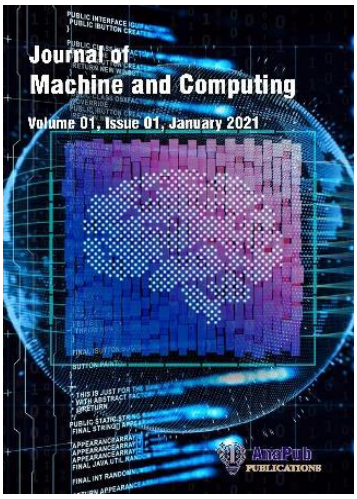
Journal Pre-proof

Applying Artificial Intelligence and the Internet of Things to the Building Sector to Improve Security and Disaster Prediction



Nanthini N, Swati Saxena, Sampathirao Suneetha, Sabaresan V, Ananda Babu T and Prakalya S B

DOI: 10.53759/7669/jmc202505087

Reference: JMC202505087

Journal: Journal of Machine and Computing.

Received 12 April 2024

Revised form 11 November 2024

Accepted 09 March 2025

Please cite this article as: Nanthini N, Swati Saxena, Sampathirao Suneetha, Sabaresan V, Ananda Babu T and Prakalya S B, “Applying Artificial Intelligence and the Internet of Things to the Building Sector to Improve Security and Disaster Prediction”, Journal of Machine and Computing. (2025). Doi: https://doi.org/10.53759/7669/jmc202505087

AnaPub
PUBLICATIONS

Applying Artificial Intelligence and the Internet of Things to the Building Sector to Improve Security and Disaster Prediction

N. Nanthini[1,*], Swati Saxena[2], Sampathirao Suneetha[3], V. Sabaresan[4], T. Ananda Babu[5], S. B. Prakalya[6]

[1]Department of Computer Science and Engineering, Sathyabama Institute of Science and Technology, Chennai, Tamil Nadu 600119, India. **** Corresponding Author Email: nanthini0294@gmail.com***

[2]Department of Computer Science and Engineering, KCG College of Technology, Chennai, Tamil Nadu, India. *Email: swati.cse@kcgcollege.com*

[3]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, 522502, India. *Email: sampath.suneetha@gmail.com*

[4]Department of Computer Science and Engineering, St. Joseph's Institute of Technology, Chennai, Tamil Nadu, 600119, India. Email: sabaresanvenugopal@gmail.com

[5]Department of Electronics and Communication Engineering, Seshadri Rao Gudlavalleru Engineering College, Gudlavalleru, Andhra Pradesh, 521356, India. *Email: anand.mits1@gmail.com*

[6]Department of Electronics and Communication Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, 600077, Tamil Nadu, India. *Email: prakalyasb.sse@saveetha.com*

Abstract

The Intelligent Infrastructure Monitoring System (IIMS) is a data-driven approach that uses Artificial Intelligence (AI) and the Internet of Things (IoT) to enhance civil engineering Disaster Management (DM). To predict future disasters, the system uses a Multi-Tiered Model (MTM) to integrate real-time data from IoT sensors, such as stress, vibration, temperature, humidity, and corrosion levels. The Temporal Graph Convolutional Network Model (TGCNM) processes this data to capture spatial and temporal dependencies across structural components, enabling proactive maintenance and risk mitigation. The TGCNM outperforms baseline models by a significant margin, and hyperparameter sensitivity analysis identifies the optimal configuration for enhanced performance. This data-driven approach is vital for monitoring and securing key infrastructure and enhancing civil engineering DM with AI and IoT.

Keywords: Intelligent Infrastructure Monitoring System, Temporal Graph Convolutional Network, Internet of Things, Artificial Intelligence, LSTM, Accuracy.

1. Introduction

As the probability of disasters caused by nature continues to increase and as vital infrastructure continues to fail, civil engineering professionals have come to understand the vital significance of cutting-edge predictive monitoring [1-2]. The infrastructure maintenance approaches frequently ignore unavoidable hazards, resulting in disastrous failures and implications [3]. Novel advances in Artificial Intelligence (AI) and the Internet of Things (IoT) have enabled it to monitor and evaluate infrastructure scenarios in real-time while simultaneously [4-6]. Data on

stress, vibration, temperature, humidity, and corrosion, among additional key parameters, can be collected via IoT sensor networks connected to buildings. AI can subsequently employ this data to predict when infrastructure will deteriorate [7-8]. This work employs a proactive model to enhance Disaster Management (DM) and infrastructure adaptability, significantly developing predictive maintenance [9–10].

The primary disadvantage of civil engineering persists in its inability to monitor reliable and periodic infrastructure [11]. Conventional methods find it particularly challenging to identify preliminary evidence of decomposition in highly intricate and large buildings [12]. This drawback impedes proper actions, contributing to increased hazards and maintenance costs. Primarily for buildings susceptible to maintenance and natural disasters, the demand for data-driven, real-time models has increased in the context of outdated infrastructure [13–14]. Architectural tests alone are insufficient to promote DM; research requires implementing a more complex model connecting IoT and AI to attain data in real-time and make change predictions [15–16].

Several researchers have studied methods such as Long Short-Term Memory (LSTM) and Autoregressive Integrated Moving Average (ARIMA) for infrastructure durability prediction and monitoring purposes in the past [17–19]. Although these models effectively identify differences over time, they frequently fall short regarding the complex spatial relationships found in massive infrastructures. Although Graph Convolutional Networks (GCN) are helpful for spatial modeling, they cannot handle time-dependent tasks discretely [20].

While the Spatial-Temporal Graph Convolutional Network (ST-GCN) can address spatial and temporal features, it might have trouble with long-term temporal relationships [21]. This is because it enhances the accuracy of predictions. Attention-based Temporal Graph Convolutional Networks (A3T-GCN) make it easier to select features, but they can make simulations difficult without constantly assessing predictions are more accurate [22]. This work must develop an improved model considering spatial and temporal dependencies to overcome these drawbacks and make accurate predictions for infrastructure resilience applications.

Accurately measuring Building Sectors (BS) is the primary objective of this paper's predictive monitoring system, which will do this by monitoring the temporal dynamics and real-time spatial dependencies among BS. This study has developed a model to address the shortcomings of modern methods in predictive DM, applying a model that combines AI and IoT for constant monitoring. The investigation revolves around developing a Temporal Graph Convolutional Network (T-GCN) that can continuously integrate infrastructure data points to provide accurate and timely predictions regarding the integrity of the infrastructure [23].

A multi-tiered Intelligent Infrastructure Monitoring System (IIMS) that employs IoT sensors for monitoring environmental variables like corrosion, stress, and vibration in real-time procedures is an element of the study objective [24]. Implementing Edge Computing (EC) to process these data flows reduces delay and enhances real-time decision-making. The key objectives of this research involve implementing the T-GCN on the buildings to recognize their spatial-temporal dependencies, verify their accuracy using standard models, and determine the optimal parameters for the proposed model [25]. This article applies IoT sensors and analytics to predict results, highlighting an IIMS that enables real-time problem-solving and BS testing. The suggested infrastructure comprises multiple layers: one for collecting data in real-time using sensors, one for preliminary analysis by an EC unit, possibly another for recent predictive modeling by an IoT server, and finally, a cloud gateway for archived data and processing. Graph-based spatial analysis and recurrent neural networks (RNN) are used in the T-GCN to predict infrastructure health metrics in various settings accurately. The resulting model is essential to the system's predictive performance.

Firstly, below are the key objectives of this study:

(a) The objective is to develop an IIMS that relies on the IoT and has the potential for perpetual data collection and real-time infrastructure condition monitoring.

(b) This study will deploy the T-GCN to perform accurate spatial-temporal analysis. This proof required the following data to accurately predict the adaptability of infrastructures under various environmental factors and functioning states.

(c) The research aims to compare the outcomes of the T-GCN to other models to determine whether there are any developments in prediction accuracy and computational performance.

(d) In order to maximize the performance of the model, it is essential to execute a hyperparameter sensitivity analysis. This study should confirm the accuracy and efficiency optimization of the Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE) performance metrics.

Here is the outline of the article: Section II lays out the recommended IIMS, separated data collection, analysis, and storage functions of each layer. In Section III, the researchers determine the basics of the T-GCN and how it integrates into the IIMS. The researcher also found out about its spatial-temporal modeling features. Section 4 provides details of the experiment, including dataset information, baseline models, evaluation metrics, and hyperparameter settings. The results and analysis will be presented in Section 5. This section will compare how well T-GCN predicts

with standard models, check the effect of spatial and temporal layers, and rate the model's sensitivity. This article reaches its conclusion in Section 6.

2. Proposed Architecture

The primary objective of the envisaged IIMS is to permit real-time decision-making by building conditions, vibration levels, and environmental variables such as temperature and humidity. A practical application of electrical power, this model triggers sensors selectively. By regulating maintenance expenses and life expectancy, the mathematical model of the network optimizes the application of sensors. Important buildings may currently experience real-time risk assessment along with predictive maintenance proposals according to the IIMS, which takes responsibility for the cause of remote building monitoring. A secure gateway integrates several key elements, including communication channels, sensor networks (external and internal), an IoT server, and cloud storage. Intelligent systems application, management, network, and device-level monitoring and management are all components of this model. The building design of the proposed system is illustrated in Figure 1. Disaster prevention and response involving IoT sensors and Decision Support Systems (DSS) presents an original viewpoint on improving building adaptability, particularly within remote and highly susceptible areas.

This Multi-Tiered Model (MTM) is portrayed in Figure 1, and it features the following key elements:

1. **Sensor Interface Layer**: For the intent of monitoring building parameters such as stress, vibration, and environmental variables, a network of intelligent sensors that are embedded within the building's design.
2. **Data Processing Unit**: A tool that performs edge processing, including a Raspberry Pi Portenta H7, is responsible for collecting and preliminary processing sensor data in order to obtain real-time information.
3. **IoT Server with Predictive Analytics**: To assess the building design strength of a building and anticipate possible hazards, a central server that hosts complex neural network models must be installed.

   User Interface on Mobile Device: A mobile application for smartphones that keeps users up to date in real-time on the condition of the building and presents ideas that can be adopted.
5. **Cloud Gateway and Server**: A secure cloud-based setting for storing data for extended time, significant analytics, and combining DM on a larger scale.

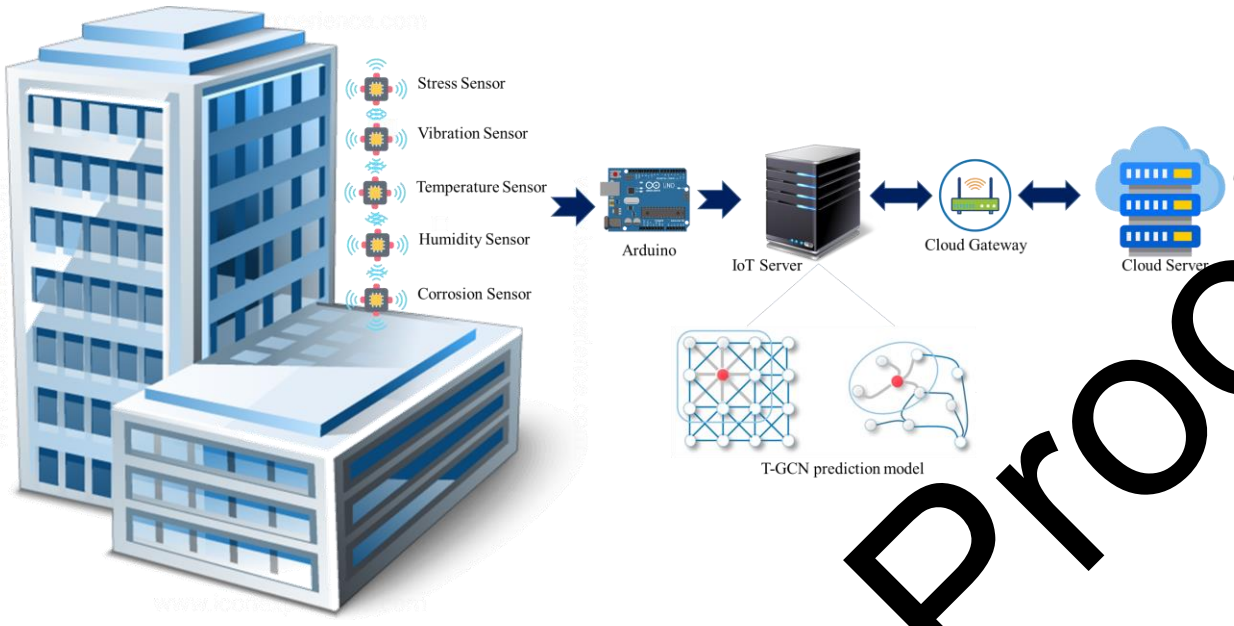


Figure 1: Proposed IoT Model

The initial layer of the architecture is known as the **Sensor Interface Layer**, and it contains a network of integrated smart sensors that actively collect building information. This data usually involves stress levels, vibrations, and factors related to the environment. This layer communicates real-time data from the buildings to the next tier through Bluetooth Low Energy (BLE) and Low Power Wide Area Network (LPWAN). An edge device, including the Arduino Portenta H7, assumes responsibility for the preprocessing and filtering of the data collected by the sensors within the **Data Processing Unit**. This device has been provided with wireless communication features, which make it possible to communicate with a central server by any of the following types of networks: Wi-Fi, 5G, and LPWAN. This third layer, the IoT Server with Predictive Analytics, collects data from a local database and correlates it with a MySQL database located in a cloud environment through encrypted channels. The data is evaluated by machine learning (ML) algorithms, which apply predictive models to assess the building design integrity of the building and determine any possible hazards. Integrating multiple sensors within this system enables it to collect an array of key data across several parameters, enhancing the accuracy of monitoring tasks. By providing immediate data collection and predictive analysis across an array of buildings, this multi-sensor methodology improves the system's applicability, presenting reliable data about the physical condition of the building.

This work aims to develop a framework that can detect and monitor building designs located remotely in real-time. This model will assist with approaches to durability and predictive maintenance.

A. **Stress Sensor**: New load sensors based on Micro-Electro-Mechanical-Systems (MEMS) provide decades of durability and accurate estimation of stress distribution across important building parts. This work was decided on these sensors because they perform accurately, are affordable, and are suitable for low-power device protocols like Bluetooth Low Energy (BLE), so we can keep track of things all day without connecting them. Their adaptability to environmental variations reduces the probability of inaccurate readings.

B. **Vibration Sensor**: Applying the ADXL1002 high-sensitivity MEMS acceleration sensors, data from vibration may be collected and remotely transmitted. The type of sensor was selected because the device is simple to install, portable, and has a long functioning lifetime. If it senses anomalous vibrations in the building, which might result from physical use, it will capture the vibrations and send them remotely to a central system for testing.

C. **Temperature Sensor**: The Sensirion SHT40 is a Bluetooth-enabled sensor that determines the outside temperature of the building's exterior. Important information about the long-term impacts of heat stress on building materials can be obtained by the real-time wireless data provided by the sensor and the computation device.

D. **Humidity Sensor**: Humidity is a further measurement that the Sensirion SHT40 measures since variations in this metric can impact the building design integrity of materials such as steel and concrete. Monitoring environmental variables that could impact building design integrity is optimal by low-noise signal processing and amplifying processes, which provide high precision.

E. **Corrosion Sensor**: New ultrasonic sound corrosion sensors maintain track of the level of corrosion, particularly on accessible steel buildings. These sensors may determine the layer thickness of materials and detect corrosion or degradation in early phases using ultrasonic sound waves. Predictive maintenance DSS is made feasible by sending data from the sensor to the central unit over an I2C connection.

F. **Arduino Portenta H7**: Several versions, such as the Arduino Due, Arduino Mega, Arduino Leonardo, and Arduino Portenta H7, have been released for use with the Arduino microcontroller chips. The Portenta H7, selected for this particular system, is suitable for processing real-time data from multiple sensors due to its robust processing speed and dual-core microcontrollers. This device is suitable with 5G, Wi-Fi, and LPWAN manufacturing communication protocols and features a range of input/output (I/O) based DSS, including analog and electronic inputs. Considering an emphasis on excellent performance edge computing, the Portenta H7 can preprocess data locally and significantly reduce delay,

developing differentiates as opposed to previous versions. As the primary controller of the monitoring system, the Portenta H7 interfaces seamlessly with various sensors and serves as a bridge between the sensors and the IoT. This open-source microcontroller is programmable using the Arduino IDE and is critical in integrating real-time sensor data into the more significant IoT infrastructure for predictive DM.

The sensor data collected by the IIMS provides the foundation for computing key functional parameters that support evaluating BS durability and predicting potential failures.

Key Measurements Include:

a) Computation of Structural Stress and Load Distribution: Structural stress (σ) across load-bearing components is estimated using data from strain and load sensors. The stress is typically calculated using Hooke's Law, assuming linear elastic behavior, Eq. (1)

$$\sigma = E \cdot \epsilon \quad (1)$$

where:

- $\sigma =$ Stress in the material (Pa),
- $E =$ Modulus of elasticity of the material (Pa),
- $\epsilon =$ Strain, obtained from strain gauge sensors.

Load distribution (F) is calculated based on the force (F) exerted across cross-sectional areas (A), providing insight into potential overloads or irregular stress concentrations, Eq. (2)

$$\sigma = \frac{F}{A} \quad (2)$$

b) **Measurement of Vibration and Dynamic Response**: The BS's vibration response ($a(t)$) is measured using accelerometers to detect shifts in dynamic behavior due to environmental forces like wind, traffic, or seismic events.

The natural frequency (f_n) of a building design component can be calculated using the Eq. (3):

$$f_n = \frac{1}{2\pi}\sqrt{\frac{k}{m}} \quad (3)$$

where:

- $f_n =$ Natural frequency (Hz),
- $k =$ Stiffness of the building design (N/m),
- $m =$ Mass of the building design (kg).

Real-time vibration data $a(t)$ from accelerometers can also detect unusual vibrations that may indicate damage or deterioration, Eq. (4).

$$a(t) = A \cdot \sin(2\pi f_n t + \phi) \quad (4)$$

where:

- $a(t) =$ Acceleration at time t,
- $A =$ Amplitude of vibration,
- $\phi =$ Phase angle.

c) Environmental Monitoring (Temperature and Humidity): Temperature (T) and humidity (H) levels are key factors influencing material properties. Sensors record these values continuously to monitor potential impacts on building design materials. Material expansion or contraction due to temperature can be estimated with the coefficient of thermal expansion (α) as follows, Eq. (5)

$$\Delta L = L_0 \cdot \alpha \cdot \Delta T \quad (5)$$

where:

- $\Delta L =$ Change in length
- $L_0 =$ Original length
- $\alpha =$ Coefficient of thermal expansion $(^{\circ}C^{-1})$
- $\Delta T =$ Change in temperature

d) Corrosion and Material Degradation Measurement: Corrosion progression (D) is monitored through ultrasonic measurements which assess the thickness (T) of steel components over time.

The corrosion rate (R) can be estimated as Eq. (6)

$$R = \frac{\Delta T}{t} \quad (6)$$

where:

- $R =$ Corrosion rate (MM/Year),
- $\Delta T =$ Change in thickness (MM),
- $t =$ Time (Years).

This helps predict the lifespan of BS and determine when maintenance is required. Real-time corrosion data supports decisions on material preservation and repair timelines, enhancing building design resilience against environmental impacts.

3. Prediction Model for DM and Structural Resilience

In this study, infrastructure resilience prediction aims to assess the BS condition of critical infrastructure over a specified time using historical sensor data collected from various BS components. Building condition data encompasses load stress, vibration, environmental conditions (temperature, humidity), and corrosion levels. For illustration, this study uses load stress as an example of building condition data in the experiment section.

Definition 1: Infrastructure Network $\mathcal{G}$. The infrastructure is represented as an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, modeling the structural relationships among components. Each critical structural element is represented as a node, where $\mathcal{V} = \{s_1, s_2, \dots, s_M\}$ is the set of nodes, with M as the total number of nodes. $\mathcal{E}$ represents the edges that define relationships (such as load distribution or adjacency) between nodes. The adjacency matrix $\mathbf{A} \in \mathbb{R}^{M \times M}$ encodes these connections with elements where $\mathbf{A}_{ij} = 1$ if nodes i and j are connected, and $\mathbf{A}_{ij} = 0$ otherwise.

Definition 2: Feature Matrix $\mathbf{X}_{M \times Q}$. The building condition data across the infrastructure network is represented by a feature matrix $\mathbf{X} \in \mathbb{R}^{M \times Q}$, where Q denotes the number of features or the length of the historical data sequence. $\mathbf{X}_t \in \mathbb{R}^{M \times j}$ represents a specific structural feature (*e.g.*, load stress) at time j for each node. Additional features could include vibration data, environmental metrics, or corrosion indicators.

The problem of predicting Spatial-Temporal BS can thus be formulated as learning a mapping function $'\phi'$, which, based on the network topology $\mathcal{G}$ and feature matrix $\mathbf{X}$, predicts the structural condition metrics over the following T time steps as shown in Eq. (7):

$$[\mathbf{X}_{t+1}, \dots, \mathbf{X}_{t+T}] = \phi\big(\mathcal{G}, (\mathbf{X}_{t-n}, \dots, \mathbf{X}_{t-1}, \mathbf{X}_t)\big) \quad (7)$$

where n is the length of the historical data sequence and T is the forecast horizon.

3.1 T-GCN

The T-GCN is designed to capture spatial and temporal dependencies in complex datasets, making it highly suitable for infrastructure resilience prediction. The integration of GCN and Recurrent Neural Networks (RNN) allows them to accurately represent structural changes' sequential, time-dependent behavior and the spatial connections among correlated building blocks. A graph of data with nodes representing structural components and edges representing interactions between them is implemented by the GCN feature of the T-GCN to exploit the infrastructure network design. The GCN can understand its overall build concerning its spatial location by applying an adjacency matrix to collect data from all the nodes' interconnected neighborhoods' nodes.

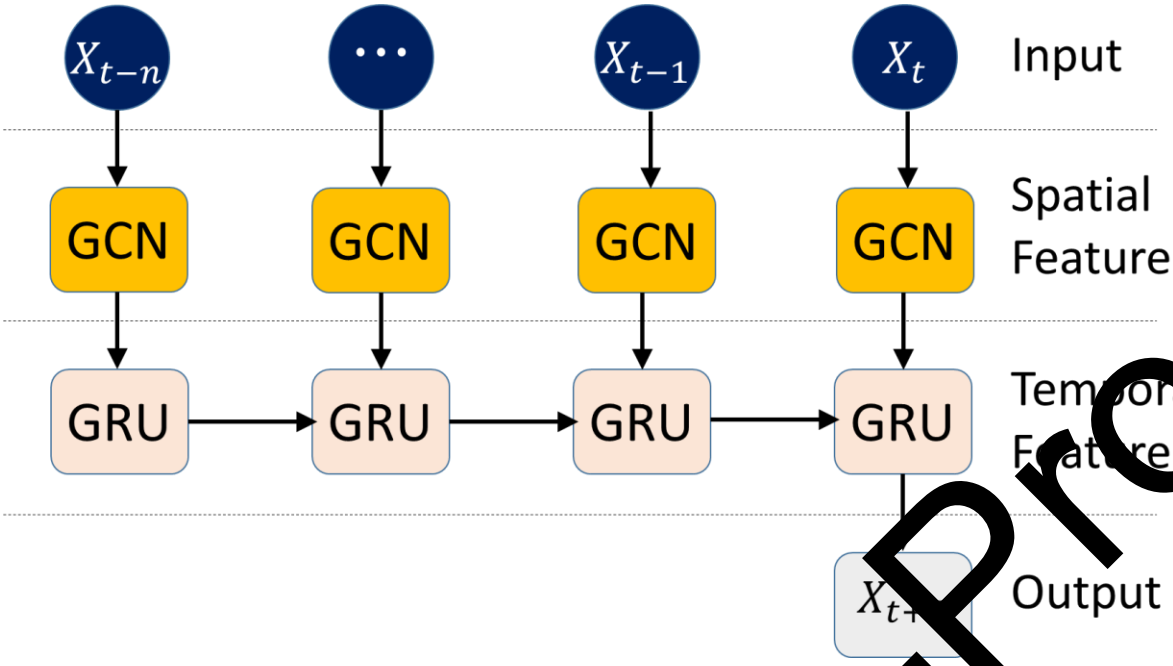


Figure 2: T-GCN Model

Data in the temporal sequence is analyzed by the RNN portion of the T-GCN, which typically functions using GRU (or) LSTM cells. In this method, the mathematical model could select any patterns or trends, including the gradual occurring building of stress or the impact of external factors, that can identify hazards. With these layers collectively, the T-GCN becomes an effective tool for monitoring building conditions in real-time, and it helps in proactive maintenance and durability testing via the use of previously collected data for predicting a structural issue.

(i) Spatial Dependence Modeling: The GCN segment of the T-GCN analyzes spatial reliance by identifying the structural connections between the physical network's different elements. The infrastructure is signified as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where every node $v_i \in \mathcal{V}$ matches complex building materials, and each edge $e_{ij} \in \mathcal{E}$ epitomizes a link between two components, such as a load-bearing connection. The spatial design of the network is fixed in an adjacency matrix $\mathbf{A} \in \mathbb{R}^{M \times M}$, where $\mathbf{A}_{ij} = 1$ if nodes i and j are linked and $\mathbf{A}_{ij} = 0$ else, with M instead of the complete nodes.

For the time $'t'$, the feature matrix $\mathbf{X}_t \in \mathbb{R}^{M \times Q}$ encompasses the features for all nodes, such as stress, vibration, and environmental factors, with Q as the number of features. The GCN layer then aggregates the data from all nodes' neighbors, rendering it to the adjacency matrix, permitting every node to integrate data about the structural environments of adjacent components.

The propagation rule for the GCN is defined as Eq. (8).

$$\mathbf{H}_t = \sigma\left(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{X}_t\mathbf{W}_g + \mathbf{b}_g\right) \quad (8)$$

where:

- $\mathbf{H}_t \in \mathbb{R}^{M \times F}$ is the matrix of node features after the GCN layer at time $'t'$,
- $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ is the adjacency matrix with added self-loops (where $\mathbf{I}$ is the identity matrix),
- $\tilde{\mathbf{D}}$ is the diagonal degree matrix of $\tilde{\mathbf{A}}$, with $\tilde{\mathbf{D}}_{ii} = \sum_j \tilde{\mathbf{A}}_{ij}$,
- $\mathbf{W}_g$ is a trainable weight matrix for the GCN layer,
- $\mathbf{b}_g$ is the bias term,
- σ is an activation function, typically ReLU.

This normalization process (multiplying by $\tilde{\mathbf{D}}^{-\frac{1}{2}}$) ensures that feature contributions from connected nodes are scaled by their respective node degrees, which prevents high-degree nodes from disproportionately influencing their neighbors. By applying this GCN layer, the model learns the spatial dependencies between structural components, integrating contextual information that reflects the connectivity and dependencies within the infrastructure network. This spatial feature extraction is crucial for accurately assessing the resilience of individual components based on the overall structural network.

(ii) Temporal Dependence Modeling: The temporal dependence modeling in the T-GCN is handled by the RNN component, which captures the sequential patterns and trends in the structural data over time. In the T-GCN, the temporal dependencies of each node's feature sequence are modeled using GRU (or) LSTM units. These recurrent layers allow the model to retain information about past states and identify temporal patterns essential for predicting future structural conditions.

At each time step t, the GCN output $\mathbf{H}_t \in \mathbb{R}^{M \times F}$, which represents spatially processed features for each node, is passed as input to the GRU. The GRU methods this temporal sequence of graph features, changing its hidden state $\mathbf{Z}_t \in \mathbb{R}^{M \times F}$ based on the prior hidden state $\mathbf{Z}_{t-1}$ and the new state of GCN output $\mathbf{H}_t$. The GRU's updates are Eq. (9) and Eq. (12).

$$\mathbf{r}_t = \sigma(\mathbf{W}_r \mathbf{H}_t + \mathbf{U}_r \mathbf{Z}_{t-1} + \mathbf{b}_r) \quad (9)$$

$$\mathbf{z}_t = \sigma(\mathbf{W}_z \mathbf{H}_t + \mathbf{U}_z \mathbf{Z}_{t-1} + \mathbf{b}_z) \quad (10)$$

$$\tilde{\mathbf{Z}}_t = \tanh(\mathbf{W}_h \mathbf{H}_t + \mathbf{r}_t * (\mathbf{U}_h \mathbf{Z}_{t-1}) + \mathbf{b}_h) \quad (11)$$

$$\mathbf{Z}_t = (1 - \mathbf{z}_t) * \mathbf{Z}_{t-1} + \mathbf{z}_t * \tilde{\mathbf{Z}}_t \quad (12)$$

Where:

- $\mathbf{r}_t$ is the reset gate, controlling the impact of the previous hidden state,
- $\mathbf{z}_t$ is the update gate, determining the balance between the previous hidden state $\mathbf{Z}_{t-1}$ and the current input,
- $\tilde{\mathbf{Z}}_t$ is the candidate's hidden state at time t,

- $\mathbf{W}_r, \mathbf{W}_z, \mathbf{W}_{h,}\mathbf{U}_r, \mathbf{U}_z$, and $\mathbf{U}_h$ are trainable weight matrices,
- $\mathbf{b}_r, \mathbf{b}_z$, and $\mathbf{b}_h$ are biased terms,
- σ and tanh are the sigmoid and hyperbolic tangent activation functions.

A GRU continually analyzes the spatially analyzed features at each time step, maintaining key details from previous phases for storing data temporal relationships. By applying structural condition measures' temporal motion, the T-GCN makes it possible to predict impending structural challenges by combining past data with recent advancements. Proactive DM and maintenance planning use this temporal Feature Extraction (FE) to assess the changing state of BS.

(iii) Loss Function: The T-GCN has been trained to determine the strength of buildings using the MSE loss function. To compute the MSE for a prediction radius of T time steps, perform Eq. (13).

$$\mathcal{L}_{\text{MSE}} = \frac{1}{T}\sum_{t=1}^{T} (y_t - \hat{y}_t)^2 \quad (13)$$

where:

- y_t is the actual structural metric at the time t
- $\hat{y}_t$ is the predicted metric at time t,
- T is the forecast horizon.

Reducing this loss is essential to maximize the T-GCN model's durability prediction precision.

4. Experiment analysis

4.1 Data Description

A significant open-source dataset for investigating building durability, Structural Health Monitoring (SHM), is used in the current research. It was generated by the Los Alamos National Laboratory (LANL). Strain, temperature, and acceleration are essential variables collected in the time series provided in real-world sensor data relating to various building elements. The data is collected from sensors embedded in buildings that respond to multiple functioning and environmental conditions. This data contains valuable information for determining the durability of the building.

The SHM dataset includes:

- **Strain Measurements:** Strain sensors attached to load-bearing elements to monitor stress distribution were employed to collect the data.
- **Acceleration Data:** The data was collected using accelerometers to measure vibration and behavior under various loading conditions.
- **Temperature Data:** Thermal sensors have been employed to monitor the material to find out how variations in temperature impact the quality of the material being monitored.

A time-series model appropriate for spatial-temporal modelling can be attained by regular measurement logging of each sensor. Because of how this dataset is organized, the T-GCN can identify the spatial connections between several building elements and variations in that building's condition as time progresses. By implementing this data, the research study tests the T-GCN's prediction accuracy in predicting probable building problems.

4.2 Metrics

Researchers apply three standard regression metrics—MSE, MAE, and RMSE—to evaluate the T-GCN's predictive accuracy for predicting building durability. To determine how effectively the model performs, these data points to test the extent to which proposed actual building condition data are from predictions for the future.

1. **MSE:** MSE computes the average squared variance between predicted data $(\hat{y}_t)$ and real data (y_t), fining more important errors more severely, Eq. (14).

$$\text{MSE} = \frac{1}{T}\sum_{t=1}^{T} (y_t - \hat{y}_t)^2 \quad (14)$$

where:

- T is the total time steps in the test set.
- y_t and $\hat{y}_t$ represent the real and predicted data at time $'t'$.

2 **MAE:** MAE measures the average absolute difference between predicted and actual data, providing an easy signal of prediction accuracy, Eq. (15).

$$\text{MAE} = \frac{1}{T}\sum_{t=1}^{T} |y_t - \hat{y}_t| \quad (15)$$

MAE is less complex to significant errors than MSE, making it cooperative in rendering general prediction accuracy.

3 **RMSE:** RMSE is the square root of MSE and signifies the prediction error in the same units as the real data, helping interpretability, Eq. (16).

$$\text{RMSE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T} (y_t - \hat{y}_t)^2} \quad (16)$$

Significant errors possess a severe impact, and RMSE provides information about the scale of the median prediction error.

4.3 Hyperparameter and Training

Several hyperparameters require being optimized for the T-GCN to predict building durability accurately. The learning rate, batch size, number of training sessions, and number of GCN and GRU layers are key hyperparameters. These hyperparameters were selected and tuned using grid search to identify the optimal configuration for minimizing prediction error on the validation set.

The training was performed using backpropagation with the Adam optimizer, which combines adaptive learning rates and momentum, enhancing convergence efficiency. The MSE loss function was used to minimize the discrepancy between predicted and actual values of structural health metrics. The model was trained for 200 epochs with early stopping based on validation loss to prevent overfitting.

Table 1: Hyperparameter for training

Hyperparameter	Description	Value
GCN Layers	Number of graph convolution layers	2
GRU Units	Number of units in the GRU layer	64
Learning Rate	The initial learning rate for an optimizer	0.001
Batch Size	Number of samples per training batch	32
Training Epochs	Total number of training epochs	200
Early Stopping	Patience for stopping training early	10

4.4 Hardware and Software Configuration

The model training and evaluation were conducted on a machine with an Intel Core i7 processor, 16 GB RAM, and an NVIDIA GTX 1080 GPU to support efficient computation of graph convolutions and temporal dependencies in large datasets. Python and PyTorch were implemented to build models and training for the application environment. The model's efficacy was demonstrated using Matplotlib, while the Deep Graph Library (DGL) addressed graph-based data structures, and NumPy and Pandas were used for the manipulation of data and planning.

4.5 Baseline Models for Comparison

(a) ARIMA: A standard analytical model for predicting time series data; it considers dependencies spanning time but cannot account for spatial connections.

(b) LSTM: An RNN that captures long-term temporal dependencies in sequential data. Although LSTM handles time-series data effectively, it does not account for spatial dependencies between nodes.

(c) GCN: A Deep Learning (DL) that captures spatial dependencies in graph-structured data. While GCNs spatial relationships, it lacks temporal modeling capabilities when used independently.

(d) ST-GCN: Combines GCN for spatial modeling with temporal convolutional layers. It captures spatial and temporal dependencies but may have limitations in capturing complex temporal sequences compared to RNN.

(e) Attention-based Temporal Graph Neural Network (A3T-GCN): By applying graph convolution and attention mechanisms to capture spatial and temporal dependencies. This model is known for focusing on key features in the sequence but may add computational complexity.

5. Result Analysis

5.1 Predictive Accuracy of T-GCN vs. Baseline Models

As shown in Figure 3, the T-GCN achieves the lowest MSE (1.42), MAE (0.92), and RMSE (1.19) values, demonstrating superior accuracy in predicting PS. The A3T-GCN closely follows MSE, MAE, and RMSE values of 1.58, 0.98, and 1.26, respectively, highlighting its effectiveness with slightly higher errors than T-GCN. ST-GCN shows moderate performance, with MSE of 1.97, MAE of 1.17, and RMSE of 1.40, benefitting from its spatial-temporal capabilities but lacking the temporal depth of T-GCN. The LSTM and GCN perform less accurately, with MSE, MAE, and RMSE values of 2.63, 1.44, and 1.62 for LSTM and 2.91, 1.56, and 1.71 for GCN, reflecting limitations in handling combined spatial-temporal dependencies. ARIMA, a purely temporal model, has the highest errors with MSE of 3.57, MAE of 1.78, and RMSE of 1.89, indicating its limited capability for complex spatial-temporal data.

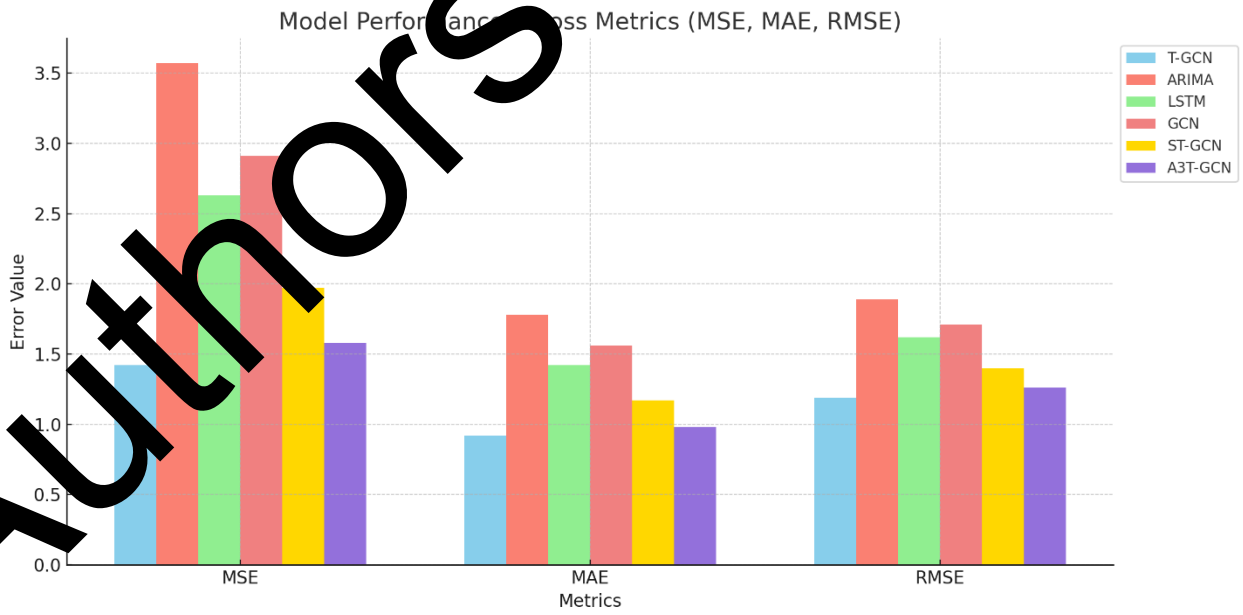


Figure 3: MSE, MAE, and RMSE results for each model

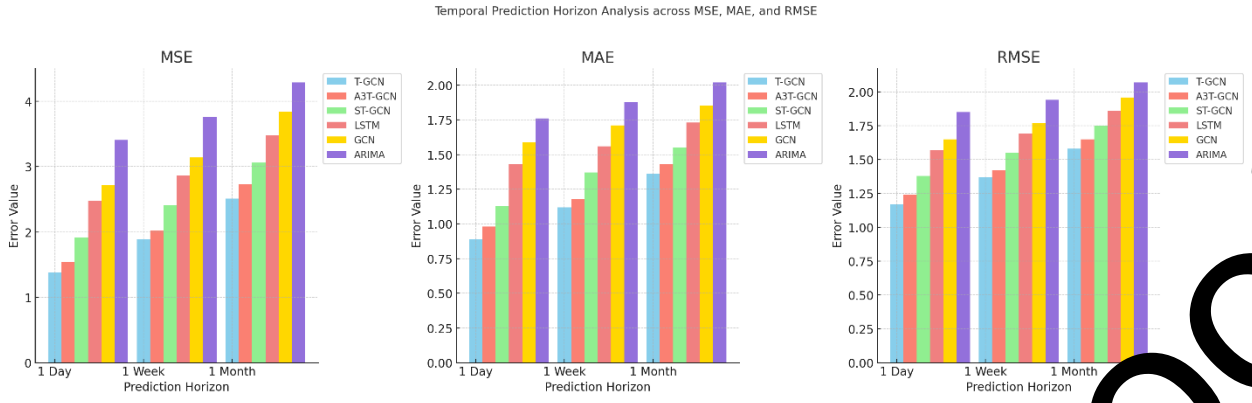


Figure 4: Temporal Prediction Horizon Analysis

5.2 Temporal Prediction Horizon Analysis

The T-GCN demonstrates the highest accuracy across all prediction horizons (Figure 4). For the 1-day horizon, T-GCN achieves an MSE of 1.38, MAE of [illegible], and RMSE of 1.17, outperforming A3T-GCN, which has MSE, MAE, and RMSE values of [illegible]54, 0.98, and 1.24, respectively. ST-GCN follows with higher errors (MSE of 1.[illegible]2), while LSTM, GCN, and ARIMA show progressively worse performance, with ARIMA showing the highest errors (MSE of 3.41, MAE of 1.76, RMSE of 1.85). For the 1-week horizon, T-GCN maintains the lowest errors (MSE of 1.89, MAE of 1.12, RMSE of 1.37). A3T-GCN and ST-GCN perform moderately well but have slightly higher errors (MSE of 2.02 and 2.41, respectively). LSTM, GCN, and ARIMA show further increases in error values, with ARIMA again demonstrating the highest errors (MSE of 3.76, MAE of 1.88, RMSE of 1.94).

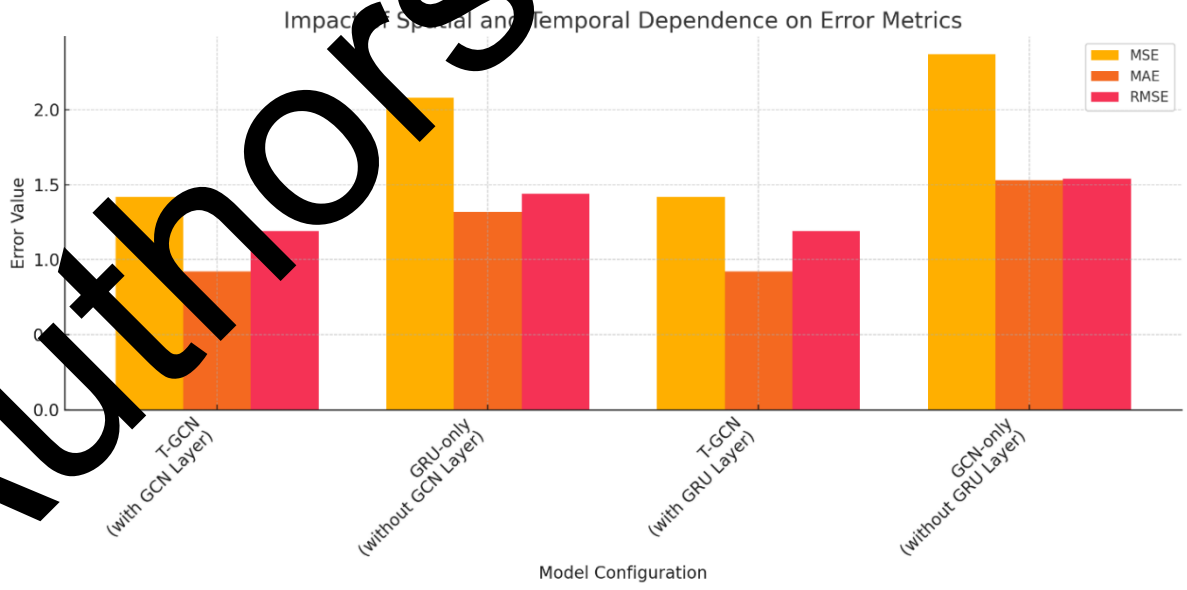


Figure 5: Impact of GCN and GRU layer

At the 1-month horizon, T-GCN remains the most accurate, with an MSE of 2.51, MAE of 1.36, and RMSE of 1.58. A3T-GCN follows with slightly higher errors (MSE of 2.73, MAE of

1.43, RMSE of 1.65), while ST-GCN has an MSE of 3.06. The LSTM and GCN show further degradation in accuracy, while ARIMA records the highest error metrics (MSE of 4.29, MAE of 2.02, RMSE of 2.07), indicating limited capability for long-term temporal predictions.

5.3 Impact of Spatial Dependence (GCN) and Temporal Dependence (GRU) Layer

Including the GCN in the T-GCN significantly enhances accuracy (Figure 5), achieving an MSE of 1.42, MAE of 0.92, and RMSE of 1.19. The GRU-only configuration (without GCN) has a higher MSE of 2.08, MAE of 1.32, and RMSE of 1.44, indicating that the GCN layer effectively captures spatial dependencies among structural components. This improvement demonstrates the GCN's contribution to reducing error by integrating spatial relationships within the model. Similarly, the GRU in the T-GCN enhances predictive accuracy by handling temporal dependencies, resulting in an MSE of 1.42, MAE of 0.92, and RMSE of 1.19. The GCN-only configuration (without GRU) shows increased error metrics, with an MSE of 2.37, MAE of 1.53, and RMSE of 1.54. This comparison underscores the GRU's role in capturing sequential patterns over time, which is essential for accurate resilience predictions in dynamic structural environments.

5.4 Hyper Parameter Sensitivity

The optimal configuration for the T-GCN (Figure 6), with 2 GCN, 64 GRU units, a learning rate of 0.001, and a batch size of 32, achieves the lowest MSE (1.42), MAE (0.92), and RMSE (1.19), indicating balanced accuracy across all metrics and increasing the GCN to 3 results in slightly higher errors (MSE of 1.43, MAE of 0.98, RMSE of 1.24), showing diminishing returns across all metrics. Adjusting the GRU units from 64 to either 32 or 128 similarly raises errors across MSE, MAE, and RMSE, signifying that 64 units best balance complexity and error minimization. The learning rate of 0.001 minimizes MSE, MAE, and RMSE, with higher (0.005) and lower (0.0005) rates leading to increased errors due to convergence challenges. For batch size, 32 yields the lowest errors (MSE, MAE, RMSE), while smaller (16) and larger (64) batch sizes slightly increase all three error metrics, showing that 32 provides the most balanced update rate for optimized performance across MSE, MAE, and RMSE.

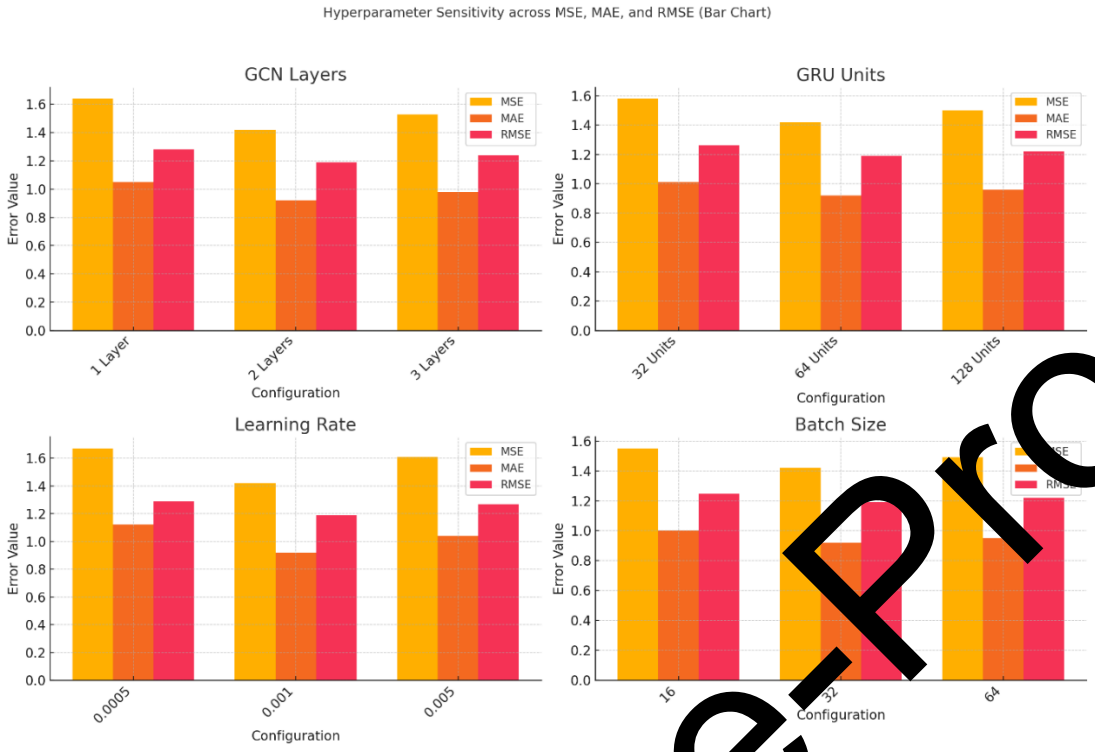


Figure 6: Hyperparameter Sensitivity Results

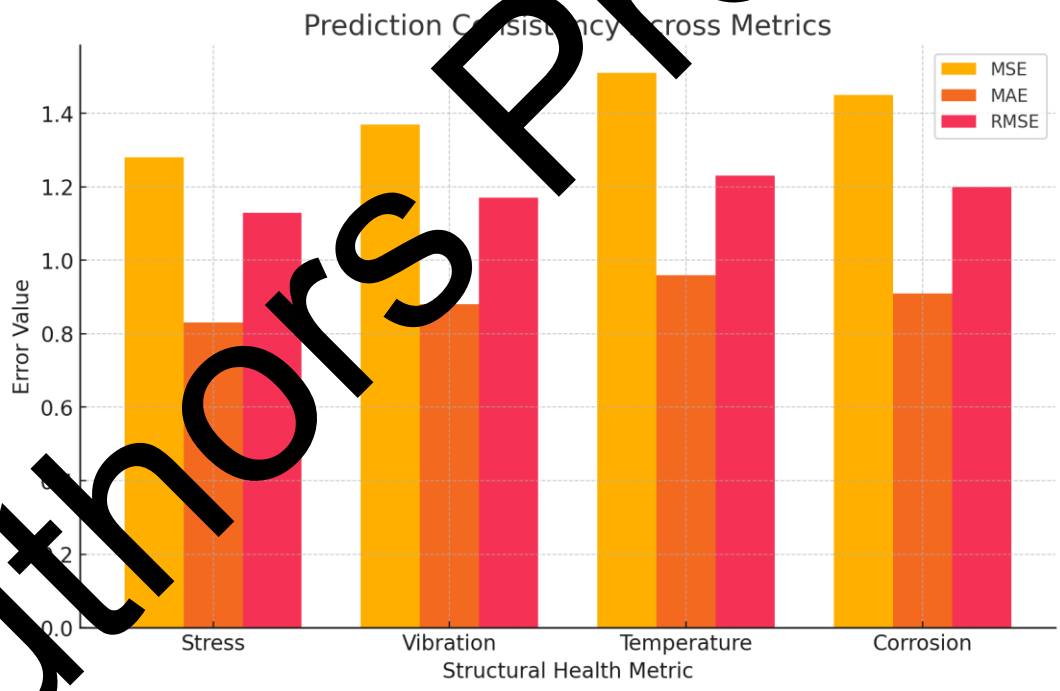


Figure 7: Prediction consistency across metrics

5.5 Prediction Consistency Across Structural Health Metrics

The T-GCN model shows strong consistency across structural health metrics (Figure 7), with the lowest errors in stress prediction (MSE of 1.28, MAE of 0.83, RMSE of 1.13), indicating high accuracy in tracking stress variations. Vibration predictions also perform well, with an MSE of 1.37, MAE of 0.88, and RMSE of 1.17, suggesting reliable capture of dynamic structural responses. Temperature predictions exhibit slightly higher errors (MSE of 1.51, MAE of 0.96, RMSE of 1.23), reflecting the impact of environmental variations on model performance. Corrosion predictions have an MSE of 1.45, MAE of 0.91, and RMSE of 1.20, demonstrating the model's ability to forecast gradual changes in material degradation.

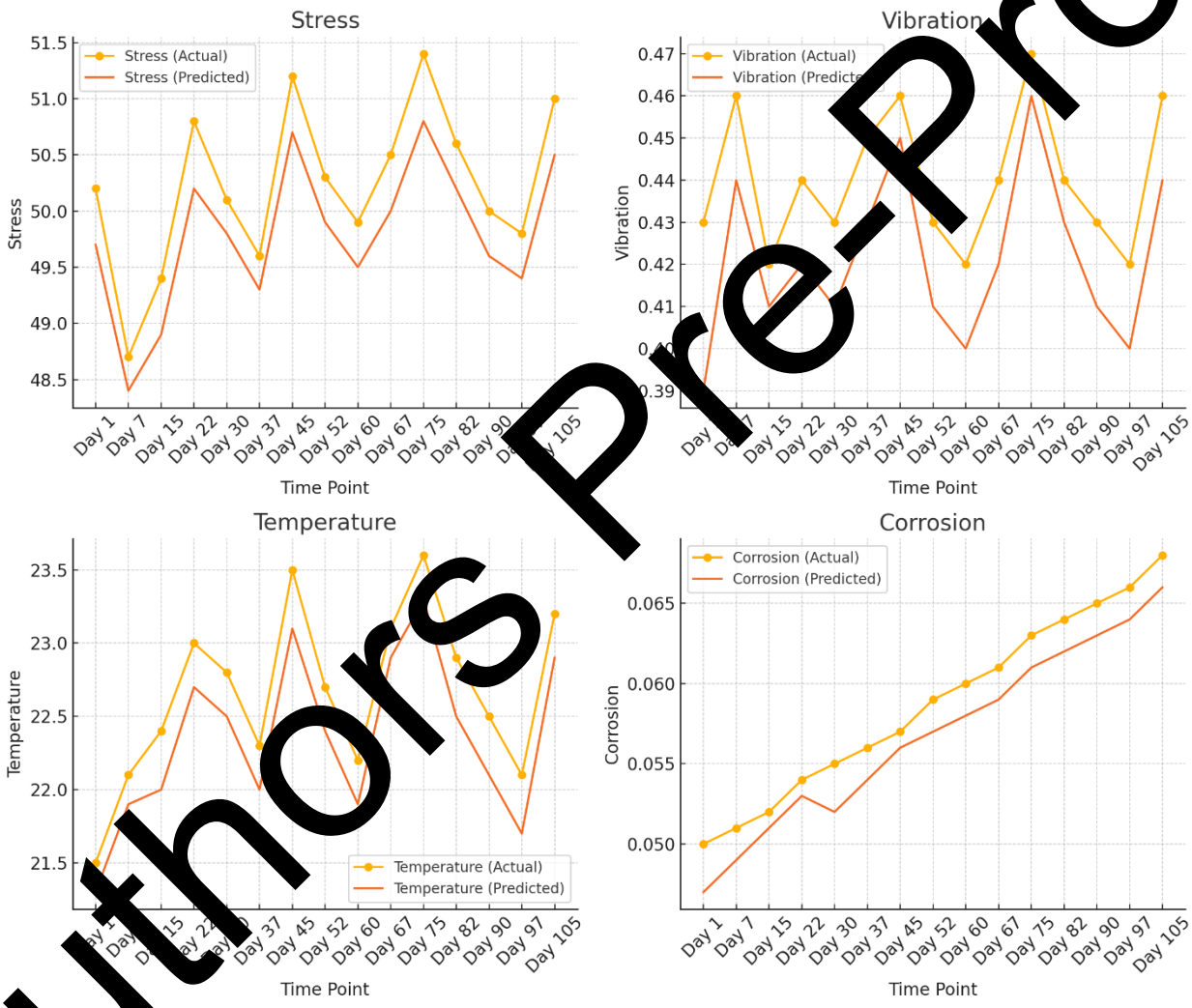


Figure 8: Predicted vs. Actual Metrics

5.6 Predicted vs. Actual Metrics Over Three Months

Over three months, the T-GCN maintains accuracy across metrics (Figure 8) with slight deviations between actual and predicted values. For stress, the predicted values closely follow

actual readings (*e.g.*, Day 1: actual 50.2, predicted 49.7), showing consistent tracking of load distribution. Vibration predictions align with minor differences, reflecting accurate detection of frequency changes (*e.g.*, Day 7: actual 0.46, predicted 0.44). Temperature predictions exhibit realistic deviations, as seen in daily values (*e.g.*, Day 22: actual 23.0, predicted 22.7). Corrosion predictions capture gradual trends with minor variations (*e.g.*, Day 60: actual 0.060, predicted 0.058), indicating reliable long-term monitoring.

6 Conclusion and Future Work

The study presents a robust model for predictive DM and BS in civil engineering, using AI and IoT to monitor structural factors. The IIMS uses real-time data from IoT sensors to provide proactive results for hazard assessment and maintenance planning. The TGCN accurately predicts durability by collecting spatial and temporal links. The model outperforms standard and modern models in essential methods. Optimizing the model's accuracy and efficiency is easy by identifying suitable environments for GCN layers, GRU, learning rate, and batch size. The research supports combining AI-based predictive measures with IoT monitoring to manage infrastructure durability proactively.

In the future, researchers prefer to develop the model further for more general use in structural engineering, explore adaptive methods to accommodate different environmental factors and render it even faster to compute for large-scale applications.

References

1 Hao, H., Bi, K., Chen, W., Pham, T. M., & Li, J. (2023). Towards next-generation design of sustainable, durable, multi-hazard resistant, resilient, and smart civil engineering structures. *Engineering Structures*, *277*, 115477.

2 Berglund, E. Z., Monroe, J. G., Ahmed, I., Noghabaei, M., Do, J., Pesantez, J. E., ... & Levis, J. (2020). Smart infrastructure: a vision for the role of the civil engineering profession in smart cities. *Journal of Infrastructure Systems*, *26*(2), 03120001.

3 Mehvar, S., Wennberg, K., Borsje, B., Kerle, N., Schraagen, J. M., Vinke-de Kruijf, J., ... & Hulscher, S. (2021). Towards resilient, vital infrastructure systems–challenges, opportunities, and future research agenda. *Natural Hazards and Earth System Sciences*, *21*(5), 1383-1407.

Rane, N., Choudhary, S., & Rane, J. (2023). Artificial Intelligence (AI) and Internet of Things (IoT)-based sensors for monitoring and controlling in architecture, engineering, and construction: applications, challenges, and opportunities. *Available at SSRN 4642197*.

5 Alahi, M. E. E., Sukkuea, A., Tina, F. W., Nag, A., Kurdthongmee, W., Suwannarat, K., & Mukhopadhyay, S. C. (2023). Integration of IoT-enabled technologies and artificial

intelligence (AI) for smart city scenario: recent advancements and future trends. *Sensors*, *23*(11), 5206.

6 Zhang, X., Shu, K., Rajkumar, S. A., & Sivakumar, V. (2021). Research on deep integration of application of artificial intelligence in environmental monitoring systems and real economy. *Environmental Impact Assessment Review*, *86*, 106499.

7 Yosefi, A., & Nasseri, H. (2024). A Review Investigation of Damage Identification in Structures via Internet of Things. *Computational Engineering and Physical Modeling*, *7*(1), 73-93.

8 Chen, Q., Cao, J., & Zhu, S. (2023). Data-driven monitoring and predictive maintenance for engineering structures: Technologies, implementation challenges, and future directions. *IEEE Internet of Things Journal*, *10*(16), 14527-14551.

9 Makhoul, N., Roohi, M., van de Lindt, J. W., Sousa, H., Santos, L. O., Argyroudis, S., ... & Zmigrodzki, S. (2024). Seismic resilience of interdependent built environment for integrating structural health monitoring and emerging technologies in decision-making. *Structural Engineering International*, *34*(1), 19-33.

10 Yodo, N., Afrin, T., Yadav, O. P., Wu, P., & Huang, Y. (2023). Condition-based monitoring as a robust strategy towards sustainable and resilient multi-energy infrastructure systems. *Sustainable and Resilient Infrastructure*, *8*(sup1), 170-189.

11 Mishra, M., Lourenço, P. B., & Ramana, G. V. (2022). Structural health monitoring of civil engineering structures by using the internet of things: A review. *Journal of Building Engineering*, *48*, 103954.

12 Mishra, M. (2021). Machine learning techniques for structural health monitoring of heritage buildings: A state-of-the-art review and case studies. *Journal of Cultural Heritage*, *47*, 227-245.

13 Addassu, K. (2022). *Data-Driven Analysis of Transportation Infrastructure Systems using Embedded Wireless Sensing and Cloud-Based Data Architectures* (Doctoral dissertation).

14 Sagar, P. Vidya, et al., Secure multi-party computation in deep learning: Enhancing privacy in distributed neural networks, Journal of Discrete Mathematical Sciences and Cryptography, Vol. 27, No. 2-A, pp. 249–259, DOI: 10.47974/JDMSC-1879, 2024.

15 Cuong, T. N., You, S. S., Long, L. N. B., & Kim, H. S. (2022). Seaport Resilience Analysis and Throughput Forecast Using a Deep Learning Approach: A Case Study of Busan Port. *Sustainability*, *14*(21), 13985.

16 Zhu, J., Han, X., Deng, H., Tao, C., Zhao, L., Wang, P., ... & Li, H. (2022). KST-GCN: A knowledge-driven spatial-temporal graph convolutional network for traffic forecasting. *IEEE Transactions on Intelligent Transportation Systems*, *23*(9), 15055-15065.

17 Ge, L., Li, S., Wang, Y., Chang, F., & Wu, K. (2020). Global spatial-temporal graph convolutional network for urban traffic speed prediction. *Applied Sciences*, *10*(4), 1509.

18 Beeking, M., Steinmaßl, M., Urban, M., & Rehrl, K. (2023). Sparse data traffic speed prediction on a road network with varying speed levels. *Transportation research record*, *2677*(6), 448-465.

19 Gulista Khan et al., Energy-Efficient Routing Algorithm for Optimizing Network Performance in Underwater Data Transmission Using Gray Wolf Optimization Algorithm, Journal of Sensors, Volume 2024, Article ID 2288527, 15 pages, https://doi.org/10.1155/2024/2288527

20 Nabeel S Alsharafa et al., An Edge Assisted Internet of Things Model for Renewable Energy and Cost-Effective Greenhouse Crop Management, Journal of Machine and Computing, Vol. 5, No. 1, pp: 576-588, 2025, https://doi.org/10.53759/7669/jmc202505045

21 Milton Lopez-Cueva et al., A Multi Moving Target Localization in Agricultural Farmlands by Employing Optimized Cooperative Unmanned Aerial Vehicle Swarm, Scalable Computing: Practice and Experience Volume 25, Issues 6, pp. 4647–4660, DOI 10.12694/scpe.v25i6.3130

22 Firas Tayseer Ayasrah et al. Strategizing Low-Carbon Urban Planning through Environmental Impact Assessment by Artificial Intelligence-Driven Carbon Foot Print Forecasting, Journal of Machine and Computing, Vol. 4, No. 04, 2024, doi: 10.53759/7669/jmc202404105.

23 Hayder M. A. Ghanimi et al., Smart Fertilizing Using IOT Multi-Sensor and Variable Rate Sprayer Integrated UAV, Scalable Computing: Practice and Experience, ISSN 1895-1767, Volume 25, Issue 5, pp. 3766–3777, DOI 10.12694/scpe.v25i5.3132

24 Shaymaa Hussein Noori et al., Genetic Algorithms for Optimized Selection of Biodegradable Polymers in Sustainable Manufacturing Processes, Journal of Machine and Computing, Vol. 4, No. 3, PP. 563-574, https://doi.org/10.53759/7669/jmc202404054.

25 Ghanimi, Hayder M. A., et al. Chebyshev polynomial approximation in CNN for zero-knowledge encrypted data analysis, Journal of Discrete Mathematical Sciences and Cryptography, 27:2-A, 203–214, DOI: 10.47974/JDMSC-1880, 2024.