Yale CPSC 483/583: Deep Learning on Graph-Structured Data (Fall 2024)

Koray Akduman
Department of Computer Science
Yale University
New Haven, CT
koray.akduman@yale.edu

1 Introduction

Past literature on traffic analysis uses a spatio-temporal approach to predicting traffic. One popular and successful model is a Diffusion Convolutional Recurrent Neural Network (DCRNN) [3]. A DCRNN models traffic flow as a diffusion process on a graph to capture both spatial and temporal dependencies. To take advantage of spatial data, previous works use a distance-based adjacency matrix, with distances based on the driving distance along the road network between two sensors. However, it remains unclear whether this assumption is always optimal. It stands to reason that two road sensors may be far apart physically but still be highly correlated (e.g., sensors on the same highway segment yet far away, experiencing similar traffic patterns).

This study presents a comparison between the widely-used distance-based approach and a new correlation-based approach, where correlations between sensors are calculated based on the traffic speeds. These correlations are then used to build the graph representing spatial features of the road network. Our findings show that constructing graph features based on speed correlations provides worse predictive ability for future traffic speeds when compared to a distance-based approach, measured by mean absolute error (MAE) and root mean square error (RMSE). This result suggests that physical spatial relationships are more important than correlational relationships that capture the functional dynamics of traffic flow. However, our results also demonstrate that there is some merit in considering correlational relations while constructing this graph, as a GCN trained using only spatial data performs better on a correlational graph as opposed to a distance-based graph. Future works may consider important correlational edges as a supplement to existing distance-based edges in traffic network graphs.

2 Related Work

Deep Learning Models. The advent of deep learning has revolutionized traffic prediction by enabling models to learn intricate patterns from large-scale traffic data. Among these models, Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRU) have been particularly prominent due to their ability to effectively capture temporal dependencies in sequential traffic data [1]. LSTMs, with their memory cell structure, can retain information over extended time periods, making them suitable for modeling long-term traffic trends. Similarly, GRUs offer a streamlined architecture with fewer parameters, facilitating faster training while maintaining performance comparable to LSTMs. However, despite their strengths in temporal modeling, both LSTM and GRU networks inherently lack mechanisms to model the spatial structure of road networks. This limitation restricts their ability to fully exploit the spatial correlations and dependencies that are critical for accurate traffic prediction.

Representing Traffic Networks as Graphs. Accurately representing road networks is fundamental to enhancing traffic prediction models. Road networks are naturally structured as graphs, where intersections and road segments are modeled as nodes and edges, respectively. Traditionally, these graphs have been constructed using distance-based adjacency matrices, wherein the edges between nodes (sensors) are weighted based on the physical driving distance along the road network [4]. This method provides a straightforward way to incorporate spatial information into the model. However, distance-based adjacency matrices may oversimplify the complex interactions inherent in traffic systems. For instance, they might not adequately capture factors such as traffic flow directionality, road capacity, or the impact of nearby intersections. Consequently, while distance-based graphs offer a foundational approach, there is a need for more sophisticated representations that can encapsulate the multifaceted nature of traffic dynamics.

Spatio-Temporal Approaches in Traffic Prediction. Addressing the dual challenges of spatial dependencies and temporal dynamics, spatio-temporal forecasting has become a central focus in traffic analysis. These approaches aim to model the intricate interplay between the spatial layout of road networks and the temporal evolution of traffic conditions [3]. Among the various models developed, the Diffusion Convolutional Recurrent Neural Network (DCRNN) has emerged as a particularly effective solution. DCRNN leverages bidirectional random walks on the graph to capture spatial dependencies, allowing the model to understand how traffic information diffuses across the network. Concurrently, it employs an encoder-decoder architecture with scheduled sampling to model temporal dependencies, effectively handling the non-linear and dynamic nature of traffic patterns. This integrated approach has demonstrated substantial improvements, achieving consistent enhancements of 12-15% over previous state-of-the-art baselines when evaluated on large-scale road network traffic datasets [3]. The success of DCRNN underscores the importance of simultaneously addressing spatial and temporal factors in traffic prediction.

Graph-based Models. Graph Convolutional Networks (GCNs) have gained prominence as powerful tools for modeling traffic on road networks, which are inherently non-Euclidean in nature [2]. GCNs excel at capturing the structural relationships between nodes in a graph, making them well-suited for representing complex road networks. Building on this foundation, spatio-temporal models have been developed to integrate GCNs with temporal processing units. For example, the DCRNN [3] combines graph diffusion convolution with recurrent units to effectively capture both spatial and temporal dependencies. Further advancements in graph-based traffic prediction include the incorporation of attention mechanisms, as seen in models like GMAN [8] and ASTGCN [9], which enhance the model's ability to focus on the most relevant parts of the graph for prediction tasks. Additionally, synchronous approaches such as STSGCN [6] have been proposed to better align spatial and temporal processing, ensuring that the model can simultaneously consider spatial correlations and temporal sequences. These graph-based advancements have significantly improved the accuracy and robustness of traffic prediction models, demonstrating the versatility and efficacy of GCNs in this domain.

3 Method

This study introduces a correlation-based graph construction method and compares it to the traditional distance-based graph construction. We evaluate the resulting graphs using multiple models that represent different aspects of spatio-temporal learning.

3.1 Model Architecture

We implemented and compared four distinct models:

Distance-based DCRNN: A DCRNN [3] using a graph constructed from the physical distances between sensors.

Correlation-based DCRNN: A modified DCRNN using a graph where edges are determined by correlations between sensor speed time series rather than physical distance.

LSTM: An LSTM network capturing temporal patterns, without explicit spatial modeling.

Distance-based GCN: A GCN using the distance-based graph, focusing only on spatial relationships without temporal modeling.

Correlation-based GCN: A GCN using the correlation-based graph, similarly focusing solely on spatial aspects.

3.2 Graph Construction

Distance-based Graph. The distance-based graph was constructed using driving distances between sensors. Edge weights are inversely proportional to distance.

Correlation-based Graph. We computed Pearson correlation coefficients between the traffic speed time series of each pair of sensors. Edges were retained if the correlation exceeded a threshold (0.6), ensuring a sparse but informative graph.

3.3 Data Processing and Training

We used the METR-LA dataset [5], which consists of traffic speed measurements at 209 sensors in Los Angeles collected between March and June of 2012. Missing values were interpolated, frontfilled, or backfilled, and the data were normalized. The dataset was split into training (70%), validation (10%), and test (20%) sets. Each model was trained for 5 epochs, using a learning rate of 0.01 halved every epoch, and a hidden dimension size of 24. We used the minimum number of layers to accomplish the goal of the underlying architecture while keeping the model lightweight enough to run locally. We additionally preprocessed our data by building a distance-based graph and correlation-based graph visualized in Figure 3. All models were trained using the same hyperparameters for fair comparison.

The models predicted the next 15 minutes of traffic, given the last hour of data, where each time step is 5 minutes.

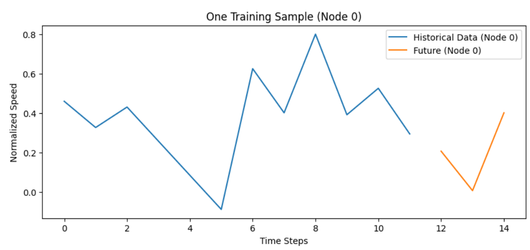


Figure 1: Example prediction for one sensor

3.4 Evaluation Metrics

We evaluated models using commonly accepted metrics [3]:

- **MAE:** Mean Absolute Error
- **RMSE:** Root Mean Square Error

3.5 Novelty and Contribution

By using correlation-based edges, we challenge the assumption that physical distance is the best proxy for sensor relationships. The correlation-based approach can capture non-trivial relationships between sensors that are not captured by physical proximity.

The distance-based GCRNN is the best possible baseline for our study of evaluating the correlation-based GCRNN. However, we also compare these models to simpler baselines (LSTM) and spatial-only baselines (GCN), providing insights into the relative importance of spatial and temporal components. These can also be thought of as pseudo-ablation studies, taking away a component of our models by defining new models that forgo either the temporal or the spatial component of our data.

4 Experiments

We analyzed the METR-LA traffic dataset, a dataset consisting of traffic speed measurements at 209 sensors in Los Angeles between March and June 2012, providing a total of 34272 time steps with 5-minute granularity. The first 100 time steps are visualized for five sensors in Figure 2. Our training hyperparameters and other details are defined in cell 2 of main.ipynb. Specifically, we used hidden layer widths of 24, trained for 5 epochs, started with a learning rate of 0.01 (and halved every epoch), and we used the random seeds 0, 1, and 2 for three runs on all models. We also used 70% of the data for training, 10% for validation, and 20% for testing, as is commonplace in previous works [3]. We used the minimum number of layers to accomplish the goal of the underlying architecture while keeping the model lightweight enough to run locally.

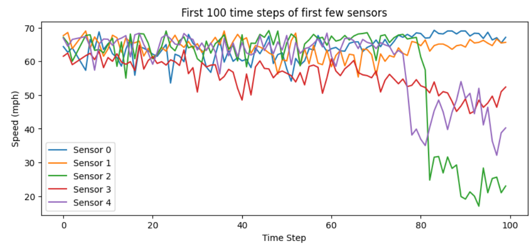
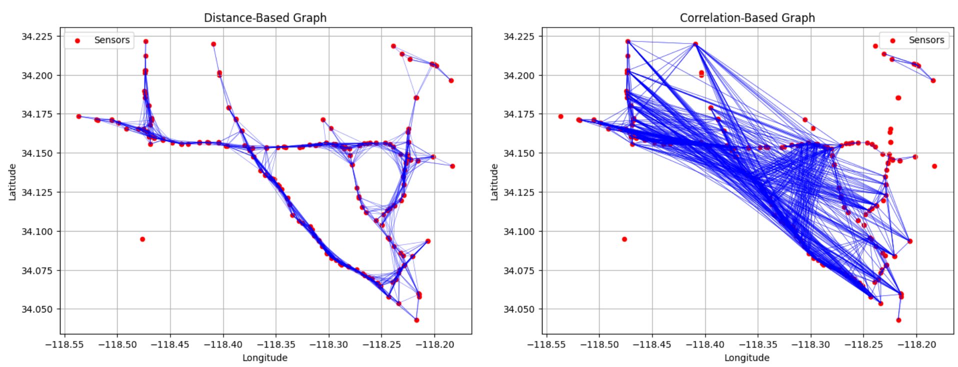


Figure 2: The first 100 time steps of traffic speed for five sensors.

We preprocessed the dataset and constructed both the distance-based and correlation-based graphs. Figures 3(a) and 3(b) show these graphs with sensors plotted at their respective coordinates. The distance graph contains 1722 edges, whereas the correlation graph contains 2099 edges.



(a) Distance-based Graph

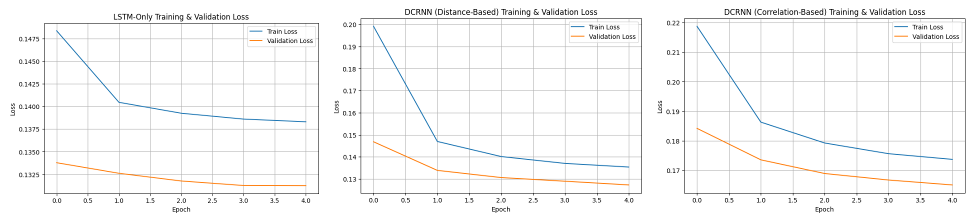
(b) Correlation-based Graph

Figure 3: Visualization of the two graphs used in this study.

These findings align with prior research with DCRNNs and RNNs that demonstrate a great capability of RNNs (and specifically LSTMs) to predict traffic data and neglect solely spatial approaches as spatial approaches would assume traffic is identical at all time steps, which is trivially false [3]. Our findings, consistent with prior work, also demonstrate a slight improvement over LSTMs by using distance-based DCRNNs [3]. One notable difference between our project and prior DCRNN work is that rather than stacking multiple recurrent layers, we rely on a single DiffusionGraphConv-based GRU cell for the encoder and another for the decoder [3]. This simplification reduces the representational capacity of the model, potentially limiting performance. Indeed, we see that the performance of the model on training data never exceeds that of the model on the validation data despite the exclusion of dropout and weight decay, indicating that the model is perhaps not complex enough to fully capture the patterns in the training data.

4.1 Results

Our results (Table 1 and Figure 5) show that the DCRNN using the distance-based graph performs better than the DCRNN using the correlation-based graph, contrary to our initial hypothesis that mapping by correlation would perform better. Interestingly, the GCN with solely correlation outperforms the distance-based graph, showing that there is some merit to including correlation as part of the graph. However this effect is likely overpowered by the temporal significance of the data. Both GCNs perform markedly worse than models representing temporal data, as trivially predicted by the largely temporal task of predicting future traffic patterns. We also show that the LSTM performs similar to the distance-based DCRNN, where while the DCRNN obtains slightly lower RMSE (our loss metric), the LSTM outperforms the DCRNN in terms of MAE. This effect may be explained in large part by not training for enough epochs, where 5 epochs shows that the DCRNN was still improving while the LSTM had reached a plateau (Figure 4). Increasing the complexity of the DCRNN would likely allow it to outperform the LSTM, as demonstrated by Li et al., 2018, but the goal of this project is to compare the distance-based GNN to the correlation-based GNN.



(a) LSTM Training Curve (b) DCRNN (Distance) Training Curve (c) DCRNN (Correlation) Training Curve

Figure 4: Representative training and validation loss curves. The DCRNN continues to improve, while the LSTM plateaus.

Table 1: Performance comparison of all models.

Model	MAE (Std Err)	RMSE (Std Err)
LSTM	2.594 ± 0.004	4.841 ± 0.001
DCRNN (Distance)	2.904 ± 0.003	4.827 ± 0.004
DCRNN (Correlation)	3.402 ± 0.004	5.510 ± 0.004
GCN (Distance)	7.387 ± 0.027	10.763 ± 0.005
GCN (Correlation)	5.537 ± 0.010	8.508 ± 0.003

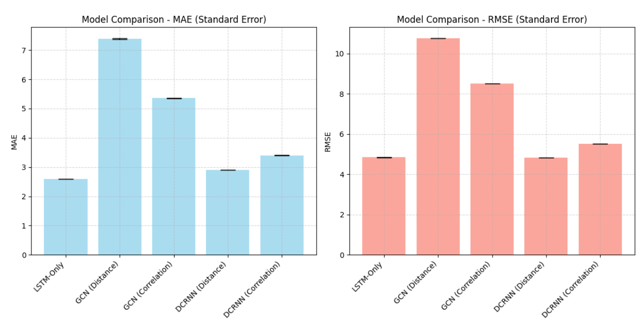


Figure 5: The performance (MAE, RMSE) for all models, with standard error shown.

5 Conclusion

We create two DCRNNs trained on a graph with distance-based edges and a graph with edges based on correlations between traffic speeds at different sensors. The DCRNN trained on the distance-based graphs outperforms the other yet performs only slightly better than the LSTM, though this is likely due to computational constraints restricting the complexity of our DCRNN. Our data also show poor performance of the two GCN models excluding temporal information, demonstrating a strong predictive power of temporal data as opposed to spatial data in traffic prediction.

Our show that while correlations cannot supplant distance-based edges, there may be merit to including some correlations, as this graph contains new information for the GNN that may not be apparent using the distance-based GNN, evidenced by a correlation-based GCN outperforming a distance-based GCN. For future projects analyzing traffic networks, then, it may be worthwhile to consider high correlations as edges alongside distances.

Reproducibility

The results can be reproduced by following the instructions in the README at `https://github.com/kakduman/dcrnn-distance-correlation`.

References

[1] Chen, K., Chen, F., Lai, B., Jin, Z., Liu, Y., Li, K., Wei, L., Wang, P., Tang, Y., Huang, J., & Hua, X.-S. (2020). Dynamic spatio-temporal graph-based cnns for traffic flow prediction. *IEEE Access*, 8, 185136–185145. `https://doi.org/10.1109/ACCESS.2020.3027375`

[2] Lee, K., & Rhee, W. (2022). DDP-GCN: Multi-graph convolutional network for spatiotemporal traffic forecasting. *Transportation Research Part C: Emerging Technologies*, 134, 103466. `https://doi.org/10.1016/j.trc.2021.103466`

[3] Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2018). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting (arXiv:1707.01926). *arXiv*. `https://doi.org/10.48550/arXiv.1707.01926`

[4] Mallick, T., Balaprakash, P., Rask, E., & Macfarlane, J. (2020). Graph-partitioning-based diffusion convolutional recurrent neural network for large-scale traffic forecasting (arXiv:1909.11197). *arXiv*. `https://doi.org/10.48550/arXiv.1909.11197`

[5] METR-LA. (2024). *Kaggle*. `https://www.kaggle.com/datasets/annnnguyen/metr-la-dataset`

[6] Sofianos, T., Sampieri, A., Franco, L., & Galasso, F. (2021). Space-time-separable graph convolutional network for pose forecasting (arXiv:2110.04573). *arXiv*. `https://doi.org/10.48550/arXiv.2110.04573`

[7] Wang, H., Chen, J., Zhang, L., Jiang, R., & Song, X. (2024). Unveiling the inflexibility of adaptive embedding in traffic forecasting (arXiv:2411.11448). *arXiv*. `https://doi.org/10.48550/arXiv.2411.11448`

[8] Zheng, C., Fan, X., Wang, C., & Qi, J. (2020). Gman: A graph multi-attention network for traffic prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(01), 1234–1241. `https://doi.org/10.1609/aaai.v34i01.5477`

[9] Zhu, J., Tao, C., Deng, H., Zhao, L., Wang, P., Lin, T., & Li, H. (2020). Ast-gcn: Attribute-augmented spatiotemporal graph convolutional network for traffic forecasting (arXiv:2011.11004). *arXiv*. `https://doi.org/10.48550/arXiv.2011.11004`