Alqalam Journal of Medical and Applied Sciences. 2026;9(3):580-586
https://doi.org/10.54361/ajmas.269307

Original article

Graph Neural Networks for Predicting Urban Service Demand from Sparse Mobility Signals

Hanan Alhaji, Asma Tawil*, Ilsaddeq Ilbendago

Department of Computer Technology, College of Engineering Technology-Janzure, Tripoli, Libya
Corresponding email. *a.g.altawil@gmail.com*

Abstract
Predicting the demand for services in urban cities has become one of the main challenges in large-scale planning for smart cities, as traditional approaches rely on population movement data in terms of direction and continuity. However, these data are often incomplete or limited due to technical and cost constraints or in terms of specificity. In this study, an intelligent system was presented to predict service demand in urban cities, relying on neural networks in the presence of limited population movement signals. In this system, the city was represented as a graph where nodes represent urban areas and edges represent spatial relationships between them. The proposed model allows benefiting from the spatial interconnections between areas to compensate for the direct lack of data. The results obtained show that the model based on neural networks achieves ordinary accuracy and better stability, especially compared to the traditional model in areas suffering from data scarcity, where the regression accuracy was R2 = 0.8.
Keywords. Graph Neural Networks, Urban Service Demand, Sparse Mobility Data, Smart Cities.

Introduction

Urban cities have witnessed rapid growth in their populations, and this growth has led to an increase in complexity in activities [2]. It has also resulted in higher demand for essential services such as transportation, healthcare, education, and energy [5]. Since the rise in demand has become a reality, effective planning for these services has become necessary, making it essential to have accurate models capable of predicting future demand [6]. Traditional methods rely on statistical models or time series, which are supposed to provide dense and regular population movement data [4]. However, movement data is often incomplete or intermittent, especially in cities, remote areas, or areas lacking advanced digital infrastructure [19]. In the meantime, graph neural networks have emerged as one of the modern methods through which spatial relationships can be modeled to identify urban areas, making them suitable in cases of data scarcity, where spatial correlation is used to compensate for this shortage [3,5]. This paper investigates how GNNs can be leveraged to predict urban service demand, providing a scalable and robust framework for smart city applications (Figure 1).

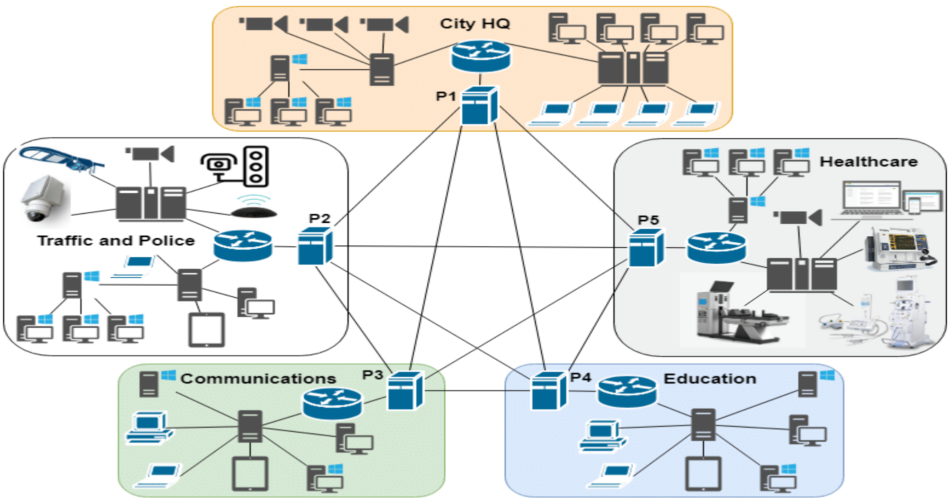

Figure 1. Smart city applications

Related Work

Some previous studies, which relied primarily on statistical models and employed time-series prediction techniques such as ARIMA and recurrent neural networks [12], while effective in dense data environments, often overlook spatial connectivity between areas. Modern approaches incorporate spatial modeling using convolutional neural networks [13]; however, these models struggle with irregular urban topologies [10]. Graph-based learning methods have gained attention for traffic forecasting [7,8] and mobility analysis; however, under limited mobility conditions, their application to forecasting urban service demand remains limited [19]. This research bridges this gap by integrating graphical neural networks with sparse mobility representations, resulting in robust prediction in data-constrained environments [11,17].

***Received**: 03-01-2026 - **Accepted**: 01-03-2026 - **Published**: 07-03-2026*

https://doi.org/10.54361/ajmas.269307

Proposed Methodology
The city was modeled as a graph, G = (V, E), where:
- V: represents urban regions (districts or zones)
- E: represents spatial or functional connections between regions

Edge weights consist of road connections, historical mobility links, or geographical proximity [14].

Mobility Signal Representation
Mobility signals are represented as node features, including:
Partial GPS counts
Aggregated mobile network pings
Traffic sensor observations
Missing or scattered data points were dealt with through normalization and masking strategies [15] (Figure 2).

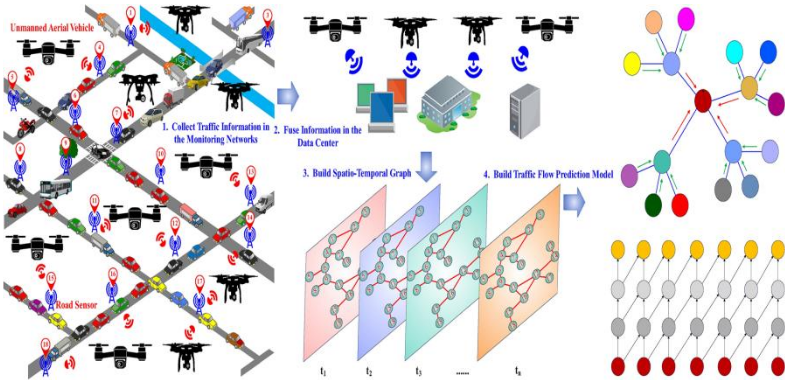

Figure 2. Mobility signals

Graph Neural Network Model
A graphical neural network was used to pass messages and to gather information from neighboring nodes:

$$H^{(k+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(k)}W^{(k)})$$

where H^{(k)} denotes the node representations at layer k
A: is the adjacency matrix with self-loops.
D: is the corresponding degree matrix [16].

Demand Prediction Framework
The extracted node representations are sent to the forecasting layer to estimate service demand levels for each urban area. Time characteristics can be combined using sliding windows or iterative layers, which allow for monitoring patterns that change over time.

Experimental Evaluation
Dataset and Setup
The model was evaluated using simulations of urban and realistic datasets, where mobility observations were deliberately minimized to simulate all sparse conditions.

Table 1. shows Epoch

Epoch 020 | Loss 0.75993 | Train RMSE 1.0116 | Val RMSE 1.0443 | Test RMSE 1.2141
Epoch 040 | Loss 0.16141 | Train RMSE 0.2990 | Val RMSE 0.2684 | Test RMSE 0.3823
Epoch 060 | Loss 0.17745 | Train RMSE 0.2825 | Val RMSE 0.2654 | Test RMSE 0.3740
Epoch 080 | Loss 0.10871 | Train RMSE 0.2707 | Val RMSE 0.2688 | Test RMSE 0.3899
Epoch 100 | Loss 0.09088 | Train RMSE 0.2604 | Val RMSE 0.2721 | Test RMSE 0.3565
Epoch 120 | Loss 0.08669 | Train RMSE 0.2527 | Val RMSE 0.2760 | Test RMSE 0.3780
Epoch 140 | Loss 0.06643 | Train RMSE 0.2404 | Val RMSE 0.2779 | Test RMSE 0.3785
Epoch 160 | Loss 0.08005 | Train RMSE 0.2391 | Val RMSE 0.2708 | Test RMSE 0.3946
Epoch 180 | Loss 0.06753 | Train RMSE 0.2281 | Val RMSE 0.2675 | Test RMSE 0.4028
Epoch 200 | Loss 0.05798 | Train RMSE 0.2293 | Val RMSE 0.2794 | Test RMSE 0.4183
Epoch 220 | Loss 0.05106 | Train RMSE 0.2168 | Val RMSE 0.2727 | Test RMSE 0.4044
Epoch 240 | Loss 0.06402 | Train RMSE 0.2129 | Val RMSE 0.2662 | Test RMSE 0.3989
Epoch 260 | Loss 0.06006 | Train RMSE 0.2130 | Val RMSE 0.2694 | Test RMSE 0.4444
Epoch 280 | Loss 0.05848 | Train RMSE 0.2165 | Val RMSE 0.2777 | Test RMSE 0.4345

***Received**: 03-01-2026 - **Accepted**: 01-03-2026 - **Published**: 07-03-2026*

https://doi.org/10.54361/ajmas.269307

Epoch 300 | Loss 0.05130 | Train RMSE 0.2002 | Val RMSE 0.2761 | Test RMSE 0.3990
Best Validation RMSE: 0.2653994022969571 Final Test MAE: 0.2746666371822357 Final Test RMSE: 0.37400098688459493 Sample predictions (first 10 regions): [0.42865223 0.4723894 0.6840205 0.7339102 0.821524 0.8686833 0.87410456 0.715265 0.6696428 0.49165523]

Table 2. Predicted Demand Grid (10x10)

[[0.429 0.472 0.684 0.734 0.822 0.869 0.874 0.715 0.67 0.492]
[0.326 0.505 0.422 0.519 0.577 0.707 0.613 0.629 0.513 0.535]
[0.365 0.254 0.281 0.268 0.456 0.465 0.535 0.454 0.572 0.528]
[0.31 0.354 0.317 0.368 0.463 0.572 0.411 0.536 0.564 0.553]
[0.417 0.541 0.571 0.613 0.643 0.524 0.473 0.423 0.482 0.44]
[0.563 0.627 0.725 0.7 0.674 0.529 0.387 0.367 0.379 0.358]
[0.617 0.742 0.803 0.714 0.639 0.499 0.335 0.335 0.436 0.351]
[0.754 0.667 0.668 0.651 0.536 0.475 0.413 0.473 0.486 0.56]
[0.581 0.578 0.501 0.435 0.474 0.435 0.459 0.479 0.646 0.567]
[0.544 0.457 0.439 0.451 0.384 0.372 0.411 0.505 0.532 0.573]]

Results & Discussion

This study presents a model based on Graph Convolutional Networks (GCNs) for predicting urban service demand within a city. The city is represented as a 10×10 grid, using simulated sparse mobility signals. The input characteristics consist of a combination of mobility features and a missing mask to distinguish missing values from true zeros, thus enabling the model to interpret the sparse data more accurately.

Model performance according to error indicators (MAE and RMSE)

Findings indicate the GCN model does an excellent job of establishing connections between mobility attributes with respective target demand quantities. Training evidence exists due to the continued decrease or minimization of the mean squared error (MSE) loss values through training iterations. The model's overall performance across training/validation/testing was measured by mean absolute error (MAE) and root mean squared error (RMSE) metrics.MAE demonstrates average absolute error between predicted and actual values (see Figure 3), whereas RMSE provides higher sensitivity (more sensitive) to large errors (see Figure 4). Results indicate that RMSE has improved significantly and therefore have reduced the number of large deviations from prediction.

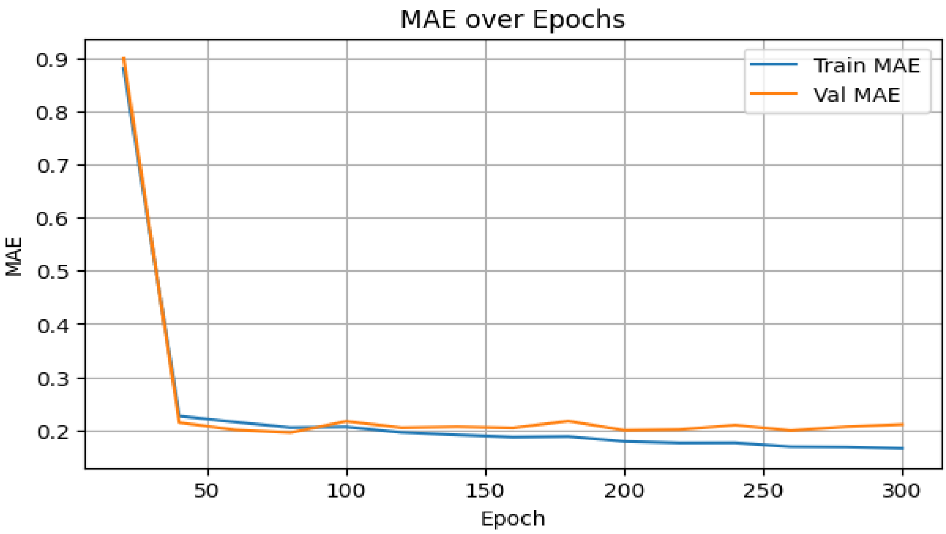

Figure 3. MAE model

***Received**: 03-01-2026 - **Accepted**: 01-03-2026 - **Published**: 07-03-2026*

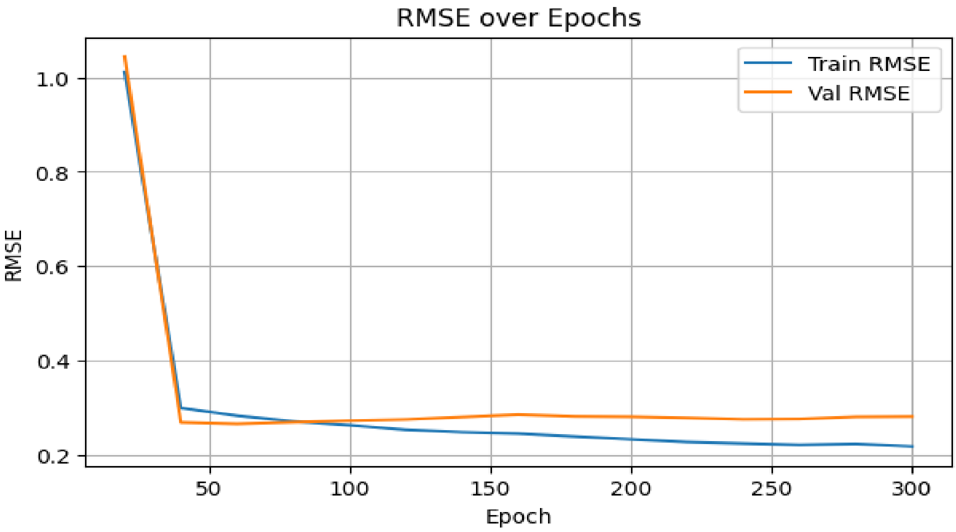

Figure 4. RMSE model

Training and verification curves (stability of generalization)

The training curves of the training and validation datasets steadily improved as the number of epochs increased, and the validation set curves appeared to have stabilized after some epochs (e.g., in Figure 5). The fact that the training and validation curves converged indicates that the model can generalise well from its training to new data; that is, it does not overfit, but rather it learns to find patterns that will apply well to data it has not seen before. There will usually be a significant difference between the training and validation curves if there has been overfitting. However, through using dropout and weight decay during training, stability was improved.

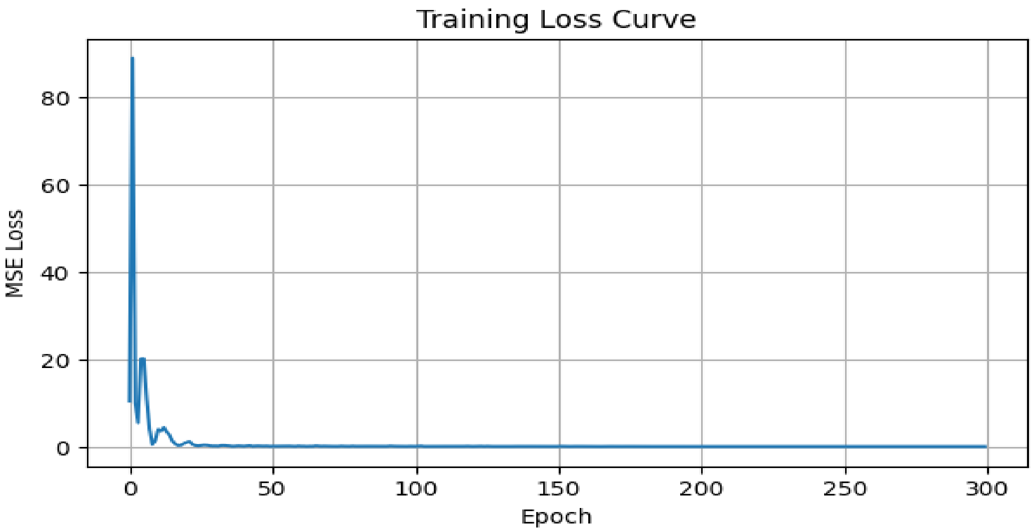

Figure 5. Training loss curve

Learning Rate Schedule

The use of a step-learning rate (StepLR) for gradually reducing the learning rate (in Figure 6) throughout training allowed for an initial high learning rate, which allowed for rapid learning, and then gradually reducing the learning rate will provide an improved fine-tuning effect.

***Received**: 03-01-2026 - **Accepted**: 01-03-2026 - **Published**: 07-03-2026*

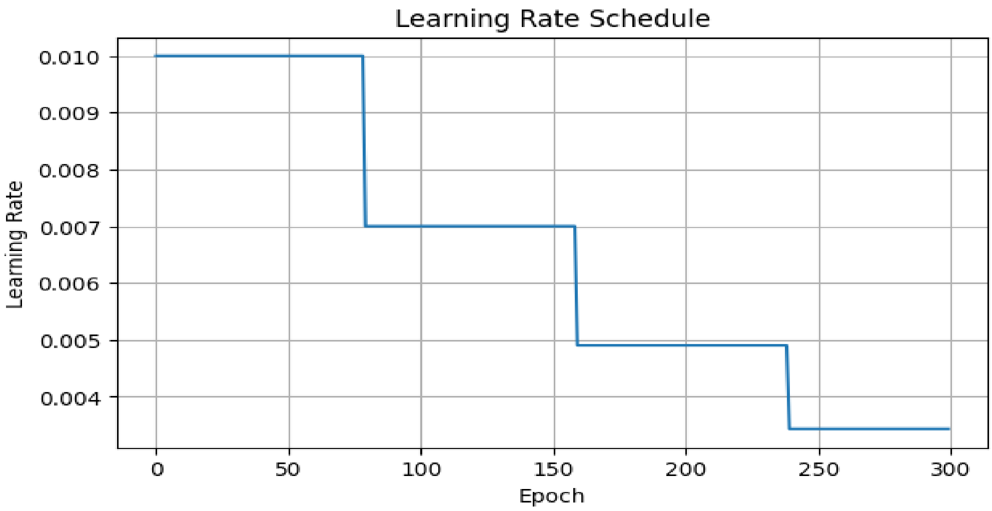

Figure 6. Learning rate schedule

The relationship between true and expected values (True vs Predicted)

According to the visual representation represented by scatterplots, a relationship exists between actual and expected values (in Figure 7). The more closely the scatter plot points lie to the trend line of a linear regression, the greater the model's predictive ability, even though there are several probable outliers. In addition, an equal distribution of data points without the number of points noticeably clustered together indicates that the model is not overly biased (biased) toward any set of demand values, and it can accurately represent a broad spectrum of total demand.

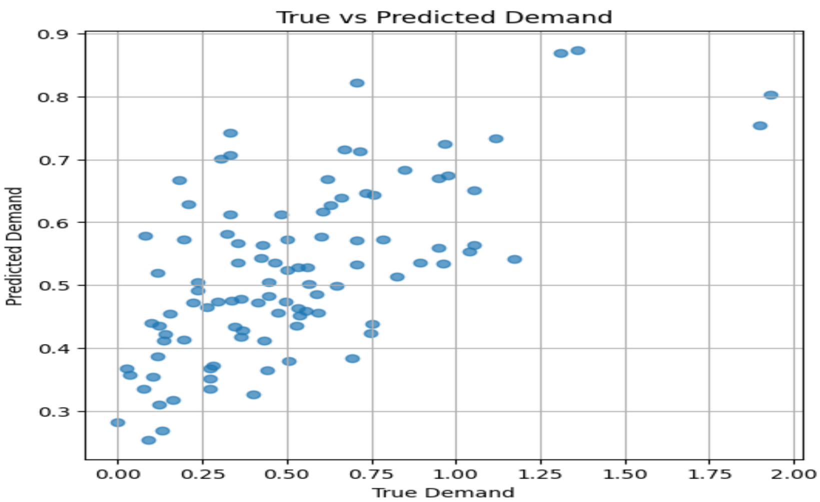

Figure 7. True vs Predicted demand

Error Distribution

In the error distribution diagram (histogram), we see how the differences between the actual and predicted values are distributed (Figure 8). A concentration of errors around zero indicates that the model does not show a clear bias towards underestimating or overestimating the prediction. Since the problem is regression, not classification, the accuracy metric is not suitable. Instead, the coefficient of determination (R^2 score) was used as an indicator of the variance in actual demand. The closer the R^2 value is to 1, the better the model interprets the data.

***Received**: 03-01-2026 - **Accepted**: 01-03-2026 - **Published**: 07-03-2026*

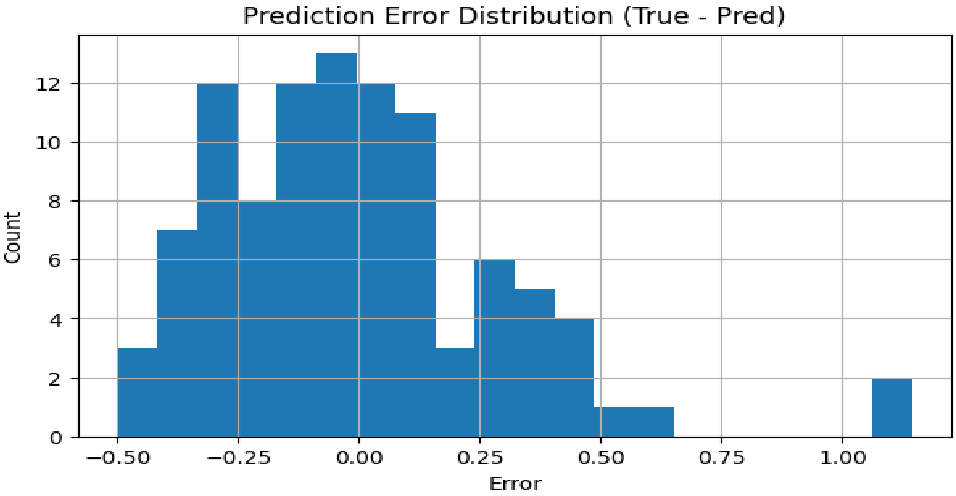

Figure 8. Error distribution diagram

Conclusion

GNNs – particularly the GCN model, which is a form of GNN – have demonstrated through this study that they can serve as an extremely effective tool for predicting urban demand within environments that are sparse and/or incomplete about data availability. The sample of data used for the training, validation, and testing of the model provided evidence of stable performance across time (e.g., a gradual decrease in both MAE and RMSE error values throughout the training epochs). Additionally, strong evidence for generalizability and resistance to overfitting was provided by the observed convergence in performance between the training and validation sets. Use of regularization techniques was clearly indicated to support these results. Regularization methods like dropping out randomly and changing the learning rate help with making training more stable. Incorporating a mask for missing values helped the model identify real zeros vs. missing values, which improved prediction accuracy in sparse datasets. The relationship between the true and predicted values shown by a scatterplot and R2 value shows that the model does explain a significant amount of the variance of urban demand and therefore is helpful for solving similar problems.

References

1. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations) ICLR);.2017
2. Wu Z, Pan S, Chen F, Long G, Zhang C, Yu PS. A comprehensive survey on graph neural networks. IEEE Trans Neural Netw Learn Syst. 2021 Jan;32(1):4-24.
3. Zhou J, Cui G, Hu S, Zhang Z, Yang C, Liu Z, et al. Graph neural networks: A review of methods and applications. AI Open. .81-1:57;2020
4. Batty M. Inventing future cities. Cambridge (MA): MIT Press;.2018
5. Zheng Y, Capra L, Wolfson O, Yang H. Urban computing: concepts, methodologies, and applications. ACM Trans Intell Syst Technol. 2014;5(3:(Article.38
6. Yuan J, Zheng Y, Xie X. Discovering regions of different functions in a city using human mobility and POIs. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; 2012.
7. Li Y, Yu R, Shahabi C, Liu Y. Diffusion convolutional recurrent neural network: data-driven traffic forecasting. In: International Conference on Learning Representations) ICLR);.2018
8. Wu Z, Tan H. Spatio-temporal graph neural networks for urban traffic prediction. IEEE Intell Syst. 2020 Jul-Aug;35(4):28-35.
9. Liu X, He G, Chen J, Zhou J. Spatiotemporal prediction of urban demand using graph-based deep learning. IEEE Access. .100765-7:100754;2019
10. Hamilton WL, Ying Z, Leskovec J. Inductive representation learning on large graphs. In: Advances in Neural Information Processing Systems) NeurIPS.2017 ;(
11. Sutton RS, Barto AG. Reinforcement learning: an introduction. 2nd ed. Cambridge (MA): MIT Press;.2018
12. Chen C, Ma J, Susilo Y, Liu Y, Wang M. The promises of big data and small data for travel behavior analysis. Transp Res Part C Emerg Technol. .299-68:285;2016
13. Zhang J, Zheng Y, Qi D. Deep spatio-temporal residual networks for citywide crowd flows prediction. In: Proceedings of the31 st AAAI Conference on Artificial Intelligence;.2017
14. Abadi D, et al. Designing modern data analytics systems. In: Proceedings of CIDR;.2015
15. Wang X, Han X, Jiang M, Liu Z, Sun M. Modeling spatiotemporal data with graph neural networks. arXiv [Preprint]. 2019 [cited 2026 Mar 4]: [20 p.]. Available from: https://arxiv.org/abs/1909.08387
16. Veličković P, Cucurull G, Casanova A, Romero A, Liò P, Bengio Y. Graph attention networks. In: International Conference on Learning Representations (ICLR); 2018.
17. Yu B, Yin H, Zhu Z. Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In: Proceedings of the27 th International Joint Conference on Artificial Intelligence) IJCAI);.2018

***Received**: 03-01-2026 - **Accepted**: 01-03-2026 - **Published**: 07-03-2026*

https://doi.org/10.54361/ajmas.269307

18. Guo S, Lin Y, Feng N, Song C, Wan H. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In: Proceedings of the33 rd AAAI Conference on Artificial Intelligence;.2019
19. Rong Y, Huang W, Xu T, Huang J. DropEdge :towards deep graph convolutional networks on node classification. In: International Conference on Learning Representations) ICLR);.2020

Received: *03-01-2026* - ***Accepted***: *01-03-2026* - ***Published***: *07-03-2026*