Scalable Structure Learning for Spatiotemporal Analysis

Qi (Rose) Yu

USC/Caltech

2017: Large-Scale Spatiotemporal Problems

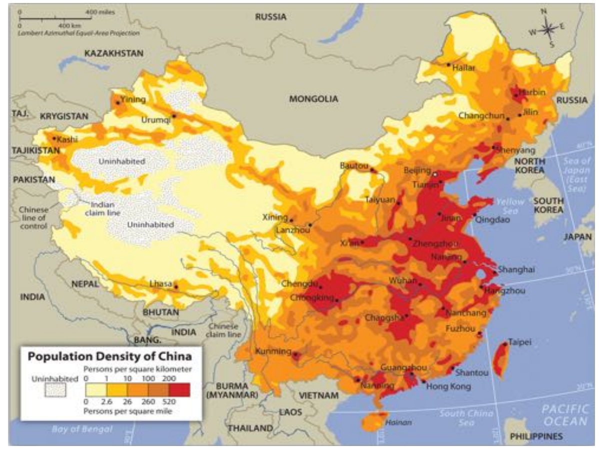
$124 billion
Congestion cost

$1.5 trillion
Disaster damage

61%
Energy waste

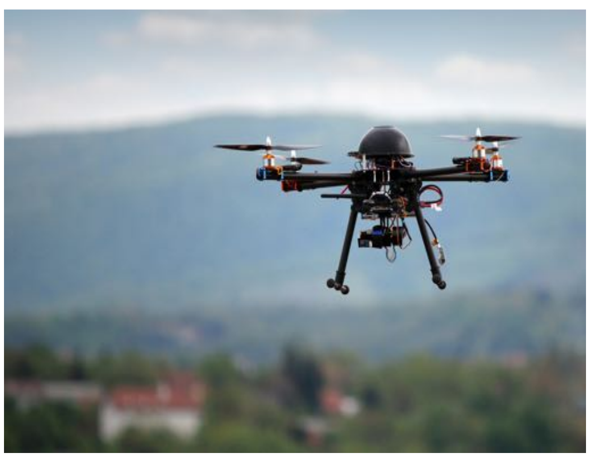
The Solution: Spatiotemporal Analysis

forecasting



clustering

interpolation

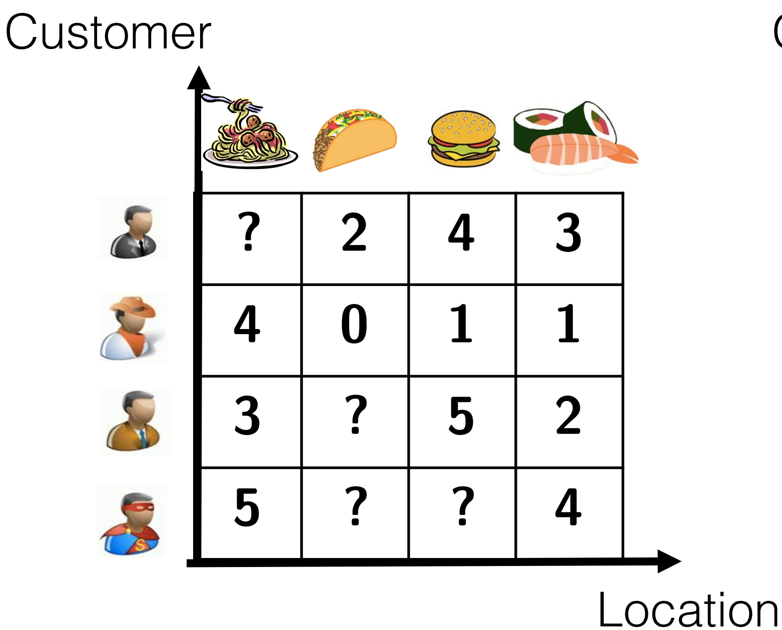


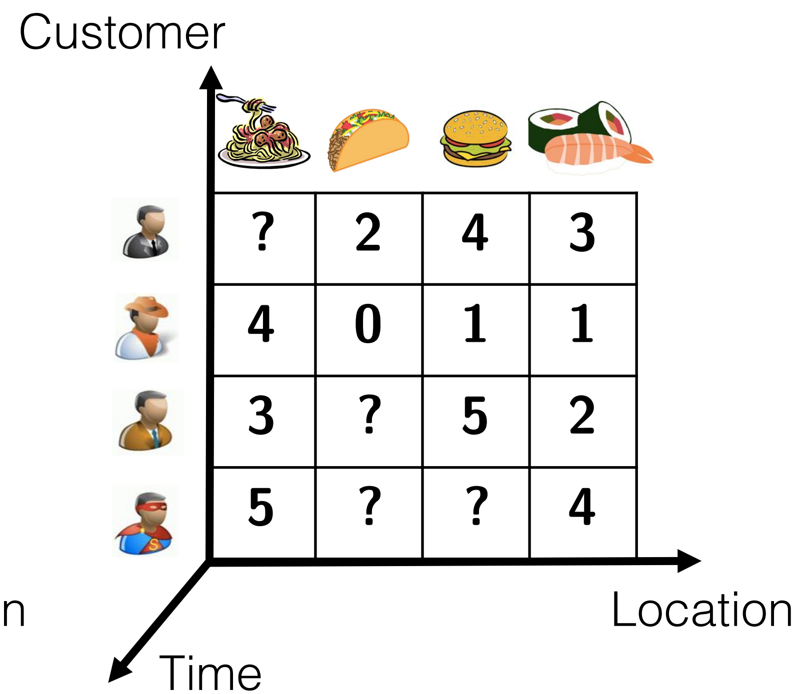
tracking

sensing

monitoring

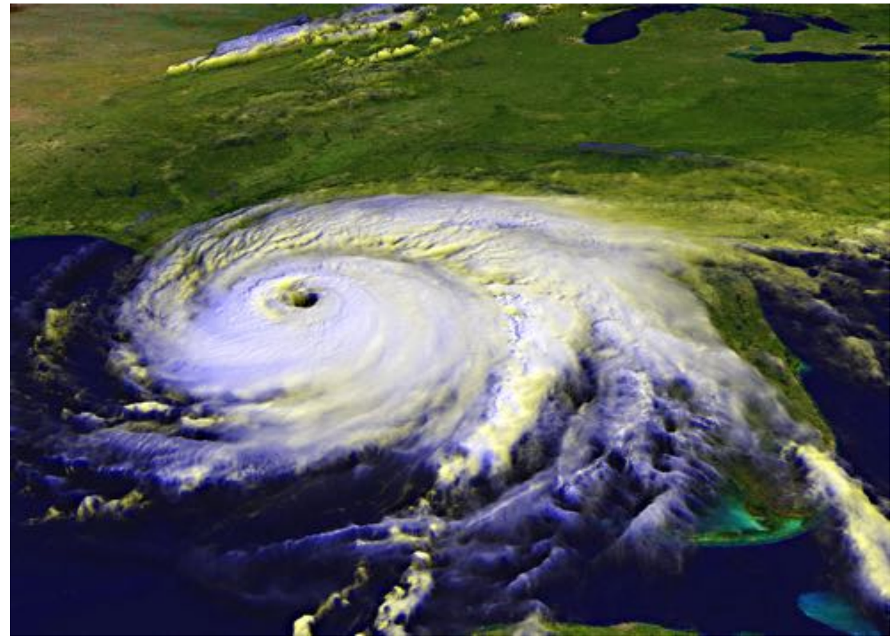
High-Order Correlation





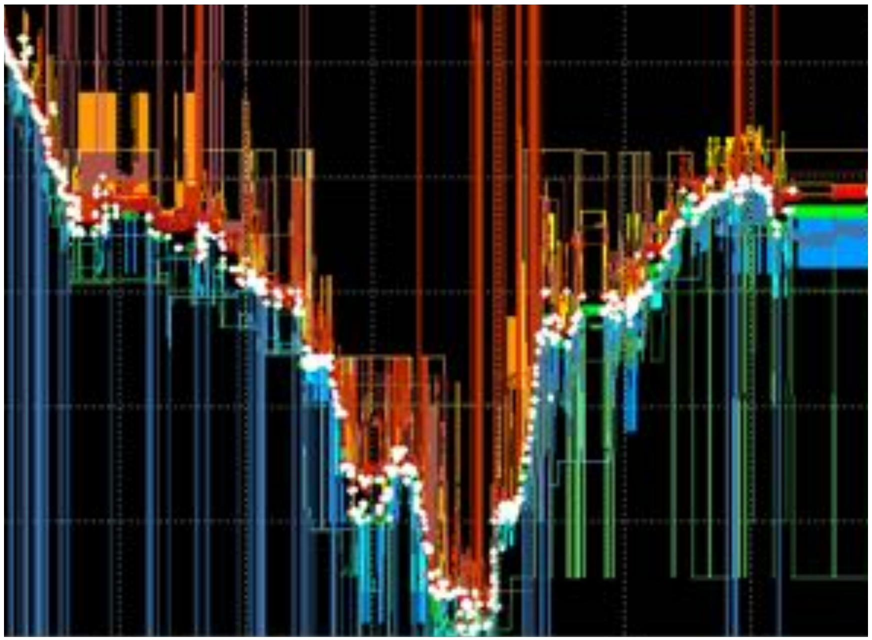
Non-Linear Dynamics

traffic flow



air turbulence

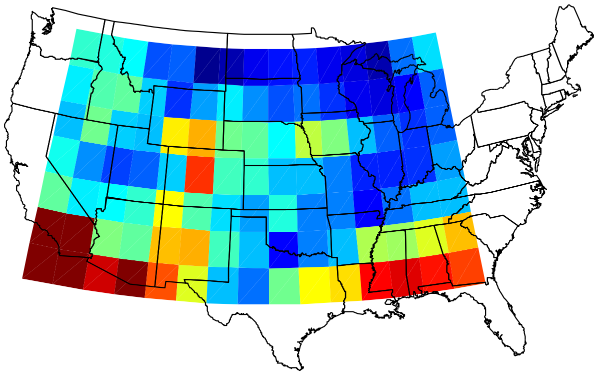
High Dimensionality



stock time series

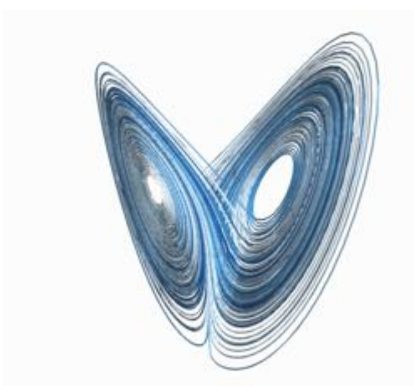
sensor network signals

Learning from Spatiotemporal Data

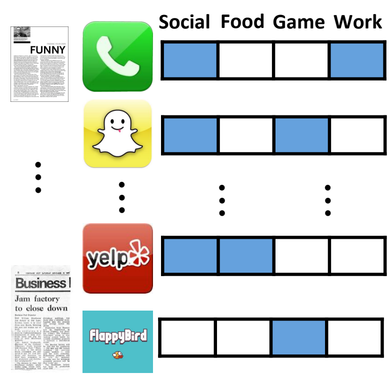


Low-Rank Tensor Learning
[NIPS 2014, ICML 2015, ICML 2016]

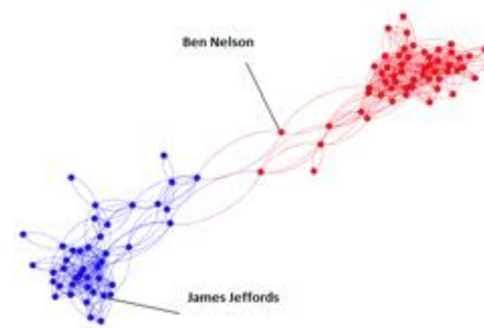
Road Network Traffic Forecasting
[KDD 2015, SDM 2017]



Learning Chaotic Dynamics
[ICML 2017 deepstruct]



Geographic User Profiling
[WSDM 2015]



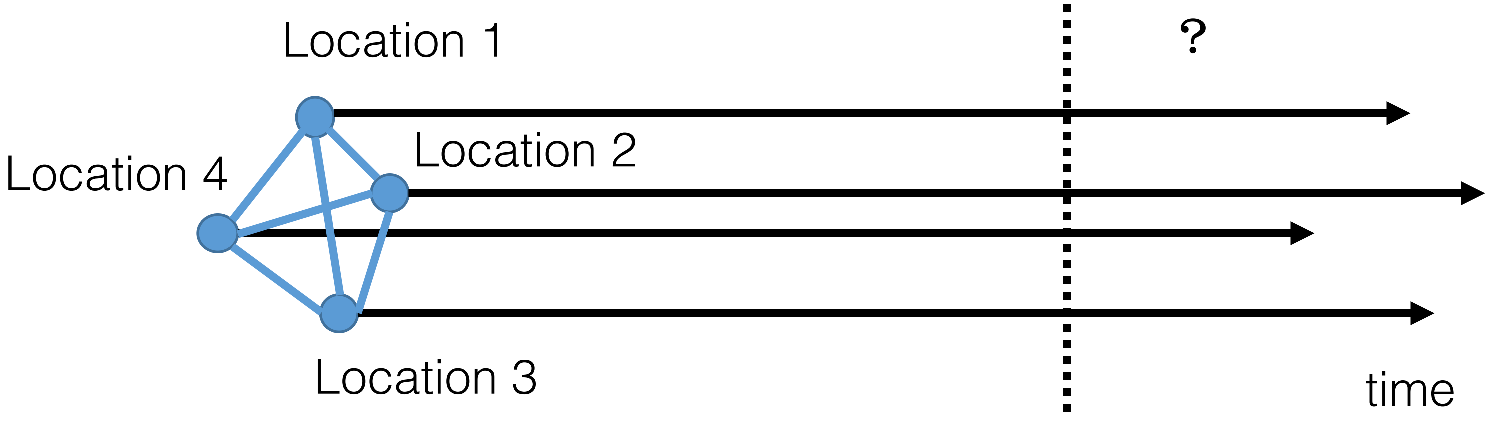
Social Media Anomaly Detection
[KDD 2014]

Graph Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting

Graph Convolutional Recurrent Neural Network: Data Drive Traffic Forecasting, joint work with Yaguang Li, Cyrus Shahabi, Yan Liu.

Introduction

- Spatiotemporal forecasting
  - Input: history from *P* locations $x_{t-1,1}, \dots, x_{t-K,P}$
  - Output: future values $x_{t,1}, \dots, x_{t+H,P}$
- Challenges
  - Non-linear dynamics
  - Non-regular graphs



Traffic Forecasting

Knowledge-driven
- Queueing theory
- Strong model assumption
- Human engineering

Data-driven
- Flexible modeling
- Quick response time
- Better generalization

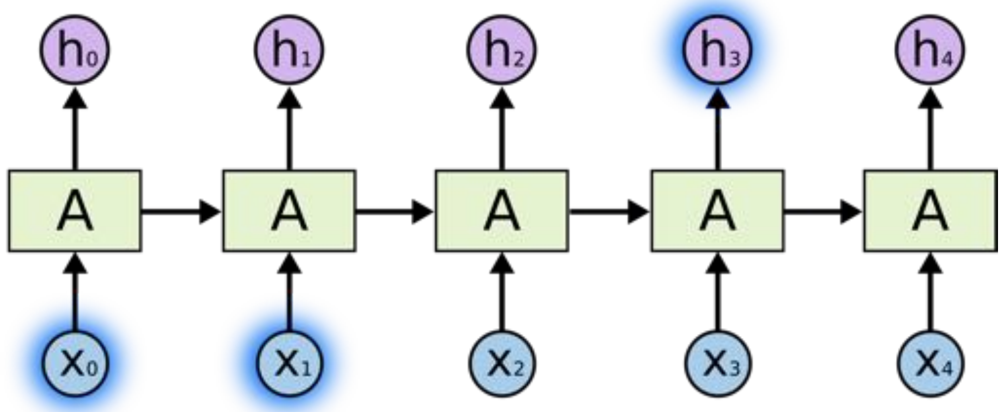

activity recognition

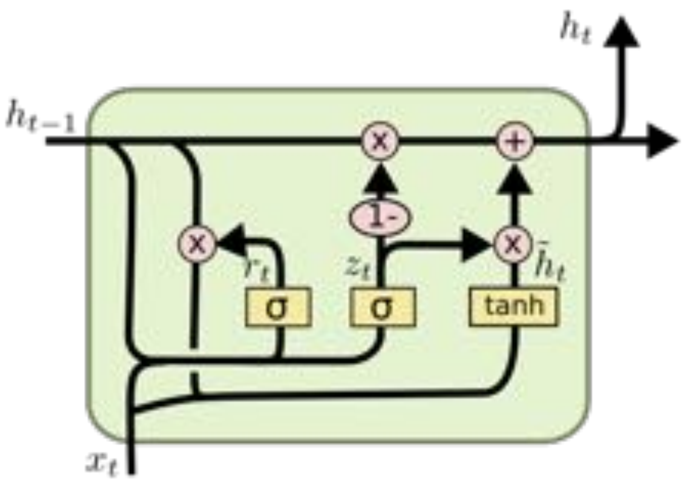

internet of things


epidemic control

Deep Recurrent Neural Networks

- RNN with Gated Recurrent Unit (GRU)





$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

† Christopher Olah, "Understanding LSTM Networks", blog post http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Spatial Dependency

- *Network connectivity*: High-way networks have sensors installed every 1-2 miles
- *Flow conservation*: The number of vehicles entering/exiting roads are approximately the same

Local spatial dependency for single sensor learned from weighted average

Network Connectivity

- Human visual attention: many animals focus on specific parts of their visual inputs
- Generalize the *attention mechanism* † for irregular graphs
- Learn to focus only on the close neighborhood instead of the entire network

$$\cancel{c_i \quad \sum_{j \in \ldots)} \quad h_j}$$

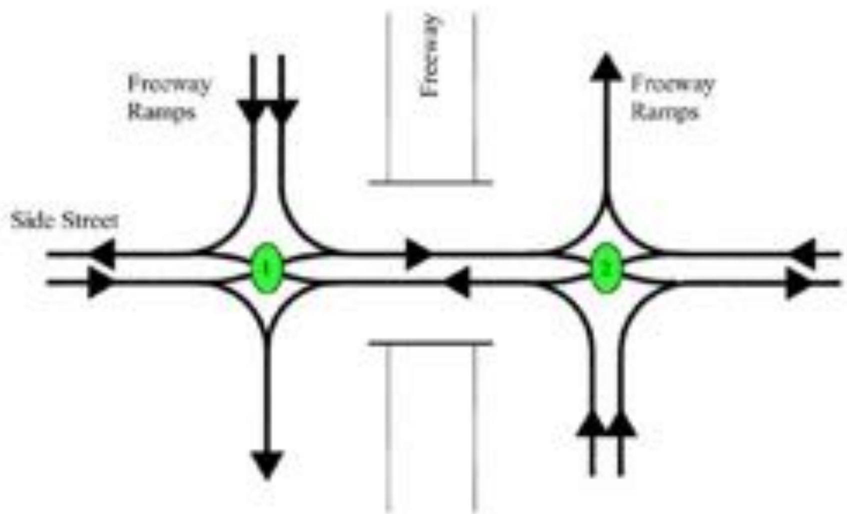
$$a_{ij} = \frac{\exp(f_{att}(h_i, h_j))}{\sum_{k \in nb(i,K)} \exp(f_{att}(h_i, h_k))}, \quad f_{att}(h_i, h_j) = h_i W_a h_j$$

† Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate." arXiv preprint arXiv:1409.0473 (2014).

Flow Conservation

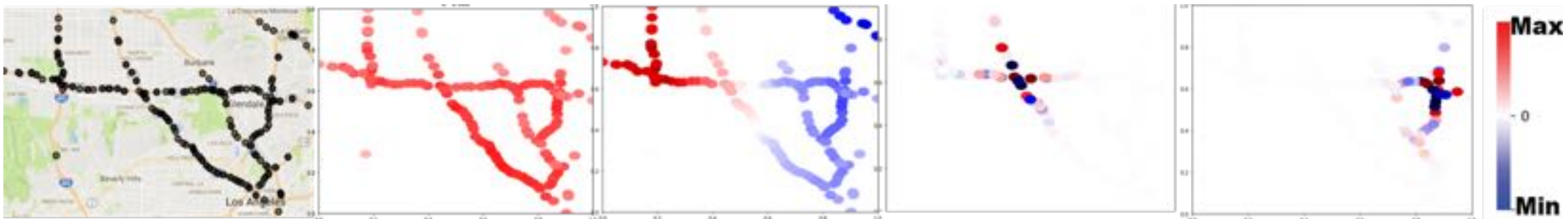
- Applying Laplacian operator $y = Lx$ represents one-step diffusion of the signal on the graph
- Similar to heat equation, which is given by the law "conservation of energy' in physics

$$\frac{\partial x_i(t)}{\partial t} = \sum A_{ij}(x_i - x_j) \qquad \frac{\partial x_i}{\partial t} = cL_i x$$



flow conservation

Graph Convolution



(a) (b) (c) (d) (e)

- Powers of Laplacian represent different spatial resolutions

$$y^t = g_w(L)x^t =: \sum_{k=0}^{K-1} w_k L^k x^t = W *_g x^t$$

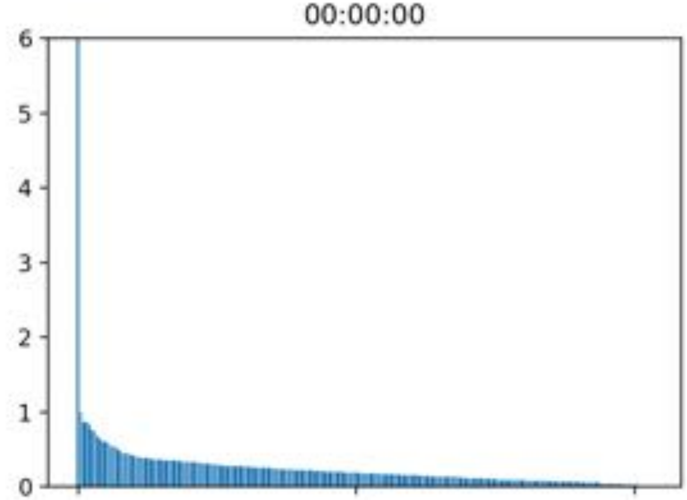
- Use Chebyshev polynomial expansion † as approximation

$$\sum_{k=0}^{K-1} g_w(L) = U \sum_{k=0}^{K-1} g_{\overline{w}}(\Lambda)U^T \approx U \sum_{k=0}^{K-1} g_{\overline{w}} T_k(\overline{\Lambda})U^T$$

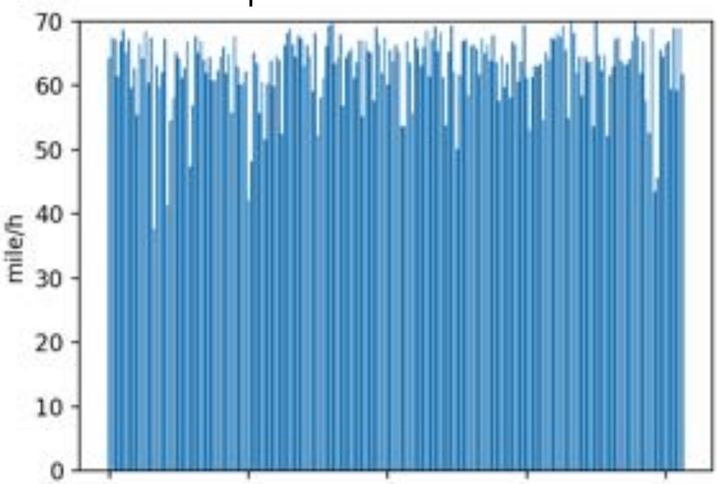
† Defferrard, Michael and Bresson, Xavier and Vandergheynst, Pierre, "Convolutional neural networks on graphs with fast localized spectral filtering", *Advances in Neural Information Processing Systems*, (NIPS) 2016

Spectral Transformation over Time

Traffic signal over time



Spectral domain



Vertex domain

$$z_t = \sigma\left(W_z *_g [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r *_g [h_{t-1}, x_t]\right)$$

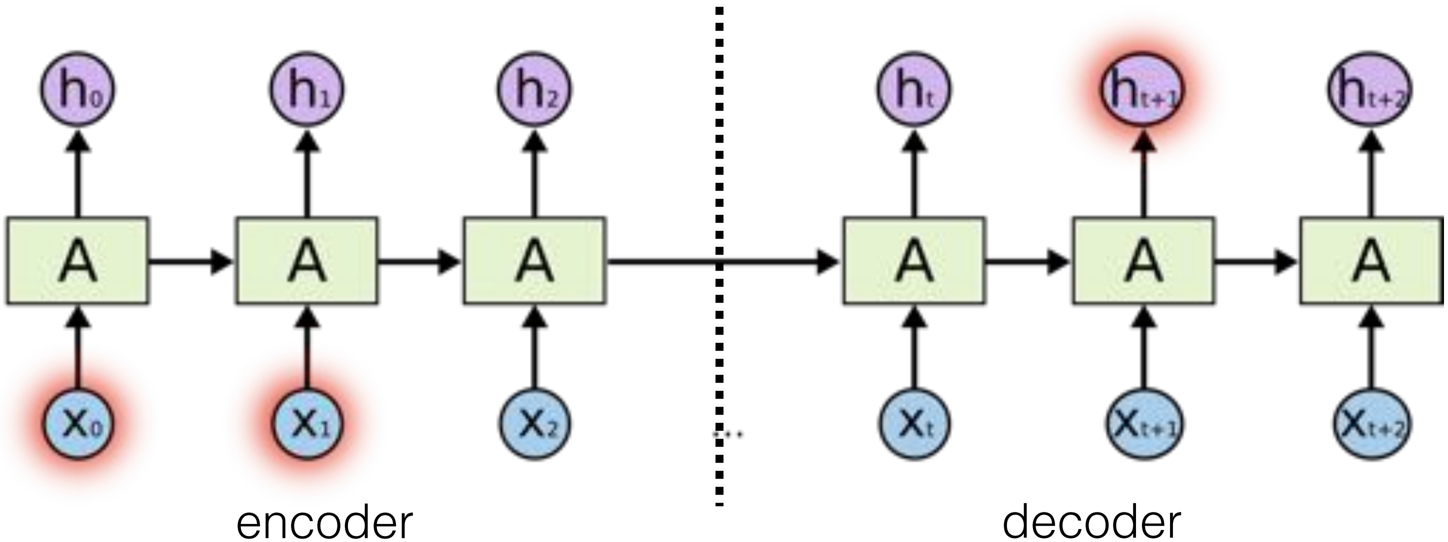
$$\tilde{h}_t = \tanh\left(W *_g [r_t * h_{t-1}, x_t]\right)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

- Spectral domain enjoys better sparsity.
- Skewness of the distribution corresponds to traffic congestion condition.

Long-Term Forecasting

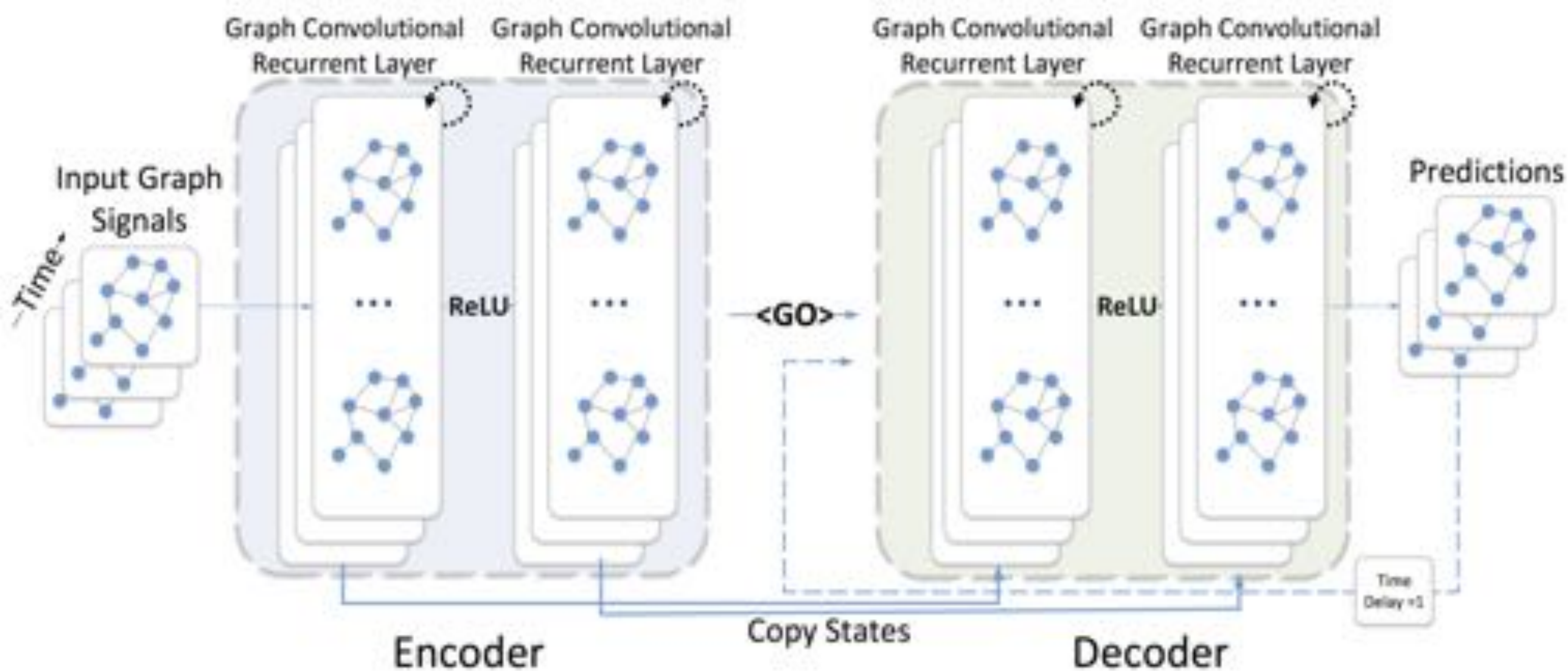
- Encoder-decoder architecture in sequence to sequence
- Mitigate error propagation with *Scheduled Sampling* †



† Bengio, Samy, et al. "Scheduled sampling for sequence prediction with recurrent neural networks." *Advances in Neural Information Processing Systems*. 2015.

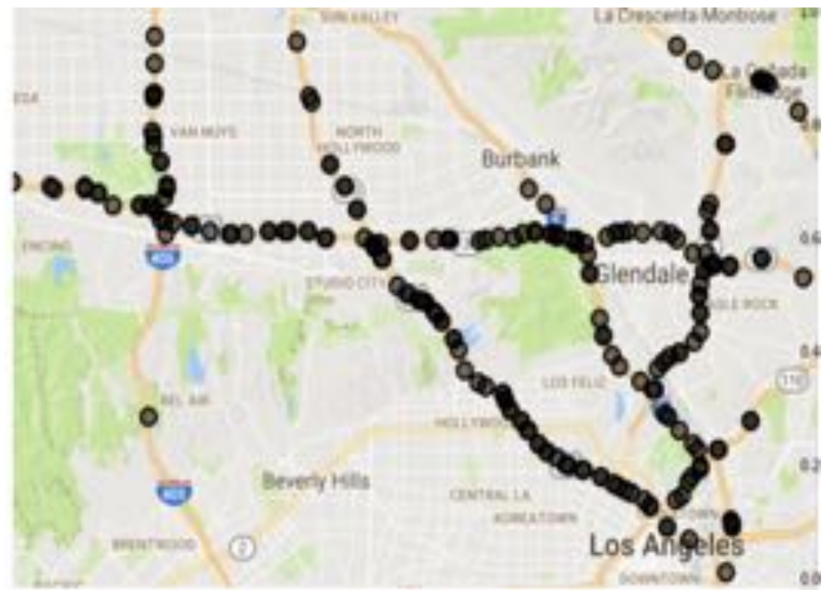
Graph Convolutional Recurrent Neural Network

- Graph convolutional kernel
- Recurrent neural network
- Encoder-decoder with scheduled sampling



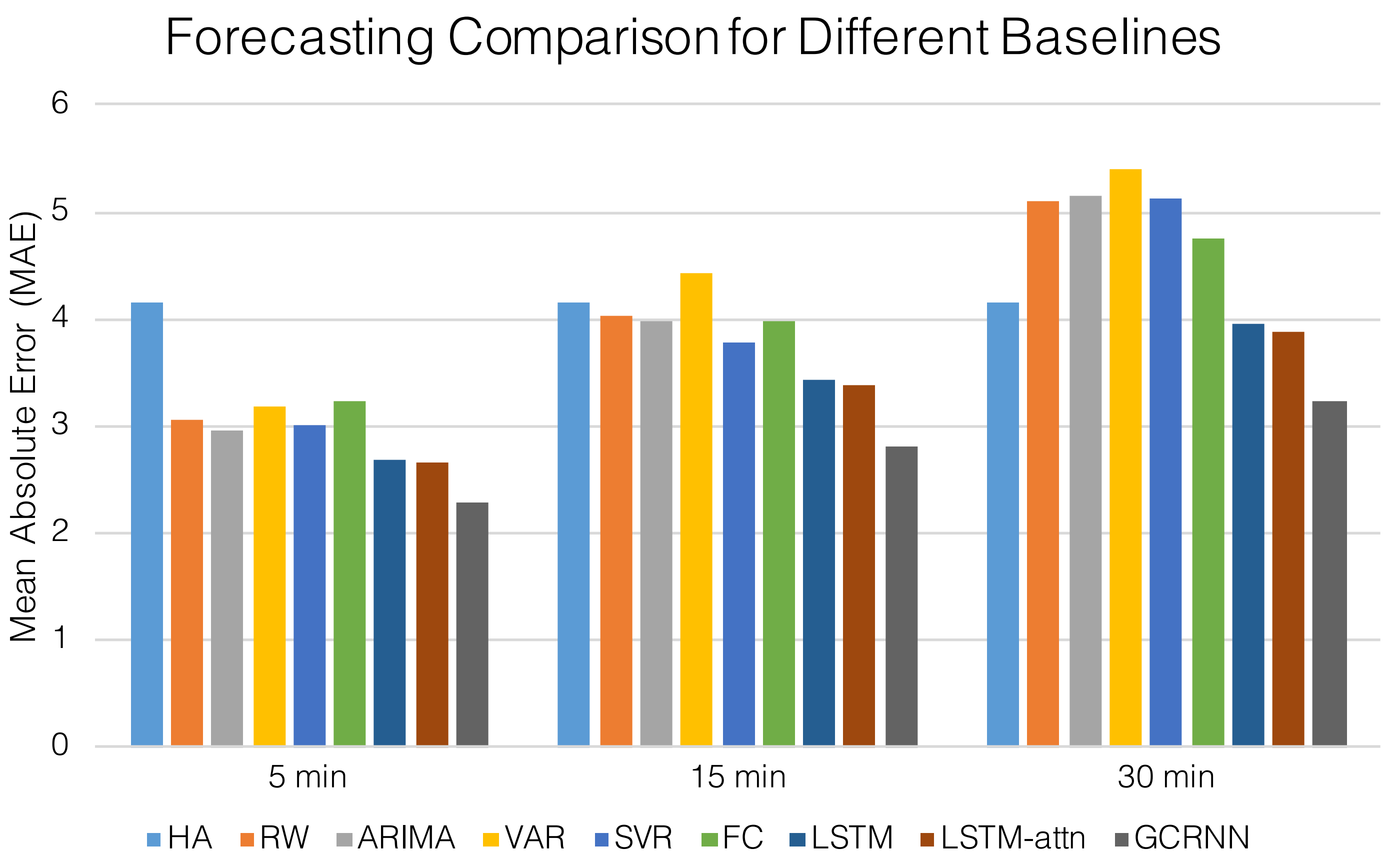
Experiment Setup

- Data:
  - 207 Highway loop detectors
  - 4 months in 2012 in Los Angeles County.
- Baselines:
  - Historical Average (HA)
  - Autoregressive Integrated Moving Average (ARIMA)
  - Random Walk (RW)
  - Support Vector Regression (SVR)
  - Vector Auto-Regression (VAR)
  - Feed forward Neural network (FC)
  - RNN with LSTM hidden units (LSTM)
  - RNN with spatial attention (LSTM-attn)



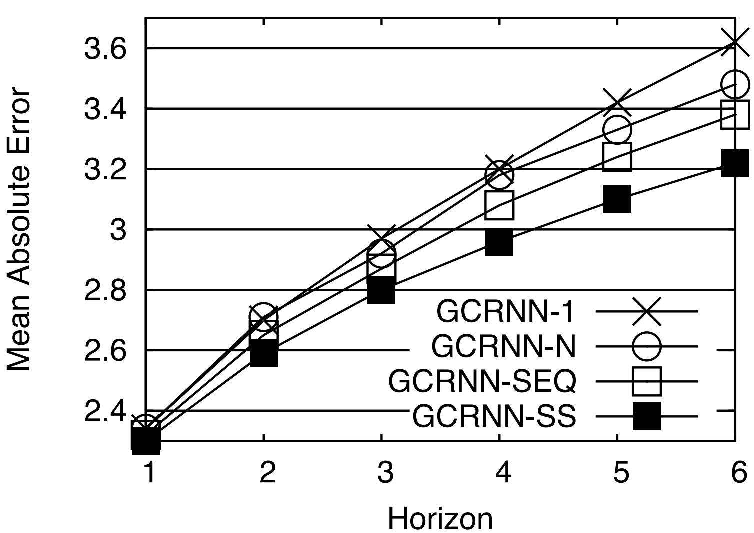
Experimental Results

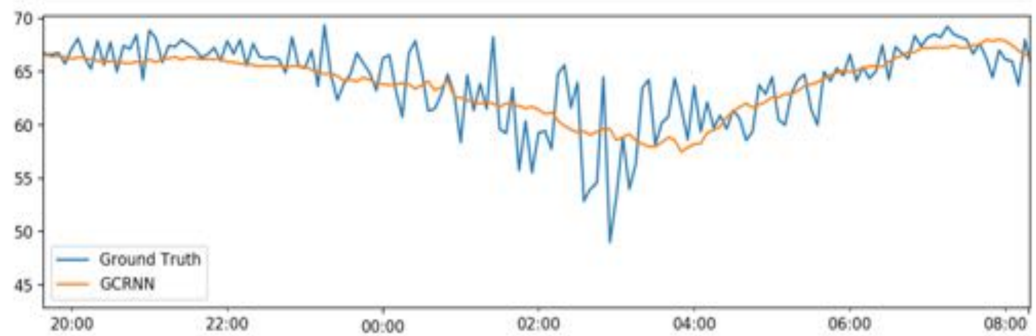
- GCRNN achieves best performance for all forecasting horizons (5 min, 15 min, 30 min)



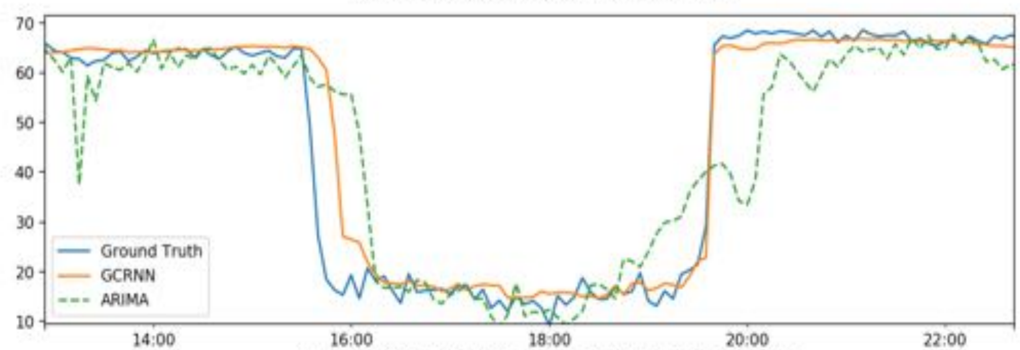
Experimental Results

- GCRNN can benefit from jointly learning multivariate times series and is less prone to error propagation
- Generates smooth prediction and is usually able to predict the start and end of peak hours.





(a). GCRNN generates smooth prediction.



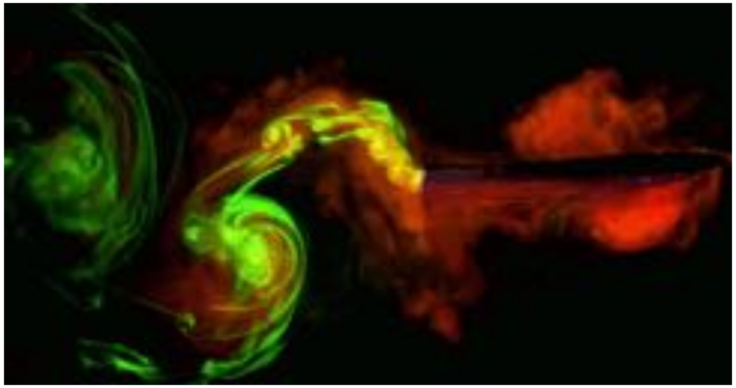
(b). GCRNN predicts the start and end of peak hours.

Learning from large-scale Spatiotemporal Data

Scalability

- How to forecast over long-range and long-term?
  - Structured prediction
  - Multi-resolution inference
- How to reduce the sampling complexity?
  - Weak supervision
  - Physical constraints

structured prediction



physical constraint

Reliablity

- How to handle messy data?
  - Error propagation
  - Adversarial corruption
- How to reason with uncertainty?
  - Probabilistic calibration
  - Stochastic optimization


chaotic dynamics


reason with uncertainty

Q&A?