DIVE: Dataset for Indian Vehicular Traffic Evaluation

Author Name
Affiliation
email@example.com

Abstract

Traffic congestion affects the country's economy directly or indirectly. Traffic congestion also takes people's valuable time, and the cost of fuel every single day is quite high. In order to demonstrate the effectiveness and robustness of the proposed DIVE dataset, different GNNs are trained on DIVE and other state-of-the-art datasets. Finally, the results calculated on each dataset at different time lags are compared. Also, the comparison of the results produced by the models against the state-of-the-art PeMS datasets at time lags of 15, 30, and 45 minutes respectively. The graph models achieve appreciable results for RMSE and MAPE on the DIVE dataset which are comparable to those obtained on PeMSD7 and PeMSD8.

1 Introduction

Mumbai is estimated to consist of around 12 million people. With the city expanding at a fast pace, more buildings rise in the city, and migration steadily increases. The city has an extensive local train system connecting all areas of Mumbai together. Yet, due to the increasing population, trains carry roughly 2.6 times more weight than what they are actually designed for. People hang on the footboard of the train to get to their workplaces. Traffic in this city is quite unmanageable.

Roads like highways, freeways, lanes, etc. connect the city together and are used by people as a means of commuting by car, bus, or auto. Apart from this, the city also has metro lines with three operational lines day and night. Despite many developments, the city has a standing congestion level of 50%. This means that the passengers take 50% more time to travel during heavy congestion than they normally would. A distance of a mere 4 km takes 1 hr in heavy traffic areas, especially during the peak hours.

Mumbai is plagued by traffic issues. It has the world's worst road traffic and is ranked fifth in the most congested city. The traffic conditions have deteriorated over time. As of 2021, the congestion rate was 53% meaning a journey of 30 min would take 20 mins longer during peak hours. Such extreme conditions traffic affect the country in various other ways. For example, Mumbai has an economic loss of 410 billion, due to jams caused along with a 121-hour delay yearly.

Apart from this, Mumbai has a linear geography which restricts the city's growth. With only two major highways, roads become heavily congested during peak hours. Besides, Mumbai has 4.1 million cars which also plays a hand in the huge traffic. Car density in Mumbai has risen to 600 cars per kilometer of the road which is the highest in India so far. Heavy encroachment and bottleneck areas cause traffic as well. Ongoing construction of the metro has also reduced the number of lanes available.

The acceleration of urbanization and the rapid growth of the urban population bring great pressure on urban traffic management. An Intelligent Transportation System (ITS) is an indispensable part of a smart city, and traffic prediction is an important ITS component. Traffic prediction means forecasting the volume and density of traffic flow, usually to manage vehicle movement, reduce congestion, and generate the optimal (least time- or energy-consuming) route. Machine Learning (ML) is one of the most important and popular emerging branches these days as it is a part of Artificial Intelligence (AI). In recent times, machine learning has become essential in the research of transportation engineering, especially in traffic prediction. The depreciation of traffic congestion issues is aided by the optimization of transportation networks which are implemented only after visual or statistical confirmation.

Traffic congestion affects the country's economy directly or indirectly. Traffic congestion also takes people's valuable time, and the cost of fuel every single day is quite high. According to a study, India suffers a huge loss of $21.3 billion annually because of delays and additional fuel consumption due to poor road conditions and frequent halts.

According to the estimates, the cost of delay was $6.6 billion per year and the cost of additional fuel consumption due to delay was $14.7 billion per year according to the study conducted jointly by logistics firm Transport Corporation of India (TCI) and IIM-Kolkata. As traffic congestion is a major problem for all classes in society, there has to be a small-scale traffic prediction for the people's sake for living their lives without frustration and worries. User ease is required in the first place, and this is possible only when the traffic flow is smooth. To deal with this, Traffic prediction is needed so that we can estimate or predict future traffic. In addition to the country's economy, pollution can also be reduced. The plot of this research is to find different machine learning algo-

rithms and speculate the models by utilizing python3. The goal of traffic flow prediction is to predict the traffic to the users as soon as possible. Nowadays the traffic becomes hectic and this cannot be determined by the people when they are on the roads. There are various challenges like External factors. Traffic spatiotemporal sequence data is also influenced by external factors, such as weather conditions, events, or road attributes.

2 Experimentation

In order to demonstrate the effectiveness and robustness of the proposed DIVE dataset, different GNNs are trained on DIVE and other state-of-the-art datasets. Finally, the results calculated on each dataset at different time lags are compared.

2.1

Dataset

Our experiments used two well-known traffic datasets from Caltrans Performance Measurement System: PeMSD7 and PeMSD8 [Roy *et al.*, 2021], along with the proposed DIVE dataset.

- PeMSD7: It is the traffic data in California's District 7 that measures traffic speed using 228 sensors for the period of May to June 2012 (only on weekdays), with a time interval of 5 minutes.
- PEMSD8: This dataset includes data from 170 detectors on 8 highways in San Bernardino, California, collected every five minutes from July through August of 2016.

Models Used

- A3TGCN [Zhu *et al.*, 2020]: Attention Temporal Graph Convolutional Network:A3TGCN stands for "Asymmetric Three-Stream Graph Convolutional Network". It is a type of deep learning architecture for graph-based tasks, such as graph classification and graph node classification.

  The "three-stream" in its name refers to the fact that it uses three different types of neural network layers to process the node features, edge features, and global graph features, respectively. The "asymmetric" in its name refers to the fact that the network uses asymmetric convolutional filters to capture different types of information in the graph structure.

  A3TGCN has shown state-of-the-art performance on various graph-based benchmarks and has been applied in various domains, including chemistry, biology, and social network analysis.
- STGCN [Yu *et al.*, 2018]: Spatio-Temporal Graph Convolutional Network: STGCN stands for "Spatial-Temporal Graph Convolutional Network". It is a type of deep learning architecture for graph-based time series data.

  The "spatial-temporal" in its name refers to the fact that the network considers both the spatial relationships between nodes in the graph and the temporal relationships between time steps. The "graph convolutional network" refers to the use of graph convolutional layers, which are specialized neural network layers that can handle graph-structured data.

  STGCN has been applied to various time series prediction tasks, such as traffic forecasting and air quality prediction, where the graph structure can capture the relationships between different spatial regions or nodes over time. The network has shown promising results and has been widely adopted in various domains.
- DCRNN [Li *et al.*, 2017]: Diffusion Convolutional Recurrent Neural Network: The "diffusion convolutional" in its name refers to the use of a diffusion convolution operation, which can effectively capture the long-range dependencies in the graph structure. The "recurrent neural network" refers to the use of recurrent layers, which can capture the temporal dependencies in the time series data.

  DCRNN has been applied to various graph-based time series prediction tasks, such as traffic forecasting and energy consumption prediction. The network has shown state-of-the-art performance on various benchmark datasets and has been widely adopted in various domains.
- SSTGNN [Roy *et al.*, 2021]: Simplified Spatio-temporal Traffic forecasting model using Graph Neural Network: The "spatial-spectral temporal" in its name refers to the fact that the network considers both the spatial relationships between nodes in the graph and the spectral representations of the graph, as well as the temporal dependencies in the time series data. The "graph neural network" refers to the use of graph neural network layers, which are specialized neural network layers that can handle graph-structured data.

  SSTGNN has been applied to various graph-based time series prediction tasks, such as traffic forecasting and energy consumption prediction. The network has shown promising results and has been widely adopted in various domains.

Evaluation Metrics

Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) are used to evaluate models on the datasets. The presented values are calculated after averaging the values calculated after five experimental rounds.

2.2 Implementation Details

Z-score normalization is applied to data in our experiments and de-normalized the predicted values to calculate the original value. We selected 80% of the data for training while the remaining 20% for testing for the models. SSTGNN and STGCN were trained and implemented using Pytorch while Pytorch Geometric Temporal was used for DCRNN and A3TGCN. We used the Adam optimizer with a learning rate of 0.01 and ran 500 epochs for each model for each dataset.

2.3 Comparison

Tables 1, 2 and, 3 show the comparison of the results produced by the models against the state-of-the-art PeMS

Models	Dataset	MAE	RMSE	MAPE
STGCN	DIVE	0.614	1.404	1.355
	PeMSD7	2.26	4.07	5.24
	PeMSD8	1.19	2.62	2.34
DCRNN	DIVE	0.005	1.677	0.511
	PeMSD7	2.22	4.25	5.16
	PeMSD8	1.49	3.56	3.21
A3TGCN	DIVE	0.002	1.832	0.187
	PeMSD7	2.58	4.52	5.77
	PeMSD8	1.27	2.35	2.31
SSTGNN	DIVE	2.04	3.53	4.77
	PeMSD7	1.03	2.08	1.86
	PeMSD8	0.95	1.76	2.01

Table 1: Results at 15 mins time-lag.

Models	Dataset	MAE	RMSE	MAPE
STGCN	DIVE	1.326	1.991	2.778
	PeMSD7	3.09	5.77	7.39
	PeMSD8	1.59	3.61	3.24
DCRNN	DIVE	0.002	1.701	0.128
	PeMSD7	3.64	7.24	9
	PeMSD8	1.71	4.13	3.83
A3TGCN	DIVE	0.0167	1.833	1.665
	PeMSD7	3.21	5.41	7.22
	PeMSD8	1.67	2.89	2.76
SSTGNN	DIVE	2.67	4.8	6.6
	PeMSD7	1.62	3.28	2.67
	PeMSD8	0.96	1.69	2.03

Table 2: Results at 30 mins time-lag.

Models	Dataset	MAE	RMSE	MAPE
STGCN	DIVE	3.334	1.524	2.594
	PeMSD7	3.79	7.03	9.12
	PeMSD8	1.19	2.62	2.34
DCRNN	DIVE	0.005	1.677	0.511
	PeMSD7	2.22	4.25	5.16
	PeMSD8	1.17	2.59	2.32
A3TGCN	DIVE	0.011	1.83	0.959
	PeMSD7	2.58	4.52	5.77
	PeMSD8	3.34	5.11	4.72
SSTGNN	DIVE	3.17	5.79	8
	PeMSD7	1.03	2.08	1.86
	PeMSD8	0.91	1.69	1.94

Table 3: Results at 45 mins time-lag.

datasets at time lags of 15, 30, and 45 minutes respectively. The graph models achieve appreciable results for RMSE and MAPE on the DIVE dataset which are comparable to those obtained on PeMSD7 and PeMSD8. The models produced very low MAE values (0.001) on the DIVE dataset. The possible reason for such low values might be the low number of training data (1345 samples). Thus, improvements can be made by collecting and training with more data samples. The test results achieved by the GNN models for an individual sensor are visualized in Fig. 1, 2, and 3 for 15, 30 and 45 minutes respectively.

Acknowledgments

We would like to express our sincere gratitude to iHub-Data Mobility, our mentor, and our college faculty for their invaluable support of this research. We would also like to acknowledge the guidance and insights shared especially by the iHub-Data Mobility team, which helped to shape the direction and outcome of our work. We believe that this collaboration with iHub Data Mobility marks a significant milestone in our research journey, and it will serve as a foundation for further exploration and discovery in the field. We hope that this research will bring about meaningful and positive impacts for the community and industry, and contribute towards the continued growth and advancement of data mobility.

References

[Li *et al.*, 2017] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting, 2017.

[Roy *et al.*, 2021] Amit Roy, Kashob Kumar Roy, Amin Ahsan Ali, M. Ashraful Amin, and A. K. M. Mahbubur Rahman. SST-GNN: Simplified spatio-temporal traffic forecasting model using graph neural network. In *Advances in Knowledge Discovery and Data Mining*, pages 90–102. Springer International Publishing, 2021.

[Yu *et al.*, 2018] Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, jul 2018.

[Zhu *et al.*, 2020] Jiawei Zhu, Yujiao Song, Ling Zhao, and Haifeng Li. A3t-gcn: Attention temporal graph convolutional network for traffic forecasting, 2020.

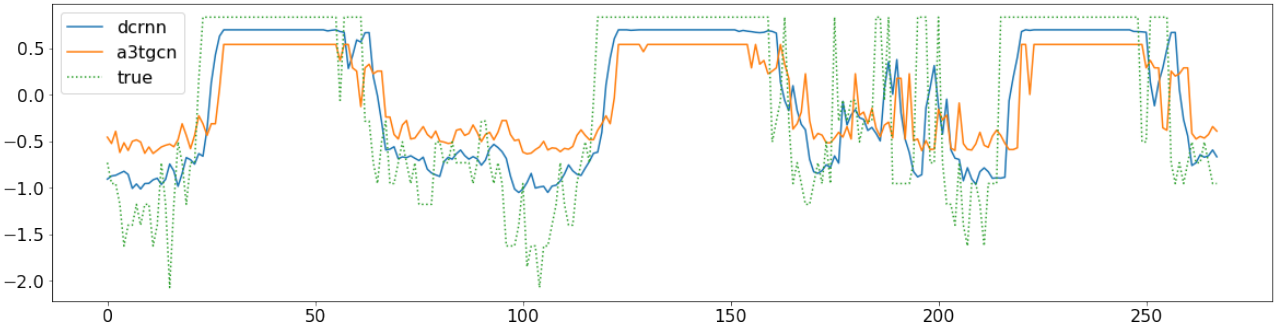

Figure 1: True Vs. Predicted Speeds at 15 mins time lag.

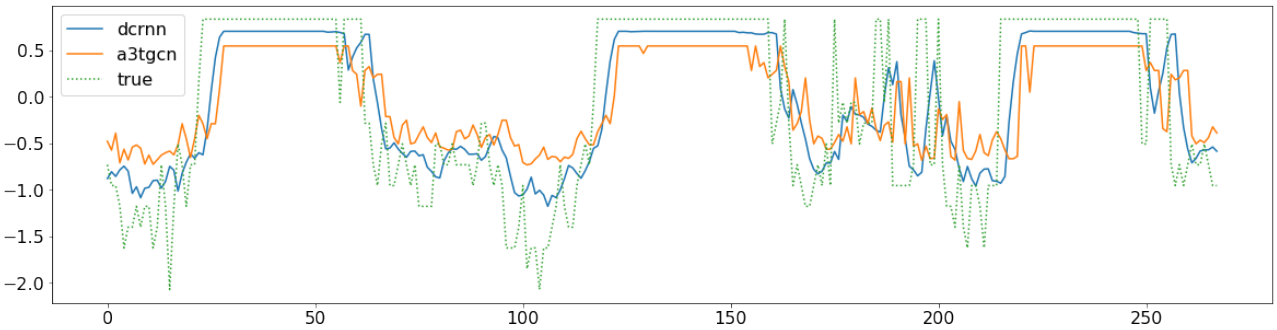

Figure 2: True Vs. Predicted Speeds at 30 mins time lag.

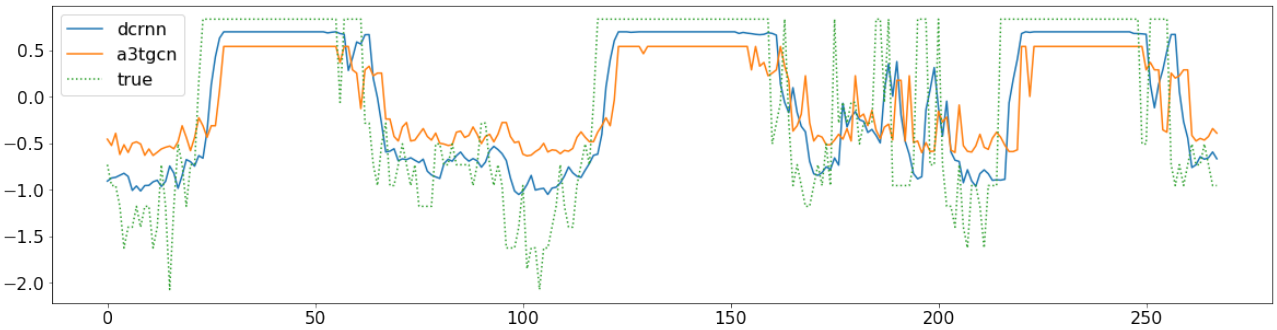

Figure 3: True Vs. Predicted Speeds at 45 mins time lag.