Machine Learning for Spatiotemporal Sequence Forecasting and Its Application to Nowcasting

Dit-Yan Yeung

Professor and Acting Head

Department of Computer Science and Engineering

Hong Kong University of Science and Technology

Outline

- Introduction
  - What is STSF? Why is it important? Why choose this research topic?
- Architectures for STSF-RG
  - Background – Deep Learning, Precipitation Nowcasting
  - ConvLSTM
  - TrajGRU
  - HKO-7
- Architectures for STSF-IG
  - Background
  - GaAN
  - GGRU
- Conclusion & Future Work

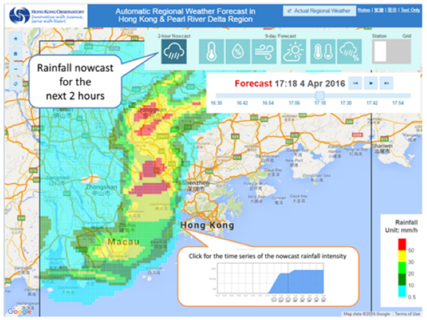
Outline

- Introduction
  - What is STSF? Why is it important? Why choose this research topic?
- Architectures for STSF-RG
  - Background – Deep Learning, Precipitation Nowcasting
  - ConvLSTM
  - TrajGRU
  - HKO-7
- Architectures for STSF-IG
  - Background
  - GaAN
  - GGRU
- Conclusion & Future Work

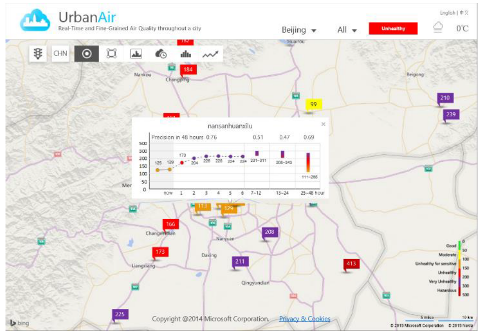
Introduction – Spatiotemporal Sequence Forecasting

- Many real-world phenomena are spatiotemporal:
  - Traffic flow, diffusion of air pollutants, regional rainfall, etc.
- Predicting the multi-step future of these spatiotemporal systems based on the past is important for many real-world applications

Typhoon alert system



Rainfall nowcasting



Forecasting air pollutants

- We call this type of problems Spatiotemporal Sequence Forecasting (STSF)

Introduction – Definition of STSF

- A length-*T* spatiotemporal sequence: $\mathbf{X}_{1:T} = [\mathbf{X}_1, \mathbf{X}_2, ...\mathbf{X}_T]$

$\mathbf{X}_t \in \mathbb{R}^{K\times(D+E)}$ *K*: number of locations, *D*: number of measurements, *E*: number of coordinates

$\mathbf{M}_t \in \mathbb{R}^{K\times D}$ Measurements (observed values at the locations)

$\mathbf{C}_t \in \mathbb{R}^{K\times E}$ Coordinates (locations)

- Spatiotemporal sequence forecasting problem:

$$\hat{\mathbf{X}}_{t+1:t+L} = \underset{\mathbf{X}_{t+1:t+L}}{\operatorname{argmax}}\, p(\mathbf{X}_{t+1:t+L} \mid \mathbf{X}_{1:t}, \mathcal{A}_t).$$

L-step future (*L*>1) past observation auxiliary information

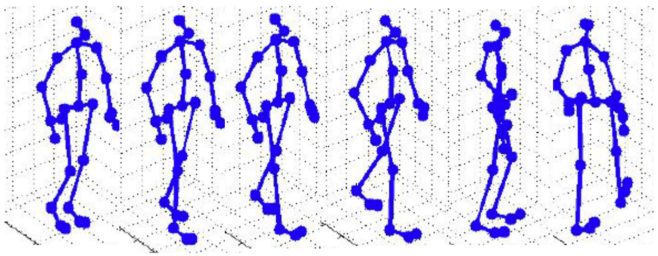
- For problems where both input and output can be spatiotemporal sequences

Introduction – Three types of STSF problems

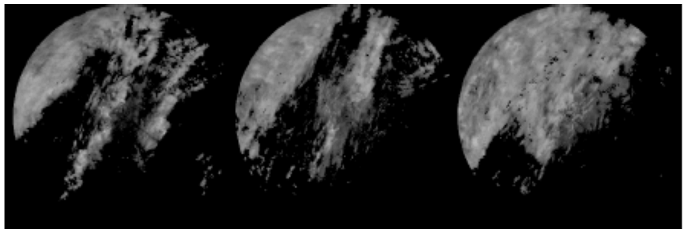
[Shi & Yeung, 2018]

	Problem Name	Coordinates	Measurements
TF-MPC	Trajectory Forecasting of Moving Point Cloud	Changing	Fixed/Changing
STSF-RG	Spatiotemporal Forecasting on Regular Grid	Fixed regular grid	Changing
STSF-IG	Spatiotemporal Forecasting on Irregular Grid	Fixed irregular grid	Changing

- Examples:



Human Motion Prediction,
Crowd Movement Prediction
TF-MPC



Rainfall Nowcasting,
Video Prediction
(Dense Observation)
STSF-RG

Weather Data Prediction,
Traffic Accident Prediction,
(Sparsely Spread)
STSF-IG

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

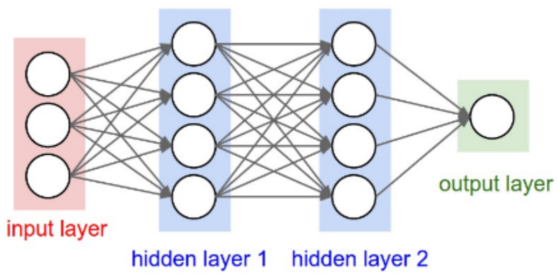
Introduction – Machine Learning for STSF

- Already have an accurate model? (Know the laws)
  - Step 1: Identify the initial condition of the model
  - Step 2: Forecast by simulation
  - Not always the case!
    - Systems with unknown dynamics – Crowd, Atmosphere, Natural Videos
- Machine learning for STSF!
  - Train a forecasting model based on the historical data – Learning the laws

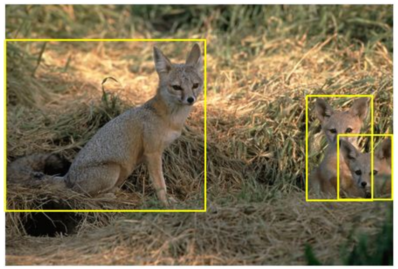
Introduction – Deep Learning for STSF

- Different types of ML methods for STSF: [Shi & Yeung, 2018]
  - Feature-based, state-space models, Gaussian process based models, etc.
  - Deep learning based
- Deep learning: Layered network structure + End-to-end training

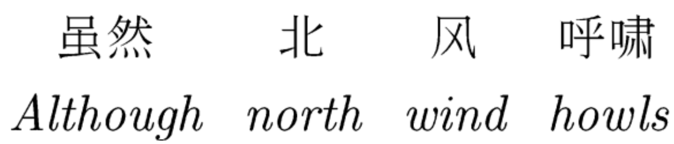
Breakthroughs in many tasks



Input → Output



Object Detection



Machine Translation

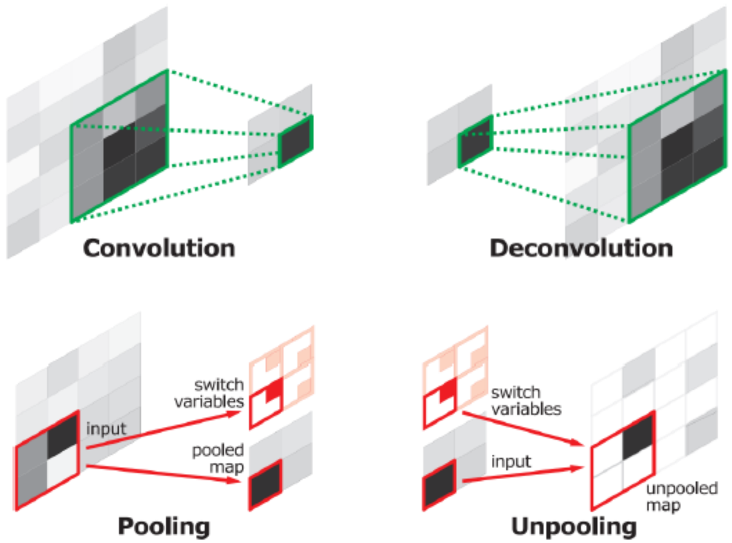
Introduction – Outline of Talk

- Exploring deep learning architectures for STSF
- Architectures for STSF-RG:
  - Tackle the precipitation nowcasting problem
  - Convolutional Long Short-Term Memory (ConvLSTM) - first machine learning based solution
  - Trajectory Gated Recurrent Unit (TrajGRU)
  - HKO-7 benchmark - first large-scale benchmark
- Architectures for STSF-IG:
  - Convert to spatiotemporal graph
  - Gated Attention Network (GaAN)
  - Graph GRU (GGRU)

Outline

- Introduction
  - What is STSF? Why is it important? Why choose this research topic?
- Architectures for STSF-RG
  - Background – Deep Learning, Precipitation Nowcasting
  - ConvLSTM
  - TrajGRU
  - HKO-7
- Architectures for STSF-IG
  - Background
  - GaAN
  - GGRU
- Conclusion & Future Work

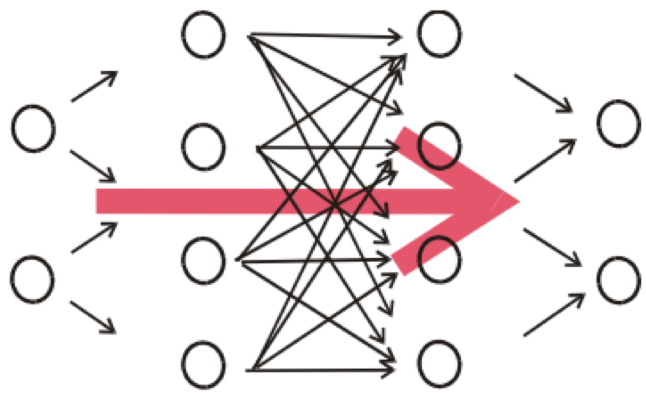
Deep Learning 101 – Basics



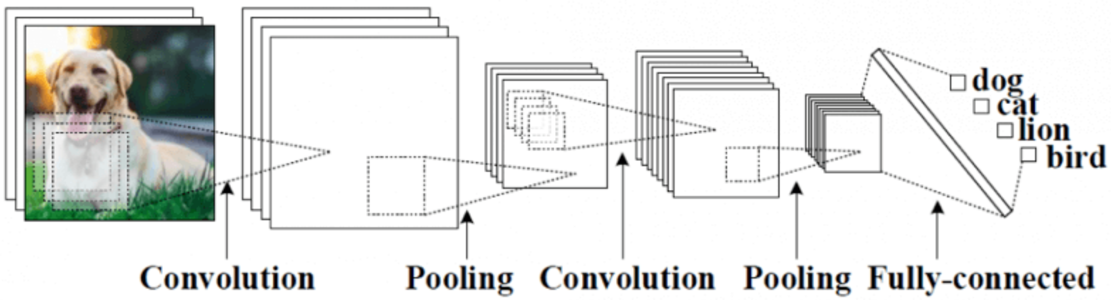
- Deep Learning
  - Layered structure (Stacking building blocks)
  - End-to-end (Input → Model → Output)
- Basic Building Blocks
  - Fully-connected Layer: $\mathbf{h} = \mathbf{W}\mathbf{x} + \mathbf{b},$
  - Activation: $\mathbf{h} = f(\mathbf{x})$
  - Convolution Layer: $\mathcal{H} = \mathcal{W} * \mathcal{X} + \mathbf{b}, \quad \mathcal{H}_{:,i,j} = \mathbf{W}\mathbf{x}^{\mathcal{N}(i,j)} + \mathbf{b}.$
  - Pooling Layer: $\mathcal{H}_{k,i,j} = g(\{\mathcal{X}_{k,s,t} \mid (s,t) \in \mathcal{N}(i,j)\})$
  - Deconvolution and Unpooling: "Backward" computation
- Feedforward Neural Networks & Recurrent Neural Networks

Deep Learning 101 – Feedforward Neural Networks

A feedforward neural network (FNN) is acyclic. There is no loop.



Convolutional neural network (CNN)

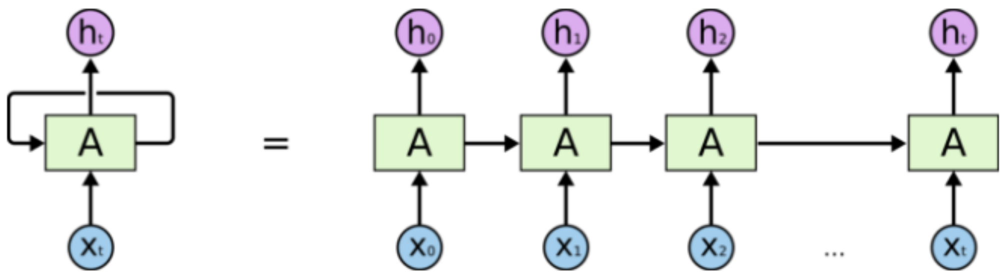


Deep Learning 101 – Recurrent Neural Networks

Cycles are allowed in a recurrent neural network (RNN)

Basic RNN: $\mathbf{h}_t = \tanh(\mathbf{W}_h \mathbf{h}_{t-1} + \mathbf{W}_x \mathbf{x}_t + b)$

After unfolding the structure, an RNN can be viewed as an FNN with shared weights.



An unrolled recurrent neural network.

Deep Learning 101 – Training

- Stochastic Gradient Descent
- Backpropagation (BP)
  - $\frac{\partial f(g(h))}{h} = \frac{\partial f(g(h))}{\partial g(h)} \frac{\partial g(h)}{h}$ Gradient from higher layer → lower layer
  - Also known as "Reverse mode of automatic differentiation"
- Backpropagation Through Time (BPTT)
  - Unfold the RNN and run BP

Deep Learning 101 – Gated Recurrent Neural Network

[Pascanul et.al, ICML2013]

- Product of Jacobians → Vanishing/exploding gradient
- Gated Recurrent Neural Network: Control information flow

Long Short-Term Memory (LSTM)

$$\boxed{\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{W}_{ci} \circ \mathbf{c}_{t-1} + \mathbf{b}_i),}$$
$$\boxed{\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_t + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{W}_{cf} \circ \mathbf{c}_{t-1} + \mathbf{b}_f),}$$
$$\mathbf{c}_t = \mathbf{f}_t \circ \mathbf{c}_{t-1} + \mathbf{i}_t \circ \tanh(\mathbf{W}_{xc}\mathbf{x}_t + \mathbf{W}_{hc}\mathbf{h}_{t-1} + \mathbf{b}_c),$$
$$\boxed{\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{W}_{co} \circ \mathbf{c}_t + \mathbf{b}_o),}$$
$$\mathbf{h}_t = \mathbf{o}_t \circ \tanh(\mathbf{c}_t).$$

[Hochreiter & Schmidhuber, 1997]

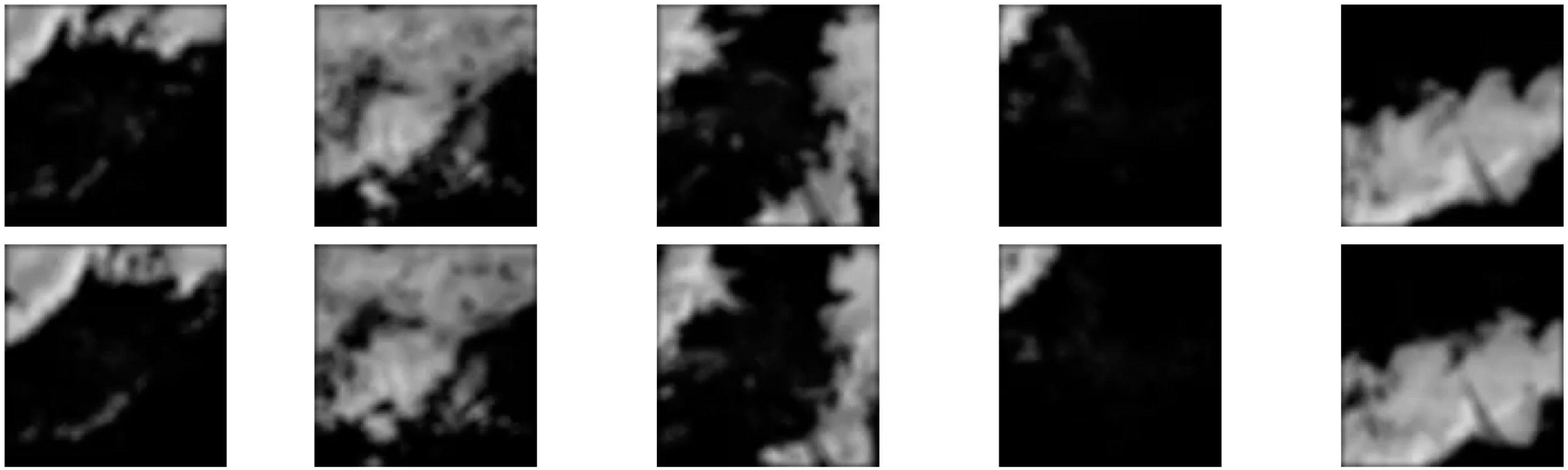
Gated Recurrent Unit (GRU)

$$\boxed{\mathbf{z}_t = \sigma(\mathbf{W}_{xz}\mathbf{x}_t + \mathbf{W}_{hz}\mathbf{h}_{t-1} + \mathbf{b}_z),}$$
$$\boxed{\mathbf{r}_t = \sigma(\mathbf{W}_{xr}\mathbf{x}_t + \mathbf{W}_{hr}\mathbf{h}_{t-1} + \mathbf{b}_r),}$$
$$\mathbf{h}'_t = f(\mathbf{W}_{xh}\mathbf{x}_t + \mathbf{r}_t \circ (\mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{b}_h)),$$
$$\mathbf{h}_t = (1 - \mathbf{z}_t) \circ \mathbf{h}'_t + \mathbf{z}_t \circ \mathbf{h}_{t-1},$$

[Chung et al., 2014]

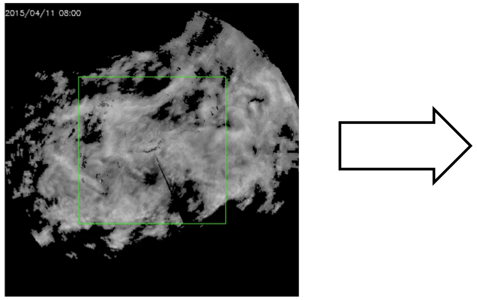
Precipitation Nowcasting – Definition

- Predict the future rainfall intensity (0-6 hours) in a local region based on radar echo maps, rain gauge and other data.
  - High resolution & high frequency (usually 6min)
  - High-dimensional spatiotemporal data



Precipitation Nowcasting – Real-world Impact + A Challenging Problem

- Precipitation nowcasting IMPACTS our daily life



Precipitation nowcasting

a) Road condition

b) Guidance for aviation

c) Rainstorm warning

- Complexities of the atmosphere + real-time, large-scale, and fine-grained nowcasting → Challenging problem!

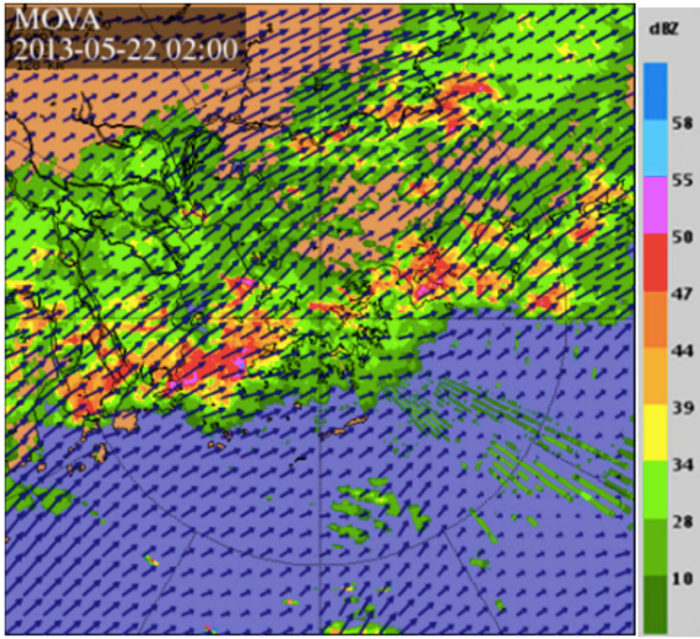
Precipitation Nowcasting – Classical Methods

- Numerical weather prediction (NWP) based methods
  - Build a model with several physical equations. Predict by simulation.
  - More accurate in the longer term
  - The first 1-2 hours of model forecasts may not be available
- Optical flow based methods
  - Optical flow estimation + Extrapolation (Semi-Lagrangian extrapolation)
  - More accurate in the first 1-2 hours
  - ROVER algorithm by HKO

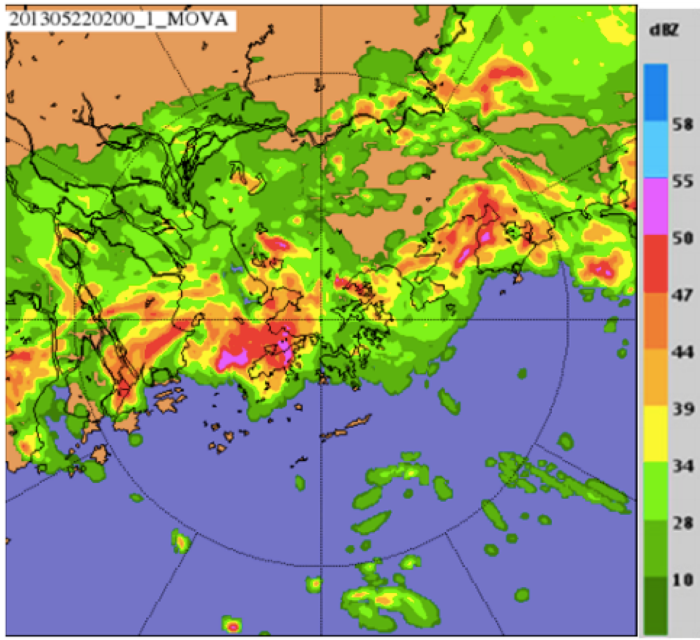
[Cheung & Yeung, 2012]

Precipitation Nowcasting – More about Optical Flow based Methods

- Step-1: Estimate the flow field based on the previous 2 frames
- Step-2: Extrapolate the last frame



Arrows denote the estimated flow field



Precipitation Nowcasting – Limitations of Optical Flow based Methods

- Limitations:
  - Flow estimation step and radar echo extrapolation step are separated, accumulative error
  - Do not benefit from our available radar-echo sequences
  - Longer-range temporal relationship (Optical flow is estimated using 2 frames)

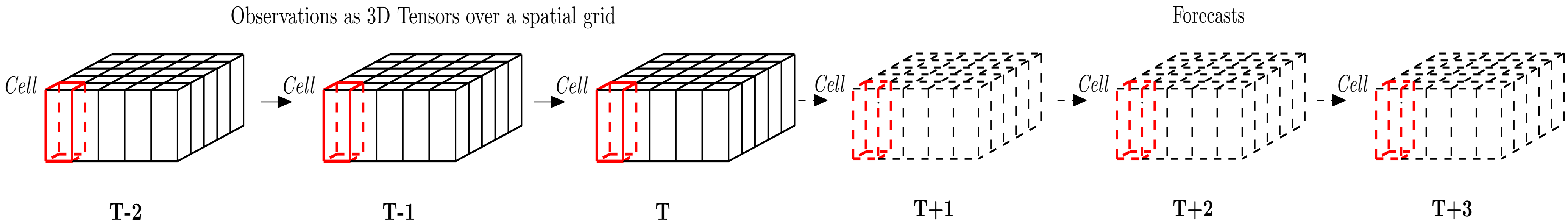
- We need a machine learning based, end-to-end approach to this problem! → How about deep learning?

Precipitation Nowcasting – Deep Learning Solution is Non-trivial

- However, solving the problem by deep learning is not trivial!
- Multi-step prediction
  - size of the search space grows exponentially
- Spatiotemporal data
  - We need to take advantage of the spatiotemporal correlation within the data

Precipitation Nowcasting – Formulated as STSF-RG

- Periodic observations taken from a dynamical system over a spatial grid → sequence of tensors



- Predict the most likely length-*K* sequence in the future given the previous *J* observations

$$\tilde{\mathcal{X}}_{t+1}, \ldots, \tilde{\mathcal{X}}_{t+K} = \underset{\mathcal{X}_{t+1}, \ldots, \mathcal{X}_{t+K}}{\arg\max} \; p(\mathcal{X}_{t+1}, \ldots, \mathcal{X}_{t+K} \mid \hat{\mathcal{X}}_{t-J+1}, \hat{\mathcal{X}}_{t-J+2}, \ldots, \hat{\mathcal{X}}_{t})$$

Precipitation Nowcasting – Encoder-Forecaster Structure

- Encoder-forecaster (EF) structure

[Sutskever et al., 2014]

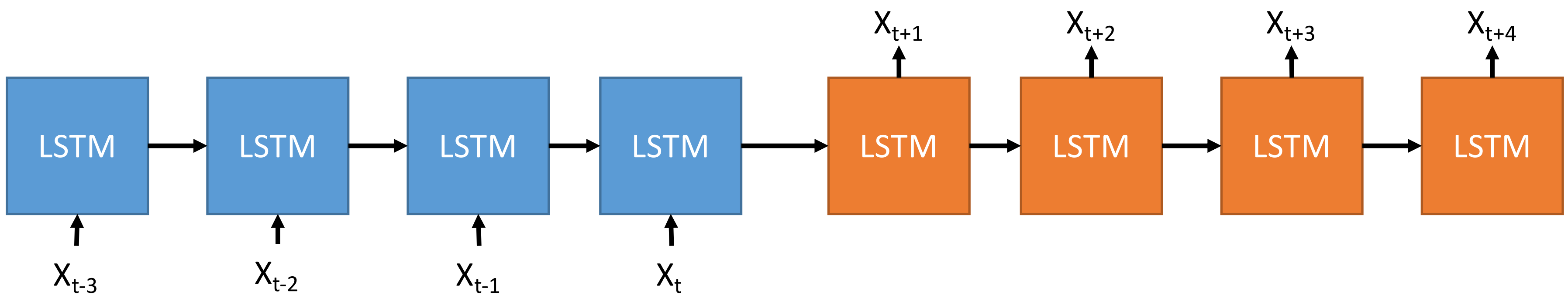
$$\tilde{\mathcal{X}}_{t+1}, \ldots, \tilde{\mathcal{X}}_{t+K} = \underset{\mathcal{X}_{t+1}, \ldots, \mathcal{X}_{t+K}}{\operatorname{argmax}} \; p(\mathcal{X}_{t+1}, \ldots, \mathcal{X}_{t+K} \mid \hat{\mathcal{X}}_{t-J+1}, \hat{\mathcal{X}}_{t-J+2}, \ldots, \hat{\mathcal{X}}_t)$$

$$\approx \underset{\mathcal{X}_{t+1}, \ldots, \mathcal{X}_{t+K}}{\operatorname{argmax}} \; p(\mathcal{X}_{t+1}, \ldots, \mathcal{X}_{t+K} \mid f_{encoder}(\hat{\mathcal{X}}_{t-J+1}, \hat{\mathcal{X}}_{t-J+2}, \ldots, \hat{\mathcal{X}}_t))$$

$$\approx g_{forecaster}(f_{encoder}(\hat{\mathcal{X}}_{t-J+1}, \hat{\mathcal{X}}_{t-J+2}, \ldots, \hat{\mathcal{X}}_t))$$

Precipitation Nowcasting – Flatten to vectors

- Naïve approach: Treat the 3D tensors as vectors and directly use LSTM as the encoder and forecaster. [Srivastava et al., 2015]



- Ignores the spatiotemporal nature of the data
- We propose Convolutional LSTM (ConvLSTM) and Trajectory GRU (TrajGRU) as new building blocks. [Shi et al., 2015], [Shi et al., 2017]

Outline

- Introduction
  - What is STSF? Why is it important? Why choose this research topic?
- Architectures for STSF-RG
  - Background – Deep Learning, Precipitation Nowcasting
  - ConvLSTM
  - TrajGRU
  - HKO-7
- Architectures for STSF-IG
  - Background
  - GaAN
  - GGRU
- Conclusion & Future Works

ConvLSTM – Motivation

- What is the characteristics of the spatiotemporal data?
- Strong correlation between local neighborhoods, i.e., nearby points tend to act similarly!
- Encode the prior knowledge by specifying the network structure
- ConvLSTM: Combine CNN & RNN by convolutional recurrence

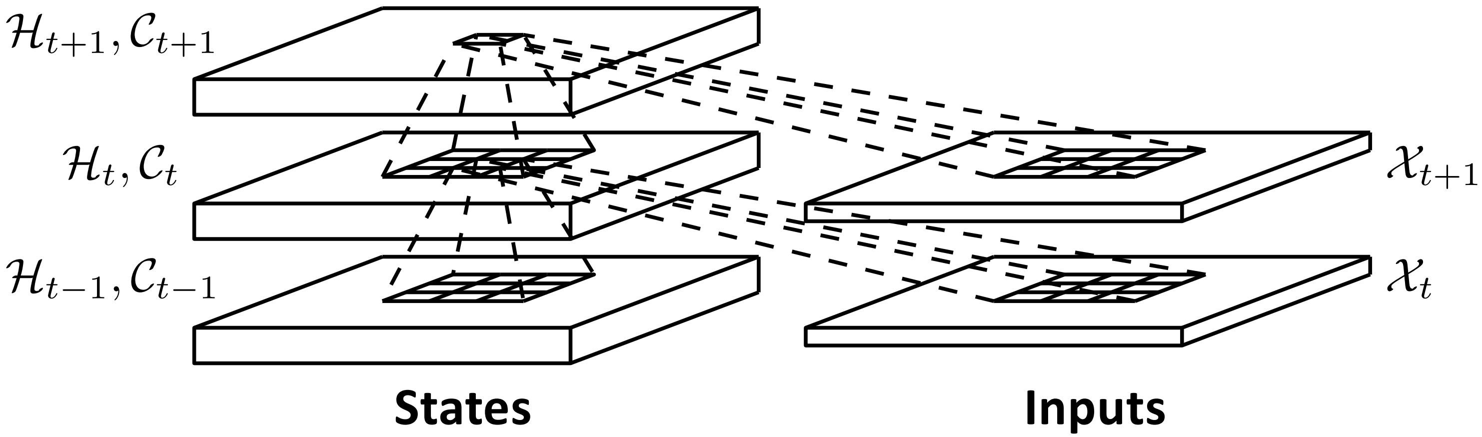
ConvLSTM – Formula

- Proposed method: Convolutional LSTM (ConvLSTM)
  - Inputs are 3D tensors rather than vectors
  - Use convolution instead of full-connection in state-to-state / input-state transition!

$$
\begin{aligned}
i_t &= \sigma(W_{xi} * \mathcal{X}_t + W_{hi} * \mathcal{H}_{t-1} + W_{ci} \circ \mathcal{C}_{t-1} + b_i) \\
f_t &= \sigma(W_{xf} * \mathcal{X}_t + W_{hf} * \mathcal{H}_{t-1} + W_{cf} \circ \mathcal{C}_{t-1} + b_f) \\
\mathcal{C}_t &= f_t \circ \mathcal{C}_{t-1} + i_t \circ \tanh(W_{xc} * \mathcal{X}_t + W_{hc} * \mathcal{H}_{t-1} + b_c) \\
o_t &= \sigma(W_{xo} * \mathcal{X}_t + W_{ho} * \mathcal{H}_{t-1} + W_{co} \circ \mathcal{C}_t + b_o) \\
\mathcal{H}_t &= o_t \circ \tanh(\mathcal{C}_t)
\end{aligned}
$$

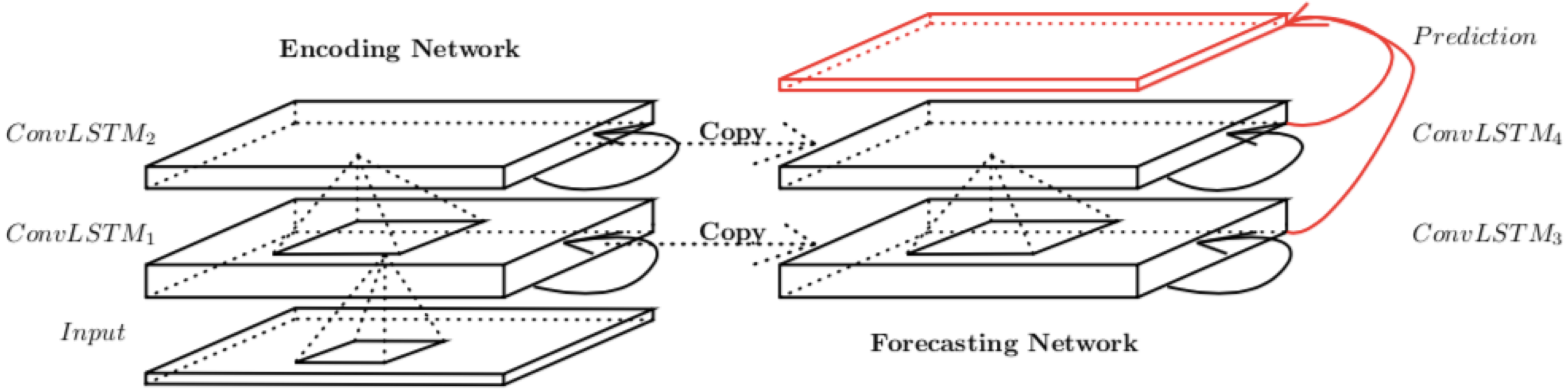
Use Hadamard product to keep the constant error carousel (CEC) property of cells

ConvLSTM – Illustration



- FC-LSTM can be viewed as a special case of ConvLSTM with all features standing on a single cell. (Size = 1x1, Kernel = 1)
- Using 'state of the outside world' for boundary grids. Zero padding is used to indicate 'total ignorance' of the outside.

ConvLSTM – EF Structure



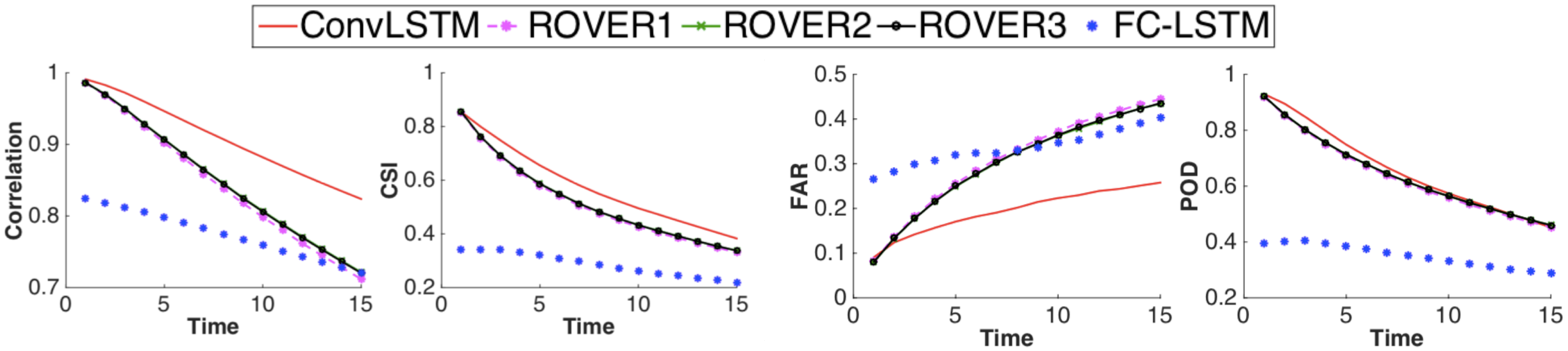
ConvLSTM – Radar Echo Dataset

- Z-R Relationship: $\mathrm{dBZ} = 10\log a + 10b\log R \qquad a = 118.239, b = 1.5241$
- 97 rainy days from 2011 to 2013 in Hong Kong
- Applies disk filter and rescales the images to be 100x100
- Number of train/val/test sequences: 8148/2037/2037
- 5 for input and 15 for prediction
- Scores: 0.5 mm threshold
  - CSI = TP / (TP + FN + FP)
  - FAR = FP / (TP + FP)
  - POD = TP / (TP + FN)
  - Correlation: $\frac{\sum_{i,j} P_{i,j} T_{i,j}}{\sqrt{(\sum_{i,j} P_{i,j}^2)(\sum_{i,j} T_{i,j}^2)} + \varepsilon}$

	Truth = 1	Truth = 0
Pred = 1	TP	FP
Pred = 0	FN	TN

ConvLSTM – Nowcasting Performance

Model	Rainfall-MSE	CSI	FAR	POD	Correlation
ConvLSTM(3x3)-3x3-64-3x3-64	**1.420**	**0.577**	**0.195**	**0.660**	**0.908**
Rover1	1.712	0.516	0.308	0.636	0.843
Rover2	1.684	0.522	0.301	0.642	0.850
Rover3	1.685	0.522	0.301	0.642	0.849
FC-LSTM-2000-2000	1.865	0.286	0.335	0.351	0.774



Outline

- Introduction
  - What is STSF? Why is it important?
  - My research topic: Deep learning for STSF
- Architectures for STSF-RG
  - Background – Deep Learning, Precipitation Nowcasting
  - ConvLSTM
  - TrajGRU
  - HKO-7
- Architectures for STSF-IG
  - Background
  - GaAN
  - GGRU
- Conclusion & Future Work

TrajGRU – Motivation

- ConvLSTM is not optimal!
- Convolution applies a location-invariant filter. Convolutional recurrence lacks the ability to model location-variant spatiotemporal correlation patterns.

- Propose a model to actively learn a location-variant connection structure.

TrajGRU – ConvGRU Recap

- ConvGRU

$$\mathcal{Z}_t = \sigma(\mathcal{W}_{xz} * \mathcal{X}_t + \mathcal{W}_{hz} * \mathcal{H}_{t-1}),$$

$$\mathcal{R}_t = \sigma(\mathcal{W}_{xr} * \mathcal{X}_t + \mathcal{W}_{hr} * \mathcal{H}_{t-1}),$$

$$\boxed{\mathcal{H}'_t = f(\mathcal{W}_{xh} * \mathcal{X}_t + \mathcal{R}_t \circ (\mathcal{W}_{hh} * \mathcal{H}_{t-1})),}$$

$$\mathcal{H}_t = (1 - \mathcal{Z}_t) \circ \mathcal{H}'_t + \mathcal{Z}_t \circ \mathcal{H}_{t-1}.$$

- Convolution applies a location-invariant filter

$$\mathcal{H}'_{t,:,i,j} = f(\mathbf{W}_{hh}\text{concat}(\langle \mathcal{H}_{t-1,:,p,q} \mid (p,q) \in \underbrace{\boxed{\mathcal{N}^h_{i,j}}}_{\text{Fixed!}} \rangle)) = f(\sum_{l=1}^{|\mathcal{N}^h_{i,j}|} \mathbf{W}^l_{hh} \mathcal{H}_{t-1,:,p_{l,i,j},q_{l,i,j}})$$

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

TrajGRU – From ConvGRU to TrajGRU

- Our goal: neighborhood set varies at different locations + timestamps.

Size of the neighborhood set

$$\mathcal{H}'_{t,:,i,j} = f\left(\sum_{l=1}^{\boxed{L}} \mathbf{W}_{hh}^{l} \mathcal{H}_{t-1,:,\underline{p_{l,i,j}(\theta),q_{l,i,j}(\theta)}}\right),$$

$(p_{l,i,j}(\theta), q_{l,i,j}(\theta))$ is the lth neighborhood

- Indexing will be non-differentiable in general. We choose to use Bilinear Sampling to warp the pixels instead (soft attention)
- $I_{c,y+dy,x+dx} = \sum_{m=1}^{H} \sum_{n=1}^{W} I_{c,m,n} \max(0, 1 - |y + dy - m|) \max(0, 1 - |x + dx - n|)$
- TrajGRU uses a parameterized network to output L (dy, dx)s for all the locations (y, x).

TrajGRU – Formula & Illustration

$$\mathcal{U}_t, \mathcal{V}_t = \underline{\gamma(\mathcal{X}_t, \mathcal{H}_{t-1})},$$

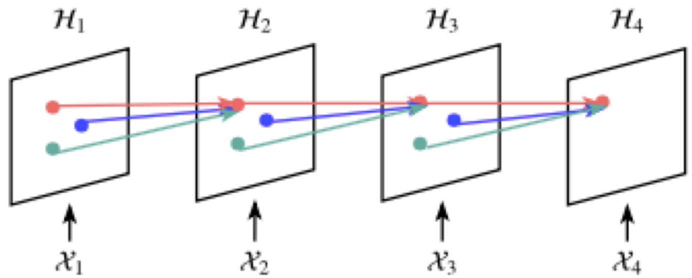
γ is a subnetwork with two conv-layers.
Generates L flow-maps.

$$\mathcal{Z}_t = \sigma(\mathcal{W}_{xz} * \mathcal{X}_t + \sum_{l=1}^{L} \mathcal{W}_{hz}^{l} * \text{warp}(\mathcal{H}_{t-1}, \mathcal{U}_{t,l}, \mathcal{V}_{t,l})),$$

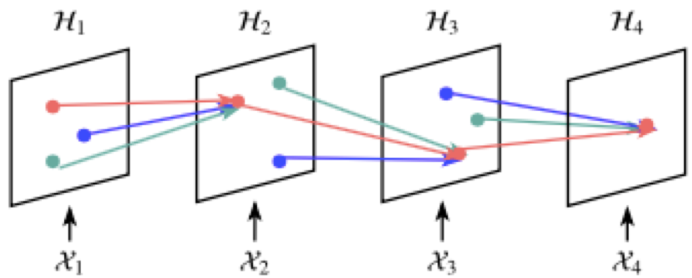
$$\mathcal{R}_t = \sigma(\mathcal{W}_{xr} * \mathcal{X}_t + \sum_{l=1}^{L} \mathcal{W}_{hr}^{l} * \text{warp}(\mathcal{H}_{t-1}, \mathcal{U}_{t,l}, \mathcal{V}_{t,l})),$$

$$\mathcal{H}_t' = f(\mathcal{W}_{xh} * \mathcal{X}_t + \mathcal{R}_t \circ (\sum_{l=1}^{L} \mathcal{W}_{hh}^{l} * \text{warp}(\mathcal{H}_{t-1}, \mathcal{U}_{t,l}, \mathcal{V}_{t,l}))),$$

$$\mathcal{H}_t = (1 - \mathcal{Z}_t) \circ \mathcal{H}_t' + \mathcal{Z}_t \circ \mathcal{H}_{t-1}.$$



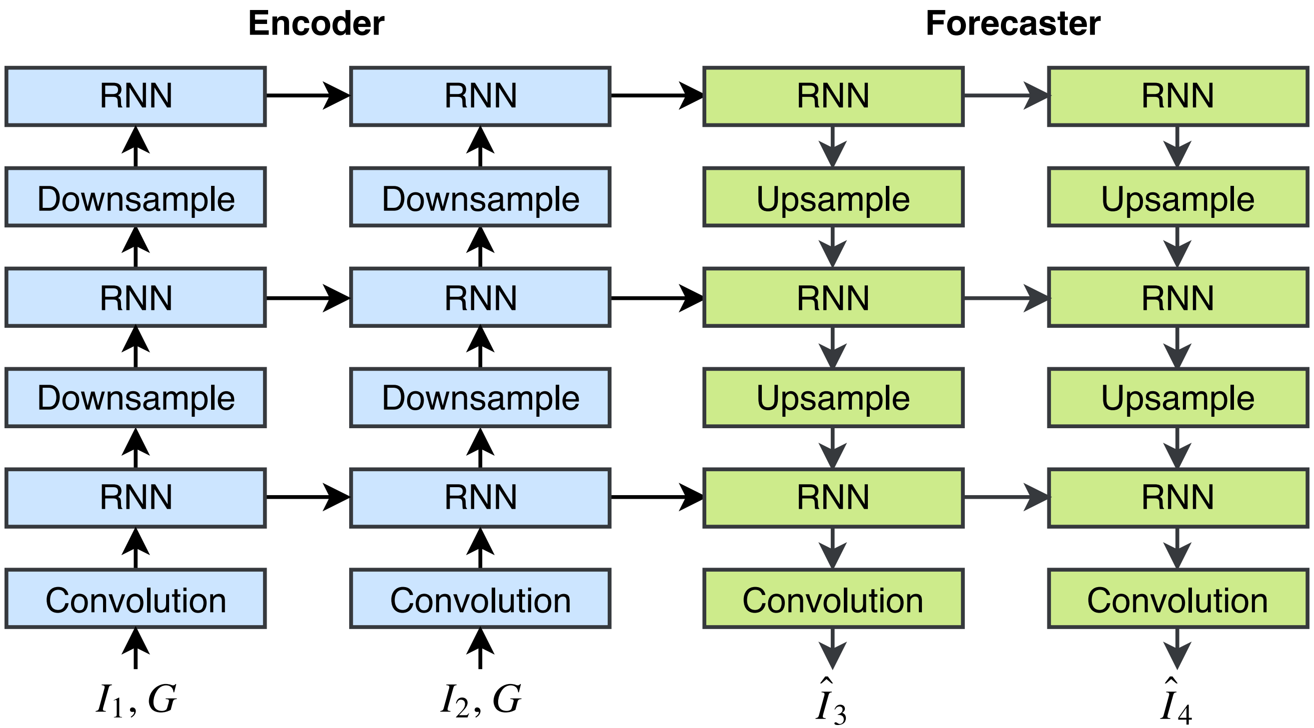
(a) For convolutional RNN, the recurrent connections are fixed over time.



(b) For trajectory RNN, the recurrent connections are dynamically determined.

TrajGRU – EF Structure

Reverse the direction of the links in the forecaster



Encoding:

Low-level to High-level

Forecasting:

High-level guides Low-level

Any valid RNN, e.g, ConvGRU, TrajGRU

TrajGRU – Findings by Visualizing the Links

- Encoder: local spatiotemporal structure → global spatiotemporal structure
- Forecaster: Coarse global motion structure → Finer details

Outline

- Introduction
  - What is STSF? Why is it important? Why choose this research topic?
- Architectures for STSF-RG
  - Background – Deep Learning, Precipitation Nowcasting
  - ConvLSTM
  - TrajGRU
  - HKO-7
- Architectures for STSF-IG
  - Background
  - GaAN
  - GGRU
- Conclusion & Future Work

HKO-7 – Motivation

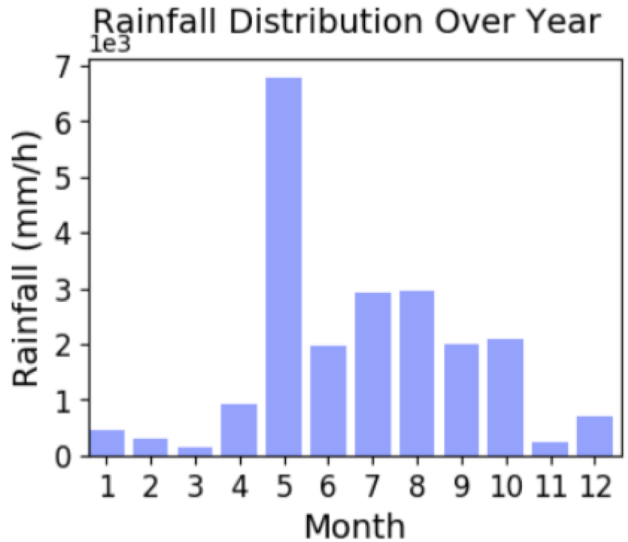
- Evaluated in a small dataset (97 days) and only the 0.5 mm/h threshold. Far from real-world requirement.
- The whole area “Deep Learning for Precipitation Nowcasting” is still in its early stage! We are still not clear how models should be evaluated to meet the need of real-world applications.
- Propose the HKO-7 benchmark to fill the gap
  - 7-year dataset
  - New evaluation scores
  - New evaluation protocols

[Shi et al., 2017]

HKO-7 – Dataset

- The radar data from 2009 to 2015 collected by HKO (only use days that have rain gauge record)
- Altitude: 2km, Spatial Range: 512km * 512 km, Resolution: 480 * 480

	Train	Validate	Test
Years	2009-2014	2009-2014	2015
#Days	812	50	131
#Frames	192,168	11,736	31,350



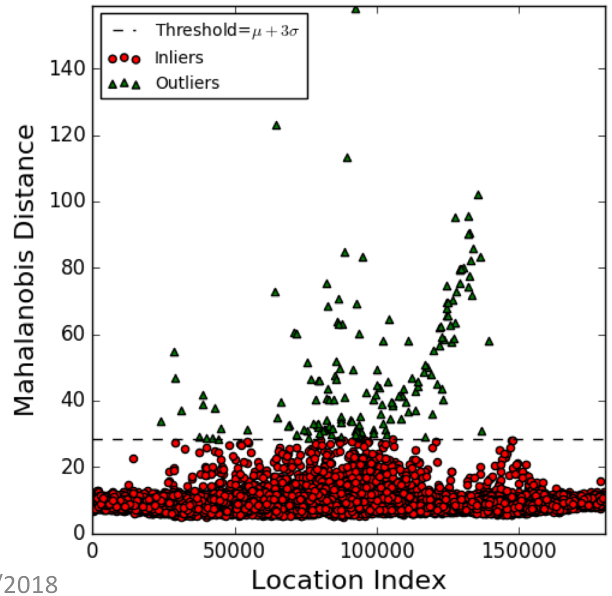
HKO-7 – Dataset

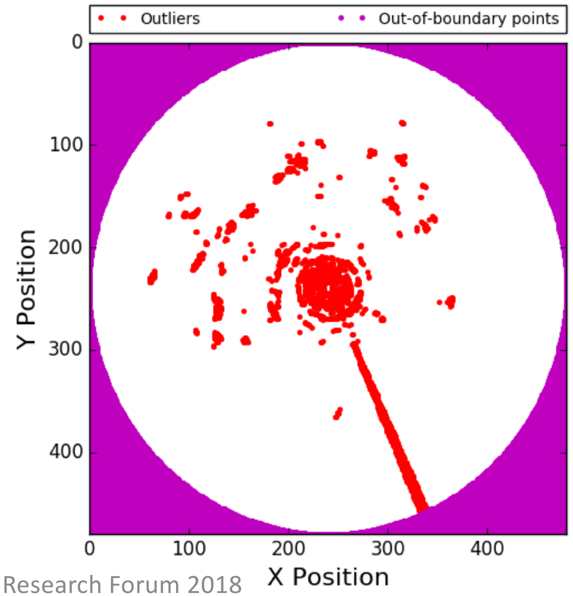
- Rain-rate statistics

Rain Rate (mm/h)	Proportion (%)	Rainfall Level
$0 \leq x < 0.5$	90.25	No / Hardly noticeable
$0.5 \leq x < 2$	4.38	Light
$2 \leq x < 5$	2.46	Light to moderate
$5 \leq x < 10$	1.35	Moderate
$10 \leq x < 30$	1.14	Moderate to heavy
$30 \leq x$	0.42	Rainstorm warning

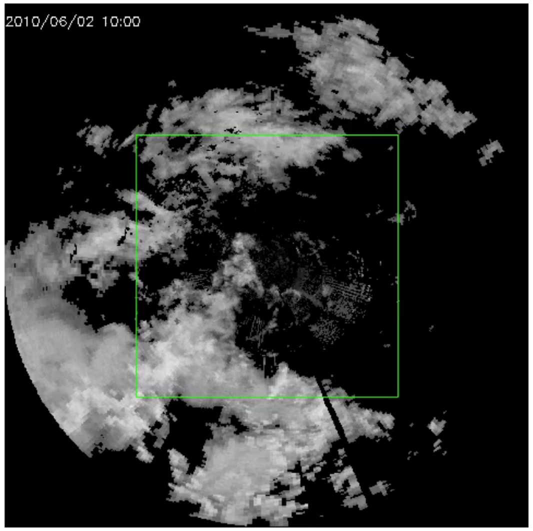
HKO-7 – Remove Noise in Data

- Radar data are noisy due to factors like ground clutter, sun spikes, sea clutter, etc.
- We detect the outliers based on the ratio of pixel values.

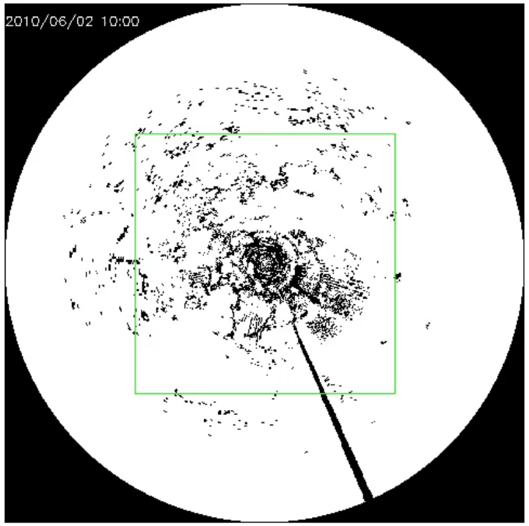




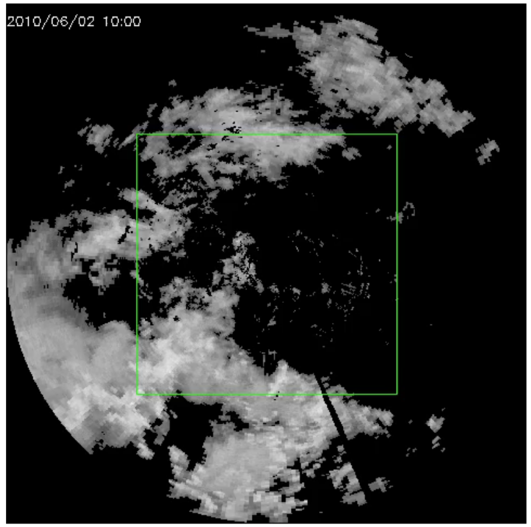
HKO-7 – Remove Noise in Data



Raw



Noise Mask



Filtered

HKO-7 – Evaluation Scores

- Heavier rainfall occurs less often but has a higher real-world impact
  - New scores: B-MSE, B-MAE
  - Assign larger weights to heavier rainfalls
  - Differentiable, can be used in training
  - Higher correlation with the classical scores: CSI, HSS

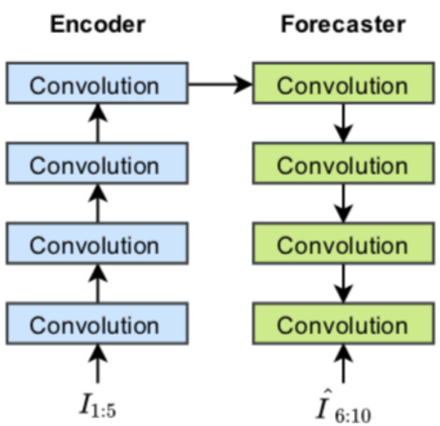
$$w(x) = \begin{cases} 1, & x < 2 \\ 2, & 2 \leq x < 5 \\ 5, & 5 \leq x < 10 \\ 10, & 10 \leq x < 30 \\ 30, & x \geq 30 \end{cases}$$

HKO-7 – Evaluation Methodology

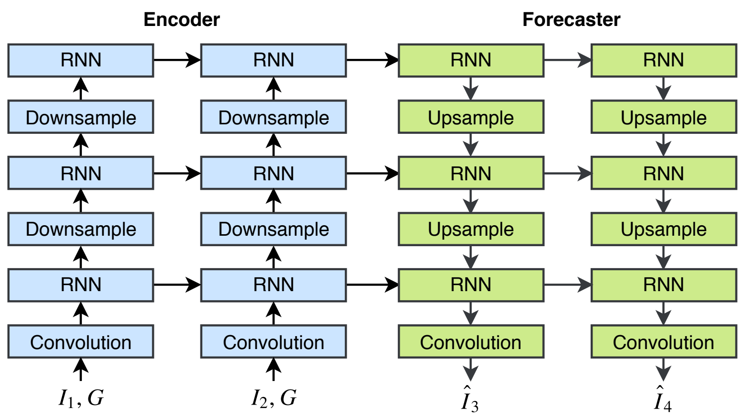
- In real-life, we can actively adapt to newly emerging patterns
  - Offline setting: Use 5 frames to predict 20 frames. Cannot use previous observations
  - Online setting: Use 5 frames to predict 20 frames. Can do online updating.

HKO-7 – Evaluated Algorithms

- No-Deep: Last-Frame, ROVER, ROVER-nonlinear
- Deep: Conv2D, Conv3D, ConvGRU, TrajGRU
- Online setting for deep models
  - We use AdaGrad with lr=1E-4 to fine-tune the models in online setting.



Conv2D
Conv3D



ConvGRU
TrajGRU

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

HKO-7 – Evaluation Results

Algorithms	CSI ↑					HSS ↑					B-MSE ↓	B-MAE ↓
	$r \geq 0.5$	$r \geq 2$	$r \geq 5$	$r \geq 10$	$r \geq 30$	$r \geq 0.5$	$r \geq 2$	$r \geq 5$	$r \geq 10$	$r \geq 30$		
Offline Setting												
Last Frame	0.4022	0.3266	0.2401	0.1574	0.0692	0.5207	0.4531	0.3582	0.2512	0.1193	15274	28042
ROVER + Linear	0.4762	0.4089	0.3151	0.2146	0.1067	0.6038	0.5473	0.4516	0.3301	0.1762	11651	23437
ROVER + Non-linear	0.4655	0.4074	0.3226	0.2164	0.0951	0.5896	0.5436	0.4590	0.3318	0.1576	10945	22857
2D CNN	0.5095	0.4396	0.3406	0.2392	0.1093	0.6366	0.5809	0.4851	0.3690	0.1885	7332	18091
3D CNN	0.5109	0.4411	0.3415	0.2424	0.1185	0.6334	0.5825	0.4862	0.3734	0.2034	7202	17593
ConvGRU-nobal	0.5476	0.4661	0.3526	0.2138	0.0712	0.6756	0.6094	0.4981	0.3286	0.1160	9087	19642
ConvGRU	0.5489	0.4731	0.3720	0.2789	0.1776	0.6701	0.6104	0.5163	0.4159	0.2893	5951	15000
TrajGRU	**0.5528**	**0.4759**	**0.3751**	**0.2835**	**0.1856**	**0.6731**	**0.6126**	**0.5192**	**0.4207**	**0.2996**	**5816**	**14675**
Online Setting												
2D CNN	0.5112	0.4363	0.3364	0.2435	0.1263	0.6365	0.5756	0.4790	0.3744	0.2162	6654	17071
3D CNN	0.5106	0.4344	0.3345	0.2427	0.1299	0.6355	0.5736	0.4766	0.3733	0.2220	6690	16903
ConvGRU	0.5511	0.4737	0.3742	0.2843	0.1837	0.6712	0.6105	0.5183	0.4226	0.2981	5724	14772
TrajGRU	**0.5563**	**0.4798**	**0.3808**	**0.2914**	**0.1933**	**0.6760**	**0.6164**	**0.5253**	**0.4308**	**0.3111**	**5589**	**14465**

- ALL deep models outperform optical-flow based models when trained with B-MSE + B-MAE
- TrajGRU attains the BEST overall performance among all the deep learning models.
- With online fine-tuning, models CONSISTENTLY perform better.

HKO-7 – Evaluation Results

- B-MSE/B-MAE correlates better with CSI/HSS at multiple thresholds than MSE/MAE. We calculate the Kendall's tau between metrics.

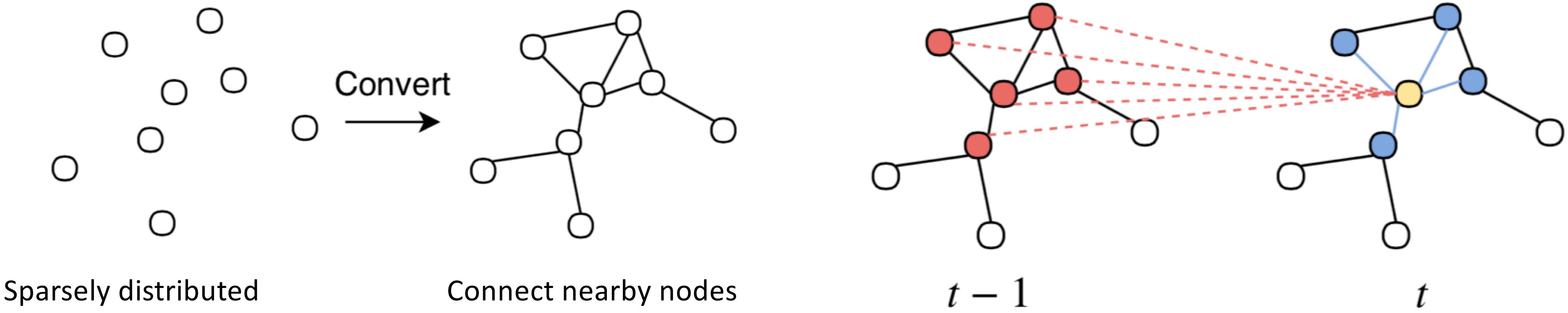
Skill Scores	CSI					HSS				
	$r \geq 0.5$	$r \geq 2$	$r \geq 5$	$r \geq 10$	$r \geq 30$	$r \geq 0.5$	$r \geq 2$	$r \geq 5$	$r \geq 10$	$r \geq 30$
MSE	-0.24	-0.39	-0.39	-0.07	-0.01	-0.33	-0.42	-0.39	-0.06	0.01
MAE	-0.41	-0.57	-0.55	-0.25	-0.27	-0.50	**-0.60**	-0.55	-0.24	-0.26
B-MSE	-0.70	-0.57	**-0.61**	**-0.86**	-0.84	-0.62	-0.55	**-0.61**	**-0.86**	-0.84
B-MAE	**-0.74**	**-0.59**	-0.58	-0.82	**-0.92**	**-0.67**	-0.57	-0.59	-0.83	**-0.92**

Outline

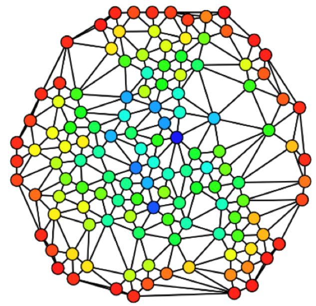
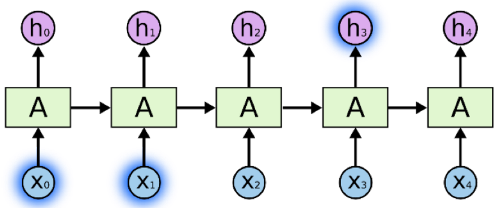
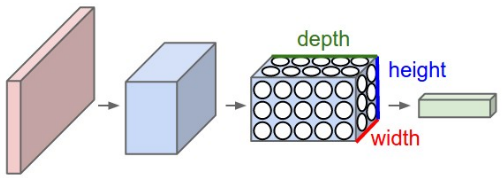
- Introduction
  - What is STSF? Why is it important? Why choose this research topic?
- Architectures for STSF-RG
  - Background – Deep Learning, Precipitation Nowcasting
  - ConvLSTM
  - TrajGRU
  - HKO-7
- Architectures for STSF-IG
  - Background
  - GaAN
  - GGRU
- Conclusion & Future Work

STSF-IG – General Strategy

- For STSF-IG, the stations are sparsely distributed! [Li et al., 2018]
- Construct a spatiotemporal graph based on these stations
- Deep learning on graphs



Deep Learning on Graphs – Graph Convolution

Text	Image	Graph
虽然 北 风 呼啸 *Although north wind howls*		
		
Recurrent Neural Network	Convolutional Neural Network	Graph Convolutional Network

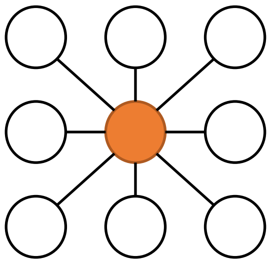
Deep Learning on Graphs – Graph Convolution

- Generalized convolution: Regular Grid → Graph Structure
- Spectral Approach & Spatial Approach
- Spectral Approach:
  - Convolution Theorem: $X * Y = F^{-1}(F(X) \circ F(Y))$
  - Graph Fourier Transform:
    - $F(X) = U^T X$ → $X * Y = U((U^T X) \circ (U^T Y))$
    - Eigen-value decomposition of the graph Laplacian: $L = U \Lambda U^T, L = I - D^{1/2} A D^{1/2}$
    - $f_\theta(X) = U((U^T X) \circ \theta) = U\ diag(\theta) U^T X$
    - High computational cost!! Can be accelerated but actually leads to the spatial approach.

[Bruna et al., 2014] [Duvenaud et al., 2015] [Kipf & Welling, 2017] [Zhang et al., 2018]

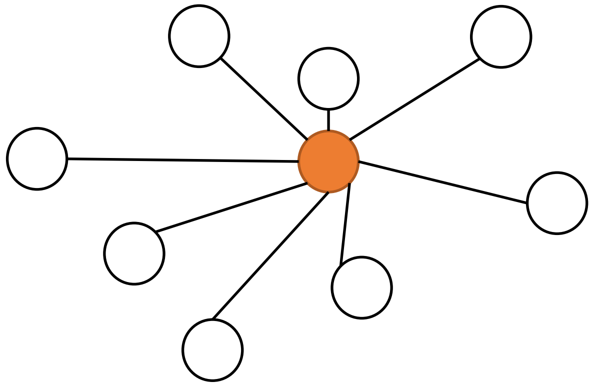
Deep Learning on Graphs – Graph Convolution

- Spatial Approach:
  - Aggregate information from the local neighborhood + share parameters
  - Graph aggregator: $y_i = r_\theta(x_i, \{z_{N_i}\})$



$y_i = f(W x_{z_{N_i}} + b)$

Spatial Convolution



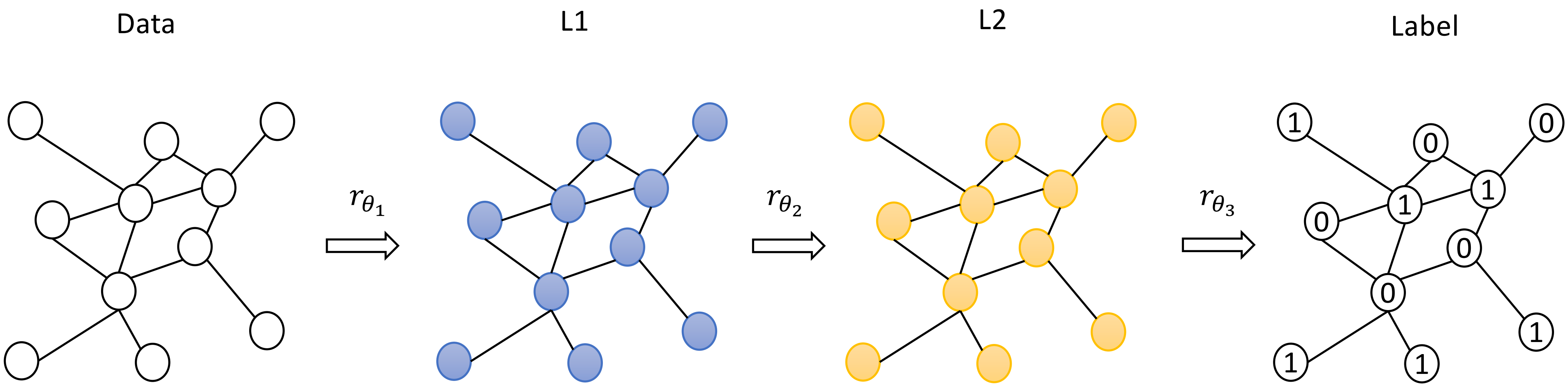
Graph Convolution

Permutation-invariant

Different sizes of N_i

Mean Pooling, Max Pooling, …

Deep Learning on Graphs – Graph Convolutional Networks



Node Classification

Outline

- Introduction
  - What is STSF? Why is it important? Why choose this research topic?
- Architectures for STSF-RG
  - Background – Deep Learning, Precipitation Nowcasting
  - ConvLSTM
  - TrajGRU
  - HKO-7
- Architectures for STSF-IG
  - Background
  - GaAN
  - GGRU
- Conclusion & Future Work

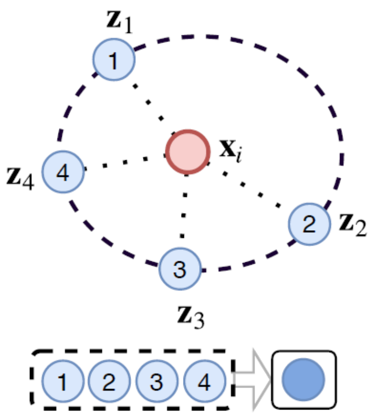
GaAN – Motivation

- Performance of graph convolutional neural networks is strongly related to the graph aggregator [Hamilton et al., 2017]
- Investigate the performance of different graph aggregators
  - Inductive node classification on large graphs
- Propose a new attention-based aggregator called Gated Attention Networks (GaAN)
  - Traditional multi-head attention based aggregator treats each head equally
  - Soft gates to control the attention heads' importance

[Zhang et al., 2018]

GaAN – Types of Graph Aggregators

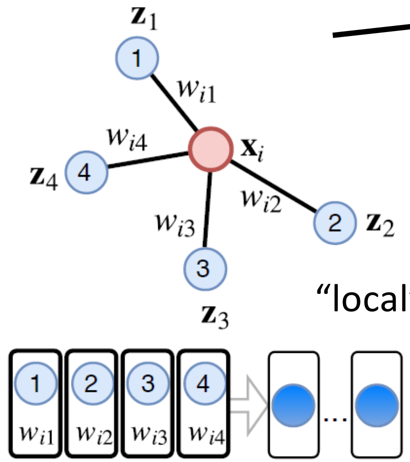
Multi-head



Pooling-based

$$\mathbf{y}_i = \phi_o(\mathbf{x}_i \oplus \mathrm{pool}_{j\in\mathcal{N}_i}(\phi_v(\mathbf{z}_j)))$$

[Hamilton et al., 2017]



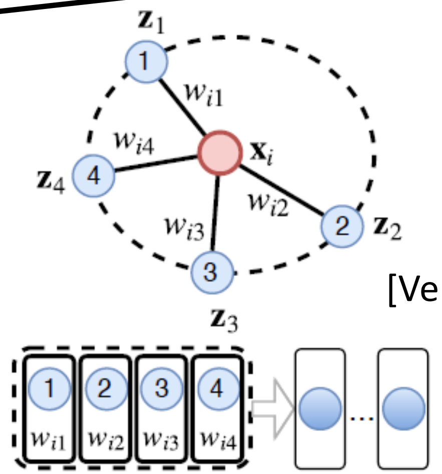
"local" context

Pairwise-sum

$$\mathbf{y}_i = \phi_o(\mathbf{x}_i \oplus \big\Vert_{k=1}^{K} \sum_{j\in\mathcal{N}_i} w_{i,j}^{(k)} \phi_v^{(k)}(\mathbf{z}_j)),$$

$$w_{i,j}^{(k)} = \phi_w^{(k)}(\mathbf{x}_i, \mathbf{z}_j).$$

[Liang et al., 2016]



[Veličković et al., 2018]

"global" context

Attention-based

$$\mathbf{y}_i = \mathrm{FC}_{\theta_o}(\mathbf{x}_i \oplus \big\Vert_{k=1}^{K} \sum_{j\in\mathcal{N}_i} w_{i,j}^{(k)} \mathrm{FC}^h_{\theta_v^{(k)}}(\mathbf{z}_j)),$$

$$w_{i,j}^{(k)} = \frac{\exp(\phi_w^{(k)}(\mathbf{x}_i, \mathbf{z}_j))}{\sum_{l=1}^{|\mathcal{N}_i|} \exp(\phi_w^{(k)}(\mathbf{x}_i, \mathbf{z}_l))},$$

$$\phi_w^{(k)}(\mathbf{x}, \mathbf{z}) = \langle \mathrm{FC}_{\theta_{xa}^{(k)}}(\mathbf{x}), \mathrm{FC}_{\theta_{za}^{(k)}}(\mathbf{z}) \rangle.$$

GaAN – Limitations of Standard Multi-head Attention

- Head → Subspace
- Traditional multi-head attention treats all subspaces equally
- For some nodes, certain subspaces are more important.
  - E.g., 7 types of relationships in total, each node has only 3 valid relationships
  - Forcing all nodes to use all 7 aggregated vectors will mislead the network
- GaAN adds soft gates on the attention heads to control their relative importance.

GaAN – Gated Attention Networks

Number of heads

Importance

$$\mathbf{y}_i = \mathrm{FC}_{\theta_o}\Big(\mathbf{x}_i \oplus \mathop{\Big\Vert}_{k=1}^{\boxed{K}} \Big(\boxed{g_i^{(k)}} \sum_{j \in \mathcal{N}_i} w_{i,j}^{(k)} \mathrm{FC}^{h}_{\theta_v^{(k)}}(\mathbf{z}_j)\Big)\Big),$$

Attention head

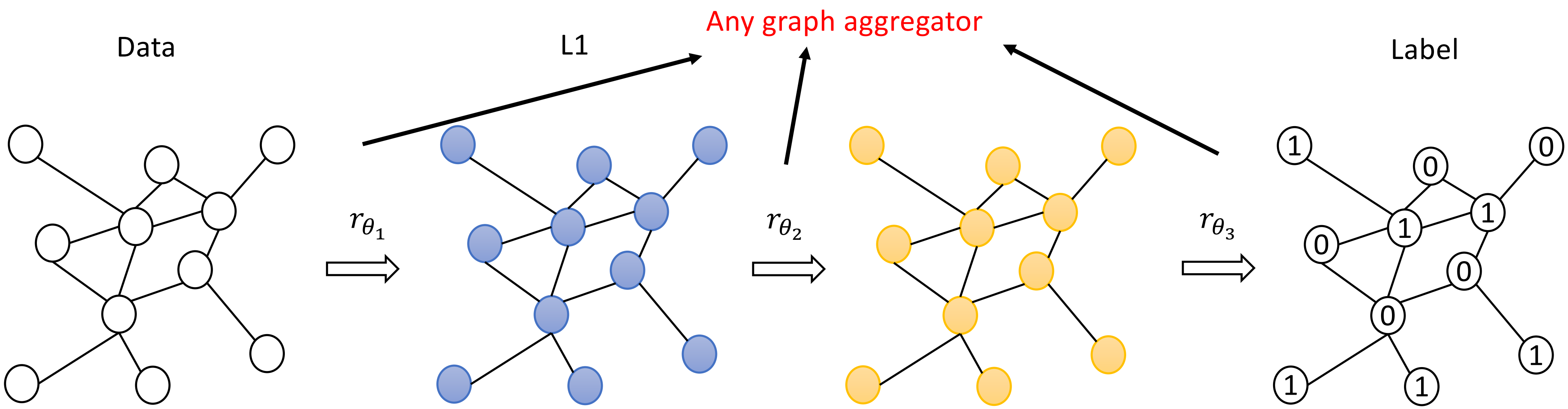
$$\mathbf{g}_i = [g_i^{(1)}, \ldots, g_i^{(K)}] = \psi_g(\mathbf{x}_i, \mathbf{z}_{\mathcal{N}_i}),$$

- g_i is between 0 (low importance) and 1 (high importance)
- We use a small convolutional network to compute g_i

$$\mathbf{g}_i = \mathrm{FC}^{\sigma}_{\theta_g}\Big(\mathbf{x}_i \oplus \max_{j \in \mathcal{N}_i}(\{\mathrm{FC}_{\theta_m}(\mathbf{z}_j)\}) \oplus \frac{\sum_{j \in \mathcal{N}_i} \mathbf{z}_j}{|\mathcal{N}_i|}\Big)$$

GaAN – Inductive Node Classification

- Compare the performance of different graph aggregators
- Goal: classify unseen testing nodes



GaAN – Datasets

- PPI: Protein-protein interaction graph. Human tissue.
- Reddit: Posts are connected if the same user commented on them.

Data	#Nodes	#Edges	#Fea	#Classes	
PPI	56.9K	806.2K	50	121(multi)	Multi-label
Reddit	233.0K	114.6M	602	41(single)	Multi-class

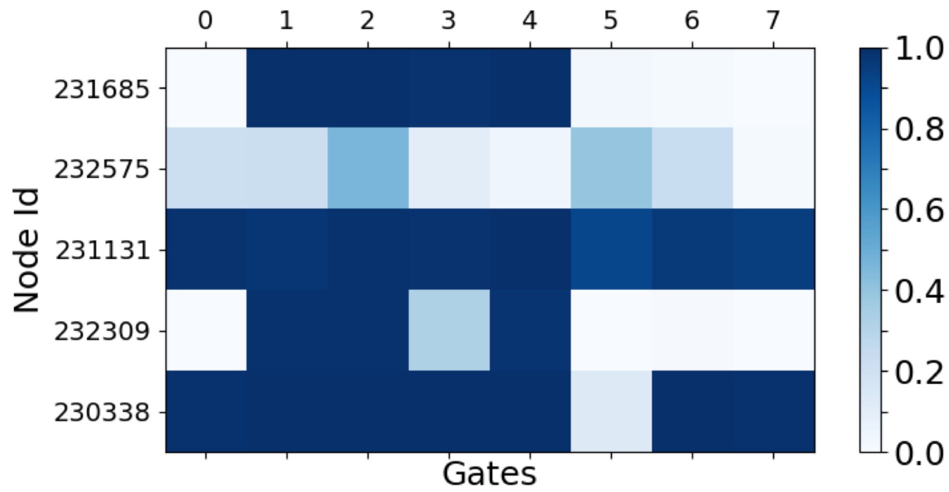
So far the largest dataset

GaAN – Main Results

	Models / Datasets	PPI	Reddit
[Hamilton et al., 2017]	GraphSAGE [38]	(61.2)[1]	95.4
[Veliˇckovíc et al., 2018]	GAT [96]	97.3 ± 0.2	-
[Chen et al., 2018]	Fast GCN [14]	-	93.7
Implemented by us	2-Layer FNN	54.07±0.06	73.58±0.09
	Avg. pooling	96.85±0.19	95.78±0.07
	Max pooling	98.39±0.05	95.62±0.03
	Pairwise+sigmoid	98.39±0.05	95.86±0.08
	Pairwise+tanh	98.32±0.18	95.80±0.03
	Attention-only	98.46±0.09	96.19±0.07
	GaAN	**98.71±0.02**	**96.36±0.03**

SOTA performance

GaAN – Visualizing the Gate Values



Reddit Dataset

- The gate-generation network can be learned to assign different importance to different heads.

Outline

- Introduction
  - What is STSF? Why is it important? Why choose this research topic?
- Architectures for STSF-RG
  - Background – Deep Learning, Precipitation Nowcasting
  - ConvLSTM
  - TrajGRU
  - HKO-7
- Architectures for STSF-IG
  - Background
  - GaAN
  - GGRU
- Conclusion & Future Work

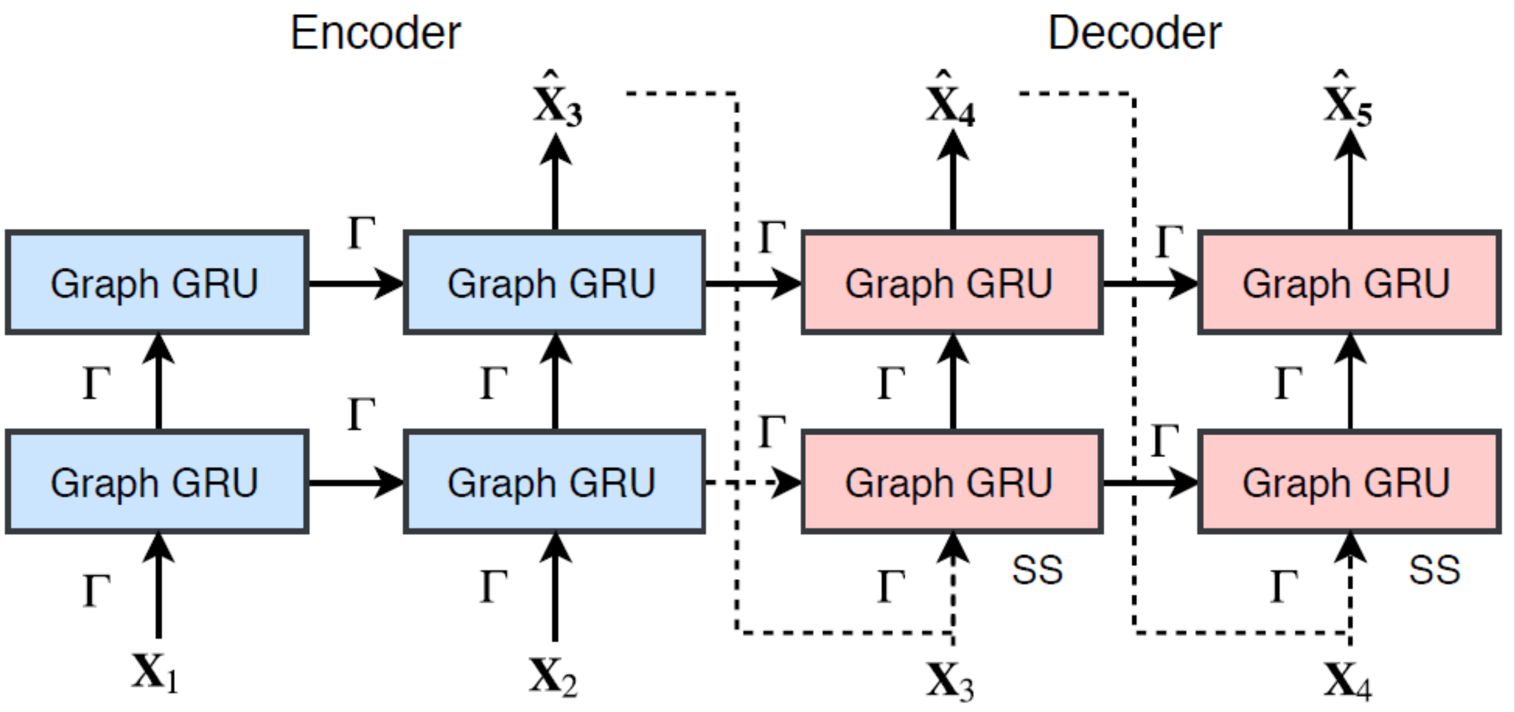
GGRU – RNNs for Spatiotemporal Graphs

- Unified framework to convert graph aggregators to RNNs for spatiotemporal graphd
- Graph GRU (GGRU)

$$
\begin{aligned}
\mathbf{U}_t &= \sigma(\Gamma_{\Theta_{xu}}(\mathbf{X}_t, \mathbf{X}_t; \mathcal{G}_s) + \Gamma_{\Theta_{hu}}(\mathbf{X}_t \oplus \mathbf{H}_{t-1}, \mathbf{H}_{t-1}; \mathcal{G}_t)),\\
\mathbf{R}_t &= \sigma(\Gamma_{\Theta_{xr}}(\mathbf{X}_t, \mathbf{X}_t; \mathcal{G}_s) + \Gamma_{\Theta_{hr}}(\mathbf{X}_t \oplus \mathbf{H}_{t-1}, \mathbf{H}_{t-1}; \mathcal{G}_t)),\\
\mathbf{H}'_t &= h(\Gamma_{\Theta_{xh}}(\mathbf{X}_t, \mathbf{X}_t; \mathcal{G}_s) + \mathbf{R}_t \circ \Gamma_{\Theta_{hh}}(\mathbf{X}_t \oplus \mathbf{H}_{t-1}, \mathbf{H}_{t-1}; \mathcal{G}_t)),\\
\mathbf{H}_t &= (1 - \mathbf{U}_t) \circ \mathbf{H}'_t + \mathbf{U}_t \circ \mathbf{H}_{t-1}.
\end{aligned}
$$

  - States/Inputs are all graphs
  - $\Gamma_\Theta(\mathbf{X}, \mathbf{Z}; \mathcal{G})$ means applying the graph aggregator for all nodes in $\mathcal{G}$,
  - $\mathbf{X}_t$: input features, $\mathbf{H}_t$: hidden states of the nodes
  - $\mathbf{U}_t$: the update gate, $\mathbf{R}_t$: the reset gate

GGRU – EF Structure for STSF-IG



Outline

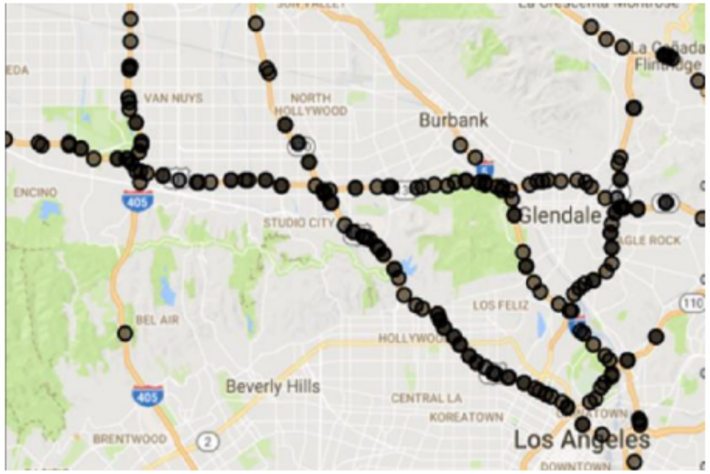
- Introduction
  - What is STSF? Why is it important? Why choose this research topic?
- Architectures for STSF-RG
  - Background – Deep Learning, Precipitation Nowcasting
  - ConvLSTM
  - TrajGRU
  - HKO-7
- Architectures for STSF-IG
  - Background
  - GaAN
  - GGRU
- Conclusion & Future Work

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Conclusion

- Architectures for STSF-RG
  - ConvLSTM
    - Convolutional recurrence
    - First ML solution for precipitation nowcasting
  - TrajGRU
    - Actively learns the recurrent connection
  - HKO-7
    - First large-scale benchmark for precipitation nowcasting
- Architectures for STSF-IG
  - GaAN
    - Soft gates to control each attention heads' importance
    - SOTA performance for inductive node classification on large graph
  - GGRU
    - Unified framework for converting graph aggregator to RNN for STSF-IG

Future Work



- Use GGRU for traffic speed forecasting
- Add a global external memory structure to the existing models
- Handle uncertainty by using probabilistic encoder/forecaster

$$\mathbf{s} = f(\mathcal{F}_t; \theta_1), \qquad \hat{\mathbf{X}}_{t+1:t+L} = g(\mathbf{s}; \theta_2).$$

$$\Rightarrow$$

$$\mathbf{s} \sim f(\mathcal{F}_t; \theta_1), \qquad \hat{\mathbf{X}}_{t+1:t+L} \sim \pi_g(\mathbf{s}; \theta_2).$$

Related Publications

- Introduction:
  - [1] **Xingjian Shi** and Dit-Yan Yeung. Machine Learning for Spatiotemporal Sequence Forecasting: A Survey. In Submission. arxiv version: https://arxiv.org/pdf/1808.06865.pdf
- Architectures for STSF-RG:
  - [2] **Xingjian Shi**, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong and Wang-chun Woo. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. in NIPS 2015.
  - [3] **Xingjian Shi**, Zhihan Gao, Leonard Lausen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong and Wang-chun Woo. Deep Learning for Precipitation Nowcasting: A Benchmark and A New Model. in NIPS 2017. **(Accepted as Spotlight)**
- Architectures for STSF-IG
  - [4] Jiani Zhang*, **Xingjian Shi***, Junyuan Xie, Hao Ma, Irwin King and Dit-Yan Yeung. GaAN: Gated Attention Networks for Learning on Large and Spatiotemporal Graphs. in UAI 2018. (*** indicates equal contribution.**)

Thank You