CS230: Precipitation Nowcasting using Deep Learning Techniques

Anika Jain, Georgios Sarmonikas, Yash Gaur
Video Link: https://youtu.be/G-aENwYe3N8

Abstract

Nowcasting Precipitation is gaining traction in the artificial intelligence community. There are many neural network techniques already tried, therefore this work explores different deep learning architectures by combining CNN and LSTM neural networks, using a radar echo dataset. We have developed three models with different architectures and tested them against the synthetic moving MNIST dataset, prior to validating on precipitation nowcasting and concluded that stacked ConvLSTMs with residual input gives satisfactory results.

Introduction

We aimed to predict the sequence of Radar images, given a sequence of input frames.

- **Input:** A sequence of consecutive 6-minute interval Radar echo frames.
- **Model:** Using Convolutional and LSTM layers to deal with the spatio-temporal structure and encoding-forecasting architecture to make the predictions.
- **Output:** Prediction of 6-10 images which translate to 36 mins- 1 hour in the future.

Experiments on Moving-MNIST dataset

We started to test different deep neural network architectures on Moving-MNIST dataset where we tried to predict one time step ahead, four step ahead, and eight time steps ahead.

Figure 1: One time step prediction

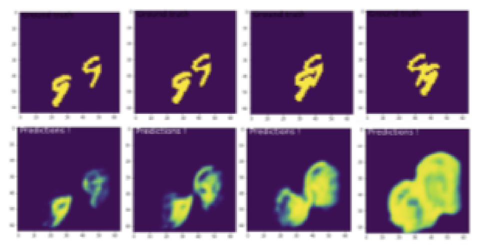

Figure 2: four time step prediction

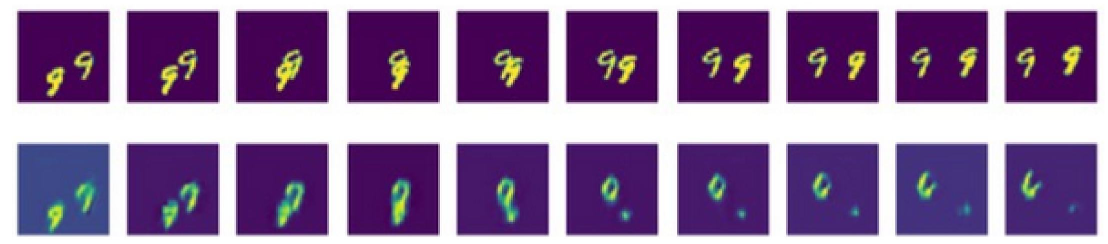

Figure 3: eight time step prediction

Radar Dataset and Features

The dataset we used is publicly available on Harvard Dataverse and is from radar echos captured from the NEXRAD radar located in Pudong,Shanghai.

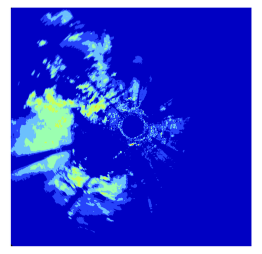

Figure 7: sample radar precipitation echo

The dataset has:

- 170,000 precipitation radar echoes frames from October 2015 to July 2018.
- 6-minutes interval between frames.
- Each frame is a 501 x 501 pixel grid image, covering almost 501 square kilometers. We have re-scaled from 501x501 to 64x64.

We chose 80 sequenced images that have a good representation of precipitation content.

Methods & Algorithm

To tackle the problem of precipitation nowcasting we used Convolutional LSTM architecture. Architecture based on stacked **ConvLSTMs**:

- Convolutions capture the spatial features,
- LSTM capture the temporal dimension

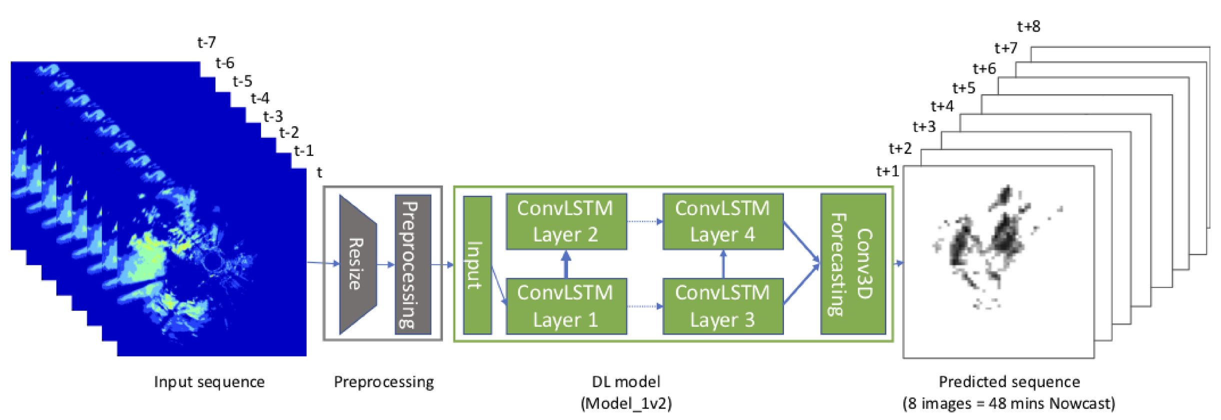

Figure 8: Model & processing pipeline

We used this architecture for both Moving MNIST and Radar datasets. Our model takes as input not just the current input example but also the historical perceived input. The model will preserve information from sequence images that has passed through it using the hidden state.

Results and Evaluation

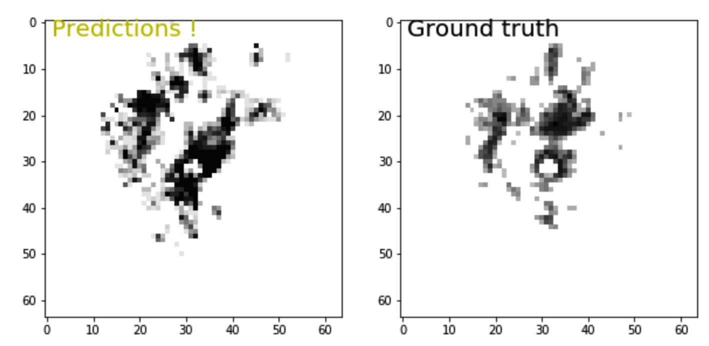

Figure 4: prediction

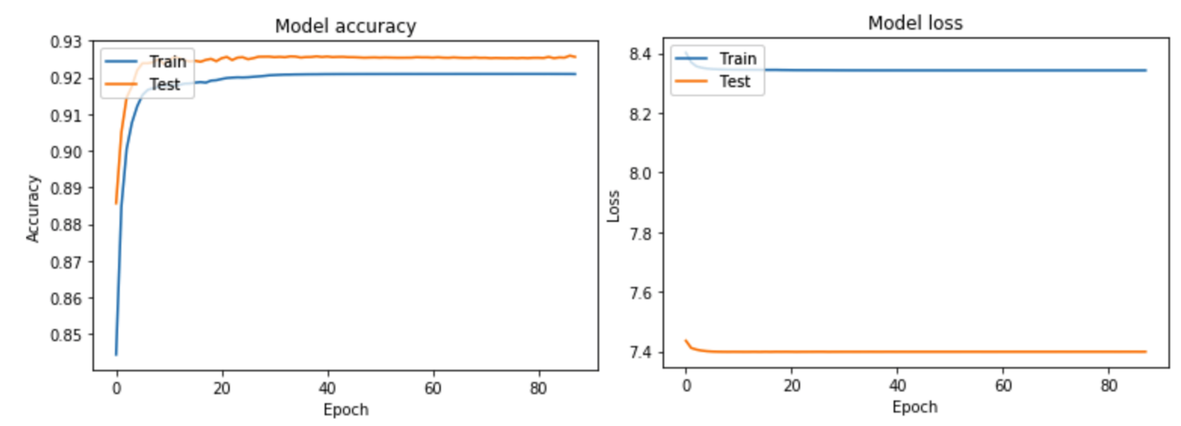

Figure 5: eight time step prediction

SSIM (Structural Similarity Index): Measures perceptual difference of two similar images x and y.
FAR (False Alarm Ratio) = False Positive / (True Positives + False Positives + smooth factor), give a measure of reliability,
POD (Probability of Detection, or Hit Rate) = (True Positives + smooth factor) / (True Positives +False Negatives + smooth factor) give a measure of discrimination.
A smooth factor with value=0.0001 is also used to smooth the results and avoid any potential division by zero. The evaluation of the models against these metrics is shown in the table below

Precipitation Models Evaluation:

		Metrics / Skills										
Model Name	Type	Accuracy	SSIM	POD	FAR	Epochs	Learning Rate	Batch Size	Loss Function	Optimizer	beta1	beta2
Model1_v1	Stacked ConvLSTM	80%	40.94%	0.97	0.22	80	0.001	2	MSE	Adam	0.9	0.999
Model1_v2	Stacked ConvLSTM	92.00%	42.71%	0.9789	0.003	80	0.001	2	Logcosh	Adam	0.9	0.999
Model2_v1	Encoder-Decoder	91.95%	43.62%	0.2	0.822	25	0.1	2	mse	Adam	0.9	0.999

Figure 6: Evaluations

References

[1] SHI Xingjian, 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 *Advances in neural information processing systems*, pages 802–810, 2015.

Future Research

In this project we focused mainly in exploring and developing several models based on ConvLSTM architectures. For future work:

- Using transfer learning and change the last few layers
- Increasing the number of training data and incorporation of additional input dimensions.
- Implemnting U-NET architecture as they can address the blurriness of ConvLSTMs.

Contact Information

Anika: anikaj@stanford.edu
Georgios: gs7776@stanford.edu
Yash: yashgaur@stanford.edu