# Multi-View Spatiotemporal Sequence to Sequence Learning for Precipitation Nowcasting

**Masayuki Hayashi** and **Rafik Hadfi** and **Takayuki Ito**
Nagoya Institute of Technology, Japan
{hayashi.masayuki, rafik}@itolab.nitech.ac.jp, ito.takayuki@nitech.ac.jp

**Ivor Tsang**
University of Technology, Sydney, Australia
Ivor.Tsang@uts.edu.au

## Abstract

Nowcasting the quantity of future precipitation is critical for many applications such as flash-flood warning and navigation safety. A new machine learning approach has been proposed for this problem in the previous work by Shi et al., where they formulated precipitation nowcasting as a spatiotemporal sequence forecasting problem, and used Convolutional LSTM in the sequence-to-sequence framework to solve it. However, while the future rainfall depends on several meteorological factors (e.g., past rainfall, cloud thickness, wind direction, etc.), only one of them is used in the prediction (i.e., past rainfall). Such multiple factors can be introduced in the prediction by using multi-view learning method. In this paper, we examine two settings to multi-view sequence to sequence spatiotemporal learning: (a) simple concatenation of input layer — where multiple modalities are fed to the encoder by just concatenating its input layer, and (b) an application of deep canonical correlation analysis (DCCA) — where each modality is fed to one encoder and DCCA is applied to the obtained representations. Our methods are validated on a radar echo and satellite image dataset, demonstrating better performance compared to the single-view learning method.

## 1 Introduction

Precipitation nowcasting has long been an important yet challenging problem in the field of meteorology. Its goal is to give a precise and timely prediction of future precipitation in a local region over a short period of time (e.g., 5, 10, ..., 60 minutes later). These forecasts are important as they are widely used in a number of hydrometeorological applications such as flash-flood warning, navigation safety, and integration with simulators aiming at urban planning. Yet, since the amount of rainfall depends on multiple meteorological factors (e.g., wind, temperature, and humidity) which can affect the rapid development of clouds, the forecasting model needs to give predictions based on those factors. Hence, the precipitation nowcasting problem is a challenging task.

Existing techniques for this problem can be largely categorized into two groups: Numerical Weather Precipitation (NWP) methods and extrapolation-based techniques. While NWP methods require expensive computations in the simulation of future weather using atmospheric models with physical equations, the prediction is done with much less computational cost in extrapolation-based methods [Kentaro Imajo and Nakano, 2011]. Thus, extrapolation-based techniques are more reasonable for the nowcasting problem [Mass, 2011]. Recently, a novel state-of-the-art precipitation nowcasting model, Convolutional LSTM Sequence to Sequence (ConvLSTM-seq2seq) model, has been proposed as a machine learning approach [Shi *et al.*, 2015]. In ConvLSTM-seq2seq, the precipitation nowcasting problem is regarded as a spatiotemporal sequence forecasting problem with the sequence of past radar maps as input and the sequence of a fixed number of future radar maps as output, and Convolutional LSTM layer, an extension of LSTM layer, is used in the sequence to sequence learning framework [Sutskever *et al.*, 2014]. While its approach is new, the ConvLSTM-seq2seq has outperformed existing operational precipitation nowcasting systems.

Despite the fact that NWP methods consider multiple meteorological factors to predict the future weather [Saltikoff *et al.*, 2015], only one modality (i.e., past rainfall) is used in ConvLSTM-seq2seq. Thus, we can estimate that the prediction could be more accurate if multiple modalities are used in the model. This issue is addressed by applying multi-view learning, where multiple measurement modalities from multiple information sources are used in its training and/or predicting phase to the spatiotemporal sequence to sequence learning model. By applying multi-view learning, we aim at obtaining better representations that give more accurate precipitation forecasts. In this paper, we introduce the multi-view learning into the spatiotemporal sequence-to-sequence model in two different ways: (a) simple concatenation of input layer — where multiple modalities are fed to the encoder by just concatenating its input layer, and (b) an application of deep canonical correlation analysis — where each multiple modality is fed to one encoder and Canonical Correlation Analysis is applied to the obtained representations.

Several approaches have been proposed for the multi-view learning on Deep Neural Networks (DNN). The simplest approach is to concatenate the units in input layer over the multiple modalities and feed them into the network. According to the prior work, such shallow approach often results in obtaining poor representations across modalities [Ngiam *et al.*, 2011]. Another approach is to use Canonical Correlation Analysis (CCA) [Hardoon *et al.*, 2004], which finds linear transformations of two modalities to form a shared representation. Furthermore, a generalized application of CCA onto DNN has been proposed as Deep Canonical Correlation Analysis (DCCA) [Andrew *et al.*, 2013]. Prior works have shown that the integration of CCA into DNN results in better shared representations to gives better performance in several tasks [Mroueh *et al.*, 2015; Yan and Mikolajczyk, 2015].

This paper is structured as follows. In the next section, we provide the building blocks of our model. In section 3, we present different ways to integrate multi-view learning into the sequence-to-sequence model. In section 4, we provide the experimental results. Finally we and conclude and highlight the future work.

# 2 Preliminaries

## 2.1 Spatiotemporal Sequence to Sequence Model

Spatiotemporal sequence to sequence models are DNN models whose input and output are variable-length sequences of spatiotemporal data. In recent study, the fundamental sequence to sequence (seq2seq) learning framework on which these models are built is proposed by [Cho *et al.*, 2014], which later leads to an extension that uses LSTM [Sutskever *et al.*, 2014]. A Seq2seq model is composed of two Recurrent Neural Networks (RNN), one of which is called encoder that encodes the variable-length input sequence to a fixed-length vector representation, and the other is called decoder that decodes the representation into another variable-length sequence to form the output. [Shi *et al.*, 2015] introduces an extension of the seq2seq model for spatiotemporal data, in which Convolutional LSTM (ConvLSTM), an extension of LSTM, is used as its encoder and decoder. We refer to this model as ConvLSTM-seq2seq model (see Fig. 1) . By introducing convolutional operations in LSTM, ConvLSTM eliminates the redundancy for handing spatial data, while keeping its advantages for temporal data, resulting in a suitable model for spatiotemporal data. ConvLSTM-seq2seq model is shown to outperform the one that uses LSTM in the task of precipitation forecasting.

### Convolutional LSTM

The key equations of ConvLSTM are shown in (1), where '$*$' denotes the convolution operator and '$\circ$' denotes the Hadamard product.

$$\begin{aligned}
i^t &= \sigma(W_{xi} * \mathcal{X}^t + W_{hi} * \mathcal{H}^{t-1} + W_{ci} \circ \mathcal{C}^{t-1} + \boldsymbol{b}_i) \\
f^t &= \sigma(W_{xf} * \mathcal{X}^t + W_{hf} * \mathcal{H}^{t-1} + W_{cf} \circ \mathcal{C}^{t-1} + \boldsymbol{b}_f) \\
\mathcal{C}^t &= f^t \circ \mathcal{C}^{t-1} + i_t \circ \tanh(W_{xc} * \mathcal{X}^t + W_{hc} * \mathcal{H}^{t-1} + \boldsymbol{b}_c) \\
o^t &= \sigma(W_{xo} * \mathcal{X}^t + W_{ho} * \mathcal{H}^{t-1} + W_{co} \circ \mathcal{C}^{t-1} + \boldsymbol{b}_o) \\
\mathcal{H}^t &= o^t \circ \tanh(\mathcal{C}^t)
\end{aligned} \tag{1}$$

At time $t$, ConvLSTM receives the input tensor $\mathcal{X}_t$, cell output $\mathcal{C}_{t-1}$, and hidden state $\mathcal{H}_{t-1}$ and produces a new cell state $\mathcal{C}_t$ and hidden state $\mathcal{H}_t$ following the equations in (1). Note all of the input and output variables are 3D tensors so that spatial information is preserved during the process. That is, $\mathcal{X}_t \in \mathbb{R}^{P \times H \times W}$, $\mathcal{C}_t \in \mathbb{R}^{K \times H \times W}$, and $\mathcal{H}_t \in \mathbb{R}^{K \times H \times W}$ for all time steps $t$.

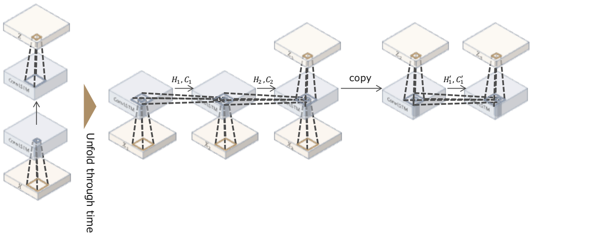

Figure 1: Convolutional LSTM Sequence to Sequence model

## 2.2 Multi-View Learning

In many machine learning applications, there are often situations where modalities or views from multiple information sources are present. These views range from inputs from different sources, such as audio + video [McGurk and MacDonald, 1976] for understanding speech, image + text [Huiskes *et al.*, 2010; Gong *et al.*, 2014] for understanding pictures, to different input modalities from the same source, such as text + text [Vinokourov *et al.*, 2002; Haghighi *et al.*, 2008; Dhillon *et al.*, 2011] for natural language processing, image + image [Zhang *et al.*, 2015] for image classification.

In the setting of multi-view learning, we have access to paired observations from two views, view 1 and view 2, in both training phase and testing phase. Let $(\boldsymbol{x}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_N, \boldsymbol{y}_N)$ denote the observations, where $N$ is the number of observations, and $\boldsymbol{x}_i \in \mathbb{R}^{D_x}$ and $\boldsymbol{y}_i \in \mathbb{R}^{D_y}$ for $i = 1, ..., N$ are the observations from view 1 and view 2 respectively. We also denote $\boldsymbol{X} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_N]$ and $\boldsymbol{Y} = [\boldsymbol{y}_1, ..., \boldsymbol{y}_N]$. The goal of multi-view learning is to extract useful representations across different modalities from the set of observations $\{(\boldsymbol{x}_i, \boldsymbol{y}_i) \mid i = 1, ..., N\}$, which allows the model to give better performances in tasks.

Several multi-view learning approaches for generative deep networks are proposed and examined in [Ngiam *et al.*, 2011]. Although the target differs from that of this paper, namely discriminative deep networks, the fundamental ideas of the ways to introduce multiple modalities can be shared. In particular, it is mentioned that the direct approach of feeding concatenated input of multiple modalities to the network, which they refer as Shalow Biomodal RBM, often results in poor representations where hidden units have strong connections to variables from individual modality but few units have connections across modalities. Among several examined models, the one that uses CCA on the features of multiple modalities is shown to learn good shared representations in the setting where all of the modalities are available in the feature learning phase.

### Deep Canonical Correlation Analysis (DCCA)

[Andrew *et al.*, 2013] propose a DNN extension of CCA named Deep CCA (DCCA; see Fig. 2). In DCCA, instead of directly applying CCA on the observed data $\boldsymbol{x}$ and $\boldsymbol{y}$, CCA is applied on the representations $\boldsymbol{f}(\boldsymbol{x})$ and $\boldsymbol{g}(\boldsymbol{y})$ which are the output of the two DNN encoders $\boldsymbol{f}$ and $\boldsymbol{g}$ using the network

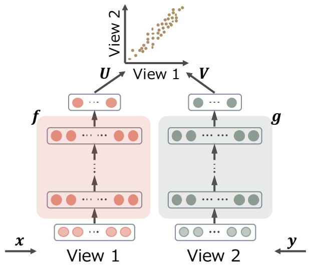

Figure 2: DCCA

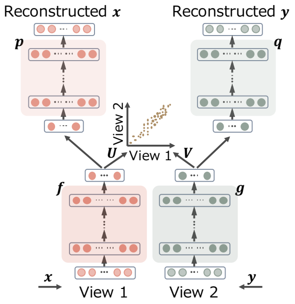

Figure 3: DCCAE

parameters $\boldsymbol{W}_f$ and $\boldsymbol{W}_g$ respectively. Under this setting, the network is trained to find the parameters that maximize the correlation between the extracted features by solving the optimization problem (2).

$$\begin{aligned}
&\max_{\boldsymbol{W_f},\boldsymbol{W_g},\boldsymbol{U},\boldsymbol{V}} && \frac{1}{N}\mathrm{tr}\left(\boldsymbol{U}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{X})\boldsymbol{g}(\boldsymbol{Y})^{\mathrm{T}}\boldsymbol{V}\right)\\
&\text{s.t.} && \boldsymbol{U}^{\mathrm{T}}\left(\frac{1}{N}\boldsymbol{f}(\boldsymbol{X})\boldsymbol{f}(\boldsymbol{X})^{\mathrm{T}}+r_x\boldsymbol{I}\right)\boldsymbol{U}=\boldsymbol{I},\\
& && \boldsymbol{V}^{\mathrm{T}}\left(\frac{1}{N}\boldsymbol{g}(\boldsymbol{Y})\boldsymbol{g}(\boldsymbol{Y})^{\mathrm{T}}+r_y\boldsymbol{I}\right)\boldsymbol{V}=\boldsymbol{I},\\
& && \boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{X})\boldsymbol{g}(\boldsymbol{Y})^{\mathrm{T}}\boldsymbol{v}_j=0,\quad\text{for}\quad i\neq j,
\end{aligned}\tag{2}$$

where $\boldsymbol{U}=[\boldsymbol{u}_1,...,\boldsymbol{u}_k]$ and $\boldsymbol{V}=[\boldsymbol{v}_1,...,\boldsymbol{v}_k]$ are the CCA directions that project the DNN outputs, and $r_x, r_y > 0$ are regularization parameters for estimating the nonsingular covariances. In DCCA, the projection $\boldsymbol{U}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{X})$ and $\boldsymbol{V}^{\mathrm{T}}\boldsymbol{g}(\boldsymbol{Y})$ are the final representations of the input $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively.

The deep network is trained using gradient-based optimization with the gradient formulas given in [Andrew *et al.*, 2013]. While Andrew et al. used full-batch optimization and mini-batch optimizations are mentioned to perform worse than full-batch optimization, it has been shown by [Wang *et al.*, 2015] that the network can still be optimized efficiently using the mini-batch optimization, Stochastic Gradient Descent (SGD), as long as large mini-batch size is used.

**Deep Canonically Correlated Autoencoders (DCCAE)**
Based on DCCA, an extended model, Deep Canonically Correlated Autoencoders (DCCAE; see Fig. 3) is proposed in [Wang *et al.*, ]. DCCAE consists of two autoencoders, each receives input of different views, and optimizes the combination of canonical correlations between the learned bottleneck representations and the reconstruction errors of the autoencoders. The objective of DCCAE is formulated as in (3).

$$\begin{aligned}
&\min_{\boldsymbol{W_f},\boldsymbol{W_g},\boldsymbol{W_p},\boldsymbol{W_q},\boldsymbol{U},\boldsymbol{V}} -\frac{1}{N}\mathrm{tr}\left(\boldsymbol{U}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{X})\boldsymbol{g}(\boldsymbol{Y})^{\mathrm{T}}\boldsymbol{V}\right)\\
&+\frac{\lambda}{N}\sum_{i=1}^{N}(\|\boldsymbol{x_i}-p(\boldsymbol{f}(\boldsymbol{x_i}))\|^2+\|\boldsymbol{y_i}-q(\boldsymbol{g}(\boldsymbol{y_i}))\|^2)\\
&\quad\text{s.t. the same constraints in (2),}
\end{aligned}\tag{3}$$

where $\lambda$ is a trade-off hyperparameter that determines how much weights to put on minimizing the reconstruction error in its training. Similarly to DCCA, stochastic optimization is shown to be applicable to train DCCAE.

# 3 Multi-View Spatiotemporal Sequence to Sequence Learning

We propose two methods to apply multi-view learning for spatiotemporal sequence to sequence learning. Following two sections describes these methods. In the following descriptions, we use the expressions $\mathcal{X}_i^t \in \mathbb{R}^{P_x\times H\times W}$ and $\mathcal{Y}_i^t \in \mathbb{R}^{P_y\times H\times W}$ to denote the observations from two views, view 1 and view 2, respectively. Also, we denote sequences by $\mathcal{X}_i = (\mathcal{X}_i^0, ..., \mathcal{X}_i^T)$ and $\mathcal{Y}_i = (\mathcal{Y}_i^0, ..., \mathcal{Y}_i^T)$, and sets of sequences by $\boldsymbol{\mathcal{X}} = \{\mathcal{X}_0, ..., \mathcal{X}_N\}$ and $\boldsymbol{\mathcal{Y}} = \{\mathcal{Y}_0, ..., \mathcal{Y}_N\}$. The goal is to estimate a $K$-length future observation sequence of one view, $(\mathcal{X}_i^{t+1}, ..., \mathcal{X}_i^{t+K})$, given the $J$-length sequences of past observations of both views, $(\mathcal{X}_i^{t-J+1}, \mathcal{X}_i^{t-J+2}, ..., \mathcal{X}_i^t)$ and $(\mathcal{Y}_i^{t-J+1}, \mathcal{Y}_i^{t-J+2}, ..., \mathcal{Y}_i^t)$.

## 3.1 Simple Concatenation of Input Layer

A straight-forward way to realize multi-view learning in spatiotemporal sequence to sequence learning is to concatenate the input of multiple views and feed it to the ConvLSTM-seq2seq model, which we refer to as the naive multi-view method. In this method, the ConvLSTM-seq2seq model is used with its input a sequence of the concatenated observations from two views, $([\mathcal{X}_i^{t-J+1}, \mathcal{Y}_i^{t-J+1}], ..., [\mathcal{X}_i^t, \mathcal{Y}_i^t])$, and its output a sequence of estimated future observations of one view, $(\hat{\mathcal{X}_i^{t+1}}, ..., \hat{\mathcal{X}_i^{t+K}})$. Note that the concatenation of 3D tensors is done on their first axis, hence, $[\mathcal{X}_i^t, \mathcal{Y}_i^t] \in \mathbb{R}^{(P_x+P_y)\times H\times W}$. Thus, the network is trained using the objective (6).

$$\min_{\boldsymbol{W_f},\boldsymbol{W_h}} Loss\big(\boldsymbol{h}(\boldsymbol{f}([\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}}])), \boldsymbol{\mathcal{Z}}\big), \tag{4}$$

where $Loss(\cdot)$ is a loss function.

As mentioned in section 2.2, the simple concatenation of input layer across modalities often results in poor representations, where limited units in the feature vector have connections across multiple modalities. Thus, this model is also supposed to result in obtaining poor representations. Hence we set this model as the baseline of our work.

## 3.2 DCCA Sequence to Sequence Learning

We introduce DCCA Sequence to Sequence method by applying it to the ConvLSTM-seq2seq model. As described in the section 2.2, DCCA tries to maximize the correlation between representations extracted from two views by encoders. In the ConvLSTM-seq2seq model, we can regard the hidden state at the last time interval in encoders, $\mathcal{H}^t$, as the representation of its input sequence, $f(\mathcal{X}_i)$. Using two ConvLSTM encoders for each view, view1 and view2, multi-view learning is achieved by applying CCA to the representations extracted by those encoders. Here, we propose two different models that integrates ConvLSTM-seq2seq with DCCA, one of which is trained to both maximize the correlation and minimize the prediction error simultaneously, and the other is trained first to find representation that maximize the correlation by pretraining and then to minimize the prediction error by finetuning.

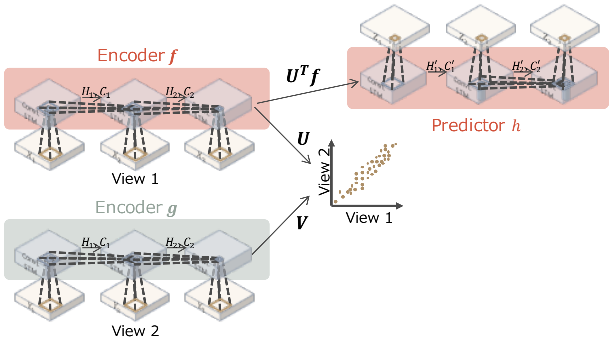

Figure 4: DCCA ConvLSTM-seq2seq model

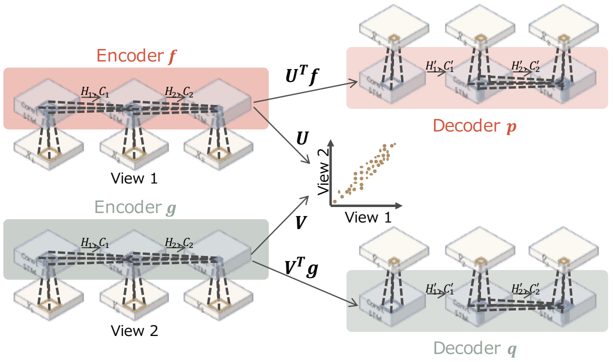

Figure 5: DCCA-Pretrain ConvLSTM-seq2seq model

**Computational optimization** When applying DCCA using the equation (2), we encounter the calculation of the correlation term, $\mathrm{tr}(\boldsymbol{U}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{\mathcal{X}})\boldsymbol{g}(\boldsymbol{\mathcal{Y}})^{\mathrm{T}}\boldsymbol{V})$, where $\boldsymbol{f}(\boldsymbol{\mathcal{X}}) \in \mathbb{R}^{N\times K\times H\times W}$ and $\boldsymbol{g}(\boldsymbol{\mathcal{Y}}) \in \mathbb{R}^{N\times K\times H\times W}$. Since the minibatch representations $\boldsymbol{f}(\boldsymbol{X})$ and $\boldsymbol{g}(\boldsymbol{Y})$ are originally expected to be minibatch of matrices, we can naively flatten the feature maps to vectors to form minibatch representations of matrices (e.g., $\boldsymbol{f}(\boldsymbol{\mathcal{X}}) \in \mathbb{R}^{N\times KHW}$ and $\boldsymbol{g}(\boldsymbol{\mathcal{Y}}) \in \mathbb{R}^{N\times KHW}$). However, this results in the projection matrices $\boldsymbol{U} \in \mathbb{R}^{KHW}$ and $\boldsymbol{V} \in \mathbb{R}^{KHW}$, which in practice does not work because the matrices are too large. Thereby, instead of flattening the feature maps $f(\mathcal{X}) \in \mathbb{R}^{K\times H\times H}$, we view each $k$-th feature map as one representation $f(\mathcal{X})_{(k)} \in \mathbb{R}^{H\times W}$ $for\ k = 1, ..., K$, and approximated as in (5) by flattening it.

$$\mathrm{tr}(\boldsymbol{U}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{\mathcal{X}})\boldsymbol{g}(\boldsymbol{\mathcal{Y}})^{\mathrm{T}}\boldsymbol{V}) \approx \sum_{k} \mathrm{tr}(\boldsymbol{U}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{\mathcal{X}})_{(k)}\boldsymbol{g}(\boldsymbol{\mathcal{Y}})^{\mathrm{T}}_{(k)}\boldsymbol{V}) \tag{5}$$

In other words, we treat each $1 \times W \times H$ sized feature map independently and use the same projection matrices over all the flattened feature maps. This approximation reduces the dimension of projection matrices to $\boldsymbol{U} \in \mathbb{R}^{WH}$ and $\boldsymbol{V} \in \mathbb{R}^{WH}$, resulting in much less expensive computation during the calculation of correlation term and hence make the problem tractable.

### DCCA ConvLSTM-seq2seq model

To accomplish multi-view learning with DCCA, we first propose DCCA ConvLSTM-seq2seq model (see Fig. 4). By adding the correlation term to the objective, DCCA is applied to ConvLSTM-seq2seq model. The network is trained using the objective (6).

$$\begin{aligned}\min_{\boldsymbol{W_f},\boldsymbol{W_g},\boldsymbol{W_h},\boldsymbol{U},\boldsymbol{V}} & -\frac{\lambda}{N}\mathrm{tr}\left(\boldsymbol{U}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{\mathcal{X}})\boldsymbol{g}(\boldsymbol{\mathcal{Y}})^{\mathrm{T}}\boldsymbol{V}\right) \\ & + Loss\big(\boldsymbol{h}(\boldsymbol{f}(\boldsymbol{\mathcal{X}})), \boldsymbol{\mathcal{Z}}\big),\end{aligned} \tag{6}$$

where $Loss\big(\boldsymbol{h}(\boldsymbol{f}(\boldsymbol{\mathcal{X}})), \boldsymbol{\mathcal{Z}}\big)$ denotes the loss between the predicted sequence $\boldsymbol{h}(\boldsymbol{f}(\boldsymbol{\mathcal{X}}))$ and the target sequence $\boldsymbol{\mathcal{Z}}$.

### DCCA-Pretrain ConvLSTM-seq2seq model

As another multi-view learning approach similar to DCCAE, we propose DCCA-Pretrain ConvLSTM-seq2seq model (see Fig. 5), where DCCA is applied in the pretraining phase, and finetuning is used to optimize the model for prediction. In this model, the network is first pretrained to find the representations whose correlations are maximized between the views, using the objective (7).

$$\begin{aligned}\min_{\boldsymbol{W_f},\boldsymbol{W_g},\boldsymbol{W_p},\boldsymbol{W_q},\boldsymbol{U},\boldsymbol{V}} & -\frac{\lambda}{N}\mathrm{tr}\left(\boldsymbol{U}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{\mathcal{X}})\boldsymbol{g}(\boldsymbol{\mathcal{Y}})^{\mathrm{T}}\boldsymbol{V}\right) \\ & + Loss\big(\boldsymbol{p}(\boldsymbol{f}(\boldsymbol{\mathcal{X}})), \boldsymbol{\mathcal{X}}\big) + Loss\big(\boldsymbol{q}(\boldsymbol{g}(\boldsymbol{\mathcal{Y}})), \boldsymbol{\mathcal{Y}}\big),\end{aligned} \tag{7}$$

where $Loss\big(\boldsymbol{p}(\boldsymbol{f}(\boldsymbol{\mathcal{X}})), \boldsymbol{\mathcal{X}}\big)$ and $Loss\big(\boldsymbol{q}(\boldsymbol{g}(\boldsymbol{\mathcal{Y}})), \boldsymbol{\mathcal{Y}}\big)$ denote the loss between the reconstructed sequence $\boldsymbol{p}(\boldsymbol{f}(\boldsymbol{\mathcal{X}}))$ and $\boldsymbol{q}(\boldsymbol{g}(\boldsymbol{\mathcal{Y}}))$, and the input sequence $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$ respectively. After the pretraining, finetuning is done using the objective (8).

$$\min_{\boldsymbol{W_f},\boldsymbol{W_g},\boldsymbol{W_h}} Loss\big(\boldsymbol{h}(\boldsymbol{f}(\boldsymbol{\mathcal{X}})), \boldsymbol{\mathcal{Z}}\big) \tag{8}$$

## 4 Experiments

We perform experiments on two meteorological datasets to compare the following two methods by precipitation forecasting task, focusing on verifying the advantage of multi-view learning compared to single-view learning.

**ConvLSTM-seq2seq**, ConvLSTM-seq2seq is used to predict one modality from one modality.

**Multi-view ConvLSTM-seq2seq**, ConvLSTM-seq2seq is used to predict one modality from two modalities. We examine three models that introduces mutli-view learning in the seq2seq model, namely, the naive multi-view ConvLSTM-seq2seq model described in section 3.1, and DCCA ConvLSTM-seq2seq model and DCCA-pretrain ConvLSTM-seq2seq model described in section 3.2.

### 4.1 Evalution Indexes

In our experiments, we use two quantitative indices for evaluations: cross-entropy loss and meteorological skill scores.

The cross-entropy loss of the predicted frame $P$ and the ground-truth frame $T$ is defined as in (9) [Shi *et al.*, 2015].

$$-\sum_{i,j,k} T_{(i,j,k)} \log P_{(i,j,k)} + (1 - T_{(i,j,k)}) \log(1 - P_{(i,j,k)}), \tag{9}$$

where $T_{(i,j,k)}$ and $P_{(i,j,k)}$ denote the element at the location $(i, j)$ in the $k$-th feature maps in $T$ and $P$ respectively.

The skill scores we use are Critical Success Index (CSI), False Alarm Rate (FAR), Probability of Detection (POD), and correlation, which are commonly used in the study of meteorology. These three skill scores are calculated as follows. We first convert the prediction and ground truth into 0/1 (sunny/rainy) matrices using the threshold of 1.0mm/h rainfall rate, and then count the number of $hits$ (prediction=1, truth=1), $misses$ (prediction = 0, truth=1), and $falsealarms$ (prediction=1, truth=0) to get the three scores as CSI $= \frac{hits}{hits+misses+falsealarms}$, FAR $= \frac{falsealarms}{hits+falsealarms}$, and POD $= \frac{hits}{hits+misses}$.

## 4.2 The Dataset

For evaluation, we used a meteorological dataset that consists of two observation modalities, radar echo (rainfall rate) and satellite image, collected in a local region in Japan. We used the observations collected in 31 rainy and sunny days in the period 01/10/2015 - 31/10/2015.

For the radar echo observation, precipitation intensities are obtained within a fixed-size local observation region, and then the actual rainfall rates (mm/h) are calculated using the Z-R relationship [Doviak and Zrnic, 2014]. Since our prediction target is the actual rainfall rate, we use the converted rainfall rates as input and output of the models. The observation region is fixed to the rectangle E$136.5°$-E$138°$ and N$34.5°$-N$35.5°$, and its spatial resolution is $45$ arcsec in width and $30$ arcsec in height, resulting in a $120 \times 120$ sized radar echo map. We regard this map as a $120 \times 120$ sized grayscaled image with $1$ channel.

For the satellite image, we use the one observed by Himawari8, a meteorological satellite in Japan. Several different types of satellite images in different regions are captured and transmitted in a data format called Himawari Standard Format (HSF). In particular, we use the infra-red images of the observation region "East Asia", in which the area observed in the radar echo data is covered. By converting count values stored in HSF to pixel brightnesses, we get a grayscale satellite image of 1 channel. In order to use the observation of exactly the same region as radar echo data, the satellite images are cropped in the way where pixels in the satellite image corresponding to the observation region of radar echo data are extracted to compose the cropped images. The spatial resolution on satellite image captured by Himawari8 is 2km per a pixel, whereas that of the radar echo data is 1km per a pixel. Thus, the resolution of satellite image is not as high as that of radar echo data, resulting in obtaining lower-resolution images compared to radar echo observations by cropping the satellite images in this way. This process results in $120 \times 120$ sized grayscale images with 1 channel.

As pre-processing, the observation value $Z$ in both radar echo map and cropped satellite image is normalized to the pixel values $P$ by setting $P = \frac{Z - \min\{Z\}}{\max\{Z\} - \min\{Z\}}$.

The radar echo data and satellite image are recorded every 5 minutes and 2.5 minutes respectively. In the multi-view learning, we expect that both view 1 (radar echo) and view 2 (satellite image) are captured at the same time, that is, we expect to use temporally synchronized data. Thus, we eliminate the satellite images that are not captured at the same time as radar echo data, and use the data recorded every 5 minutes. Ideally, we obtain 8928 observations in the 31 days, and then make sequences by sliding a 20-frame-wide window over the sequential observations of 5-minute intervals. However, due to the observation schedule of satellite image[1], some data are not captured, resulting in obtaining 7105 sequences. We then split the sequences into 3 blocks, 5684 sequences for training block, 710 sequences for validation block, and 711 sequences for testing block, all of which are 20 frames long (10 frames for the input and 10 frames for the prediction).

## 4.3 Training Details

Referring to the work by [Shi *et al.*, 2015], we set the model parameters as follows: all models use 2-layer ConvLSTM networks with each layer consisting of 8 hidden states and $3 \times 3$ kernels as their encoder and decoder. For the DCCA models, we use two 2-layer ConvLSTM networks as encoder and one 2-layer ConvLSTM as decoder, with the same number of hidden states and kernel sizes. Also, we set the hyperparameter $\lambda = 1.0$ for the DCCA models so that correlation and loss are equally optimized. Both models use the patch size of 4, so $1 \times 120 \times 120$ sized images are divided into $16 \times 30 \times 30$ sized patches and fed to the networks. We trained all the models by minimizing the cross-entropy loss (9) using back-propagation through time (BPTT) [Werbos, 1990] and RMSProp [Tieleman and Hinton, 2012] with the learning rate of $10^{-3}$, decay rate of $0.9$, and minibatch size of 16.

## 4.4 Results

The results are shown in Table 1 and Fig. 6. In addition, we show an example sequence of the prediction in Fig 7. For Table. 1, since our testing set mostly contains observations on sunny days, we get high CSI and POD, and low FAR for the skill scores.

We can find the POD scores on DCCA-Pretrain ConvLSTM outperforms ConvLSTM in the first half of the entire predicted sequence. Interestingly, though, the performance for the rest of the predicted sequence becomes worse than the other models. On the other hand, the correlation between the predicted sequence and the ground-truth are relatively higher on DCCA-Pretrain ConvLSTM compared to ConvLSTM throughout the sequence. Similar tendency is observed in the result of our preliminary experiment, where we used the same radar maps with spatially lower-resolution satellite images. From these observations, we suspect that the DCCA enables the model to capture the correlation between different modalities, but its contribution present only in the very short time period (i.e., 1-5 predicted frames). This feature can also be observed visually from the output sequences of the models (Fig. 7). Both ConvLSTM and DCCA-Pretrain ConvLSTM output relatively blurry images compared to the ground-truth, and ConvLSTM tend to output more blurry images as the forecasting time increases. To the contrary, DCCA-Pretrain ConvLSTM outputs images that have comparably clearer light and shade throughout the sequence. We can see the first half output sequence of DCCA-Pretrain ConvLSTM, compared to that of ConvLSTM, is closer to the corresponding part of the ground-truth sequence. Specifically,

[1] The observation schedule of satellite images by Himawari is published by Japan Meteorological Agency at http://www.data.jma.go.jp/mscweb/ja/operation8/status/satellite.html

Table 1: **Comparison of the scores of different models on radar echo and satellite image dataset.** All scores are the average score over the five/ten predicted frames. Cross-entropy is the sum of cross-entropy losses over all the instances in the dataset.

| Model | Cross-entropy | Validation |  |  |  |  |  | Testing |  |  |
|---|---|---|---|---|---|---|---|---|---|---|
|  |  | First 5 frames |  |  | All 10 frames |  |  | All 10 frames |  |  |
|  |  | CSI | FAR | POD | CSI | FAR | POD | CSI | FAR | POD |
| **DCCA-Pretrain ConvLSTM** | **187.64** | **0.555** | **0.077** | **0.591** | **0.539** | **0.080** | **0.575** | **0.825** | **0.000** | **0.826** |
| Naive Multi-view ConvLSTM | 185.02 | 0.554 | 0.069 | 0.571 | 0.540 | 0.066 | 0.558 | 0.826 | 0.000 | 0.826 |
| DCCA ConvLSTM | 311.38 | 0.475 | 0.131 | 0.546 | 0.483 | 0.112 | 0.542 | 0.761 | 0.072 | 0.827 |
| ConvLSTM | 187.91 | 0.557 | 0.100 | 0.589 | 0.538 | 0.099 | 0.577 | 0.826 | 0.000 | 0.826 |

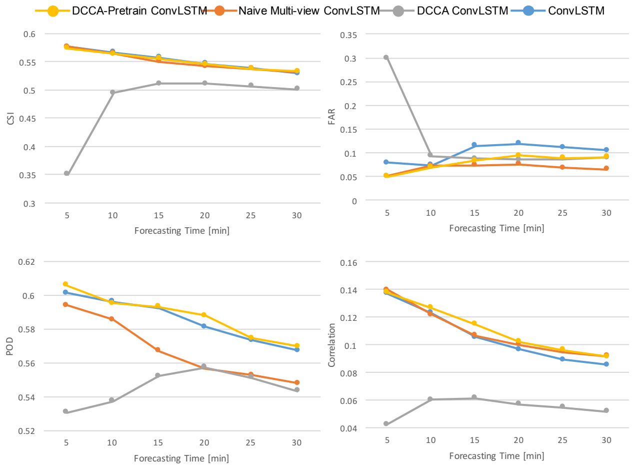

Figure 6: Comparison of the first 6 output frames of different models on four precipitation nowcasting metrics over time, evaluated by the validation set of Radar Echo and Satellite Image dataset

the difference in the transformation of the rain area over time causes this higher similarities in the output of DCCA-Pretrain ConvLSTM. We can suppose that this difference comes from the integration of features extracted from the input sequence of satellite images.

Another interesting point is that Naive Multi-view ConvLSTM constantly gives lower FAR, resulting in high CSI with relatively low POD. This is because Naive Multi-view ConvLSTM always gives smaller rain area and smaller amount of precipitation. Low FAR gives the merit of avoiding unnecessary alarm on rain, but at the same time, this model is insensitive as the POD is constantly lower than the other models. We suspect that the naive approach of inputs' concatenation results in poor shared representations which lead to this low.

Obviously, DCCA ConvLSTM performs, by far, worse than the other models in all the evaluation metrics. This clearly shows that maximizing the correlation between modalities while minimizing the prediction error does not lead to the convergence of the model parameters that gives accurate prediction. Observing Fig. 7, we can see this model cannot correctly capture the spatial correlation of the sequence over time, as the first frame of the DCCA ConvLSTM output sequence does not retain most of the intensities that exist in the frames of the input sequence. For this approach, further more investigation is needed.

From these results, we conclude both Naive Multi-view ConvLSTM and DCCA-Pretrain ConvLSTM captures the correlation between multiple modalities, and DCCA-Pretrain ConvLSTM is considered to have an advantage of obtaining better shared representation that gives better performance compared to Single-view ConvLSTM, in the precipitation nowcasting task for a short time period. In real life, sudden rainfalls such as a torrential rain is caused by a rapid growth of clouds within just a few minutes. Hence the advantage of DCCA-Pretrain ConvLSTM is significant for the prediction of such rainfalls.

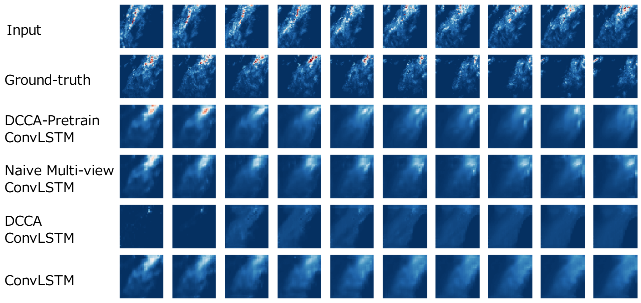

Figure 7: **An example sequence of prediction on Radar Echo and High-resolution Satellite Image dataset.** From left to right, all of the input, ground-truth, and prediction frames are sampled with their time interval of 5 minutes.

## 5 Conclusion

Multiple measurement modalities should be integrated to solve the precipitation nowcasting task by machine learning approaches. In this paper, we have applied multi-view learning to ConvLSTM sequence to sequence model in two different methods, namely, the naive concatenation of input and the application of DCCA. The experiment of the precipitation nowcasting on a radar echo and satellite image dataset shows the multi-view learning methods can capture the correlation between radar maps and satellite images. In particular, the proposed DCCA-Pretrain ConvLSTM is shown to have an advantage of obtaining better performance in precipitation nowcasting of a short time period. For future work, we investigate furthermore revision of integrating DCCA in the sequence-to-sequence model.

## Acknowledgments

We thank Weather Service Co. Ltd. for providing the radar echo and satellite image dataset. We also thank Xingjian Shi and Weiran Wang for the insightful discussions and advices.

## References

[Andrew *et al.*, 2013] Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *Proceedings of the 30th International Conference on Machine Learning*, pages 1247–1255, 2013.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[Dhillon *et al.*, 2011] Paramveer Dhillon, Dean P Foster, and Lyle H Ungar. Multi-view learning of word embeddings via cca. In *Advances in Neural Information Processing Systems*, pages 199–207, 2011.

[Doviak and Zrnic, 2014] Richard J Doviak and Dusan S Zrnic. *Doppler Radar & Weather Observations*. Academic press, 2014.

[Gong *et al.*, 2014] Yunchao Gong, Liwei Wang, Micah Hodosh, Julia Hockenmaier, and Svetlana Lazebnik. Improving image-sentence embeddings using large weakly annotated photo collections. In *Computer Vision–ECCV 2014*, pages 529–545. Springer, 2014.

[Haghighi *et al.*, 2008] Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. Learning bilingual lexicons from monolingual corpora. In *ACL*, volume 2008, pages 771–779, 2008.

[Hardoon *et al.*, 2004] David R Hardoon, Sandor Szedmak, and John Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural computation*, 16(12):2639–2664, 2004.

[Huiskes *et al.*, 2010] Mark J Huiskes, Bart Thomee, and Michael S Lew. New trends and ideas in visual concept detection: the mir flickr retrieval evaluation initiative. In *Proceedings of the international conference on Multimedia information retrieval*, pages 527–536. ACM, 2010.

[Kentaro Imajo and Nakano, 2011] Yoshiaki Taniguchi Kentaro Imajo, Go Hasegawa and Hirotaka Nakano. Short-term precipitation forecasting using an optical flow algorithm. *IEICE technical report. Technical Committee on Commuication Quality*, 110(455):93–98, feb 2011.

[Mass, 2011] Clifford Mass. Nowcasting: The next revolution in weather prediction. *Bulletin of the American Meteorological Society*, 2011.

[McGurk and MacDonald, 1976] Harry McGurk and John MacDonald. Hearing lips and seeing voices. 1976.

[Mroueh *et al.*, 2015] Youssef Mroueh, Etienne Marcheret, and Vaibhava Goel. Deep multimodal learning for audio-visual speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 2130–2134. IEEE, 2015.

[Ngiam *et al.*, 2011] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 689–696, 2011.

[Saltikoff *et al.*, 2015] Elena Saltikoff, Philippe Lopez, Antti Taskinen, and Seppo Pulkkinen. Comparison of quantitative snowfall estimates from weather radar, rain gauges and a numerical weather prediction model. *Boreal Environment Research*, 20:667–678, 2015.

[Shi *et al.*, 2015] 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. *arXiv preprint arXiv:1506.04214*, 2015.

[Sutskever *et al.*, 2014] Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[Tieleman and Hinton, 2012] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4, 2012.

[Vinokourov *et al.*, 2002] Alexei Vinokourov, Nello Cristianini, and John S Shawe-Taylor. Inferring a semantic representation of text via cross-language correlation analysis. In *Advances in neural information processing systems*, pages 1473–1480, 2002.

[Wang *et al.*, ] Weiran Wang, Raman Arora, Karen Livescu, and Jeff Bilmes. On deep multi-view representation learning.

[Wang *et al.*, 2015] Weiran Wang, Raman Arora, Karen Livescu, and Jeff A Bilmes. Unsupervised learning of acoustic features via deep canonical correlation analysis. In *Proceedings of ICASSP*, 2015.

[Werbos, 1990] Paul J Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

[Yan and Mikolajczyk, 2015] Fei Yan and Krystian Mikolajczyk. Deep correlation for matching images and text. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3441–3450, 2015.

[Zhang *et al.*, 2015] Wenlu Zhang, Rongjian Li, Houtao Deng, Li Wang, Weili Lin, Shuiwang Ji, and Dinggang Shen. Deep convolutional neural networks for multi-modality isointense infant brain image segmentation. *NeuroImage*, 108:214–224, 2015.