KIAPS
KOREA INSTITUTE OF ATMOSPHERIC PREDICTION SYSTEMS

European Meteorological Society

SCREEN CAPTURE WELCOME

Conditioned Forecasting Model based on Self-Supervised Learning

Sojung An (sojungan@kiaps.org), Tae-Jin Oh, Inchae Na, Jiyeon Jang, Wooyeon Park, Sang-wook Kim, and Ilseok Noh
Korea Institute of Atmospheric Prediction systems (KIAPS), Operational System Group, Republic of Korea

1 Background

- Temporal variation of the precipitation and topographic features are important factors to short-term precipitation forecast.
- Previous studies applied Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) to predict quantitative precipitation prediction (QPF) to capture spatio-temporal characteristics of precipitation.
- State-of-the-art (SOTA) approaches focused on learning a single latent representation of QPF. To describe every echo with only one latent variable might be an overly restrictive assumption, impeding effective learning of the precipitation features.

2 Methods

- We propose a conditioned forecasting model based on self-supervised learning (SSL), that generalizes diverse precipitation types which would enable various latent representations with two main components.

A generator for precipitation forecast

- Generator is a high resolution network (HRNet) that is to train on the task of learning information across multi-resolution representations. Note that our generator architecture utilizes a spatio-temporal self-attention mechanism, which can highlight precipitation patterns. It starts from a latent vector clustered as y, and gradually add low-to-high resolution streams.
- The training process consists of **(i)** clustering the condition features of radar observations using the sub-network with convolution layers in the generator, **(ii)** generalizing diverse precipitation types according to the clustered labels, and **(iii)** fusing the four resolution representations in ConvGRU cells, enabling various latent representations.
  - $\mathcal{L}_{\mathbf{G}}(\boldsymbol{\theta}) = \mathbb{E}_{X_{1:i}\sim P_y}\left[\mathbb{E}_{\mathrm{Z}_y}\left[\mathcal{S}(G_\theta(\mathrm{Z}_y; X_{1:i})\right] + \mathcal{T}\left(X_{1:i};\, G_\theta(\mathrm{Z}_y; X_{1:i})\right)\right] - \alpha \cdot \mathcal{L}_{\mathbf{R}}(\boldsymbol{\theta}) + \beta \cdot \mathcal{L}_{IoU}(\boldsymbol{\theta})$.
  - $\mathcal{L}_{\boldsymbol{R}}(\boldsymbol{\theta}) = \frac{1}{hwt}\left\|(\mathbb{E}_{\mathrm{Z}_y}[\,G_\theta(\mathrm{Z}_y; X_{1:i})] - X_{1+i:i+\sigma}) \odot w(X_{1+i:i+\sigma})\right\|$, where $w(\varphi) = \max(\varphi, 24)$.
  - $\mathcal{L}_{\boldsymbol{IoU}}(\boldsymbol{\theta}) = \sum_{r\in\mathbf{R}}(1 - \frac{|G_\theta(\mathrm{Z}_y;X_{1:i})\cap X_{1+i:i+\sigma}|}{|G_\theta(\mathrm{Z}_y;X_{1:i})\cup X_{1+i:i+\sigma}|})$, where $\mathbf{R}$ is a threshold set of rainfall.
  - $\mathcal{L}_{\mathbf{y}}(\{\boldsymbol{\pi_n}\}_{y=1}^{k}) = \mathbb{E}_{X_{1:i}\sim P_\pi}\left[\mathbb{E}_{X\sim\pi_y}\left[\sum_{y=1}^{k}\sum_{S\subset C_y}\left\|\boldsymbol{C_f}(\boldsymbol{\mu}) - \epsilon_y\right\|_2^2\right]\right]$, where $\boldsymbol{C_f}(\boldsymbol{\phi}) = {f_\phi}^{d}(\sum_{\ell=1}^{d-1}\mathbf{RELU}({f_\phi}^{\ell}\mathrm{H}_1^{\ell}))$.

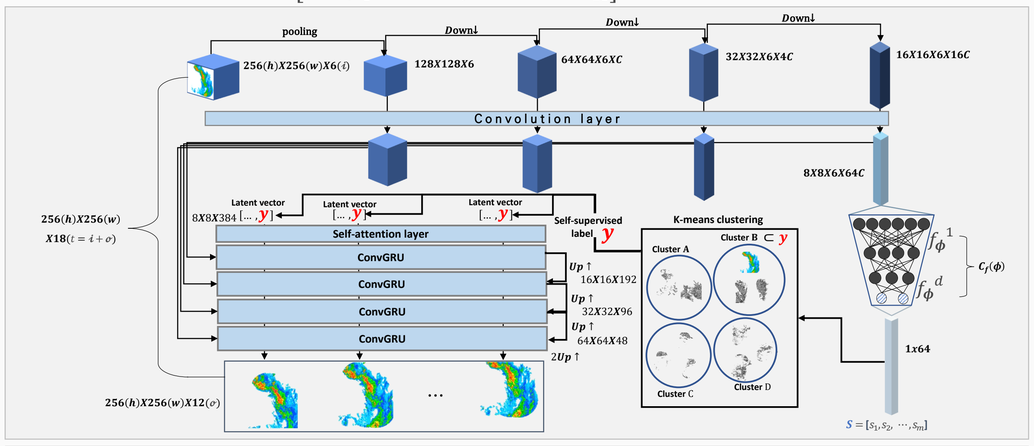

Fig 1. Illustrating how the generator predict radar sequences using self-supervised labels.

Adversarial training

- Discriminators consist of two modules to capture spatial and temporal features. The module learns to distinguish real samples from the samples generated with the automatically discovered cluster label y.
- The structure of the discriminator is based on the DGMR discriminator (Ravuri et al. 2021). One major difference is the updating ratio of the generator to the discriminators, which depends on the number of clusters.
- We address these problems with a novel **clustered Z** using hinge loss as follows:
  - $\mathcal{L}_{\mathcal{S}}(\boldsymbol{\mu}) = \mathbb{E}_{X_{1:t},\mathrm{z}_y}\left[\mathrm{RELU}(1 - \mathcal{S}_{\boldsymbol{\mu}}(X_{1+i:i+\sigma})) + \mathrm{RELU}(1 + \mathcal{S}_{\boldsymbol{\mu}}(\mathrm{G}(\mathrm{Z}_y; X_{1:i})))\right]$.
  - $\mathcal{L}_{\mathcal{T}}(\boldsymbol{\psi}) = \mathbb{E}_{X_{1:t},\mathrm{z}_y}\left[\mathrm{RELU}(1 - \mathcal{T}_{\boldsymbol{\psi}}(X_{1+i:i+\sigma})) + \mathrm{RELU}(1 + \mathcal{T}_{\boldsymbol{\psi}}(X_{1:i};\, \mathrm{G}(\mathrm{Z}_y; X_{1:i})))\right]$.

3 Data

- Experimental dataset is Korea summer (JJA) precipitation with 4km resolution for 2012 to 2021, converted from radar reflectivity of Constant Altitude Plan Position Indicator (CAPPI) to rainfall intensity.
- The images consist of 256x256 pixels from scaling down to a resolution of 4km, and extracted at 10-minute intervals. We select sequences with more than a spatial distribution of 3% of precipitation during 3 hours, which use as input for the first one hour and the output for the other.
- The number of examples in the dataset is roughly 10,000, and the training dataset consists of 7,000 sequences.

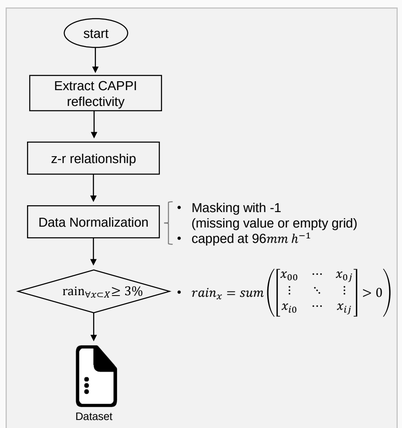

Fig 2. Flowchart for data processing

4 Results

- All the experiments were implemented using Python 3.8 on a single NVIDIA A100 (80G) GPU. The PyTorch 1.11.0, and the NVIDIA™ CUDA Toolkit 11.3 were used for all experiments.
- Parameters setting for training

Parameter	Value	Parameter	Value
Optimization	Adam	α	20
Learning rate	5e-5 (G_θ) / 2e-4 ($\mathcal{S}_\mu$, $\mathcal{T}_\psi$)	β	10
Batch size	16	k	32
Latent channel	384	**Cluster interval**	5,000 iterations

Table 1. Summary of the training parameters

- We used the following comparison baselines in our work: Optical flow (Ayzel et al. 2020), ConvLSTM (Shi et al. 2015), HRNet (An et al. 2021), TrajGRU (Shi et al. 2017), DGMR (Ravuri et al. 2021).
  - Our method was the best scores after 50-minute based on brier scores and peak signal-to-noise ratio (PSNR).
  - TrajGRU outperformed based on critical success index (CSI), followed by HRNet.

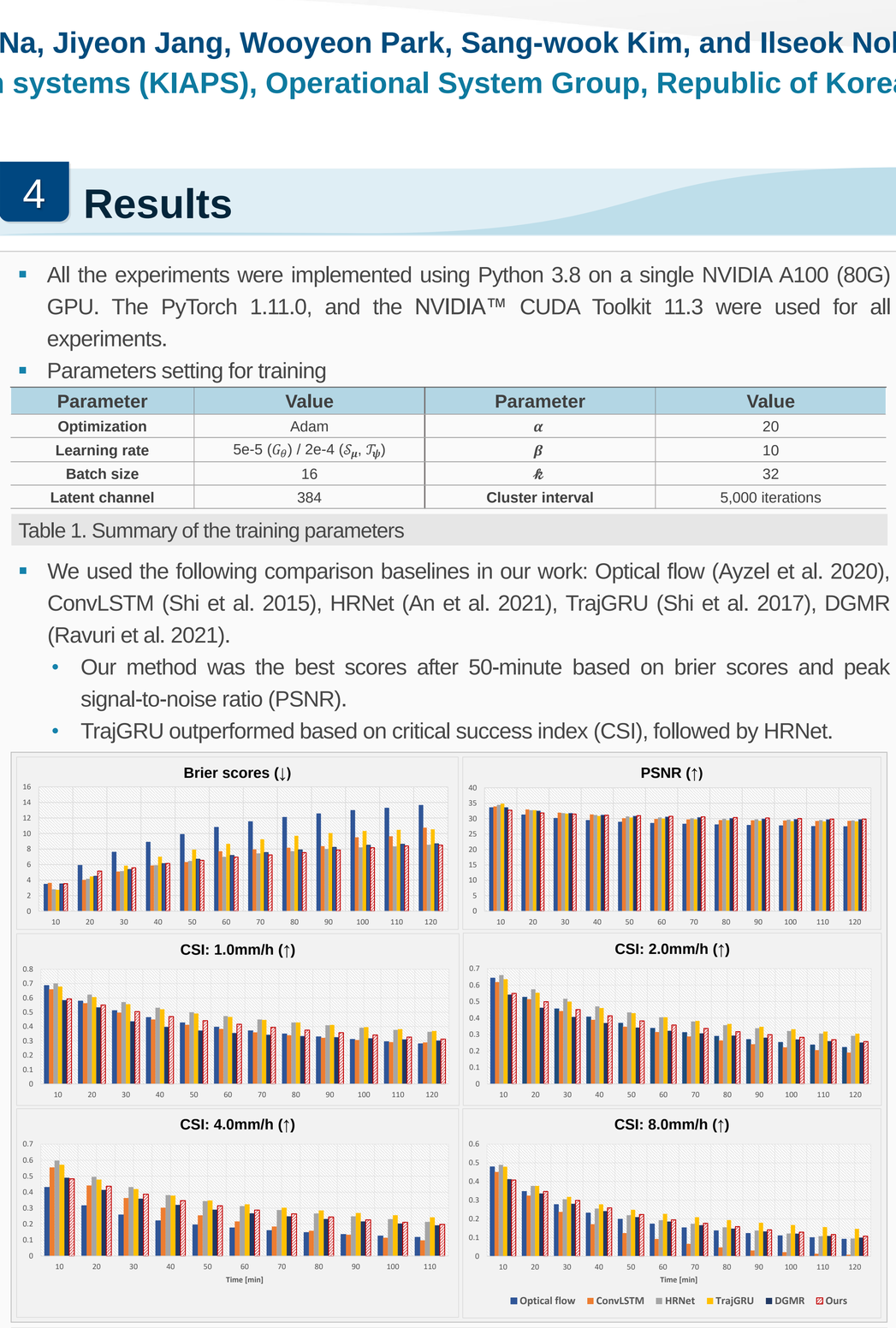

Fig 3. Comparison scores with SOTA models on CAPPI.

- Our method simulated the most plausible images from light rain to heavy rain.
- According to **Cluster A**, unlike most models that are smoothed over time, GAN-based DGMR and our model predicted precipitation in high-resolution textures.
- In **Cluster B**, the proposed model showed the best performance simulating an elongated narrow precipitation and precipitation near the southern shoreline.

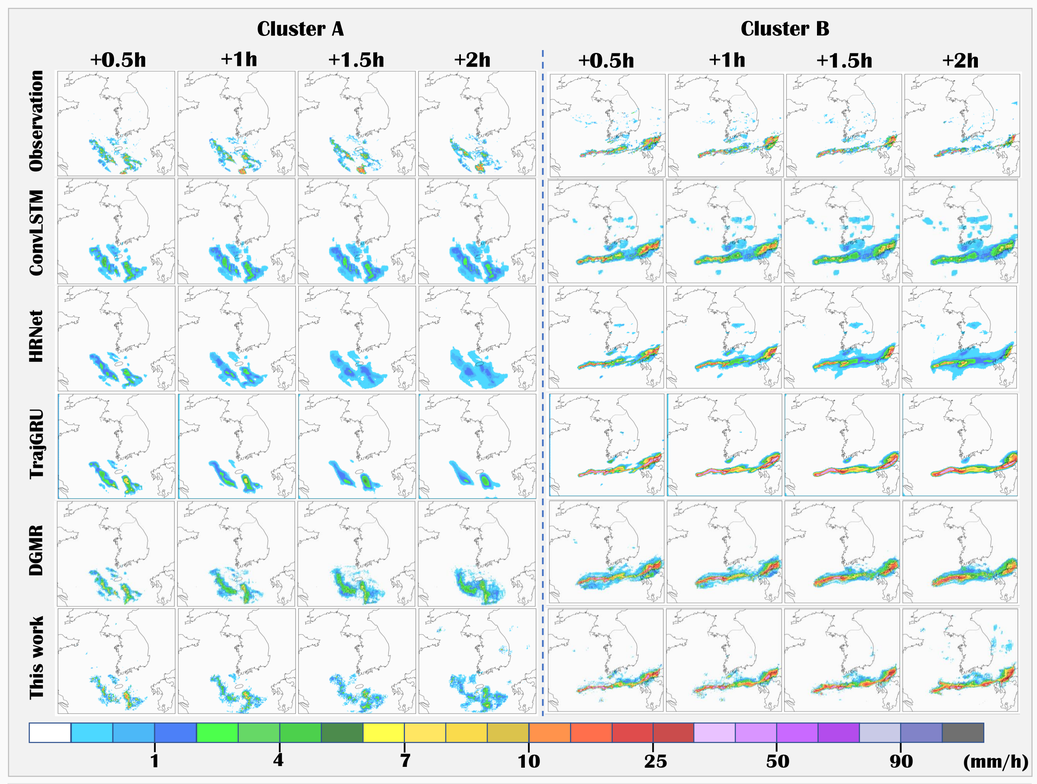

Fig 4. Qualitative comparisons between architectures. Up to 2-hour prediction results of different methods with random sampling from the test set of clusters.

5 Conclusions

- We present a self-conditioned forecasting model based on SSL to approximate variety intensity of rainfall.
- Through evaluation and comparison to SOTA approaches, we found that training GANs by clustering more than sixteen condition types is effective on preventing mode collapse where many GAN models suffer from.
- Our framework can simulate radar echo derived precipitation with high resolution texture and is robust to on various precipitation intensity.