Sequential Images Prediction Using Convolutional LSTM with Application in Precipitation Nowcasting

Mingkuan Wu

Supervisor: Prof. Gemai Chen

Department of Mathematics and Statistics
University of Calgary

mingkuan.wu@ucalgary.ca

August 15, 2019

Overview

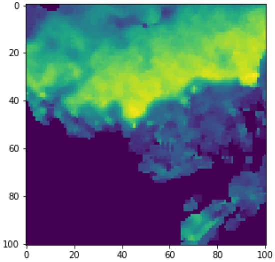
1. Introduction to Precipitation Nowcasting
2. Motivation and Model Development
3. Experiments
4. Conclusions

Overview

Goal of Precipitation Nowcasting

Goal

Give precise and timely prediction of rainfall intensity in a local region over a relatively short period of time (e.g., 0-6 hours) based on radar echo maps, rain gauge and other data.



Example of radar echo maps.

Applications:

- Provide road information under extreme weather conditions.
- Guidance for aviation at airport.
- Provide rainfall warning.

Classic Approaches

- Numerical Weather Prediction (NWP) based method.
  - Build a atmosphere model using several physical equations.
  - More accurate in the longer terms.
  - The first 2 hours of the model forecasts may not be available.
- Radar echo extrapolation based method.
  - Step 1: Optical flow estimation + Step 2: Radar echo extrapolation.
  - More accurate in the first 2 hours.
  - Example: ROVER algorithm by Hong Kong Observatory.

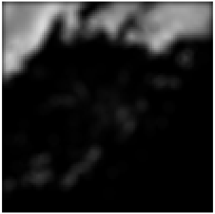
Overview

Motivation

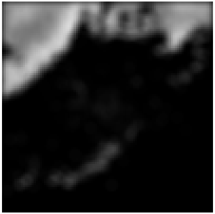
Limitation of radar echo extrapolation based method:

- Optical flow estimation & radar echo extrapolation are separated which will result in accumulative errors (i.e. Fail in decaying & growing cases of radar echo maps).

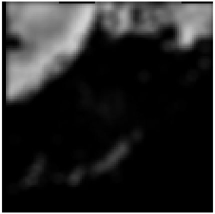
Failure example of predicting growing radar echo maps using ROVER algorithm:



First 5 input frames



15 ground truth frames



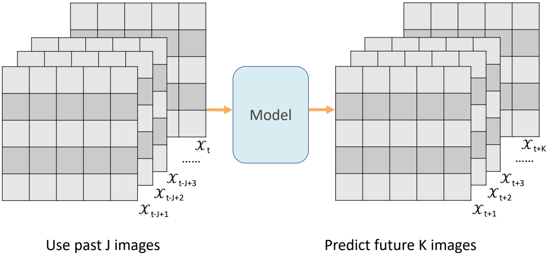
15 ROVER predictions

Machine Learning Approach

- Machine learning approach requirements:
  - Enough data.
  - End-to-end structure.
- Machine learning approach for precipitation nowcasting is not trivial:
  - Multi-step prediction
    Need a sequence of accurate rainfall intensity information at each time point in the short future.
  - Spatialtemporal data
    Take advantage of the spatial correlation within each frame (i.e. similar trends of neighbors) and temporal correlation among a sequence of input frames.

Formulation of Precipitation Nowcasting Problem

- Periodic observations taken from a dynamic system. Each radar echo map is a spatial $N_1 \times N_2$ grid. Each pixel has an integer between 0 and 255 representing the rainfall intensity.



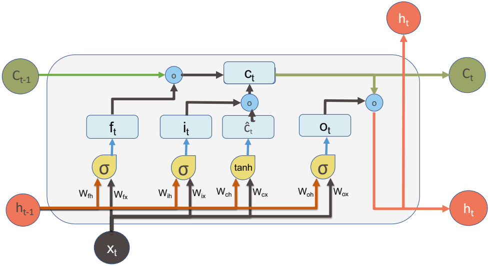
- Predict the most likely length-K sequence in the future given the previous length-J sequence:

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

How to perform multi-step prediction

Long Short-Term Memory network (LSTM) for sequential data

LSTM block at time t:



updated memory: $\tilde{\mathbf{c}}_t = tanh(\mathbf{W}_{cx}\mathbf{x}_t + \mathbf{W}_{ch}\mathbf{h}_{t-1} + \mathbf{b}_c)$

update gate: $\mathbf{i}_t = \sigma(\mathbf{W}_{ix}\mathbf{x}_t + \mathbf{W}_{ih}\mathbf{h}_{t-1} + \mathbf{b}_i)$

forgotten gate: $\mathbf{f}_t = \sigma(\mathbf{W}_{fx}\mathbf{x}_t + \mathbf{W}_{fh}\mathbf{h}_{t-1} + \mathbf{b}_f)$

new memory: $\mathbf{c}_t = \mathbf{i}_t \circ \tilde{\mathbf{c}}_t + \mathbf{f}_t \circ \mathbf{c}_{t-1}$

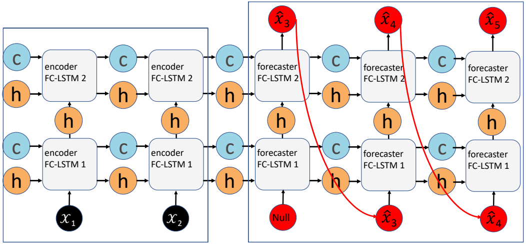
output gate: $\mathbf{o}_t = \sigma(\mathbf{W}_{ox}\mathbf{x}_t + \mathbf{W}_{oh}\mathbf{h}_{t-1} + \mathbf{b}_o)$

new hidden state: $\mathbf{h}_t = \mathbf{o}_t \circ tanh(\mathbf{c}_t)$

- LSTM with multivariate input is also called fully-connected LSTM (FC-LSTM).
- Each input x_t is in vector form.
- Memory cell c_t and hidden state h_t are also vectors of length n_{state}.

How to perform multi-step prediction

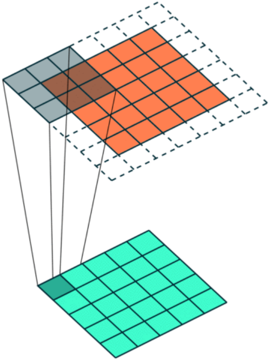
Incorporate an encoder-forecaster structure into FC-LSTM:



$$\begin{aligned}
\tilde{\mathcal{X}}_{t+1}, \ldots, \tilde{\mathcal{X}}_{t+K} &= \underset{\mathcal{X}_{t+1},\ldots,\mathcal{X}_{t+K}}{\operatorname{argmax}} P(\mathcal{X}_{t+1}, \ldots, \mathcal{X}_{t+K} | \mathcal{X}_{t-J+1}, \mathcal{X}_{t-J+2}, \ldots, \mathcal{X}_t) \\
&\approx \underset{\mathcal{X}_{t+1},\ldots,\mathcal{X}_{t+K}}{\operatorname{argmax}} P(\mathcal{X}_{t+1}, \ldots, \mathcal{X}_{t+K} | f_{encoder}(\mathcal{X}_{t-J+1}, \mathcal{X}_{t-J+2}, \ldots, \mathcal{X}_t)) \\
&\approx g_{forecaster}(f_{encoder}(\mathcal{X}_{t-J+1}, \mathcal{X}_{t-J+2}, \ldots, \mathcal{X}_t)).
\end{aligned}$$

How to deal with spatiotemporal data

Convolution for spatial correlations:



Example of sliding a 3×3 filter on a 5×5 image [1].

- The convolution of a patch of image $\mathbf{I}$ and the filter $\mathbf{K}$ is defined as:

$$\mathbf{I} * \mathbf{K} = \sum_{j=1}^{n} \sum_{k=1}^{n} x_{jk} w_{jk}.$$

- Apply zero padding to keep the convolution result as the same dimension of the input image.

[1] [1]. CNTK 103: Part D - Convolutional Neural Network with MNIST. https://cntk.ai/pythondocs/CNTK_103D_MNIST_ConvolutionalNeuralNetwork.html

How to deal with spatiotemporal data

- We lose spatial information since FC-LSTM flattens all inputs into vectors.
- Add convolution to FC-LSTM $\Rightarrow$ Convolutional LSTM.

FC-LSTM

$$
\begin{aligned}
\tilde{\mathbf{c}}_t &= tanh(\mathbf{W}_{cx}\mathbf{x}_t + \mathbf{W}_{ch}\mathbf{h}_{t-1} + \mathbf{b}_c) \\
\mathbf{i}_t &= \sigma(\mathbf{W}_{ix}\mathbf{x}_t + \mathbf{W}_{ih}\mathbf{h}_{t-1} + \mathbf{b}_i) \\
\mathbf{f}_t &= \sigma(\mathbf{W}_{fx}\mathbf{x}_t + \mathbf{W}_{fh}\mathbf{h}_{t-1} + \mathbf{b}_f) \\
\mathbf{c}_t &= \mathbf{i}_t \circ \tilde{\mathbf{c}}_t + \mathbf{f}_t \circ \mathbf{c}_{t-1} \\
\mathbf{o}_t &= \sigma(\mathbf{W}_{ox}\mathbf{x}_t + \mathbf{W}_{oh}\mathbf{h}_{t-1} + \mathbf{b}_o) \\
\mathbf{h}_t &= \mathbf{o}_t \circ tanh(\mathbf{c}_t)
\end{aligned}
$$

Input and state at a time step are all flattened as 1D tensors.

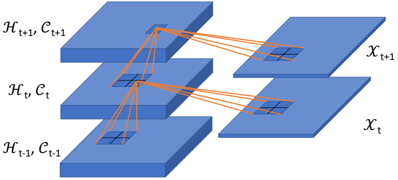
Convolutional LSTM (ConvLSTM)

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

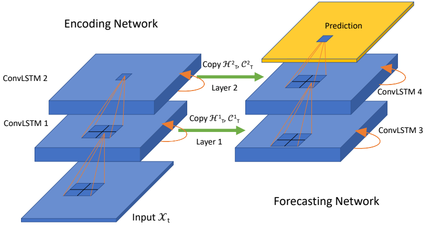
Input and state at a time step are 3D tensors. Convolution is used in both input-to-state and state-to-state transitions.

ConvLSTM Encoder-Forecaster network

Inner structure of ConvLSTM



Final model: stacked ConvLSTM encoder-forecaster network:



Training process:

- Cost function: Pixel-wise MSE.
- Back-Propagation Through Time (BPTT) to derive the derivatives of cost function with respect to each parameter & Adam optimization algorithm to update parameters.
- Apply early-stopping on validation set to avoid overfit on training set.

Overview

Experiments

- Experiment on a synthetic Moving-MNIST data.
  - Gain basic understanding of ConvLSTM.
  - Test the effectiveness of ConvLSTM on synthetic data.
- Experiment on the real-life Shenzhen radar echo data.
  - Test if the ConvLSTM based model is effective for the precipitation nowcasting task.

Moving-MNIST data

- 2 digits bouncing within 5 pixels each time step inside a 64×64 box.
- 8000 training sequences + 2000 validation sequences + 2000 test sequences, first 7 frames for input and last 7 frames to predict.
- Performance of different models:

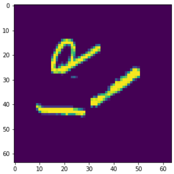
Model	# parameters	MSE_{train}	MAE_{train}	MSE_{val}	MAE_{val}
2-layer FC-LSTM $-2048-2048$	92,295,168	814.40	10.12	838.52	9.92
1-layer ConvLSTM $(5 \times 5) - 5 \times 5 - 128$	3,311,617	694.40	7.82	734.76	8.14
2-layer ConvLSTM $(5 \times 5) - 5 \times 5 - 64$ $-5 \times 5 - 64$	2,475,521	631.53	7.10	659.33	7.32
3-layer ConvLSTM $(5 \times 5) - 5 \times 5 - 64$ $-5 \times 5 - 32 - 5 \times 5 - 32$	1,858,049	582.79	6.62	625.55	6.90

Table: '-5×5' is the filter size for the state-to-state transition. '-128', '-64' and '-32' represent number of filters for the hidden states and memory cells in each layer. '(5×5)' means the filter size for the input-to-state transition.

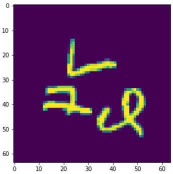
- Shallower ConvLSTM based network with much less parameters outperforms the deeper FC-LSTM based method.
- Deeper ConvLSTM based network provides better forecast.

Out-of-domain case

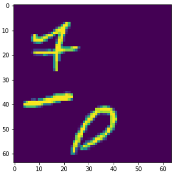
Apply the best trained 3-layer ConvLSTM model on new generated data with 3 moving digits. Use the first 7 frames as input and last 7 frames as ground truth.



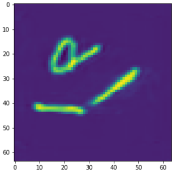
ground truth



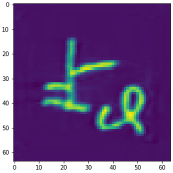
ground truth



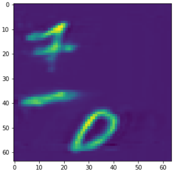
ground truth



prediction



prediction



prediction

Shenzhen radar echo maps data

- 10000 sequences. Each sequence has 15 frames. First 8 frames for input and last 7 frames to predict.
- Each radar echo map is a 101×101 grid. Each pixel value is a transformation of rainfall intensity in an area of 1 km^2.
- Select 5485 most variant sequences with correlation between the first frame P and last frame Q less than 0.75.

$$correlation(P, Q) = \frac{\sum_{i,j} P_{i,j} Q_{i,j}}{\sqrt{(\sum_{i,j} P_{i,j}^2)(\sum_{i,j} Q_{i,j}^2)} + \epsilon}$$

- To have enough data, we rotate the all frames counter-clockwise by 90 degrees and reverse the order of every sequence. We add the new data to have 10970 sequences in total.
- 6582 training sequences + 2194 validation sequences + 2194 test sequences.

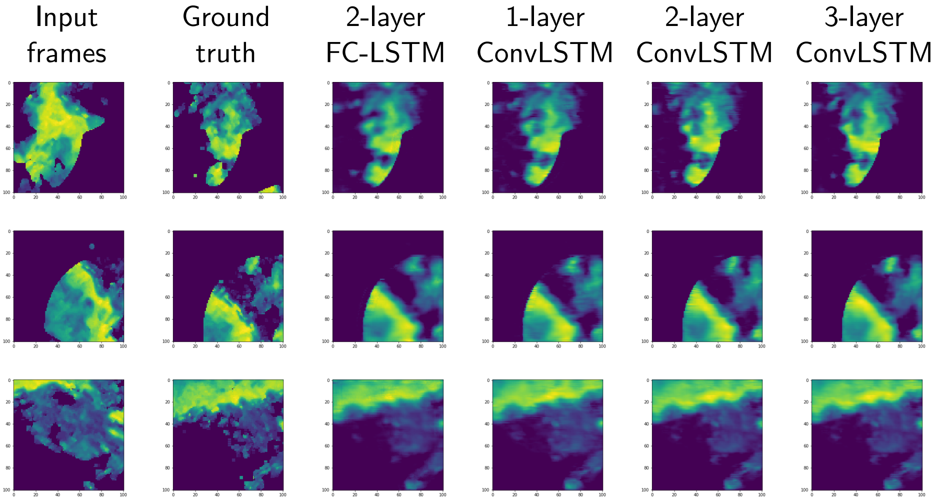
Shenzhen radar echo maps data

Performance of different models:

Model	# parameters	MSE_{train}	MAE_{train}	MSE_{val}	MAE_{val}
2-layer FC-LSTM $-2048-2048$	154,816,473	699.87	13.78	754.69	14.77
1-layer ConvLSTM $(3\times3)-3\times3-128$	1,191,937	656.74	12.39	688.59	13.71
2-layer ConvLSTM $(3\times3)-3\times3-64$ $-3\times3-64$	894,465	625.97	12.19	655.87	13.17
3-layer ConvLSTM $(3\times3)-3\times3-64$ $-3\times3-32-3\times3-32$	670,209	609.33	11.81	638.69	12.83

- The ConvLSTM encoder-forecaster network still outperforms the FC-LSTM encoder-forecaster network for this more complex task.
- Deeper ConvLSTM is more effective in precipitation nowcasting.

Shenzhen radar echo maps data



Overview

Conclusions

- ConvLSTM encoder-forecaster network breaks the limitation of classic radar echo extrapolation based method by predicting the decaying and growing radar echo maps successfully.
- ConvLSTM outperforms FC-LSTM in capturing spatiotemporal correlation by adding convolutional operation in the input-to-state and state-to-state transitions.
- Deeper ConvLSTM encoder-forecaster network provides better forecasts.
- The predicted images become more and more blurred as time step increases.