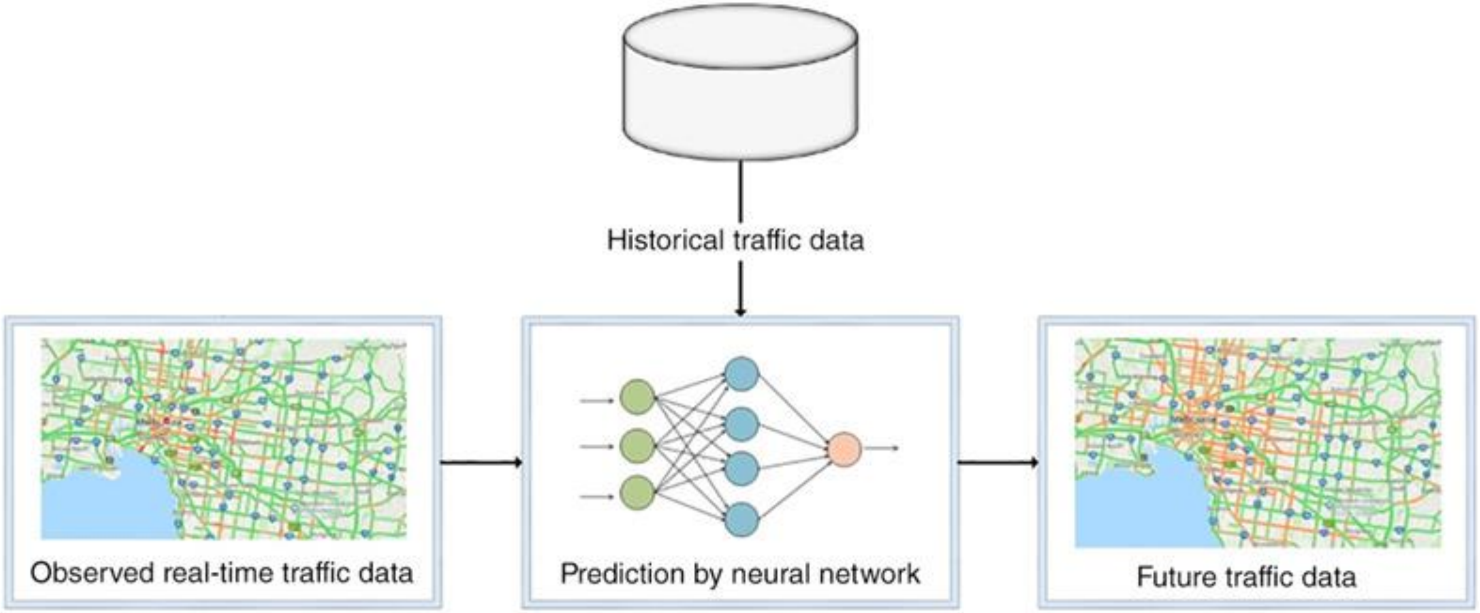
Dynamic Spatial-Temporal Graph Convolutional Neural Networks for Traffic Forecasting

Zulong Diao,[1] Xin Wang,[2] Dafang Zhang,[1*] Yingru Liu,[2] Kun Xie,[1] Shaoyao He[3]
[1]The College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China
[2]The Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY, 11794
[3]The College of Architecture, Hunan University, Changsha 410082
{zldiao, dfzhang, xiekun}@hnu.edu.cn, {x.wang, yingru.liu}@stonybrook.edu, syhe829@163.com

Problem

- Want to forecast traffic speed of road segments using historical traffic data



Motivation

- Intelligent Transportation Systems (ITS) rely on traffic forecasting.
- Accurate forecasting assists in route guidance, and reduces congestion.

Related work

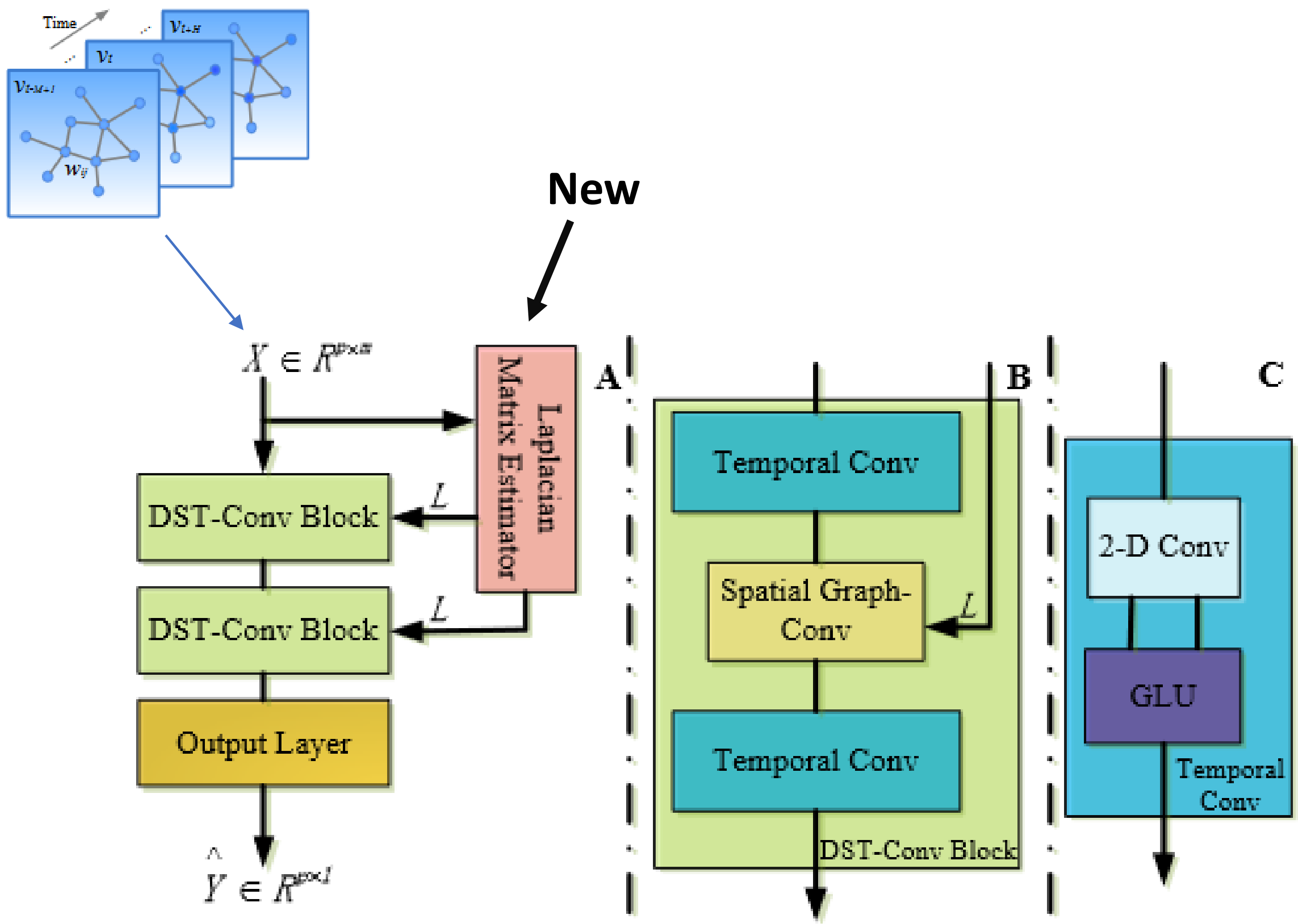
- Auto-regressive and moving average model (ARMA) (Williams and Hoel 2003)
- Support vector regression (SVR) (Chen et al. 2015)
  - Unsuitable for volatile traffic data

- Recurrent Neural Networks (RNN) (Wu and Tan 2016)
- Convolution Neural Network (CNN) (Zhang et al. 2016; Zhang, Zheng, and Qi 2016; Ma et al. 2017)
  - Not suitable for data points with irregular graph relations.

- Graph Convolutional Neural Network (STGCN) (Yu, Yin, and Zhu2017a)
  - Handle spatial relations well.
  - Assumes Laplacian is available and constant, while in the real world this is not true.

Method

- Based on GCNN architecture
- Primary contribution:
  Dynamic Laplacian via
  Laplacian Matrix Estimator
- Laplacian is crucial for determining receptive field in Graph Convolution
- |G| = N -> L = N x N



Dynamic Laplacian Matrix Estimator

- Real time Laplacian fluctuates around global Laplacian
- Observations:
  - Perturbations mainly come from short-term traffic patterns
  - Long-term traffic data forms a low rank tensor
  - Short-term data forms a sparse tensor in the temporal dimension
- Idea:
  - Compute dynamic Laplacian as: Global + Local
  - Global computed from distances between road sensors
  - Local computed in Tensor Decomposition Layer (TDL)

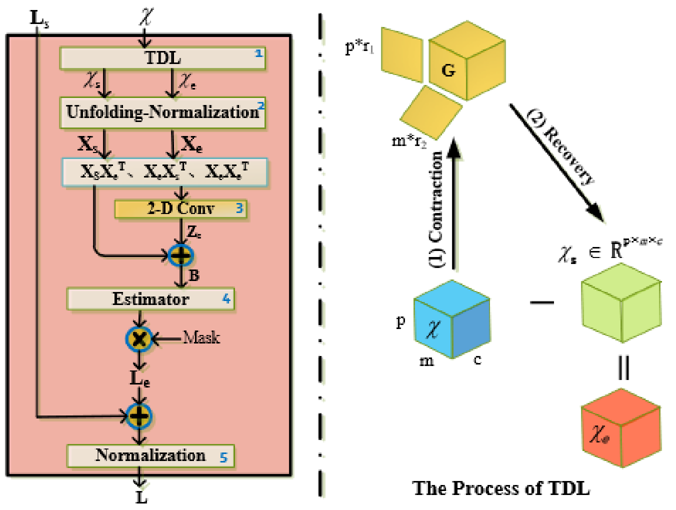


Figure 2: Laplacian Matrix Estimator

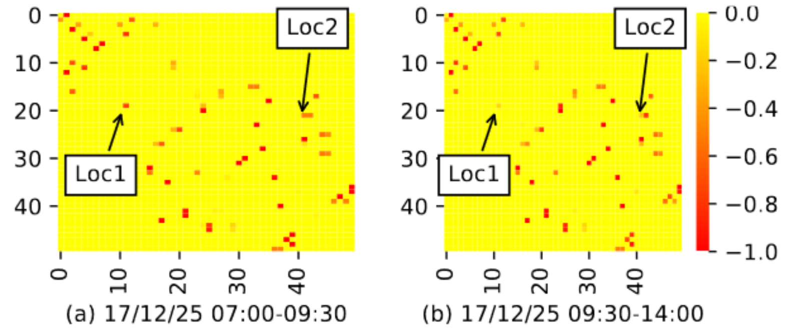


Figure 6: Spatial dependencies learning on two consecutive time spans.

Results

- Two datasets: NYC and California (PeMS).
- Four baselines:
  1. Vector Autoregression (VAR)
  2. Feed-Forward NN (FNN)
  3. Graph Conv GRU (GC-GRU)
  4. Spatio-temporal Graph CNN (STGCN)

Table 1: Forecasting error given by MAE and RMSE on NYC dataset (15/30/45 min)

	MAE	RMSE
VAR	4.14/ 4.84/ 5.29	6.01/ 6.63/ 7.45
FNN	3.60/ 3.79/ 4.16	5.80/ 5.87/ 6.20
GCGRU	3.20/ 3.36/ 3.49	5.23/ 5.31/ 5.50
STGCN	3.18/ 3.31/ 3.44	5.18/ 5.30/ 5.42
DGCNN	3.06/ 3.14/ 3.29	5.02/ 5.22/ 5.30

Performance Comparison

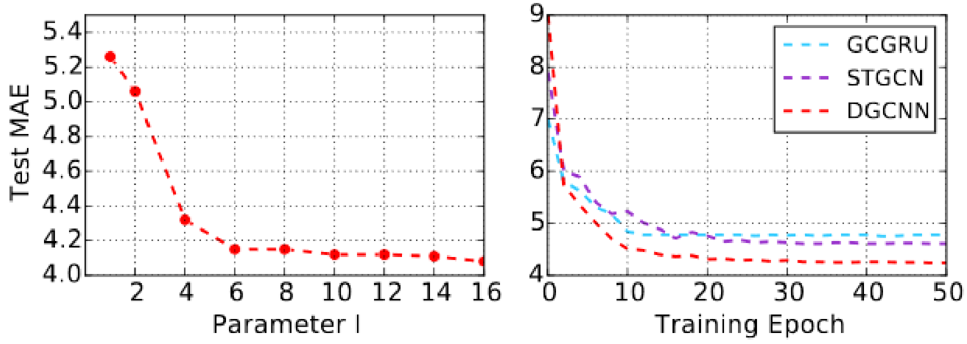


Figure 3: Test MAE versus the parameter I in DGCNN (left); Test MAE versus the number of training epochs (right). (PeMS-50)

Conclusion

- Achieves on average 10%-25% higher accuracy compared to other models, beating out other “state of the art” methods.

- No mention of inference time despite performance being important
- No mention of training time, and only brief parameter exposition