Project Title: Federated GNNs Learning from Dynamic Graphs for Traffic Forecasting.

by Muhammad Usman

This is the updated and continued form of our previous report. Last portion of the technical section contains the updated information.

Introduction:

This project, conducting by Muhammad Usman, focuses on the critical role of traffic forecasting in optimizing transportation systems and urban planning. Traffic forecasting is a critical aspect of urban planning and transportation management, with the growing complexity of urban environments necessitating advanced predictive models. Traditional methods often fall short in capturing the intricate dynamics of traffic patterns, especially in scenarios where the underlying graph structures evolve over time. This research addresses the limitations of existing approaches by proposing a novel solution that combines Federated Learning (FL) and Graph Neural Networks (GNN) to enhance traffic forecasting accuracy. By leveraging federated learning, which enables model training across decentralized data sources without sharing raw data, and incorporating dynamic graph structures into the GNN framework, our approach aims to provide a more robust and adaptable solution for predicting traffic flow in dynamic urban environments.

Objectives:

The primary objective of this project is to develop an advanced algorithm using Federated Graph Neural Networks (GNNs) to predict traffic patterns in scenarios characterized by evolving network structures

Challenges:

Training distributed dynamic GNNs poses several challenges, including massive parameter communication, model convergence issues, and workload imbalance among workers. Addressing these challenges is crucial for achieving efficient and accurate traffic forecasting.

Data Sources:

The project utilizes openly available dynamic graph benchmark datasets, including METR_LA, PEMS_D7M, EXPY-TKY and PEMS_Bay, as primary data sources.

Methodologies:

The project's approach centers on Federated Graph Neural Networks (GNNs) as the core machine learning framework for traffic forecasting. The methodology comprises data collection from real-world traffic historical speed data, dynamic graph embedding, unsupervised data partitioning, the implementation of a standalone Federated Learning Framework, and model evaluation using metrics such as MAE, RMSE, and MAPE.

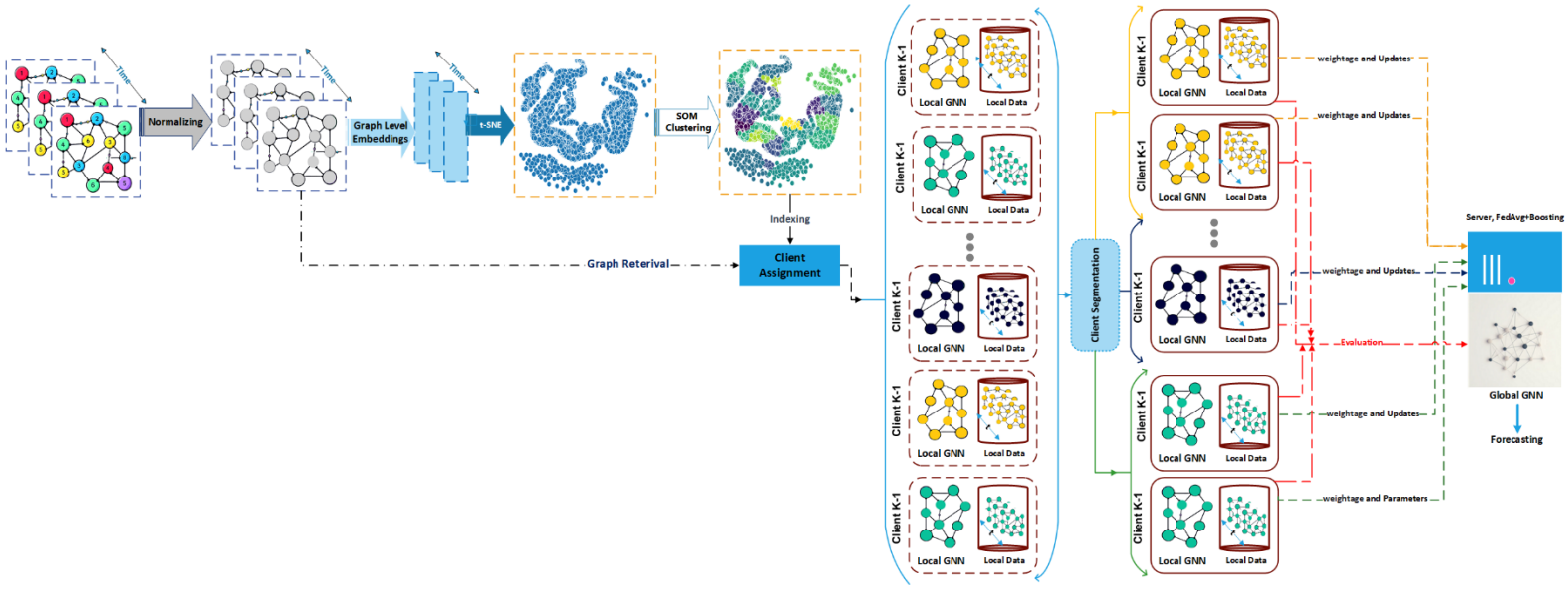
Timeline:

The project is expected to span a duration of 4-6 months to complete the coding tasks and achieve the defined objectives.

Project Summarization.

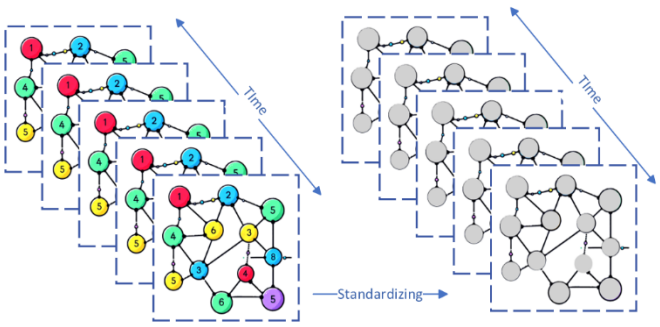
In “Federated GNNs Learning from Dynamic Graphs for Traffic Forecasting" project the focus is on enhancing traffic speed forecasting in the context of complex and ever-changing transportation networks. The project's objectives include developing an advanced methodology that utilizes Federated Graph Neural Networks (GNNs) to predict traffic patterns. Challenges related to distributed GNN training are addressed, and openly available benchmark datasets like METR_LA and PEMS_Bay are used as primary data sources. The proposed methodology encompasses data collection, dynamic graph embedding, unsupervised data partitioning, grouping relevant clients based on their previous performance and its data distribution, the implementation of a standalone Federated Learning algorithm, and model evaluation. This project is anticipated to span 4-6 months, aiming to provide valuable insights into traffic forecasting techniques using state-of-the-art machine learning approaches.

Technical Report



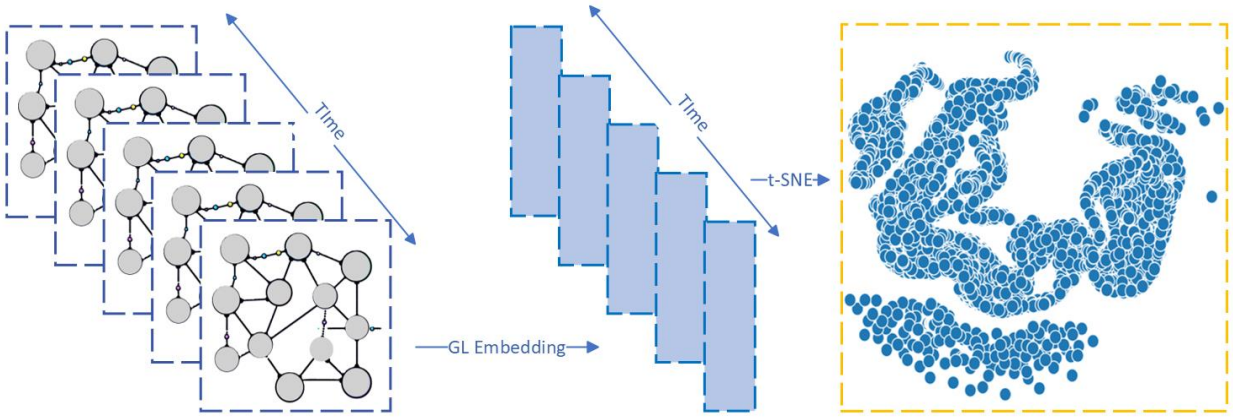
1. Graph Standardization and Normalization:

We first standardize all the graph features (X) by zero mean and unit standard deviation. Afterward, the overall features are further normalized using the maximum value of the entire dataset.



2. Graph-Level Embedding with A3TGCN:

We then convert all the standardized graphs into Graph Level Embeddings for low-level vector representation. The embeddings of all the graphs have been plotted and are displayed in the following image.



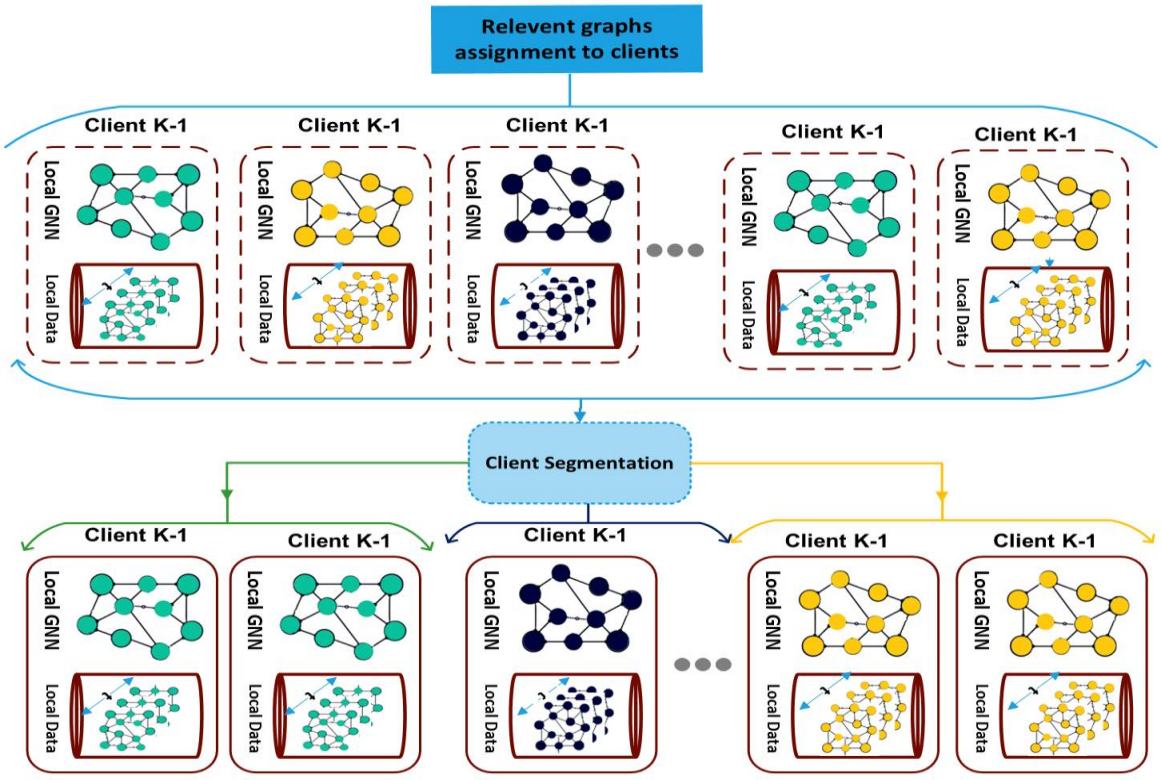
3. Unsupervised Partitioning of Participant Datasets Based on Similar Traffic Patterns:

We then employ unsupervised learning to generate clusters of embeddings, assigning each cluster to a client. Consequently, all graph embeddings within a particular cluster serve as an index for retrieving graphs from the primary dataset and allocating them to the respective client. The detailed results of graph-level embedding clustering using the Self-Organizing Map (SOM) method are presented for clarity.

Each color on the map corresponds to a cluster, with the image showcasing nine distinct colors, indicating nine clusters. Given the temporal nature of the graphs, identical patterns occurring at specific timestamps and belonging to the same cluster are color-coded consistently. The recurrence of identical colors at different locations on the map denotes the association of dynamic patterns with a specific cluster.

4. Client Grouping/Segmentation by its Data Distribution and Historical Performance

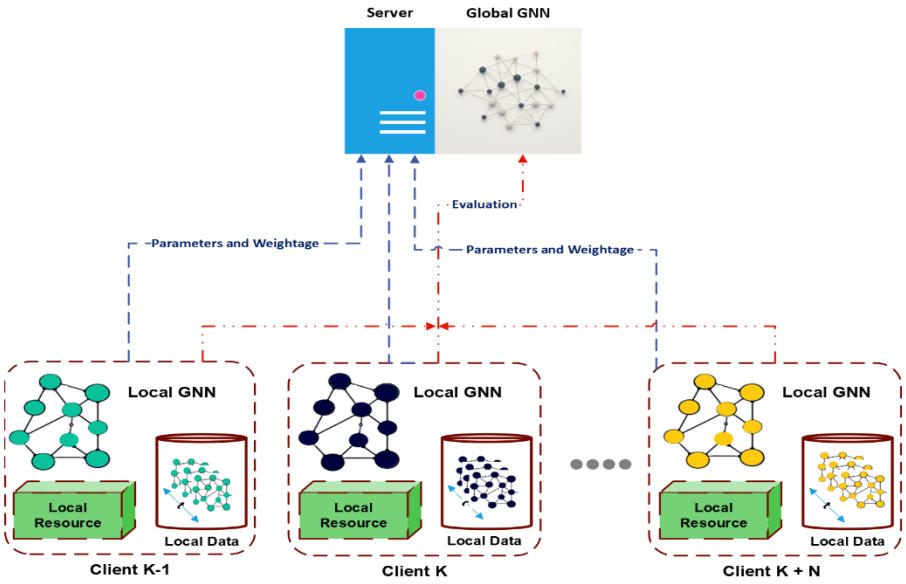
After the initial communication round between the server and clients, the server categorizes clients based on their past performances and data distribution. Fourteen statistical features are extracted from each client, and a k-means algorithm is employed to assign labels, grouping clients with similar characteristics. The optimal number of clusters is determined using the Elbow method. An illustrative example indicates that setting "K=2" groups may yield improved results. A snapshot of the output is also included for better comprehension.



5. Federated or Distributed Client-Based Global GNN Model Training.

Finally, we trained a federated global GNN model with the assistance of 9 local clients i.e., participants using the Metr_LA and PEMS_Bay datasets. Initially, the dataset was partitioned among these 9 clients based on their shared characteristics. Each client possessed its own local dynamic graph dataset and utilized our developed algorithm, DFDG, which is based on A3TGCN+ Neural ODE + Linear for modeling. Clients trained their models locally and periodically updated the server (global model) with their local parameters as well as their weightage(importance). The server, in turn, incorporated these updates from each client. Throughout this project, all clients employed the DFDG based topology.

The server, in response, integrated these updates from each client using the FedAvg mechanism to aggregate parameters from the client groups. The ensuing figures illustrate the algorithm's performance based on unseen data, showcasing 15- and 30-minute forecasting results using METR_LA and PEMSBAY datasets.



6. Utilizing Adaptive Boosting for Participant Importance Weighting.

Ada boosting in federated learning refers to the process of enhancing or optimizing the training process to improve model performance using Weighted Updates. Assign different weights to clients' updates based on their performance. This allows us to give more importance to clients that provide better results and reduce the influence of clients with noisy or unreliable data and results.

1. **For each communication rounds (t = 1, 2, ..., K)**:

   a. **Train a client**:

   - Train a participant DFDG model using the current sample weights. The client model should aim to minimize the local error.

   b. **Calculate the Weighted Error**

   - Calculate the weighted error of the participant model over the training set.

$$w_i^{(t+1)} = w_i^{(t)} \cdot \exp\left(-\alpha_t \cdot y_i \cdot h_t(x_i)\right)$$

   c. **Compute the Client weight**:

   - Compute the weight of the participant model in the final model:

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$$

d. **Update Client Weights before FedAvg at server side.**

- Update the sample weights for the next iteration:

$$\epsilon_t = \sum_{i=1}^{N} w_i^{(t)} \cdot 1(h_t(x_i) \neq y_i)$$

7. Evaluation and validation:

The Performance of algorithm is evaluated on PEMS_BAY and METR-LA dataset. Two or Three more datasets will be added to check the performance.

a. METR-LA Performance:

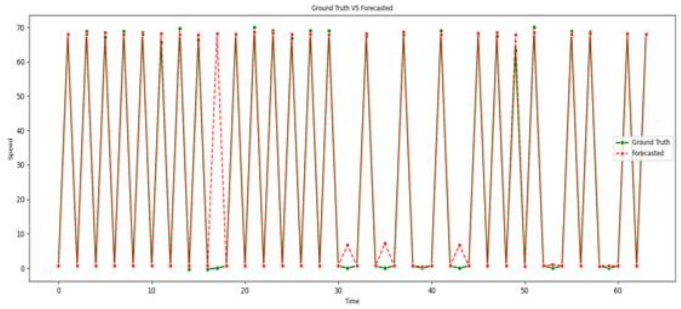
METR-LA		15min/horizon3			30min/horizon6			60min/horizon12	
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
HA(Lietal.2018)	4.16	7.80	13.00%	4.16	7.80	13.00%	4.16	7.80	13.00%
STGCN(Yu,Yin,andZhu2018)	2.88	5.74	7.62%	3.47	7.24	9.57%	4.59	9.40	12.70%
DCRNN(Lietal.2018)	2.77	5.38	7.30%	3.15	6.45	8.80%	3.60	7.59	10.50%
GW-Net(Wuetal.2019)	2.69	5.15	6.90%	3.07	6.22	8.37%	3.53	7.37	10.01%
STTN(Xuetal.2020)	2.79	5.48	7.19%	3.16	6.50	8.53%	3.60	7.60	10.16%
GMAN(Zhengetal.2020)?	2.80	5.55	7.41%	3.12	6.49	8.73%	3.44	7.35	10.07%
MTGNN(Wuetal.2020)	2.69	5.18	6.86%	3.05	6.17	8.19%	3.49	7.23	9.87%
StemGNN(Caoetal.2020)†	2.56	5.06	6.46%	3.01	6.03	8.23%	3.43	7.23	9.85%
AGCRN(Baietal.2020)	2.86	5.55	7.55%	3.25	6.57	8.99%	3.68	7.56	10.46%
CCRNN(Yeetal.2021)	2.85	5.54	7.50%	3.24	6.54	8.90%	3.73	7.65	10.59%
GTS(Shang,Chen,andBi2021)?	2.65	5.20	6.80%	3.05	6.22	8.28%	3.47	7.29	9.83%
PM-MemNet(Leeetal.2022)	2.65	5.29	7.01%	3.03	6.29	8.42%	3.46	7.29	9.97%
MegaCRN(AAAI-23)	2.52	4.94	6.44%	2.93	6.06	7.96%	3.38	7.23	9.72%
Latest Ours	1.83	5.59	6.96%	2.02	6.21	7.65%	2.52	7.60	9.43%

b. PEMS_Bay Performance:

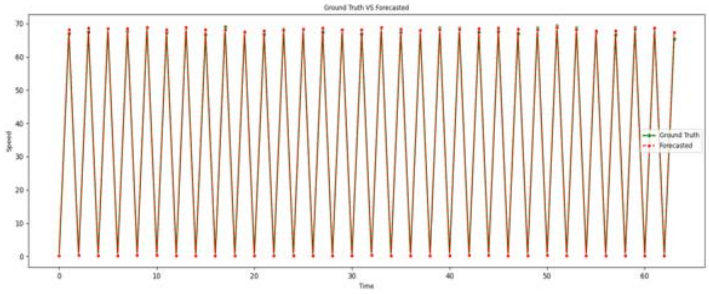
PEMS-BAY		15min/horizon3			30min/horizon6			60min/horizon12	
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
HA(Lietal.2018)	2.88	5.59	6.80%	2.88	5.59	6.80%	2.88	5.59	6.80%
STGCN(Yu,Yin,andZhu2018)	1.36	2.96	2.90%	1.81	4.27	4.17%	2.49	5.69	5.79%
DCRNN(Lietal.2018)	1.38	2.95	2.90%	1.74	3.97	3.90%	2.07	4.74	4.90%
GW-Net(Wuetal.2019)	1.30	2.74	2.73%	1.63	3.70	3.67%	1.95	4.52	4.63%
STTN(Xuetal.2020)	1.36	2.87	2.89%	1.67	3.79	3.78%	1.95	4.50	4.58%
GMAN(Zhengetal.2020)?	1.35	2.90	2.87%	1.65	3.82	3.74%	1.92	4.49	4.52%
MTGNN(Wuetal.2020)	1.32	2.79	2.77%	1.65	3.74	3.69%	1.94	4.49	4.53%
StemGNN(Caoetal.2020)†	1.23	2.48	2.63%		NA			NA	
AGCRN(Baietal.2020)	1.36	2.88	2.93%	1.69	3.87	3.86%	1.98	4.59	4.63%
CCRNN(Yeetal.2021)	1.38	2.90	2.90%	1.74	3.87	3.90%	2.07	4.65	4.87%
GTS(Shang,Chen,andBi2021)?	1.34	2.84	2.83%	1.67	3.83	3.79%	1.98	4.56	4.59%
PM-MemNet(Leeetal.2022)?	1.34	2.82	2.81%	1.65	3.76	3.71%	1.95	4.49	4.54%
MegaCRN(AAAi-2023)	1.28	2.72	2.67%	1.60	3.68	3.57%	1.88	4.42	4.41%
Latest Ours	0.96	2.38	3.10%	1.04	2.61	3.37%	1.17	3.05	3.75%

Forecasted Results

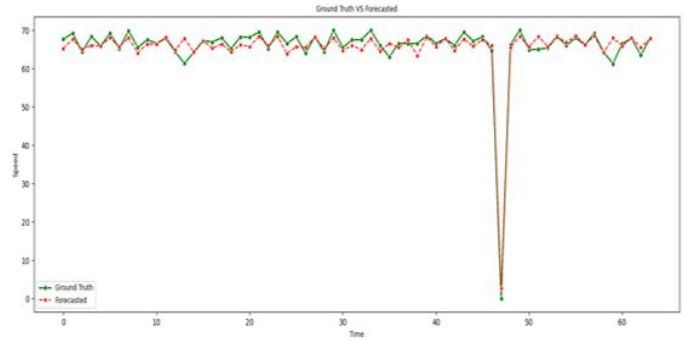
METR LA: 15minutes Forecasting



PEMS_Bay: 15minutes Forecasting



METR LA: 30 minutes Forecasting



PEMS_Bay: 30minutes Forecasting