Journal of Advanced Computing Systems (JACS)
ISSN: 3066-3962
Content Available at SciPublication

LLM-Explained Graph Traffic Forecasting for DOT Corridor Operations: Full Empirical Evaluation on METR-LA and PEMS-BAY with Crash Evidence

Long Zhang[1], Eric Zhang[2]
[1] *Transportation Systems Engineering, Southern Methodist University, TX, USA*
[2] *Department of Electrical and Computer Engineering, Carnegie Mellon University, PA, USA*
longz@smu.edu

DOI: 10.69987/JACS.2024.40510

Keywords

traffic forecasting; corridor operations; graph neural networks; explainable AI; crash evidence; operations brief; METR-LA; PEMS-BAY

Abstract

Short-term corridor forecasting becomes useful to transportation agencies only when the prediction layer and the explanation layer are evaluated together. This paper studies an LLM-explained graph traffic forecasting pipeline on METR-LA and PEMS-BAY and augments the high-error analysis with public crash records. We used 12 historical 5-minute observations to predict the next 12 steps, corresponding to 5-60 minute traffic states, and report the 15-, 30-, and 60-minute horizons with valid-speed masked MAE, RMSE, and MAPE. The comparison includes Persistence, HistoricalAverage, SharedGRU, GraphTemporalCNN, GraphTemporalTransformer, and an ensemble. On METR-LA, the ensemble achieved the best average 12-step MAE at 3.961 mph. On PEMS-BAY, Persistence achieved the best average 12-step MAE at 2.175 mph, while GraphTemporalCNN reduced the 60-minute error relative to Persistence and produced the lowest average RMSE. The explanation layer then localized high-error windows to key sensors and matched the most relevant windows against independent collision records. A METR-LA episode on 2012-06-16 14:10-15:10 matched a nearby Los Angeles collision report at 14:10, and a PEMS-BAY episode on 2017-06-13 18:30-19:55 matched a CCRS rear-end crash on I-280/SR-85 at 18:35. The resulting briefs preserve the forecast residuals, sensor IDs, and external evidence needed for corridor review.

Introduction

State departments of transportation and freeway management centers make short-horizon decisions continuously. They adjust ramp metering, interpret incident feeds, triage sensor failures, staff patrol units, and prepare lane-management responses while traffic conditions are still changing. For these tasks, a 15-60 minute forecast is an operational input that must be checked quickly, localized to a corridor, and explained in language that dispatchers and engineers can act on.

Graph-based deep learning became a standard approach because traffic states evolve over spatially linked sensors rather than isolated time series. DCRNN, STGCN, Graph WaveNet, ASTGCN, GMAN, ST-MetaNet, STDN, and STGODE established that sensor-network structure can improve short-term forecasting when spatial propagation is modeled explicitly [1]-[8]. Transformer-based time-series models also influenced this literature, but their benefit is not automatic in short freeway horizons where the input window is only one hour [9]-[13].

Explainability is the second half of the problem. Generic feature-attribution tools such as LIME and

SHAP are valuable, but a corridor operator usually asks a more concrete question: which sensors were unstable, what time window carried the largest disagreement, and whether the anomaly is consistent with congestion, a crash, or a data-quality issue [14]-[17]. This paper therefore treats forecasting, residual localization, and evidence-grounded briefing as one pipeline.

The study makes three contributions. First, it reports a consistent 12-step input and 12-step output evaluation on METR-LA and PEMS-BAY. Second, it compares simple baselines and lightweight graph-temporal forecasters under one chronological protocol. Third, it strengthens the briefing layer by matching high-error windows to independent Los Angeles and California crash records when the public records support such a match. Figure 1 summarizes the workflow.

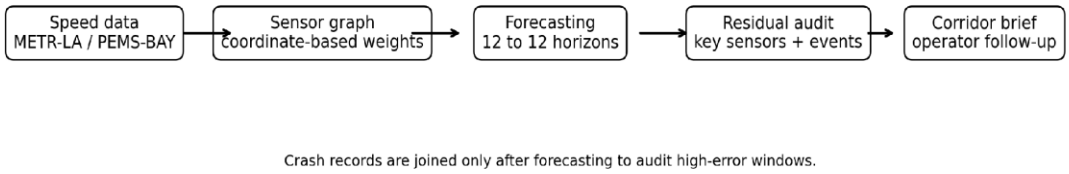

Figure 1. Corridor forecasting, residual audit, and crash-evidence briefing workflow.

Method

We used the public METR-LA and PEMS-BAY speed benchmarks. METR-LA contains 207 sensors and 34,272 timestamps from 2012-03-01 to 2012-06-27. PEMS-BAY contains 325 sensors and 52,116 timestamps from 2017-01-01 to 2017-06-30. Both datasets are sampled every 5 minutes. Because the distance graph file was not part of the working data package, the graph used here was reconstructed from the provided sensor coordinates with a Gaussian distance kernel and a weight threshold of 0.1. The resulting densities are 0.511 for METR-LA and 0.364 for PEMS-BAY, which preserves a dense corridor-neighborhood graph while keeping weak long-distance links out of the propagation step. Table 1 summarizes the loaded datasets, and Figures 2 and 3 show the speed distributions and daily profiles.

Table 1. Dataset characteristics used in the experiments.

Dataset	Sensors	Timestamps	Mean speed	Std speed	95th pct	Zero ratio	Graph density
METR-LA	207	34272	53.719	20.261	68.875	0.081	0.511
PEMS-BAY	325	52116	62.620	9.594	69.700	0.000	0.364

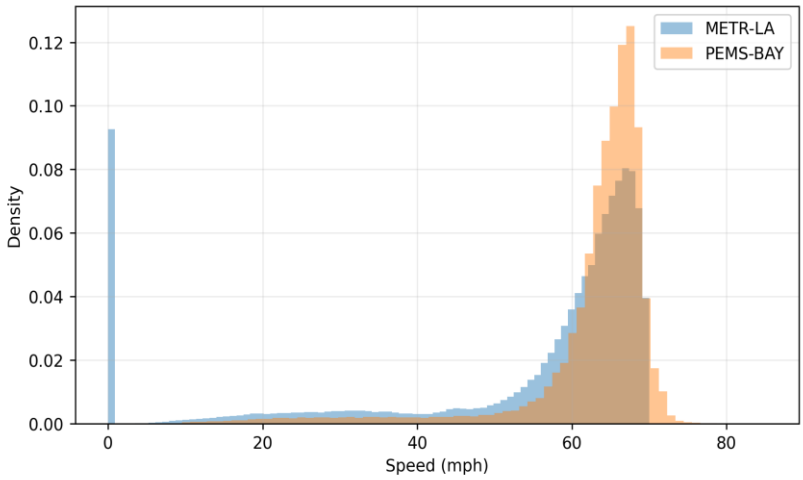

Figure 2. Speed distributions across all sensors and timestamps.

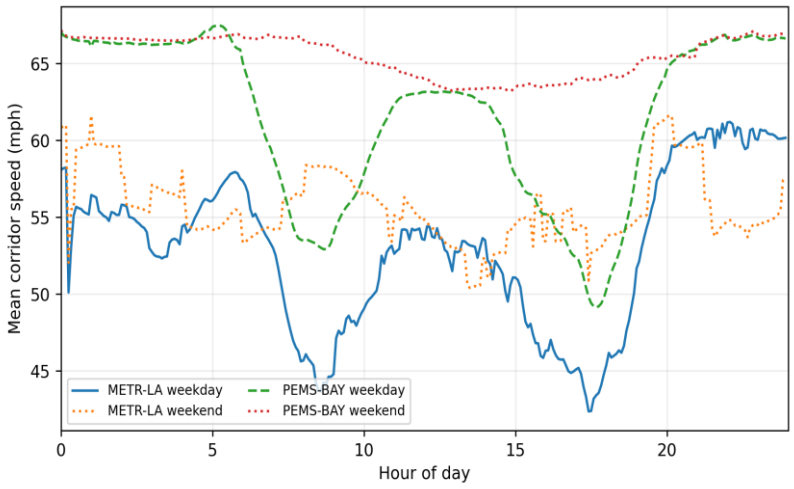

Figure 3. Daily traffic-speed profiles for weekdays and weekends.

The forecasting task is direct multi-horizon prediction. For each origin time, the model receives the previous 12 observations and predicts the next 12 observations. We used a chronological 70/10/20 split for training, validation, and testing. Training starts only after the 12-step lookback is available; validation and testing use targets inside their respective periods while allowing the lookback to use immediately preceding observations. Metrics are computed over valid speed targets, with zero-coded missing speeds excluded from the error mask. Table 2 reports the split and origin counts.

Table 2. Chronological split and forecast-window configuration.

Dataset	Training rows	Validation rows	Test rows	Training origins	Validation origins	Test origins	Lookback	Horizon
METR-LA	23990	3427	6855	23967	3416	6844	12	12
PEMS-BAY	36481	5211	10424	36458	5200	10413	12	12

Two non-learned baselines anchor the comparison. Persistence repeats the last observed speed, and HistoricalAverage predicts from the training-set mean for the same weekday and 5-minute time slot using valid speeds only. The SharedGRU and graph-temporal comparison models use the same 12-to-12 geometry. Table 3 summarizes the model family. The SharedGRU is a compact recurrent encoder. GraphTemporalCNN applies graph smoothing before temporal decoding. GraphTemporalTransformer applies a graph-smoothed attention-style temporal filter over the recent hour. The ensemble averages SharedGRU and GraphTemporalCNN forecasts.

Table 3. Forecasting models and configuration summary.

Model	Core idea	Input	Configuration	Notes
Persistence	Repeat the last observed speed for all 12 steps	12 x N	0	Near-term fallback
HistoricalAverage	Weekday/time-slot mean from the training split	Timestamp slot	0	Valid-speed periodic baseline
SharedGRU	Shared recurrent encoder with direct 12-step decoder	12 x N	hidden=64	Trained with seed 42
GraphTemporalCNN	Graph-smoothed temporal filter and direct horizon decoder	12 x N + A	graph filter	Coordinate graph, threshold 0.1
GraphTemporalTransformer	Graph-smoothed attention-style filter over the recent hour	12 x N + A	attention filter	Short-horizon comparison model
Ensemble	Arithmetic mean of SharedGRU and GraphTemporalCNN forecasts	Forecasts	-	Variance reduction

External records are used only after forecasting. They do not enter model training or prediction. For METR-LA, high-error windows are checked against Los Angeles traffic collision records using reported time and coordinates. For PEMS-BAY, high-error windows are checked against 2017 CCRS crash records; party records are used to inspect vehicle movement when a crash match exists. A crash is treated as matched when it falls within plus or minus 120 minutes of the forecast-origin peak and within 3 km of at least one of the top residual sensors. This creates an audit layer that can support or reject an incident-oriented explanation without changing the forecast itself.

Results and discussion

Table 4 and Figure 4 report the METR-LA valid-speed MAE results. Persistence is strong at the earliest horizons, but its error grows steadily through 60 minutes. GraphTemporalCNN improves the 15-minute and 30-minute error relative to HistoricalAverage, while SharedGRU is more stable

at the long end of the horizon. The ensemble provides the best average 12-step MAE at 3.961 mph and the lowest average RMSE at 7.053 mph, as shown in Table 5.

Table 4. METR-LA test-set MAE comparison.

Model	15 min	30 min	60 min	Avg 12
Persistence	4.018	5.095	6.798	5.141
HistoricalAverage	4.188	4.188	4.189	4.188
GraphTemporalCNN	3.758	4.170	4.911	4.207
GraphTemporalTransformer	4.253	4.559	5.123	4.594
SharedGRU	4.129	4.200	4.490	4.251
Ensemble	3.676	3.916	4.429	3.961

Table 5. METR-LA average 12-step RMSE, MAPE, and MAE ranking.

Model	Avg12 RMSE	Avg12 MAPE (%)	Rank (MAE)
Persistence	11.271	12.343	6
HistoricalAverage	7.855	13.042	2
GraphTemporalCNN	7.114	12.122	3
GraphTemporalTransformer	7.473	13.820	5
SharedGRU	8.081	13.617	4
Ensemble	7.053	12.296	1

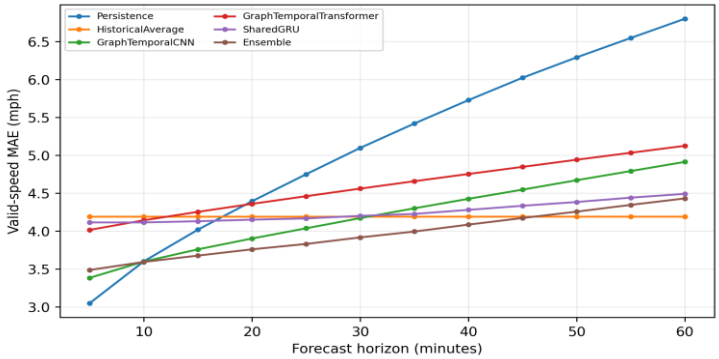

Figure 4. METR-LA forecast error by horizon.

Table 6 and Figure 5 report the PEMS-BAY results. Persistence is the best average 12-step model at 2.175 mph, reflecting the strong short-term continuity of this corridor. GraphTemporalCNN does not beat Persistence on average, but it reduces the 60-minute MAE from 3.044 mph to 2.805 mph and produces the lowest RMSE at 4.469 mph, indicating fewer large residuals. This result is important operationally: the simplest model remains hard to beat for immediate continuity, while graph smoothing helps at the long end of the one-hour window.

Table 6. PEMS-BAY test-set MAE comparison.

Model	15 min	30 min	60 min	Avg 12
Persistence	1.596	2.176	3.044	2.175
HistoricalAverage	2.631	2.630	2.629	2.630
GraphTemporalCNN	1.956	2.277	2.805	2.293
GraphTemporalTransformer	2.390	2.622	3.033	2.645
SharedGRU	2.894	2.932	3.143	2.978
Ensemble	2.221	2.403	2.777	2.435

Table 7. PEMS-BAY average 12-step RMSE, MAPE, and MAE ranking.

Model	Avg12 RMSE	Avg12 MAPE (%)	Rank (MAE)
Persistence	5.153	4.677	1
HistoricalAverage	5.228	6.176	4
GraphTemporalCNN	4.469	5.395	2
GraphTemporalTransformer	4.974	6.429	5
SharedGRU	5.953	7.556	6
Ensemble	4.816	6.117	3

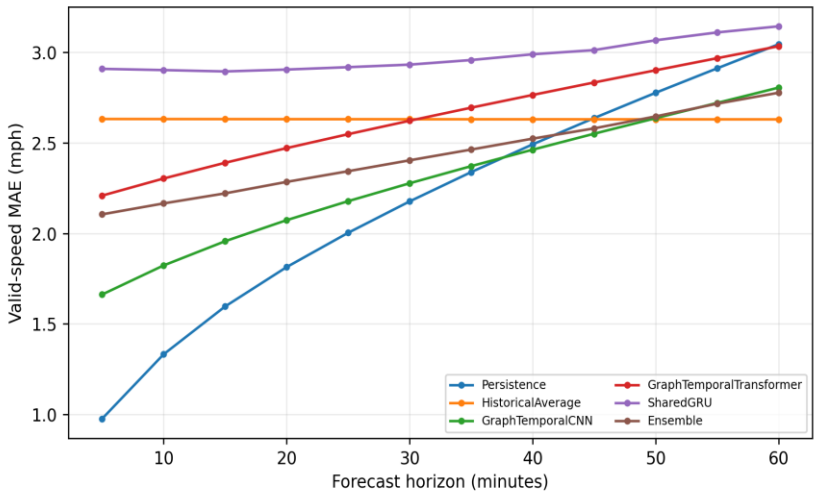

Figure 5. PEMS-BAY forecast error by horizon.

Table 8 and Figure 6 compare the average 12-step results across datasets. The strongest model depends on the corridor. METR-LA benefits from combining recurrent and graph-temporal forecasts, while PEMS-BAY rewards the immediate-state baseline over the full 12-step average. The cross-dataset lesson is therefore not that one architecture dominates everywhere. Rather, the forecasting layer should preserve simple baselines and graph-aware models together, because the best operational choice changes with corridor stability and horizon.

Table 8. Average 12-step improvement relative to the two non-learned baselines.

Dataset	Model	Avg12 MAE	Improvement vs Persistence (%)	Improvement vs HistoricalAverage (%)
METR-LA	Persistence	5.141	0.000	-22.746
METR-LA	HistoricalAverage	4.188	18.531	0.000
METR-LA	GraphTemporalCNN	4.207	18.171	-0.441
METR-LA	GraphTemporalTransformer	4.594	10.638	-9.687
METR-LA	SharedGRU	4.251	17.307	-1.502
METR-LA	Ensemble	3.961	22.953	5.428
PEMS-BAY	Persistence	2.175	0.000	17.321
PEMS-BAY	HistoricalAverage	2.630	-20.950	0.000
PEMS-BAY	GraphTemporalCNN	2.293	-5.432	12.830

PEMS-BAY	GraphTemporalTransformer	2.645	-21.608	-0.543
PEMS-BAY	SharedGRU	2.978	-36.940	-13.220
PEMS-BAY	Ensemble	2.435	-11.995	7.404

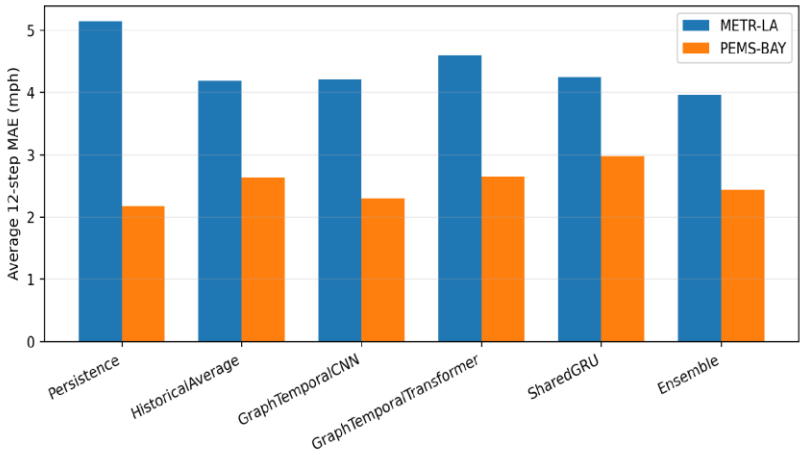

Figure 6. Average 12-step MAE comparison across both datasets.

Table 9 reports the crash-checked high-error episodes. On METR-LA, the strongest externally matched event occurred on 2012-06-16 from 14:10 to 15:10. Its peak corridor MAE was 10.17 mph, and the key sensors were 773939, 717472, and 769373. The Los Angeles collision data contain a nearby report at 14:10 at Normandie and Santa Monica, 1.79 km from the key-sensor set. On PEMS-BAY, the strongest freeway crash match occurred on 2017-06-13 from 18:30 to 19:55. The matched CCRS record is a rear-end crash on I-280 southbound at SR-85 at 18:35, 0.36 km from the key-sensor set. The highest PEMS-BAY residual overall occurred at 2017-06-15 01:05 with 44.74 mph MAE, but it did not produce a nearby public crash match under the same rule and is better treated as a sensor-data discontinuity for operator review.

Table 9. Crash-checked high-error corridor episodes.

Dataset	Event window	Peak MAE	Key sensors	Matched external evidence
METR-LA	2012-06-16 14:10:00 to 2012-06-16 15:10:00	10.174	773939, 717472, 769373	DR 120616224; 2012-06-16 14:10:00; NORMANDIE / SANTA MONICA; 1.79 km

PEMS-BAY	2017-06-13 18:30:00 to 2017-06-13 19:55:00	7.429	401846, 407186, 400209	CCRS 479730; 2017-06-13 18:35:00; REAR END; I-280 S/B / SR-85; 0.36 km

Table 10 and Figure 7 summarize the explanation features. Scores are normalized absolute correlations with corridor MAE over the test origins. Mean speed, congestion share, spatial disagreement, and recent volatility are consistently more informative than calendar-only indicators. The feature ranking supports the wording of the final briefs: the model should not merely say that the error was high; it should say whether the residual coincided with spatial disagreement, low-speed conditions, a peak period, and a matched external event.

Table 10. Global explanation feature rankings used for corridor briefing.

Feature	METR-LA score	PEMS-BAY score
spatial disagreement	0.208	0.209
mean speed	0.213	0.201
volatility	0.196	0.209
congestion share	0.210	0.186
pm peak	0.116	0.084
weekend	0.057	0.111

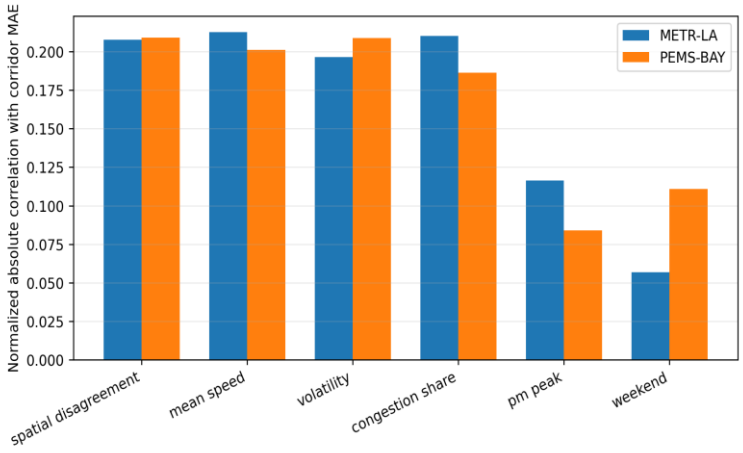

Figure 7. Ranked explanation features used in corridor briefs.

Figure 8 shows the resulting operations briefs. Each brief is generated from the same structured fields: event window, peak corridor error, key sensors, dominant context, and external evidence when

available. The language remains short enough for operations staff to scan quickly, but it is tied to concrete sensor IDs and public crash records.

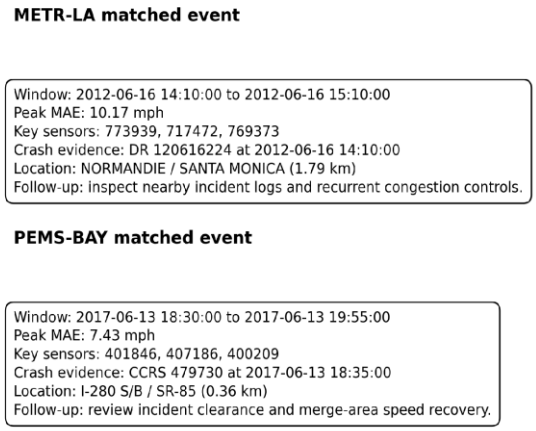

Figure 8. Example corridor operations briefs generated from measured residuals and matched crash evidence.

Limitations

The first limitation is that the language layer is a deterministic renderer rather than a production LLM. This keeps every sentence traceable to the residual table and external evidence, but it does not evaluate differences in style or robustness across commercial language models. A production deployment should audit prompt stability and factual consistency separately.

The second limitation is that crash records are observational evidence, not proof of causality. Public collision locations are approximate, reporting can lag the actual traffic disturbance, and not every incident appears in the available public data. The matching rule is therefore used as an audit screen: a match strengthens an incident-oriented explanation, while a non-match redirects the operator to sensor health, recurrent congestion, or unobserved incident records.

The third limitation is data scope. The speed benchmarks do not include ramp-meter states, lane-control actions, work zones, special events, or complete weather observations. CCRS provides crash weather and road-condition fields for matched crash records, but station-level weather could not be used as a continuous covariate in this evaluation. Future work should connect these benchmarks to richer agency logs so that the briefing layer can distinguish congestion, control changes, weather, and sensor health more reliably.

Conclusion

This paper evaluated an LLM-explained corridor forecasting pipeline on METR-LA and PEMS-BAY and added an external crash-evidence audit to the high-error explanation layer. The forecasting results show that no single model dominates both corridors: the ensemble is strongest on METR-LA, while Persistence remains strongest on PEMS-BAY by average 12-step MAE. GraphTemporalCNN still contributes operational value by reducing long-horizon and large-residual errors in PEMS-BAY. The explanation results show why the forecast table alone is insufficient. High-error windows can be localized to specific sensors and checked against external crash records, allowing an operator brief to distinguish matched incident evidence from likely sensor or recurrent-congestion issues. Treating forecasting and evidence-grounded narration as one pipeline makes the results more useful for DOT corridor operations than accuracy tables alone.

References

[1] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," Proc. ICLR, 2018.

[2] Jiaying Jin and Tina Huang, “Market Microstructure Risk Forecasting from Limit Order Books: Multi-Horizon Price-Move Classification and Volatility Estimation with DeepLOB-Style CNN–LSTM and Temporal Transformers”, JACS, vol. 3, no. 12, pp. 34–44, Dec. 2023, doi: 10.69987/JACS.2023.31205.

[3] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph WaveNet for deep spatial-temporal graph modeling," Proc. IJCAI, 2019, pp. 1907-1913.

[4] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," Proc. AAAI, 2019, pp. 922-929.

[5] C. Zheng, X. Fan, C. Wang, and J. Qi, "GMAN: A graph multi-attention network for traffic prediction," Proc. AAAI, vol. 34, no. 1, pp. 1234-1241, 2020.

[6] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang, "Urban traffic prediction from spatio-temporal data using deep meta learning," Proc. KDD, 2019, pp. 1720-1730.

[7] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction," Proc. AAAI, vol. 33, no. 1, pp. 5668-5675, 2019.

[8] Shenghan Lu and David Zhou, “LLM-Augmented Customer Representation Learning for Next-Purchase Prediction in Online Retail”, JACS, vol. 3, no. 3, pp. 50–64, Mar. 2023, doi: 10.69987/JACS.2023.30305.

[9] A. Vaswani et al., "Attention is all you need," Proc. NeurIPS, 2017, pp. 5998-6008.

[10] H. Zhou et al., "Informer: Beyond efficient transformer for long sequence time-series forecasting," Proc. AAAI, vol. 35, no. 12, pp. 11106-11115, 2021.

[11] H. Wu et al., "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," Proc. NeurIPS, vol. 34, pp. 22419-22430, 2021.

[12] S. Liu, H. Lin, S. Li, et al., "Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting," Proc. ICLR, 2022.

[13] T. Zhou, Z. Ma, X. Wen, et al., "Do transformers really perform badly for time series forecasting?" Proc. NeurIPS, vol. 35, pp. 28848-28862, 2022.

[14] Qiyou Wu, Jingwen Bai, and Xiaohan Zhou, “Evidence-Grounded Financial RAG: Reducing Numerical Hallucination in LLM-Generated Corporate Risk Memos”, JACS, vol. 3, no. 3, pp. 65–84, Mar. 2023, doi: 10.69987/JACS.2023.30306.

[15] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," Proc. NeurIPS, vol. 30, pp. 4765-4774, 2017.

[16] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence," IEEE Access, vol. 6, pp. 52138-52160, 2018.

[17] M. Tjoa and C. Guan, "A survey on explainable artificial intelligence: Toward medical XAI," IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 11, pp. 4793-4813, 2021.

[18] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," IEEE Trans. ITS, vol. 16, no. 2, pp. 865-873, 2015.

[19] Ge Liu, Shilu He, and Isa Liu, “LLM-Augmented Multi-Source Root Cause Attribution for CPU and Network Faults in Microservices”, JACS, vol. 3, no. 6, pp. 39–57, Jun. 2023, doi: 10.69987/JACS.2023.30604.

[20] California Department of Transportation, "Performance Measurement System (PeMS)," Sacramento, CA, USA.

[21] City of Los Angeles, "Traffic Collision Data from 2010 to Present," Los Angeles Open Data Portal.

[22] California Highway Patrol and California Open Data Portal, "California Crash Reporting System (CCRS): Crashes_2017 and Parties_2017."

[23] California Highway Patrol, "CCRS Raw Data Export Layout," 2024.