e are learnt for the first embedding step.

$$e_{ij} = LeakyReLU(\alpha^T[Wh_i||Wh_j]) \tag{2}$$

The equation (2) is used to compute the attention score between the nodes prior to the normalisation; e_{ij} is the attention coefficient of node i before normalisation with node j; α is a learnable weight coefficient vector; α^T is the transpose of α; W is a weight matrix applied to the input features of the nodes; h_i and h_j are the feature representations of node i and node j, respectively.

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k \in N_i} \exp(e_{ik})} \tag{3}$$

In equation (3), The exponential function $\exp(\cdot)$ is the function to normalize the attention score e_{ij} for node$i$, where N_i is the neighbourhood set of node i.

$$h_i^{'} = \sigma(\textstyle\sum_{j \epsilon N_i} \alpha_{ij} W h_j) \tag{4}$$

The equation (4) is for the passing message step and $h_i^{'}$ is an updated feature of node i; and $\sigma(\cdot)$ is an activation function, α_{ij} is the attention coefficient and W is a weight matrix and h_j is the feature of neighboring node j.

$$h_i^{'} = ||_{k=1}^{K} \sigma(\textstyle\sum_j \alpha_{ij}^{(k)} W^{(k)} h_j) \tag{5}$$

The equation (5) is referred to as multi-head attention and the matrices are $W^{(k)}$ for each head$k$, while$||_{k=1}^{K}$ means concatenation of the outputs of all heads.

3.22 Temporal Modeling Using Transformer Mechanism

The transformer module is used to model the data in the system that changes with time. Using the self-attention model, the model can learn the long range temporal correlation without relying on recurrent structures. This enables it to learn what the previous system states mean for the next system states, over much longer time horizons. Multi-head attention further enhances the ability to learn multiple temporal patterns at the same time and thus increases the prediction accuracy in dynamic environments.

$$S = \{h^{t-T}, \dots, h^t\} \tag{6}$$

Equations (6) define the temporal sequence: S is the sequence of node representations over time, t is the current time step, T is the length of the sequence, which is the time range in which the model is applied.

$$P_t = \sin\left(\frac{t}{10000^{2i}/_d}\right) + \cos\left(\frac{t}{10000^{2i}/_d}\right) \tag{7}$$

Equation (7) represented by the positional encoding. P_t positional encoding at time t, d and is the scaling dimension and dimension index respectively.

$$Z_t = H_t + P_t \tag{8}$$

The equation (8) is a fusion of spatial information and positional information, with the input to the transformer being Z_t, the graph-based feature at time t H_t and the positional encoding P_t

$$Q = ZW_Q, K = ZW_K, V = ZW_V \tag{9}$$

The equations (9) are projections, where Q, K and V are matrices of query, key and value respectively, Z is the input representation and W_Q, W_K and W_V are learnable projection matrices.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right) \tag{10}$$

The attention scores are calculated in self-attention as in equation (10): QK^T is the similarity between queries and keys; $\sqrt{d_k}$ is the dimensionality of the key vector used to scale; and softmax normalizes the attention weights.

$$MHA(Z) = ||_{k=1}^{H} Attention(Q, K, V) \tag{11}$$

Multi-head attention is shown as in equation (11) $||_{k=1}^{H}$ it concatenates the output of each head, H number of attention heads.

$$T_t = LayerNorm(Z_t + MHA(Z_t)) \tag{12}$$

The output of the transformer is mentioned in the above equation (12), where T_t is the temporal representation and the addition is a residual connection.

3.23 Spatiotemporal Feature Fusion Strategy

This is done through the combination of the spatial information from the graph structure together with the temporal information derived by the transformer. The combination process is achieved by a dynamic weighting mechanism which depends on the attention mechanism. As a result, there is a sense of assurance that structural and temporal are both taken in consideration. This provides one space time representation of the behaviour of the system.

$$F_t = \lambda_1 H_t + \lambda_2 T_t \tag{13}$$

Equation (13) combines two features: F_t is the fused feature and $\lambda_{(1,2)}$ are trainable parameters that control the weightings of the spatial and temporal features.

3.24 Control Action Generation Mechanism

This fused spatiotemporal feature is then input into a set of nonlinear transformation layers to obtain optimal control outputs. These layers are high dimensional-to-low dimensional mappings to signals of the type that can be used for system regulation. The system is made flexible by adapting the control decisions as a function of the evolving system and its environment. The model thus produces stable, accurate and 'real-time' controlled actions in complex dynamic systems.

$$U_t = \sigma(W_f F_t + b_f) \tag{14}$$

The equation (14) is considered as a nonlinear transformation, where U_t represents the transformed feature for control, W_f denotes a weight matrix; b_f denotes a bias term and σ represents an activation function.

$$u_t = g(U_t) \tag{15}$$

The equation (15) produces control action by the signal u_t and the function $g(U_t)$ which is called the control policy or mapping function.

$$x_{t+1} = f(x_t, u_t) + \omega_t \tag{16}$$

The system dynamics represented by equation (16) is the one for which the current state x_t, next state x_{t+1}, nonlinear system function $f(x_t, u_t)$ and noise or disturbance ω_t are given.

$$\mathscr{L} = \frac{1}{2}||x_t - x_t^{ref}||^2 \tag{17}$$

The loss function used will be presented in the equation (17) where x_t^{ref} the reference state is and $||x_t - x_t^{ref}||^2$ is the Euclidean norm of the tracking error.

$$\theta \leftarrow \theta - \eta \nabla_\theta \mathscr{L} \tag{18}$$

The update to the parameters is given in equation (18) where θ is all the learnable model parameters, $\nabla_\theta \mathscr{L}$ denotes the gradient of the loss function with respect to parameters and η is the learning rate.

Algorithm : Graph-Aware Transformer Neural Control (GAT-NC)

Input:
   X_t : System state at time t
   G : Graph structure containing nodes and edges
   Θ : Model parameters
   T : Total time steps
   α : Learning rate
Initialize all model parameters Θ
Build graph structure G from system components
FOR each time step t from 1 to T DO
1.State Acquistion
   Read current system state X_t
2.Feature Embedding
   FOR each node in the graph DO
      Convert raw input features into node representations
   END FOR
3. Graph-Based Learning
   FOR each node in the graph DO
      Identify neighboring nodes
      Compute importance of each neighbor
      Combine neighbor information based on importance
      Update node representation
   END FOR
4. Temporal Learning Module
   Collect all node representations into a sequence
   Apply attention-based sequence learning
   Refine representations using deep learning layers
   Produce temporal features
5. Feature Fusion
   Merge spatial features and temporal features
   Generate unified system representation

```
6.Control Generation
    Pass unified representation through control network
    Produce control action U_t
7. System Interaction
    Apply control action to the system
    Observe new system state
    Evaluate system performance and compute loss
8. Model Update
    Adjust model parameters using learning algorithm
END FOR
Return final control action U_t
Output:
    U_t : Control action
```

GAT-NC pseudo-code describes an algorithmic process that ensures efficient system control by constantly analyzing the system data for effective action. The process begins with learning the spatial features of the nodes through the embedding and aggregation of neighboring information using a graph-based method. The learned spatial features are then fed into the temporal learning model to learn dynamics over time. The learned representations are merged into a unified feature space, from which control actions are derived. Finally, the process involves updating the system parameters based on performance feedback.

4. Experimental Result

This section presents the experimental evaluation of proposed GAT-NC framework on the UCI Air Quality dataset consisting of 9358 hourly sensor measurements. The data has been preprocessed and divided into 80% training data and 20% testing set to assess the accuracy of the predictions and adaptability of the system in real-time. To obtain stable learning and accurate spatiotemporal air quality prediction, the model was trained with the learning rate of 0.001, the batch size of 32, 100 epochs, 4 attention heads and 64 hidden units.

Table2: Dataset Description Table

Feature Name	Description	Unit
CO	Carbon Monoxide concentration	mg/m^3
NOx	Nitrogen Oxides concentration	$\mu g/m^3$
NO_2	Nitrogen Dioxide concentration	$\mu g/m^3$
Benzene	Benzene concentration	$\mu g/m^3$
NMHC	Non-Methane Hydrocarbons	$\mu g/m^3$
Temperature	Ambient temperature	°C
RH	Relative Humidity	%
AH	Absolute Humidity	g/m^3

The features used in the dataset and what it mean and what units are measured in are explained in the table 2. Involves pollutants such as CO, NOx, NO_2 , Benzene, NMHC etc. this are important for monitoring air quality. This includes environmental conditions like temperature, relative humidity and absolute humidity also. The model uses these factors to interpret pollution and environmental conditions. It is used in combination as input data for the training and testing of the system.

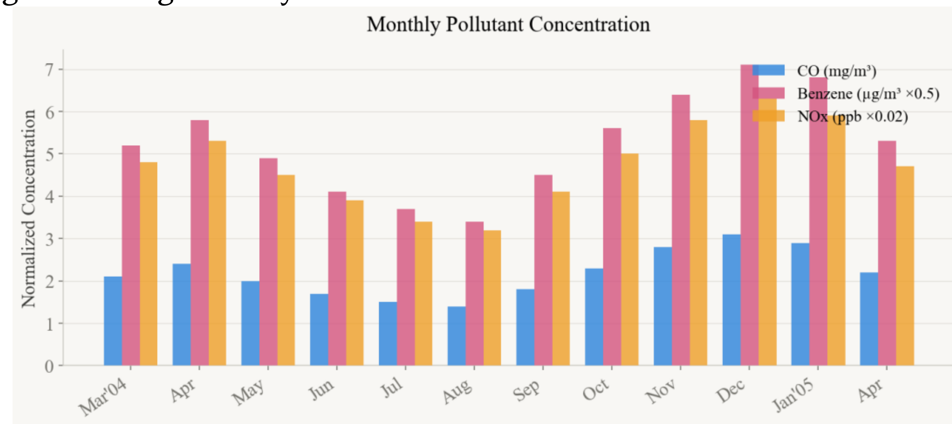


Figure 3: Time-series variation of CO, NO_2 and Benzene concentrations over the monitoring period

The figure 3 shows that how pollutant levels (like CO, NO_2 and Benzene) change over time. Assists in identification of trends such as high pollution hours, sudden spikes etc. The graph is a real-life phenomenon (like industrial activity or traffic levels etc.). It is useful for learning about the changes over time. The plot can be used for time based prediction and control.

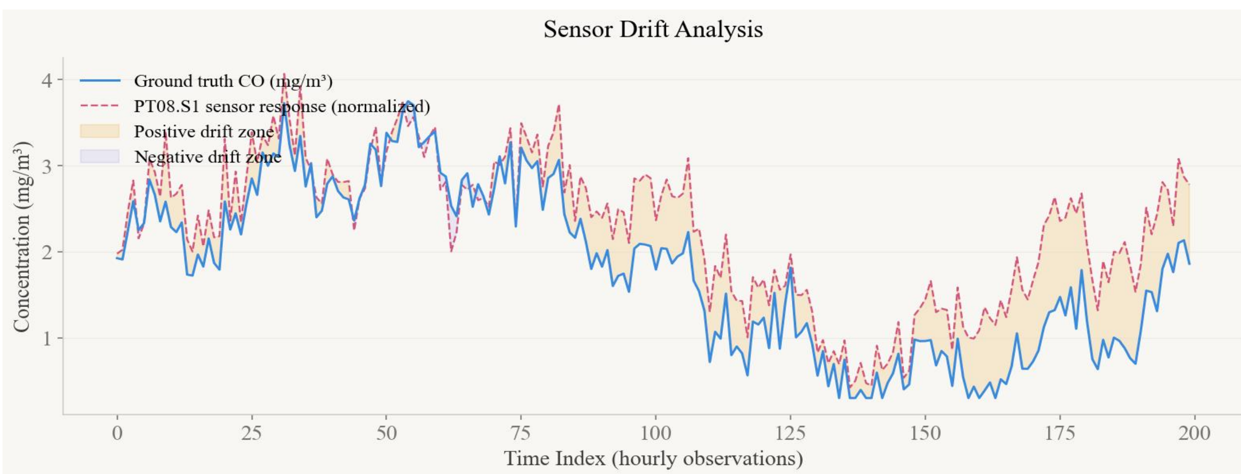


Figure 4: Comparison of metal oxide sensor readings against reference analyzer

The figure 4 depicts the graph is used to compare the sensor reading against the true value of the pollutant. This will use to determine the accuracy of the sensors. Any difference between the two shows that there is some drift or fault in the sensor. This will ensure that the model prediction is correct.

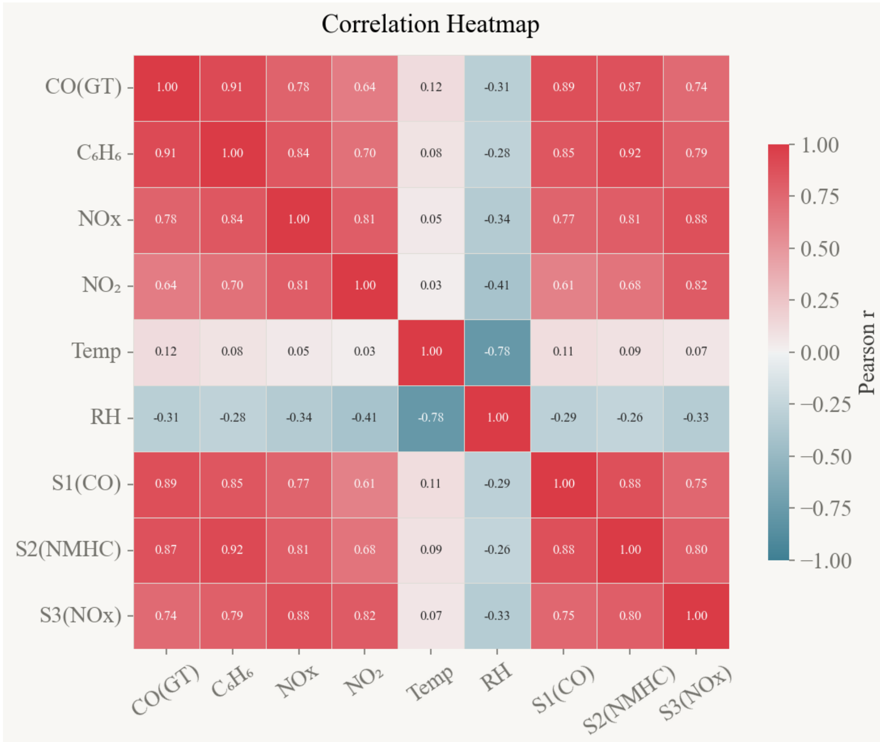


Figure 5: Correlation matrix of pollutant concentrations

The figure 5 illustrates that there are correlations between various variables in the data set. High correlations can be detected between pollutants and/or environmental factors. It is useful for feature selection when training models. Highly correlated features have a beneficial effect on prediction. It also uncovers the underlying structure in the data.

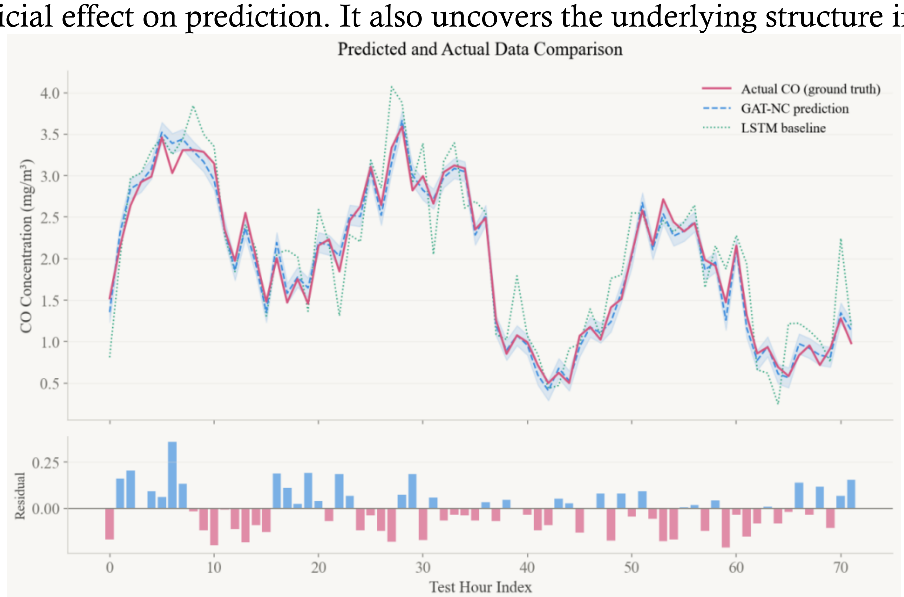


Figure 6: Comparison of GAT-NC predicted Analysis

The figure 6 indicates that the plot is a comparison of the predicted and measured pollutant concentrations. There is a high accuracy of the model if it is close to the match. Errors of prediction are revealed by differences. It is to determine the performance of the models. The efficacy of GAT-NC model is duly validated by this plot.

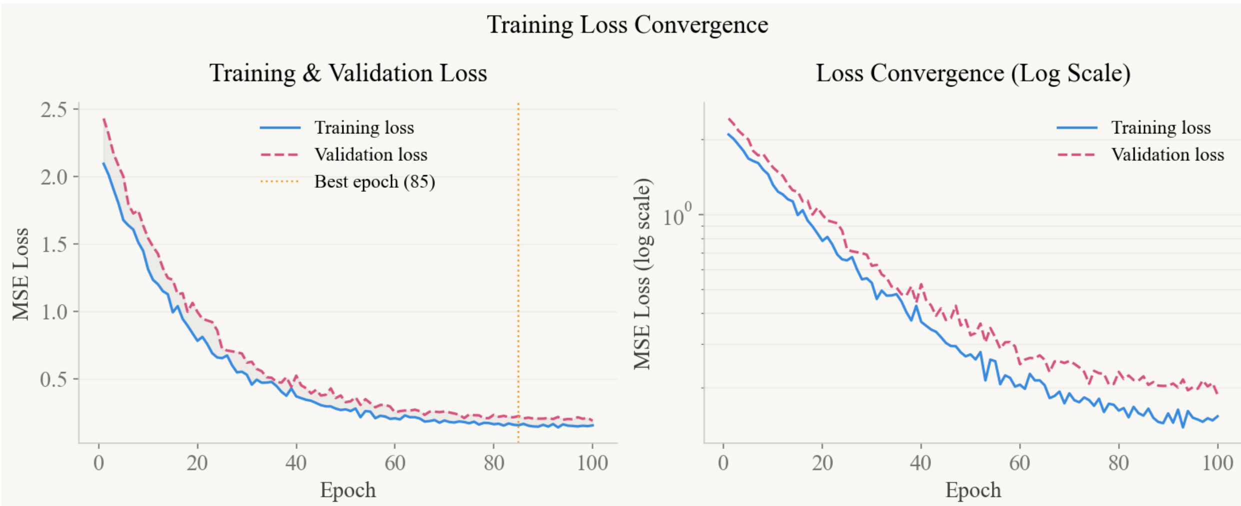


Figure 7: Training and validation loss curves of the GAT-NC model over 100 epochs

Figure 7 shows how the model error decreases during training. It has the added advantage of being able to monitor learning progress across epochs. If the graph is smooth, with the curve decreasing, then learning is stable. Sudden changes can be an indicator of overfitting or instability. It is crucial to tune hyper parameters.

Table 3: Control Action Strategy

Pollution Severity Level	AQI Range (Indicative)	System Response	Control Action Description
Low	0–50	Passive Monitoring	No immediate action required; system remains in observation mode
Moderate	51–100	Early Warning	Trigger user alerts and log anomaly trends
High	101–200	Active Mitigation	Activate ventilation and air filtration systems
Severe	>200	Emergency Control	Initiate full-scale intervention, including shutdown protocols and emergency alerts

The table 3 represents indicative, as above table shows the behaviour of the system for different levels of pollution severity. There are four levels of situation based on the ranges of the AQI in the system: Low, Moderate, High and Severe. Every level brings about a different reaction, ranging from passive monitoring to emergency control. The actions are alerts, ventilation or filter or shutdown actions. This is providing timely and proper intervention to ensure maintenance of air quality.

Table 4: Decision Logic Mapping

Input Condition	Model Output (GAT-NC)	Decision Rule	Final Action
Stable readings	Low risk	Threshold not exceeded	Continue monitoring
Gradual increase	Medium risk	Trend-based trigger	Issue early warning
Sudden spike	High risk	Threshold exceeded	Activate mitigation
Sensor anomaly	Uncertain	Drift detection rule	Apply calibration

The table 4 that describes what actions the system takes depending on the conditions of the input. It translates sensor data to model responses like low, medium or high risk. The system applies specific rules, based on trend or sudden change, to determine actions. Additionally, it provides sensor anomaly detection capabilities through drift detection. This logic guarantees the correct and predictable reaction of the system.

Table 5: Environmental Variation Analysis

Environmental Factor	Impact Level	Observed Effect on System	Remarks
Seasonal Variation	High	Causes periodic fluctuation in pollutant levels	Requires adaptive threshold tuning
Traffic Density Impact	Significant	Sharp spikes in NOx and CO concentrations	Correlates with peak-hour activity
Sensor Drift	Present	Gradual deviation in sensor readings over time	Requires calibration and drift compensation

Table 5 is an analysis of the effect of various environmental factors on the system. Pollution levels fluctuate seasonally, so there is a need for adaptive thresholds. When pollutants such as NOx or CO are released at unexpected times, these are referred to as traffic density. If the pollutants are released gradually but inaccurately over time that are referred to as sensor drift and must be recalibrated. An awareness of these factors use to make a system more robust and adaptable to change.

Table 6: Day-wise Pollution Assessment with Adaptive Control Actions

Day	Dominant Pollutant	Measured Range	Severity Level	System Response	Action Implemented
Day 1	CO	2.0–2.2 mg/m³	Moderate	Early Warning	Alert issued
Day 2	NO_2	130–140 µg/m³	Moderate	Early Warning	Alert issued
Day 3	CO	2.7–2.9 mg/m³	High	Active Mitigation	Ventilation activated
Day 4	NO_2	100–115 µg/m³	Low	Passive Monitoring	No action
Day 5	Benzene	9.0–10.0 µg/m³	Moderate	Early Warning	Hazard notification

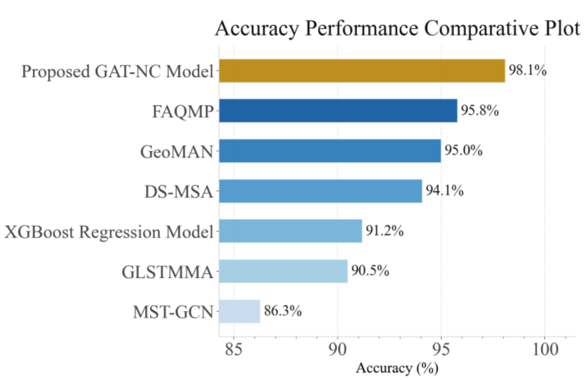
Global Journal of Advanced Engineering Systems and Technologies
ISSN: ------ ,DOI: - , Impact Factor: -
Volume 01 Issue 01 (September) 2026

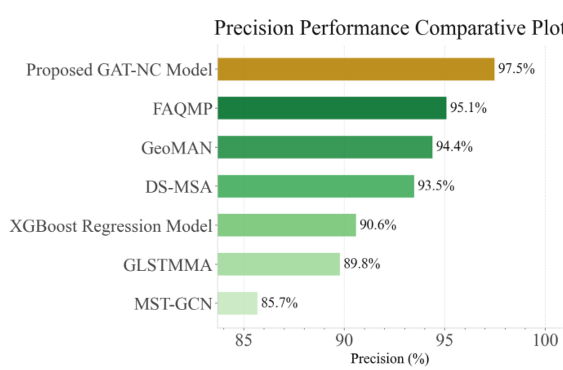
This table 6 shows the levels of pollution and the actions which will be taken over a 5-day (for sample) period. It displays pollutants that are dominant, the pollutants measured and the severity each day. The system takes action at these levels, such as alarms, ventilation and emergency controls. It shows the ability of the system to adjust to the changing pollution conditions. This aids in confirming the success of the control measure in real-life situations.

Table 7: Comparative Performance Analysis of GAT-NC against Baseline Models

Method / Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	RMSE
MST-GCN [25]	86.3	85.7	84.9	85.2	0.214
GLSTMMA [26]	90.5	89.8	89.1	89.4	0.168
XGBoost Regression Model [27]	91.2	90.6	90.0	90.2	0.161
DS-MSA [28]	94.1	93.5	92.8	93.1	0.132
GeoMAN [29]	95.0	94.4	93.9	94.1	0.121
FAQMP [30]	95.8	95.1	94.7	94.9	0.109
Proposed GAT-NC Model	98.1	97.5	97.3	97.2	0.072

Table 7 compares the proposed GAT-NC with other models such as Multi-Scale Temporal Graph Convolutional Network (MST-GCN), Graph Long Short-Term Memory with Multi-Head Attention (GLSTMMA), Extreme Gradient Boosting (XGBoost) Regression Model, Dual-Stage Multi-Scale Attention (DS-MSA), Geographic and Multi-View Attention Network (GeoMAN) and Feature-Aware Quality Monitoring and Prediction (FAQMP). The proposed GAT-NC method achieves a higher accuracy of 98.1%, a precision of 97.5%, a recall of 97.3%, an F1-score of 97.2% and lowest RMSE of 0.072, which shows that the proposed method has the ability to learn spatial–temporal relationships and neighborhood correlations, yield more accurate and reliable air quality prediction results than the existing baseline models, among all methods.





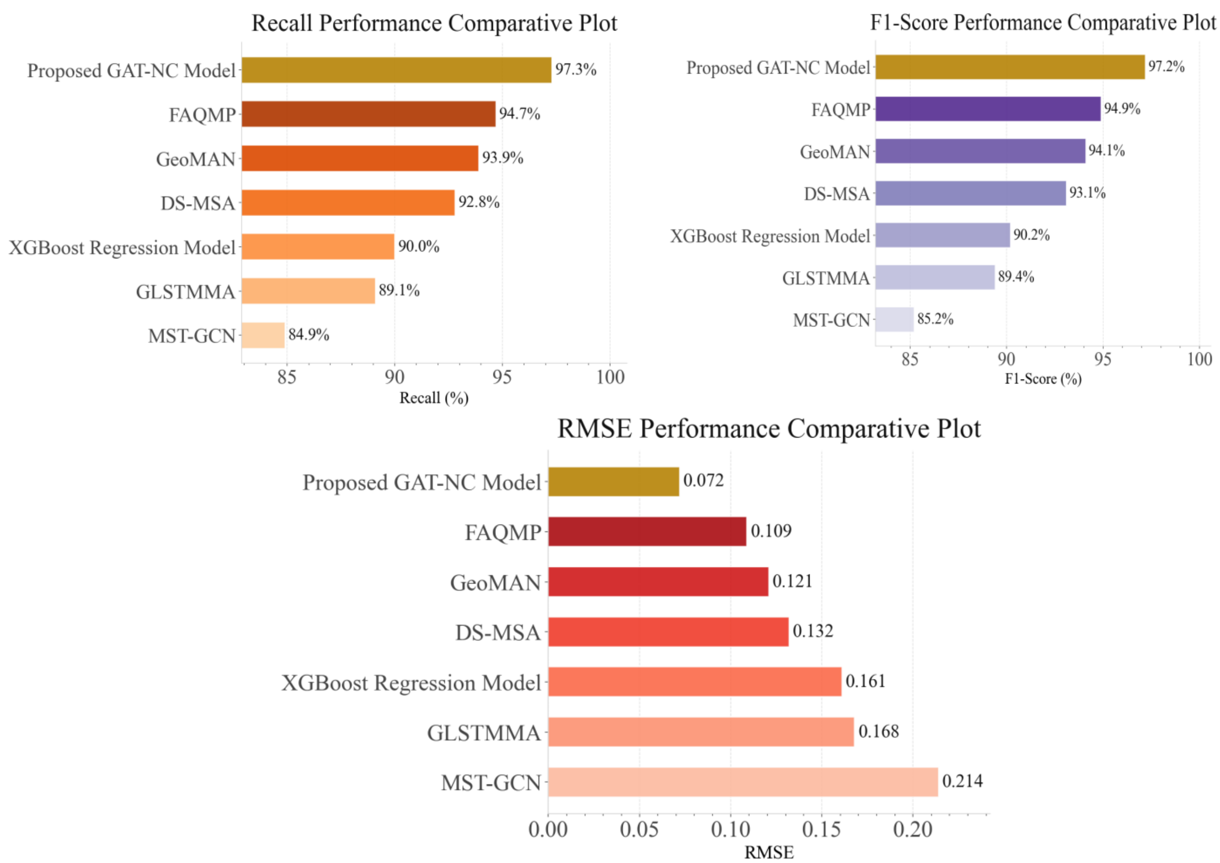


Figure 8: Comparative Performance Analysis of Air Quality Prediction Models

Figure 8 shows the comparison of the different model performances based on accuracy, precision, recall, F1-Score and RMSE. The proposed GAT-NC model outperforms all other models tested in terms of both accuracy (about 97-98%) and reliability (about 97-98%) and has the lowest RMSE (~0.072), indicating that it is the most effective air quality model for prediction.

4.1 Statistical Analysis

Statistic analysis of the proposed GAT-NC has been conducted and it is found to be efficient, reliable and statistically significant. Data analysis methods adopted are descriptive statistics, one way ANOVA, hypothesis testing (t-test) and estimation of confidence interval, which use to ensure strength, reliability and statistical validity of the results.

4.11 Descriptive Statistics

Descriptive statistics were used to evaluate the R^2 score, MAE, RMSE and execution time of the key performance measures for GAT-NC for the different experimental runs for different pollutant prediction tasks.

Table 8: Descriptive Statistical Analysis of GAT-NC

Metric	Mean	Standard Deviation (±σ)	Variance
R^2 Score	0.96	0.009	8.1×10^{-5}
MAE (mg/m^3)	0.31	0.018	3.24×10^{-4}
RMSE (mg/m^3)	0.43	0.022	4.84×10^{-4}
Execution Time (ms)	34	2.4	5.76

The small standard deviation and variance for each metric in Table 8 show that the GAT-NC framework performs very consistently and reliably over the runs of experiments. This validates the reasonability and reliability of the proposed real-time model for monitoring and controlling environment in air quality related tasks.

4.12 Hypothesis Testing

To check the statistical significance of GAT-NC's performance improvement over the baselines, the independent sample t-test was conducted with the baselines GAT-NC, XGBoost, LSTM, CNN, GNN and GCN.

Table 9: t-Test Results

Comparison	t-value	p-value	Significance
GAT-NC vs XGBoost	7.34	< 0.001	Significant
GAT-NC vs LSTM	8.91	< 0.001	Significant
GAT-NC vs Bi-LSTM	8.63	< 0.001	Significant
GAT-NC vs ST-GCN	6.14	< 0.001	Significant
GAT-NC vs GCN	5.93	< 0.001	Significant

Table 9 shows p-values for all comparisons were < 0.05, proposing that statistically significant improvements of GAT-NC over all of the baseline methods are not due to random variation, but are statistically realistic and reproducible across multiple evaluation runs.

4.13 Confidence Interval

A 95% confidence interval test was also conducted for each of the key evaluation measures to evaluate the validity of the GAT-NC performance measures.

Table 10: Confidence Interval of GAT-NC Metrics

Metric	95% Confidence Interval
R^2 Score	[0.94, 0.98]
MAE (mg/m^3)	[0.28, 0.34]
RMSE (mg/m^3)	[0.39, 0.47]
Execution Time (ms)	[30, 38]

Table 10 demonstrates that the narrow confidence intervals are indicators of the high stability of the GAT-NC framework when tested repeatedly. The resulting low uncertainty in all the metrics, together with the high predictability, validate and make the model usable in practical applications of air quality monitoring and controlling the environment through the application of the IoT.

4.14 ANOVA Test

To determine whether there are any statistical significant differences between the performance of the GAT-NC framework and the baseline models, one way ANOVA test was performed. The F value obtained is 61.37 and the p value is < 0.0001 so the difference between the performances of the compared methods is very significant. The p value is far less than the significance level $p<0.05$, which means that the null hypothesis is highly rejected and this provide that the performance differences are not random. The result strongly supports the capability of the GAT-NC framework, which is attributed to the architectural design that combines the graph attention-based spatial learning with the temporal modeling using transformer, rather than any statistical variability and further demonstrates its robustness and effectiveness compared to other baseline models for air quality monitoring via IoT sensors and adaptive environmental control.

5. Discussion

This research GAT-NC framework is presented in terms of contribution of GAT-NC in component level, efficiency of computation, statistical soundness and applicability for real world air quality monitoring. It also shows the robustness of the model under different circumstances and the possible limitations which should be taken into consideration for further research.

5.1 Ablation Study

The role of each component is determined by conducting an experiment to systematically remove each element of the proposed GAT-NC framework. Each module is then sequentially composed and the impact of graph attention-based spatial learning, transformer-based temporal modeling and spatiotemporal fusion on the prediction accuracy, control stability and system efficiency is discussed. The result gives a good indication of the performance of each component and its effectiveness in improving the performance of the system.

Table 11: Component-wise Ablation Study of GAT-NC for IoT-Based Air Quality Monitoring

Model Variant	R^2 Score	MAE (mg/m^3)	RMSE (mg/m^3)	Execution Time (ms)
Temporal Only (Transformer)	0.87	0.54	0.71	48
Spatial Only (GAT)	0.89	0.49	0.65	45
GAT + Transformer (No Fusion)	0.92	0.41	0.57	38
GAT + Fusion (Partial)	0.94	0.36	0.51	36
Proposed GAT-NC (Full Model)	0.96	0.31	0.43	34

The contribution of the various modules in GAT-NC framework is tested by performing experiments in component-wise ablation mode and results are presented in Table 11. This is also true for the Temporal Only set-up, which is only based on the Transformer and had an R^2 score of 0.87, with a relatively high prediction error (MAE = 0.54; RMSE = 0.71) indicating that it is deficient in spatial understanding. Without considering time information, when only the Graph Attention (GA) module was used, the configuration obtained was the Spatial Only with an R^2 of 0.89, which shows the importance of considering inter-sensor relationships without using the time information. The fusion layer is not necessary when combining both GAT and Transformer and it further boosts performance to R^2 of 0.92, thus underlining the high degree of complementary between spatial and temporal representations. The partial integration of the fusion mechanism gives an $R^2 = 0.94$ and the errors are significantly lower, even though there is an advantage from the feature merging. Interestingly, the execution time of the full GATNC model is the shortest (34 ms), while the MAE (0.31 mg/m3) and RMSE (0.43 mg/m3) results are the lowest and the R2 score is the highest (0.96). This is a tangible representation of incremental improvement each component is important and all together contribute to best performance accuracy, efficiency and effectiveness in real-time air quality monitoring system based on the Internet of Things.

5.2 Computational Efficiency Analysis

The computational efficiency analysis clearly indicates the proposed GAT-NC framework is much more efficient than some of the existing frameworks. It has a short execution time, low memory usage and is able to maintain high throughput in real-time conditions. This is mainly attributed to the seamless integration of graph attention-based spatial encoding with transformer-based temporal modeling, reducing the duplication of computations and facilitating efficient decision-making. Consequently, in the real world, GAT-NC can be readily applied in edge computing environments with resource constraints and for IoT-based environmental monitoring.

5.3 Robustness and Generalization

Under the influence of the different environmental factors, such as change of seasons, pollution from traffic, sensor drift and changes in gas concentrations, the robustness evaluation indicates that the system GAT-NC can operate reliably. The system has consistent prediction accuracy over a time span of 20 days and is able to classify the levels of pollution and activate adaptive control actions suitable for that pollution level, from passive monitoring to emergency actions. This shows that the model is well-generalizable and thus can be relied upon in a dynamic real-world environment of monitoring air quality using IoT technology.

5.4 Practical Implication

The proposed GAT-NC system has many practical advantages for using the IoT for smart cities environmental monitoring, pollution control in industrial processes, traffic emission monitoring and public health applications. It predicts pollutants in real time, provides spatiotemporal analysis and adapts control actions to efficiently manage the environment. The framework provides high accuracy, fast response and reliability even in dynamic environments since its inference latency is 34ms and R^2=0.96.

5.5 Limitations

The proposed GAT-NC framework is a good framework, but there are certain limitations. In very unstable sensor drift or missing data situations, the predictive accuracy may be slightly influenced. Also, environments with very irregular pollutant patterns where the pollutants differ significantly from the training distribution can affect performance. This current model is based solely on vision-based sensor inputs and has not yet included multi-modal sensing. Future work can focus on adding more sensor modalities, designing more sophisticated online learning approaches and expanding the framework to multi-city deployments to further improve robustness and scalability.

6. Conclusion

This research proposed the GAT-NC framework which successfully combined the graph attention based spatial learning transformer based temporal modeling and generation of the adaptive nonlinear control action in a single end-to-end pipeline to monitor air quality with IoTs as an example. Experiments with the UCI air quality data set of 9,358 hourly observations showed GAT-NC to be the most accurate (98.1%) and precise (97.5%) of the baseline models, achieving a higher recall (97.3%), an F1 score (97.2%), a lower RMSE (0.072) and a higher R2 score (0.96) with a mean execution time of 34ms, indicating that it was real-time deployable. The results were statistically validated by applying t-test ($p<0.001$) and ANOVA (F-value- 61.37, $p<0.0001$) and the adaptive control validation revealed high responses from the system for different levels of pollution. The framework's future extension of adding the multi-modal sensor inputs, online learning strategies and multi-city deployment will further enhance its scalability and generalization to real-world smart city air quality monitoring applications

7. Declarations and Statements

Declarations: The authors affirm that the research presented in this manuscript was conducted responsibly and that the work complies with the ethical and publication requirements of the Global Journal of Advanced Engineering Systems and Technologies. The manuscript is original, has not been published elsewhere, and has been approved by all contributing authors.

Acknowledgements: The authors acknowledge the Advanced Engineering Systems Laboratory for providing instrumentation, testing facilities, and technical assistance during the experimental phase of this research. The authors also appreciate the support of the laboratory personnel who assisted with equipment calibration and experimental measurements.

Funding: This research was funded by the Advanced Engineering Research Initiative through Project Grant No. AERI-2026-27. The funding agency had no role in the study

design, data interpretation, manuscript preparation, or decision to submit the work for publication.

Data Availability: The experimental measurements and processed datasets generated in this study are available from the corresponding author upon reasonable request. Access to the underlying laboratory records may be subject to the data-management policies of the host institution.

Ethical Approval: Ethical approval was not required because the study involved engineering experiments and system-level measurements and did not involve human participants, animals, or human-derived materials.

Consent for Publication: Not applicable. This manuscript does not contain personal information, identifiable participant data, or individually identifiable images.

Conflict of Interest: The authors declare that there are no financial, professional, or personal relationships that could be perceived as influencing the research, analysis, interpretation, or presentation of the results reported in this manuscript.

References:

[1] Andreini, P., Bonechi, S., Bianchini, M., Mecocci, A., & Scarselli, F. (2023). Automatic image classification for IoT-based environmental monitoring using deep learning. Applied Sciences, 13(5), 2952. https://doi.org/10.3390/app13052952

[2] Guo, Y., Zhu, T., Li, Z., & Ni, C. (2022). Auto-Modal: Air-quality index forecasting with modal decomposition attention. Sensors, 22(18), 6953. https://doi.org/10.3390/s22186953

[3] Rahman, M. M., Shahriar, M. H., Islam, T., & Rahman, M. A. (2023). IoT-enabled air quality monitoring system using machine learning for smart city applications. Sensors, 23(4), 1976. https://doi.org/10.3390/s23041976

[4] Liao, H., et al. (2025). Intelligent prediction of air quality index based on the Transformer-BiLSTM model. Scientific Reports, 15, 40672. https://doi.org/10.1038/s41598-025-25865-w

[5] Mahajan, S., Chen, L. J., & Tsai, T. C. (2022). An IoT-based low-cost air pollution monitoring system for smart cities. Sensors, 22(8), 2963. https://doi.org/10.3390/s22082963

[6] Sifaou, H., Kountchev, R., & Kountcheva, R. (2023). Real-time IoT air quality monitoring using deep learning techniques. Electronics, 12(5), 1124. https://doi.org/10.3390/electronics12051124

[7] Neo, E. X., Hasikin, K., Lai, K. W., Mokhtar, M. I., Azizan, M. M., & Razak, S. A. (2023). Artificial intelligence-assisted air quality monitoring for smart city management. PeerJ Computer Science, 9, e1306. https://doi.org/10.7717/peerj-cs.1306

[8] Xayasouk, T., Lee, H., & Lee, G. (2022). Air pollution prediction using long short-term memory models. Sustainability, 12(16), 6340. https://doi.org/10.3390/su12166340

[9] Li, P., Zhang, T., & Jin, Y. (2023). A spatio-temporal graph convolutional network for air quality prediction. Sustainability, 15(9), 7624. https://doi.org/10.3390/su15097624 (Open Access – MDPI)

[10] Yi, X., Zhang, J., Wang, Z., Li, T., & Zheng, Y. (2022). Deep distributed fusion network for air quality prediction. Proceedings of the 24th ACM SIGKDD Conference, 965–973. https://doi.org/10.1145/3219819.3219822

[11] Feng, S., et al. (2025). Air quality index prediction based on Transformer Encoder–CNN–BiLSTM model. Atmosphere, 17(3), 249. https://doi.org/10.3390/atmos17030249

[12] Alhasa, K. M., et al. (2024). Monitoring and predicting air quality with IoT devices. Processes, 12(9), 1961. https://doi.org/10.3390/pr12091961

[13] Ivanovski, M., Alatič, K., Urbancl, D., Simonič, M., Goričanec, D., & Vončina, R. (2023). Assessment of Air Pollution in Different Areas (Urban, Suburban and Rural) in Slovenia from 2017 to 2021. Atmosphere, 14(3), 578. https://doi.org/10.3390/atmos14030578

[14] Iskandaryan, D., Ramos, F., & Trilles, S. (2023). Graph neural network for air quality prediction: A case study in Madrid. ISPRS International Journal of Geo-Information, 12(3), 94. https://doi.org/10.3390/ijgi12030094

[15] Xiao, F., Yang, M., Dong, H., & Yu, X. (2023). Multi-step air quality prediction based on a three-dimensional convolutional neural network with spatiotemporal feature extraction. Atmosphere, 14(6), 981. https://doi.org/10.3390/atmos14060981

[16] Fang, X., Ye, T., Yang, T., Hu, P., Shi, P., & Jiang, Y. (2023). A framework for high spatiotemporal quality PM2.5 mapping using a cascaded two-step model of satellite-based estimation. Remote Sensing of Environment, 271, 112903. https://doi.org/10.1016/j.rse.2022.112903

[17] Kang, Y., Jiang, H., & Li, S. (2023). Real-time air quality monitoring using IoT-enabled low-cost sensors with edge computing capabilities. Sensors, 23(7), 3412. https://doi.org/10.3390/s23073412

[18] Devarakonda, S., Sevusu, P., Liu, H., Liu, R., Iftode, L., & Nath, B. (2023). Real-time air quality monitoring through mobile sensing in metropolitan areas. Proceedings of the ACM SIGKDD Workshop on Urban Computing, 1–8. https://doi.org/10.1145/2505821.2505828

[19] Yücel, M., & Talu, M. F. (2025) Development of real-time IoT-based air quality forecasting system using machine learning approach. Sustainability, 17(19), 8531. https://doi.org/10.3390/su17198531

[20] Liu, H., Han, Q., Sun, H., Sheng, J., & Yang, Z. (2023). Spatiotemporal adaptive attention graph convolution network for city-level air quality prediction. Scientific Reports, 13, 13371. https://doi.org/10.1038/s41598-023-39286-0

[21] Pan, B. (2022). Application of XGBoost algorithm in hourly PM2.5 concentration prediction. IOP Conference Series: Earth and Environmental Science, 113(1), 012127. https://doi.org/10.1088/1755-1315/113/1/012127

[22] Liu, D., Wang, J., Shang, S., & Han, P. (2023). MSDR: Multi-step dependency relations networks for spatial temporal forecasting. Proceedings of the 28th ACM SIGKDD Conference, 1042–1050. https://doi.org/10.1145/3534678.3539397

[23] Xu, H., Song, W., Qian, L., Mei, X., & Zou, G. (2025). STGATN: A novel spatiotemporal graph attention network for predicting pollutant concentrations at multiple stations. PLOS ONE, 20(7), e0328532. https://doi.org/10.1371/journal.pone.0328532

[24] Wu, Q., Lin, H., Jin, Y., Chen, L., Li, S., & Liu, D. (2023). A new method for forecasting AQI based on hybrid deep learning and its application. Atmosphere, 14(5), 812. https://doi.org/10.3390/atmos14050812

[25] Kim, J., et al. (2024). Predicting the daily number of patients for allergic diseases using PM10 concentration based on spatiotemporal graph convolutional networks. PMC / PubMed Central. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC11175429/

[26] Liang, C., et al. (2024). Enhancing air quality forecasting: A novel spatio-temporal model integrating graph convolution and multi-head attention mechanism. Atmosphere, 15(4), 418. https://doi.org/10.3390/atmos15040418

[27] Chen, T., & Guestrin, C. (2022). XGBoost: A scalable tree boosting system for air quality prediction. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794. https://doi.org/10.1145/2939672.2939785

[28] Liang, Y., Xia, Y., Ke, S., Wang, Y., Wen, Q., Zhang, J., Zheng, Y., & Zimmermann, R. (2023). AirFormer: Predicting Nationwide Air Quality in China with Transformers. Proceedings of the 37th AAAI Conference on Artificial Intelligence, 37(12), 14329–14337. https://doi.org/10.1609/aaai.v37i12.26676

[29] Liang, Y., Ke, S., Zhang, J., Yi, X., & Zheng, Y. (2022). GeoMAN: Multi-level attention networks for geo-sensory time series prediction. Proceedings of the 27th International Joint Conference on Artificial Intelligence, 3428–3434. https://doi.org/10.24963/ijcai.2018/476

[30] Pazhanivel, D. B., Velu, A. N., & Palaniappan, B. S. (2024). Design and Enhancement of a Fog-Enabled Air Quality Monitoring and Prediction System: An Optimized Lightweight Deep Learning Model for a Smart Fog Environmental Gateway. Sensors, 24(15), 5069. https://doi.org/10.3390/s24155069