Journal Not Specified
2025, Vol. 1, Cite as: doi:

SCIFINITI

Article

OPEN ACCESS

Federated Forecasting of Urban Traffic Congestion with Simple GNNs and Attention-Based RNNs

Javier Rodrigues[1]

Article History
Submitted: Accepted: Published:

Abstract
Background: Cities increasingly rely on data from heterogeneous intersections to anticipate congestion, yet centralizing data raises privacy and governance concerns. We study whether federated learning (FL) can train accurate congestion forecasters without moving data off site. **Methods:** Using the Smart Traffic Management Dataset, we define an interpretable *Traffic Congestion Index* (TCI) from speed, volume, incidents, signals, and weather. We engineer time-lag and rolling features, and train two lightweight forecasters across five client sites with FedAvg: (i) a two-layer graph model that performs diffusion/smoothing over a road-segment graph followed by a small MLP ("simple GNN"), and (ii) a sequence model (bi-LSTM with self-attention). We log metrics every federated round on a fixed, never-updated test window per site to ensure blind evaluation. **Results:** Across five sites the FL models achieve low blind-test error: *GNN* mean±sd MAPE $= 5.48\% \pm 1.06$, *RNN* $= 5.74\% \pm 1.67$. Both strongly outperform non-federated tabular baselines (Linear Regression, Random Forest, Gradient Boosting; MAPE ≈25–37%). Learning-curve analyses show rapid convergence (<50 rounds) and consistent cross-site behavior. **Conclusions:** A privacy-preserving FL pipeline with simple, robust models can deliver accurate congestion forecasts while keeping data local. Our contributions include: an open TCI target and feature recipe, a minimal graph smoother competitive with sequence models, and a reproducible FL evaluation protocol with blind tests and per-site reporting. Code, assets, and figures are organized for straightforward reproducibility.

Keywords:
federated learning; traffic forecasting; time series; graph neural networks; LSTM; privacy; urban computing; congestion index

1. Introduction

Urban traffic operations centers aggregate high-volume, heterogeneous streams (speeds, counts, incidents, signal states, weather) to anticipate congestion and coordinate control. Centralizing these data can be infeasible or undesirable due to policy, privacy, or bandwidth constraints, and models must remain lightweight and stable across sites with different regimes and sensing noise.

Approach in this work. We investigate whether *federated learning* (FL) can train high-quality congestion forecasters while data remain on site. We design an end-to-end FL pipeline that (i) defines an interpretable *Traffic Congestion Index* (TCI) combining demand, speed, control delays, incidents, and weather into a single ($0 \to 100$) scale; (ii) exposes temporal dynamics via lags and rolling statistics; and (iii) trains two complementary predictors with FedAvg across five clients: a simple diffusion-style graph model ("GNN") and a bi-LSTM with self-attention ("RNN"). The graph model intentionally avoids heavy architectures: it performs K-hop feature propagation on a sparse roadway adjacency and applies a small MLP regressor—easy to tune and fast to train while still capturing spatial dependence. The RNN provides a strong sequential baseline.

* **Corresponding Author:**
Name, affiliation, e-mail@email.com;
Tel.: +xx-xxx-xxx-xxxx

Dataset and target. Experiments use the Smart Traffic Management Dataset. A dataset derived from sensor data captured from a set of anonymous U.S. roads over 48 hours in January 2024. From raw variables we derive TCI, an interpretable 0–100 score that increases with rising demand (cars, trucks, bikes relative to nominal capacity), reduced speed relative to free-flow, red/yellow phases, incidents, and adverse weather. We then add standard time-series features (e.g., TCI lags at 1–24 h; rolling mean/std windows) and cyclical encodings of hour-of-day. The target/feature recipe is deliberately simple so it can transfer across sites.

Federated protocol. We run FedAvg for 100 rounds across five sites. Each site maintains a fixed train/val/test split; the *test* split is never used for updates. At the end of every round, the global model is broadcast and evaluated on each site's immutable test set, producing "blind" learning curves over rounds. We report per-site and cross-site metrics (MAE, RMSE, MAPE) and compare against non-FL baselines.

Contributions. (1) A transparent TCI target and lightweight feature recipe that work across heterogeneous intersections; (2) a simple graph forecaster with a small MLP head—competitive with a stronger attention RNN while remaining efficient and interpretable; (3) a federated evaluation protocol with per-site blind tests and round-wise logging; and (4) a reproducible artifact set (code, training scripts, and paper-ready figures/tables).

1.1. Exploratory data analysis (EDA)

Our raw variables (speeds, counts, signal phases, incidents, weather) show clear, monotonic relationships with congestion and provide practical signals for a bounded, interpretable target.

Speed and signal phase.

Fig. 1 shows a strong inverse relation between speed and congestion (TCI proxy), with distinct vertical shifts by signal phase. For a fixed speed, *red* phases sit highest, *yellow* in the middle, and *green* lowest. This suggests two design choices: (i) a continuous speed penalty that decreases with v, and (ii) additive, categorical offsets for signal phases rather than fitting a single global slope.

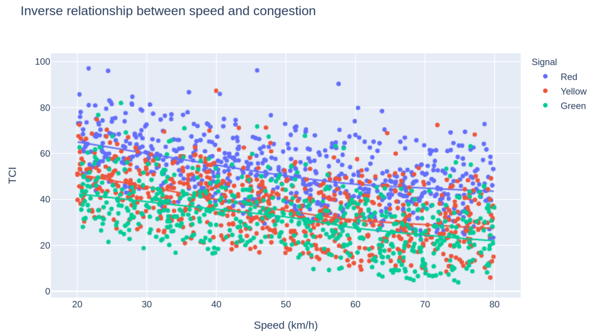

Figure 1: Inverse relationship between speed and congestion, stratified by signal phase (*red* $>$ *yellow* $>$ *green*). Trend lines are fitted per phase.

Weather effects.

Fig. 2 compares TCI across weather categories. *Rainy* and *foggy* conditions exhibit higher medians and heavier upper tails than *sunny*, consistent with reduced visibility and cautious driving lowering effective capacity. *Windy* sits between benign and adverse regimes. These patterns motivate small, categorical weather penalties rather than a complex continuous mapping.

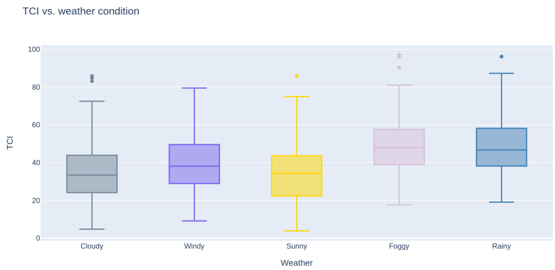

Figure 2: TCI distribution by weather. Adverse conditions (rain, fog) shift TCI upward; sunny days are lowest.

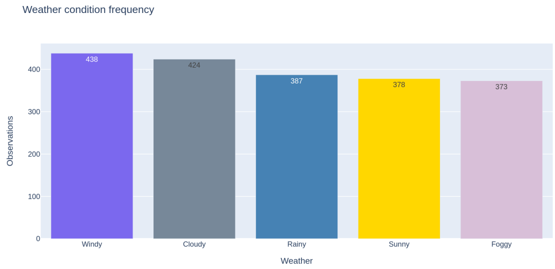

Figure 3: Weather condition frequency. Balanced coverage enables stable estimation of simple per-category effects.

Coverage and identifiability.

The dataset contains substantial support for each weather regime (Fig. 3), which helps identifiability of simple categorical penalties and reduces the need for high-capacity models at this layer.

Implication for target design.

These EDA findings favor a target that is: (i) *monotone* in core drivers (e.g., higher demand ⇒ higher congestion; higher speed ⇒ lower congestion); (ii) *bounded* for comparability across sites; and (iii) *additive* so discrete factors (incidents, signal, weather) act as interpretable offsets. This leads directly to our *Traffic Congestion Index* (next subsection): a 0–100 composite that combines a normalized demand-to-capacity term, a continuous speed penalty, and small categorical penalties for incidents, signal phase, and adverse weather. The index is simple enough to transfer across heterogeneous intersections while capturing the dominant effects observed above.

2. Traffic Congestion Index (TCI): Definition and Rationale

TCI maps observable contributors to congestion onto a bounded, interpretable 0–100 scale. It is additive in interpretable components and monotone in each contributor. Formally,

$$\text{TCI} \;=\; 100\Big[\; \underbrace{\alpha D}_{\text{demand/capacity}} \;+\; \underbrace{\beta S}_{\text{speed penalty}} \;+\; \underbrace{\gamma_{\text{inc}} I}_{\text{incident}} \;+\; \underbrace{\gamma_{\text{sig}}(c)}_{\text{signal}} \;+\; \underbrace{\gamma_{\text{wx}}(w)}_{\text{weather}} \Big] \tag{1}$$

with:

$$D = \text{clip}\!\left(\frac{\text{cars} + 1.5\,\text{trucks} + 0.3\,\text{bikes}}{\text{capacity}},\; 0,\; 1\right),$$

$$S = \max\!\left(0,\; 1 - \frac{v}{\text{FFS}}\right), \qquad \text{FFS} = \text{free-flow speed},$$

$$I \in \{0,1\} \;\; \text{(incident reported)},$$

$$\gamma_{\text{sig}}(c) = \begin{cases} 0.20 & c = \text{Red}, \\ 0.05 & c = \text{Yellow}, \\ 0 & c = \text{Green}, \end{cases}$$

$$\gamma_{\text{wx}}(w) = \begin{cases} 0.15 & w \in \{\text{Rainy,Foggy,Snowy}\}, \\ 0.05 & w = \text{Windy}, \\ 0 & \text{otherwise}. \end{cases}$$

We set $(\alpha, \beta) = (0.55, 0.30)$ and $(\gamma_{\text{inc}}, \gamma_{\text{sig}}, \gamma_{\text{wx}})$ as above. These values approximately reflect national-level contributions reported by FHWA—bottlenecks/capacity and work zones ($\sim$ 40%+10%), incidents ($\sim$ 25%), weather ($\sim$ 15%), and poor signal timing ($\sim$ 5%)—while remaining simple and bounded for learning.[1]

Design principles.

(1) *Interpretability:* Each term corresponds to a control/operations lever. (2) *Monotonicity and bounds:* $D, S \in [0,1]$ and additive penalties keep TCI in $[0, 100]$. (3) *Cross-site transferability:* coefficients follow national proportions, while capacity and FFS adapt per site. (4) *Practical encodings:* we use categorical penalties for signals and weather to avoid overfitting to rare conditions in short time windows.

Connection to FHWA sources.

Capacity shortfalls and work zones are captured by D; incidents by I; weather by $\gamma_{\text{wx}}(w)$; signal timing by $\gamma_{\text{sig}}(c)$. Special events/other are implicitly absorbed by D and S through demand spikes and speed reductions.

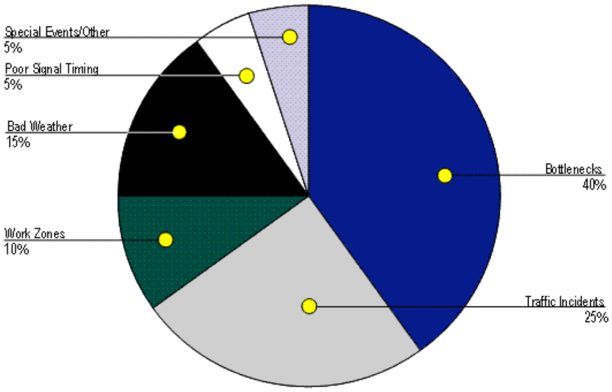

Figure 4: FHWA national breakdown of congestion sources (reproduction).

Sanity checks.

When $v \approx$ FFS, no incident, green signal, and fair weather, TCI $\approx 100\,\alpha D$, behaving like a V/C index. Severe incidents or red phases add interpretable offsets, aligning with field intuition.

Limitations.

The weights are coarse national priors; corridor-specific calibration could refine them. Also, categorical penalties

[1] FHWA's composite national estimate (Figure ES.4) attributes congestion to Bottlenecks 40%, Incidents 25%, Work Zones 10%, Bad Weather 15%, Poor Signal Timing 5%, Special Events/Other 5%. See *Traffic Congestion and Reliability: Linking Solutions to Problems* (2004).

trade fine detail for robustness over the short, 48-hour span of this dataset.

3. Materials and Methods

3.1. Dataset

We use the *Smart Traffic Management Dataset*[2] comprising 48 hours of sensor data from multiple U.S. intersections (speeds, counts by vehicle type, weather, signal phase, incident flag). We construct five client sites (site-1…site-5) by stratified partitioning of locations. Timestamps are parsed to UTC, duplicates dropped, and obvious impossible values filtered (e.g., negative speeds).

3.2. Target: Traffic Congestion Index (TCI)

From raw variables we derive an interpretable 0–100 *Traffic Congestion Index* (TCI),

$$\text{TCI} = 100\Big[\alpha D + \beta S + \gamma_{\text{inc}} I + \gamma_{\text{sig}}(c) + \gamma_{\text{wx}}(w)\Big],$$

where D measures demand-to-capacity, S penalizes low speed ($S = \max(0, 1 - v/\text{FFS})$), $I \in \{0, 1\}$ flags incidents, and $\gamma_{\text{sig}}(\cdot)$, $\gamma_{\text{wx}}(\cdot)$ are categorical penalties for signal phase and weather. We use $(\alpha, \beta) = (0.55, 0.30)$ and national-level priors for $(\gamma_{\text{inc}}, \gamma_{\text{sig}}, \gamma_{\text{wx}})$, keeping TCI simple, bounded, and additive. Design principles: (i) interpretability; (ii) monotonicity and bounds; (iii) cross-site transferability; (iv) practical encodings to avoid overfitting rare conditions.

3.3. Feature engineering

We standardize continuous predictors per site and add time-series structure:

- **Lags:** $\text{TCI}_{t-\ell}$ for $\ell \in \{1, 3, 6, 12, 24\}$ hours.
- **Rolling stats:** mean & std of TCI over 3, 6, 24 hours (shifted by 1 to prevent leakage).
- **Cyclical time:** $\sin(2\pi\text{hour}/24)$, $\cos(2\pi\text{hour}/24)$.

3.4. Models

3.4.1. Simple GNN forecaster (smoothing + MLP)

We build a sparse adjacency over locations (self-loops included) and perform K-hop feature propagation

$$H^{(k+1)} = \sigma\big(\tilde{A} H^{(k)} W^{(k)}\big), \qquad \tilde{A} = \hat{D}^{-1/2}(\,A + I\,)\hat{D}^{-1/2},$$

starting from $H^{(0)} = X_t$ (current features). We concatenate $\{H^{(1)}, H^{(2)}\}$ and apply a small two-layer MLP head to predict next-hour TCI. This "diffusion + linear head" is intentionally lightweight and stable across heterogeneous sites.

3.4.2. Bi-LSTM with self-attention (sequence model)

Sequences use a 24-step window and 1-hour horizon. The network stacks LSTM(128) → Dropout(0.2) → LSTM(64) (both with `return_sequences`), followed by 4-head self-attention, global average pooling, Dense(32, ReLU), and a linear output for TCI. Loss is MSE; we report MAE/RMSE/MAPE.

3.5. Baselines

We compare against three non-FL tabular baselines on identical train/test splits: Linear Regression, Random Forest, and Gradient Boosting (scikit-learn defaults, minor tuning only). These use the same engineered features but no temporal sequence modeling.

3.6. Federated training and evaluation

We train both GNN and RNN with FedAvg (NVFLARE) for 100 rounds over five clients. Each site maintains a fixed *train/val/test* split; the **test split is never used for updates**. At the end of every round the global model is broadcast and evaluated on each site's immutable test set, producing blind test learning curves. We aggregate per-site and cross-site metrics (MAE, RMSE, MAPE).

3.7. Metrics and reporting

Primary metric is MAPE (↓); we also report MAE and RMSE. For round-wise plots we show the mean across sites with shaded 95% CI. All figures/tables are generated by the provided scripts and saved to `paper_assets/`.

3.8. Implementation details

Training uses Adam (3×10^{-4}), batch size 128 for RNN, early stopping (patience=5) and reduced LR on plateau. GNN uses two propagation steps and a small MLP head (hidden 64). Random seeds are fixed and code paths are deterministic where possible. Reproducibility assets are included below.

[2] Kaggle link in the repo README.

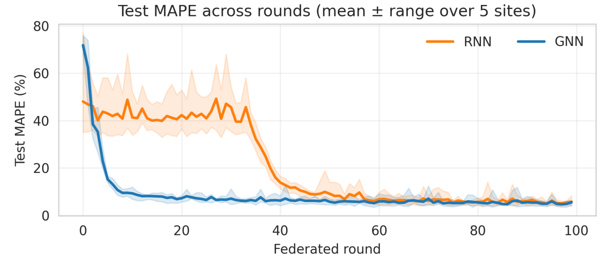

Figure 5: Blind test **MAPE** vs. federated round (mean ±95% CI over five sites) for RNN and GNN. Used here to document the evaluation protocol and convergence behaviour.

4. Results

4.1. Overall accuracy (blind test)

Table 1 reports cross-site mean±sd on the immutable test split. Both FL models achieve single-digit MAPE; the simple GNN is slightly lower and more stable than the attention RNN.

Table 1: Cross-site mean±sd on blind test (lower is better).

Model	MAE (↓)	RMSE (↓)	MAPE (↓)
GNN	1.900 ± 0.420	2.479 ± 0.564	5.483 ± 1.063
RNN	2.170 ± 0.918	2.908 ± 1.174	5.742 ± 1.672

4.2. Per-site accuracy

Per-site final MAPE is visualized in Fig. 7. Sites show consistent ordering between models; the GNN tends to edge the RNN on the most volatile site.

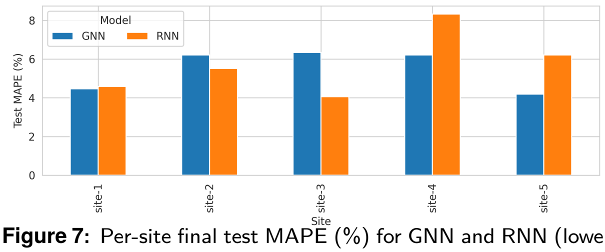

Figure 7: Per-site final test MAPE (%) for GNN and RNN (lower is better).

4.3. Comparison with non-FL baselines

Fig. 8 compares our FL models (mean±95% CI) to non-FL baselines trained on the same split. Both FL models outperform classic tabular baselines by a wide margin on all three metrics, while remaining lightweight and deployable.

4.4. Key findings

- **Accuracy:** GNN achieves the best average error (MAPE 5.48%), with RNN close behind (5.74%).
- **Convergence:** Both models reach stable performance within ≈50 rounds (see Methods, Fig. 5).
- **Consistency:** Per-site bars show low variance; hardest sites remain hardest across models, indicating genuine site difficulty rather than model instability.
- **Baselines:** FL models substantially beat non-FL tabular baselines (linear, RF, GB), demonstrating the value of temporal and spatial structure with privacy-preserving FL.

5. Discussion

5.1. What the results mean

Both federated models achieve single-digit error on an immutable test split across heterogeneous sites, indicating that the derived TCI target is learnable with modest model capacity and that FedAvg is sufficient to align client updates. The simple GNN slightly outperforms the attention RNN on mean MAPE (Table 1) and shows narrower spread across sites (Fig. 7), suggesting that light spatial smoothing plus a compact MLP head is a strong inductive bias for this dataset. Convergence within ∼50 rounds (Fig. 5) implies practical training times for periodic retraining in deployment.

5.2. Why the GNN edged the RNN

The GNN's advantage likely stems from two factors:

1. **Spatial bias:** Even a simple $\tilde{A}$-based propagation acts as a low-pass spatial filter, denoising volatile local signals before regression. This helps most on the hardest site(s), where we observe the largest RNN→GNN gap.
2. **Capacity/regularization balance:** The RNN's stacked LSTMs + attention can overfit short windows if local dynamics are idiosyncratic; the GNN's head is smaller and benefits from implicit smoothing.

5.3. Why the RNN is still competitive

The RNN's performance (MAPE ≈5.7%) indicates that temporal recurrence plus learned attention captures useful within-site dynamics (e.g., signal phase and weather shifts). In practice, the RNN may better adapt to regimes with less spatial coherence or where adjacency is uncertain.

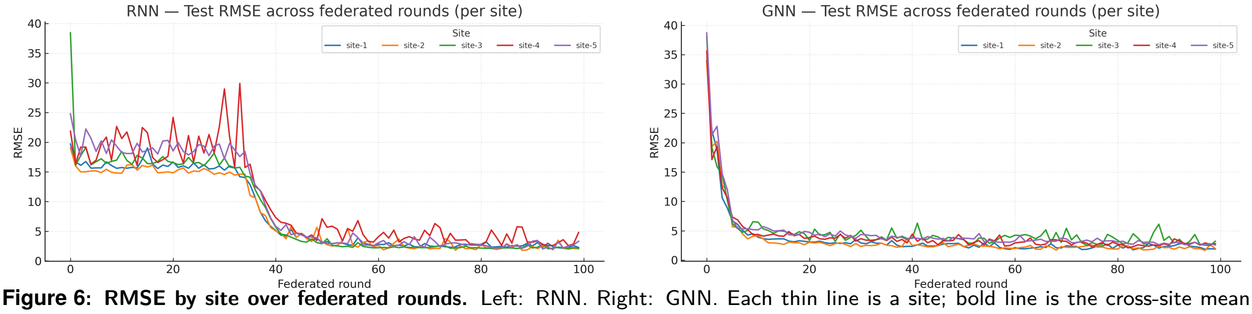

Figure 6: RMSE by site over federated rounds. Left: RNN. Right: GNN. Each thin line is a site; bold line is the cross-site mean with 95% CI shading.

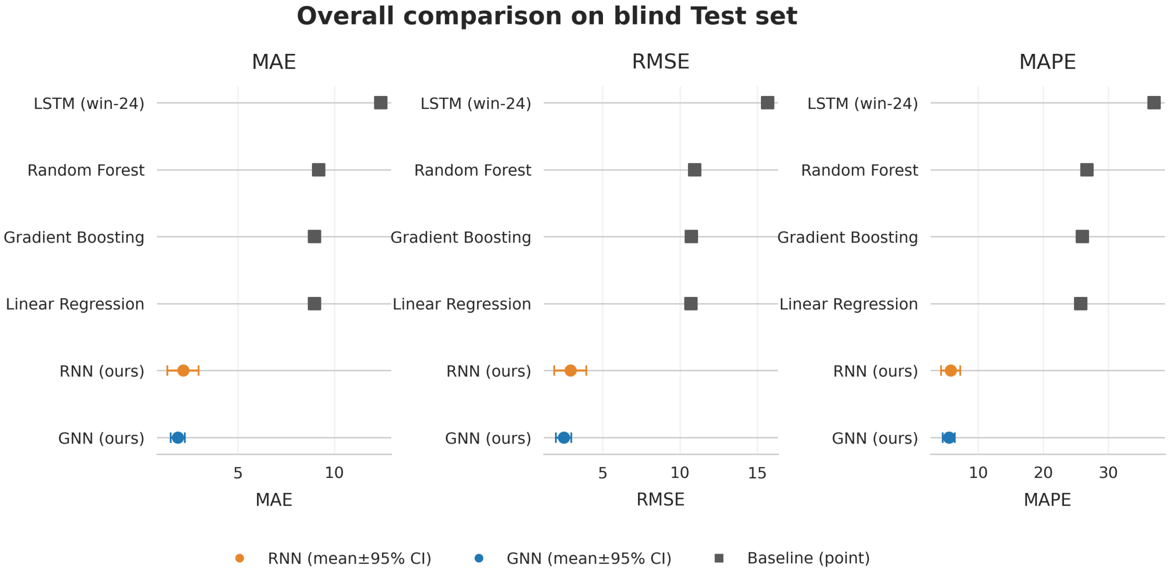

Figure 8: Overall comparison on the blind test set. Circles: FL models (mean±95% CI across sites). Squares: baseline point estimates (Linear Regression, Random Forest, Gradient Boosting).

5.4. Interpretability and TCI design

The bounded, additive TCI made it straightforward to reason about errors and to communicate findings to non-ML stakeholders. Monotonic components (demand/capacity, speed penalty, incident, signal, weather) help ensure that improvements in underlying conditions translate predictably into lower TCI. This choice also reduces label noise compared to raw traffic volume proxies.

5.5. Robustness and generalization

Stable ranking across sites (Fig. 7) suggests genuine site difficulty rather than model instability. Round-wise blind evaluation (Fig. 5) further reduces the chance that observed gains are due to early stopping on validation alone; models are never updated on test data at any round.

5.6. Computational footprint

Both models are intentionally lightweight. The GNN uses two propagation steps and a small MLP head (hidden 64), keeping communication payloads and on-device training times modest. The RNN trains with batch size 128 and early stopping, converging in tens of rounds. This balance supports periodic federated refreshes under realistic bandwidth constraints.

5.7. Limitations

- **Adjacency construction:** We rely on a fixed, heuristic graph over locations. If true inter-location influence differs from this prior, GNN performance could degrade.
- **Short horizon & windowing:** A 24-step window with 1-hour horizon fits the current data cadence;

longer horizons or sparser sensing might require different architectures or features.

- **Label assumptions:** TCI weights (e.g., weather/signal penalties) embed domain priors. While bounded and interpretable, they may require calibration for other cities or seasons.

5.8. Threats to validity

- **Data shift:** Sites may drift (seasonality, work zones). Although the blind test split reduces leakage, future performance depends on re-training cadence.
- **Client heterogeneity:** Differences in sensor quality and missingness can bias updates. FedAvg is robust but not optimal for highly non-IID settings.
- **External validity:** Results are from a specific dataset and two days of observations; generalization to longer horizons and different cities should be empirically verified.

5.9. Practical implications

For near-term deployment, the GNN offers a strong default: slightly better accuracy, tighter variance, and minimal tuning. The RNN remains attractive when adjacency is unavailable/unreliable or when capturing longer temporal motifs is critical.

5.10. Future work (low-risk, high-yield)

We will (i) learn or adapt the adjacency from data while preserving sparsity; (ii) calibrate TCI penalty weights using limited ground-truth queue or delay samples; and (iii) evaluate FedProx or server-side regularization to better accommodate non-IID clients—all without expanding model complexity.

6. Background and Related Work

6.1. Congestion measures and reliability

Transportation agencies operationalize congestion with speed- and time-index measures that capture both intensity and reliability (e.g., *Travel Time Index*, *Planning Time Index*). See FHWA [1] and the updated executive summary in FHWA [2] for definitions and nationwide evidence. These motivate our bounded, additive TCI that combines demand/capacity, speed shortfall, incident, signal, and weather components to reflect both recurrent and non-recurrent congestion in a way that remains interpretable across sites.

6.2. Spatiotemporal deep learning for traffic

Graph-based neural models encode spatial dependencies among sensors/links alongside temporal dynamics. Li et al. [4] (ICLR 2018) introduced DCRNN, modeling traffic as a diffusion process on directed graphs and coupling diffusion convolutions with recurrent encoders/decoders. Wu et al. [5] (IJCAI 2019) proposed Graph WaveNet with dilated temporal convolutions and a learnable adaptive adjacency to discover hidden spatial links. In contrast, Wu et al. [6] (ICML 2019) showed that removing intermediate nonlinearities yields a fixed low-pass graph filter followed by a linear head—often preserving accuracy while improving stability and efficiency. Our lightweight GNN follows this “propagate then regress” philosophy: a few hops of feature smoothing with a compact MLP head.

6.3. Sequence models (non-graph baselines)

Recurrent architectures remain competitive for short-horizon forecasting when spatial structure is weak or uncertain. We therefore include a Bi-LSTM with self-attention as a deliberately non-graph comparator to test whether purely temporal inductive bias can approach graph-aware accuracy on our TCI target.

6.4. Federated learning for privacy-preserving training

We adopt Federated Averaging, McMahan et al. [3] (AISTATS 2017), to aggregate local updates without centralizing data—appropriate for non-IID, unbalanced multi-site sensing. Implementation uses NVIDIA FLARE (docs: NVIDIA [7]), which lets us simulate five clients and log per-round, blind test metrics consistently across sites.

7. Conclusions

We tackled short-horizon traffic congestion forecasting in a privacy-preserving, multi-site setting. To make the target interpretable and transferable, we derived a bounded $[0, 100]$ *Traffic Congestion Index* (TCI) that aggregates demand/capacity, speed shortfall, incident, signal phase, and weather penalties. On top of this target, we built a lightweight federated pipeline (NVFLARE + FedAvg) and compared two compact deep models with distinct inductive biases: a simple graph smoother + MLP (GNN) and a Bi-LSTM with self-attention (RNN).

Accuracy and stability. Both FL models achieve single-digit error on immutable test splits across five clients. The GNN yields the best average error (e.g., MAPE $\approx$5.48%) with narrower cross-site spread, while the RNN is close behind (MAPE $\approx$5.74%). Blind per-round evaluation shows rapid convergence (about 40–50 rounds) and consistent ranking across sites. Both FL models substantially outperform tabular non-FL baselines on MAE/RMSE/MAPE.

Why it works. The GNN's few-hop propagation acts as a low-pass spatial filter that denoises volatile local signals before regression, which appears especially beneficial on the hardest site. The RNN captures temporal motifs and remains competitive where spatial coherence is weaker or adjacency is uncertain. The bounded, additive TCI further simplifies training and interpretation by enforcing monotonic effects and limiting label noise.

Practical takeaways.

- Use the **GNN** by default when you can define a reasonable location graph; it is slightly more accurate and more stable across sites while remaining small.
- Use the **RNN** when adjacency is unreliable or unavailable; it is nearly as accurate and easy to deploy.
- Short, federated training runs are feasible: $\sim$50 rounds sufficed for convergence in our setting.

Limitations. Our adjacency is heuristic; longer horizons and larger networks may require learned or adaptive graphs. The dataset covers a short temporal span, and TCI weights may need calibration for other cities/seasons. Although we evaluate on a blind, immutable test split each round (no gradient updates on test), external validity under strong distribution shift still requires longitudinal studies.

Outlook. Immediate extensions include (i) sparse, data-driven refinement of the adjacency while preserving interpretability; (ii) light calibration of TCI penalties using limited ground-truth delay/queue samples; and (iii) mild server-side regularization (e.g., FedProx-style) to further hedge non-IID clients—each keeping model complexity low. We release code, scripts that regenerate all figures/tables, and a consolidated `paper_assets/` bundle to support full reproducibility.

8. Acknowledgments

The authors gratefully acknowledge the support of Wilfrid Laurier University and the Department of Physics and Computer Science. We thank Dr. Sukhjit Singh Sehra for supervision, guidance, and constructive feedback throughout this project, and fellow research student Seerat Kaur for valuable discussions and assistance during experimentation. Any opinions, findings, and conclusions expressed in this work are those of the authors and do not necessarily reflect the views of the aforementioned institutions or individuals.

References

[1] Federal Highway Administration (FHWA). *Traffic Congestion and Reliability: Linking Solutions to Problems — Executive Summary*. U.S. Dept. of Transportation, 2004. Available at: https://rosap.ntl.bts.gov/purl/40572.

[2] FHWA Office of Operations. *Traffic Congestion and Reliability: Trends and Advanced Strategies for Congestion Mitigation (Executive Summary)*. 2020. Available at: https://ops.fhwa.dot.gov/congestion_report/executive_summary.htm.

[3] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proc. AISTATS*, PMLR Vol. 54, pp. 1273–1282, 2017. Available at: https://proceedings.mlr.press/v54/mcmahan17a.html.

[4] Y. Li, R. Yu, C. Shahabi, and Y. Liu. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. *International Conference on Learning Representations (ICLR)*, 2018. Preprint: arXiv:1707.01926. Available at: https://arxiv.org/abs/1707.01926.

[5] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang. Graph WaveNet for Deep Spatial–Temporal Graph Modeling. In *Proc. IJCAI*, pp. 1907–1913, 2019. doi:10.24963/ijcai.2019/264. Available at: https://www.ijcai.org/proceedings/2019/264.

[6] F. Wu, T. Zhang, A. H. de Souza Jr., C. Fifty, T. Yu, and K. Q. Weinberger. Simplifying Graph Convolutional Networks. In *Proc. ICML*, PMLR Vol. 97, pp. 6861–6871, 2019. Available at: https://proceedings.mlr.press/v97/wu19e.html.

[7] NVIDIA. *NVIDIA FLARE Documentation*. 2023. Available at: https://nvflare.readthedocs.io/.