Reliable precipitation nowcasting using probabilistic diffusion model

congyi nai[1], Baoxiang Pan[2], Jiarui Hai[3], Xi Chen[4], Qiuhong Tang[1], Guangheng Ni[3], Qingyun Duan[5], Bo Lu[6], Ziniu Xiao[4], and Xingcai Liu[1]

[1]Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences
[2]Chinese Academy of Sciences
[3]Tsinghua University
[4]Institute of Atmospheric Physics, Chinese Academy of Sciences
[5]Hohai University
[6]National Climate Center, China Meterological Administration

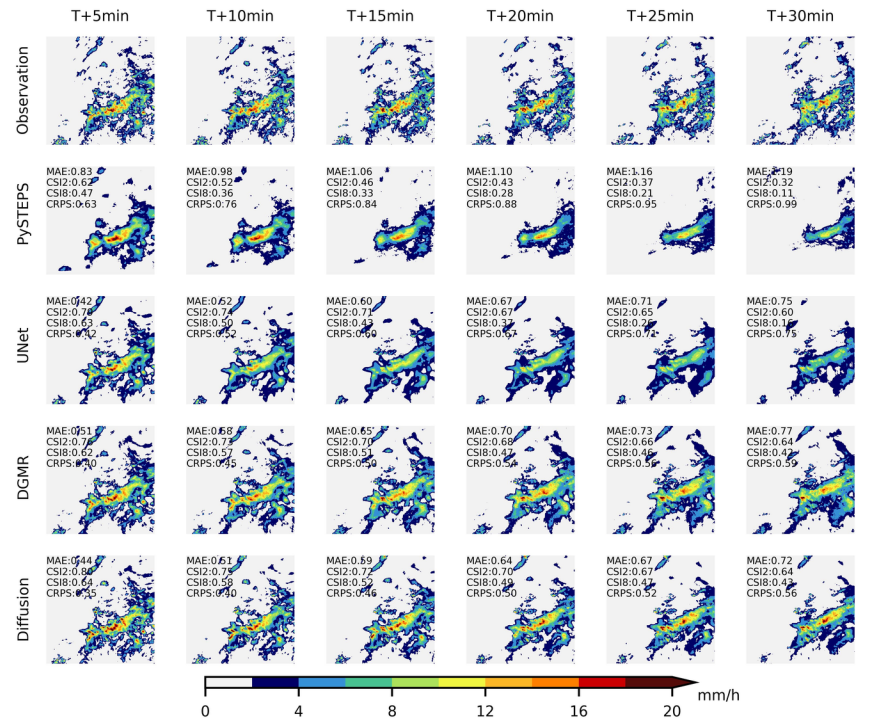
November 8, 2023

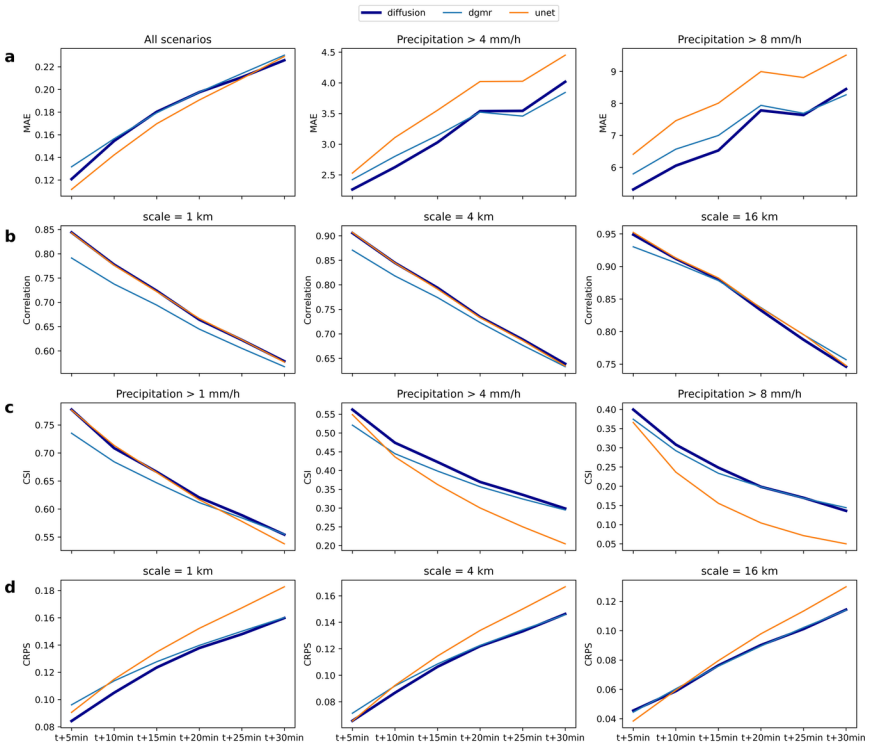
Abstract

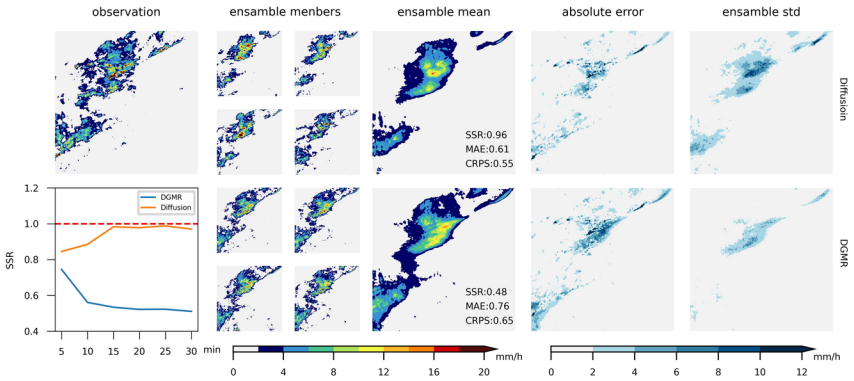
Precipitation nowcasting is a crucial element in current weather service systems. Data-driven methods have proven highly advantageous, due to their flexibility in utilizing detailed initial hydrometeor observations, and their capability to approximate meteorological dynamics effectively given sufficient training data. However, current data-driven methods often encounter severe approximation/optimization errors, rendering their predictions and associated uncertainty estimates unreliable. Here we develop a probabilistic diffusion model-based precipitation nowcasting methodology, overcoming the notorious blurriness and mode collapse issues in existing practices. Our approach results in a 3.7% improvement in continuous ranked probability score compared to state-of-the-art generative adversarial model-based method. Critically, we significantly enhance the reliability of forecast uncertainty estimates, evidenced in a 68% gain of spread-skill ratio skill. As a result, our approach provides more reliable probabilistic precipitation nowcasting, showing the potential to better support weather-related decision makings.

Hosted file

`978266_0_art_file_11552069_s3p5mp.docx` available at https://authorea.com/users/695859/articles/684370-reliable-precipitation-nowcasting-using-probabilistic-diffusion-model







Reliable precipitation nowcasting using probabilistic diffusion model

Congyi Nai[1,3], Baoxiang Pan[2], Jiarui Hai[4], Xi Chen[2], Qiuhong Tang[1,3],Guangheng Ni[4], Qingyun Duan[5], Bo Lu[6], Ziniu Xiao[2], Xingcai Liu[1,3,*]

[1] Key Laboratory of Water Cycle and Related Land Surface Processes, Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences, Beijing, China

[2] Institute of Atmospheric physics, Chinese Academy of Sciences, Beijing, China

[3] University of Chinese Academy of Sciences, Beijing, China

[4] State Key Laboratory of Hydro-science and Engineering, Department of Hydraulic Engineering, Tsinghua University, Beijing 100084, China

[5]The National Key Laboratory of Water Disaster Prevention, Hohai University, Nanjing, China

[6]Laboratory for Climate Studies and CMA-NJU Joint Laboratory for Climate Prediction Studies, National Climate Center, China Meteorological Administration, Beijing, China

*Corresponding author. E-mail address: xingcailiu@igsnrr.ac.cn

Abstract

Precipitation nowcasting is a crucial element in current weather service systems. Data-driven methods have proven highly advantageous, due to their flexibility in utilizing detailed initial hydrometeor observations, and their capability to approximate meteorological dynamics effectively given sufficient training data. However, current data-driven methods often encounter severe approximation/optimization errors, rendering their predictions and associated uncertainty estimates unreliable. Here we develop a probabilistic diffusion model-based precipitation nowcasting methodology, overcoming the notorious blurriness and mode collapse issues in existing practices. Our approach results in a 3.7% improvement in continuous ranked probability score compared to state-of-the-art generative adversarial model-based method. Critically, we significantly enhance the reliability of forecast uncertainty estimates, evidenced in a 68% gain of spread-skill ratio skill. As a result, our approach provides more reliable probabilistic precipitation nowcasting, showing the potential to better support weather-related decision makings.

Key Points

- We develop a probabilistic diffusion model-based precipitation nowcasting method.
- Our model enhances probabilistic and deterministic nowcasting skill.
- Our model yields accurate uncertainty quantification ensuring reliable forecast.

Plain language summary

Precipitation nowcasting is the task of predicting when and where it will rain in the upcoming hours. It allows people to plan their activities and make decisions based on expected weather conditions. As we do not always have a whole picture of current

weather information, and cannot process this information in time, the task of precipitation nowcasting is challenging. We take advantage of a novel machine learning approach to learn what possible precipitation conditions are, given current precipitation condition observed from radar. Our results offer accurate precipitation prediction. More importantly, this method assigns high uncertainty to predictions where predictions are more biased. This accurate estimate of prediction uncertainty is crucial for weather related decision makings.

1 Introduction

Precipitation nowcasting is the task of predicting upcoming precipitation (e.g, 0-2 hours) at high spatiotemporal resolutions. Reliable precipitation nowcasting, especially for storm cases, is crucial for risk and crisis preparation, water resource management, and many other societal sectors (Zhang et al., 2023).

Numerical weather prediction provides the most reliable short-to-medium range (6 hours to 2 weeks) forecasts. It makes predictions by first inferring the initial weather state, followed by calculating the state evolution, using numerical solvers of atmospheric fluid dynamics, and associated parameterization schemes that account for unresolved processes. Despite its theoretical soundness, numerical weather prediction offers poor precipitation nowcasting, due to difficulty in assimilating hydrometeor observations, limited spatiotemporal resolution, and high computation cost.

Empirical methods can make flexible use of detailed initial hydrometeor observations, such as those from radar and satellite. Vanilla forecasts can therefore be achieved by simply propagating the initial observations along time, such as the optical flow approach (Cheung & Yeung.,2012; Pulkkinen et al., 2019; Sakaino., 2013). More advanced approaches try to better simulate the dynamical processes by "learning" from data. These data-driven models are highly parameterized functions, for which the functional design is guided by inductive biases of the considered process, and the parameters are optimized by fitting the data to the model, guided by a learning objective function.

The design of learning objective functions is vital for data-driven prediction. A popular option is to minimize the mean squared error between predictions and observations. This objective function is based on the assumption that plausible predictions subject to a conditional Gaussian distribution, where the mean vector is a learnable function of the initial state, and covariance matrix is independent of the initial state:

$$P_\theta(y|x) = N(y; \mu_\theta(x), \sigma^2), \quad (1)$$

here $\mu_\theta(x)$ serves as the deterministic forecast. This formulation comes with two shortcomings. Firstly, it prohibits the exploration of the spatial structure of predictions, making it difficult to leverage data-informed prior knowledge for achieving structurally reasonable predictions. Secondly, it assumes a deterministic outcome, despite the absence of a full-profile and strictly accurate initial state estimate. As a result, deterministic models tend to yield poorly structured, blurry estimates, missing extreme cases and uncertainty quantification. These deficiencies are evident in models such as ConvLSTM, ConvGRU and Unet (Shi et al., 2015, 2017; Ayzel et al., 2019, 2020).

To fully explore the spatial structure of data and provide predictions along with uncertainty information, it is imperative to free our predictive model from a pre-defined distributional form. Instead, it is preferrable to deploy generative models to learn empirical distribution that maximize the likelihood of the observations:

$$\hat{y} \sim P_\theta(y|x),\ where\ \theta = argmax_\theta P(y|x;\theta). \quad (2)$$

A landmarking work along this direction is the Deep Generative Models of Radar (DGMR, Ravuri et al., 2021), which achieves state-of-the-art performance regarding the forecast skill and value. We believe the key contribution of DGMR is that, it marks a pioneering attempt to bridge probabilistic forecast and generative modeling: a probabilistic forecast should encapsulate all plausible outcomes (requirement of calibration), thereafter maximize the sharpness of its predictive distribution (requirement of sharpness, Gneiting et al., 2007). DGMR employs a spatial and a temporal discriminator neural network to guarantee that observation stays within the predictive distribution. Meanwhile, it implicitly enhances the sharpness of its predictive distribution by having the ensemble mean stay close to observation. There are two potential drawbacks here. First, the two objectives in DGMR can be in conflict, making it tricky to maximize the sharpness of the predictive distribution while guaranteeing the model is well calibrated. Second, due to unneglectable optimization errors, generative adversarial net (GAN) tends to miss plausible modes in approximating complicated distributions, resulting in biased probabilistic forecast (Prafulla Dhariwal & Alex Nichol, 2021; Ali Razavi et al., 2019).

To address these challenges, we introduce diffusion models (Sohl-Dickstein et al., 2015; Song & Ermon, 2020b; Ho et al., 2020) for precipitation nowcasting. Unlike GANs, probabilistic diffusion models are likelihood-based generative models, that is, they are trained to directly maximize the probability assigned to the observed samples. This enables a full coverage of the target distributions (Ali Razavi et al., 2019; Dhariwal & Nichol, 2021). Moreover, their iterative generation nature allows us to flexibly control the resulting distribution using initial state information. As a result, we can gradually enhance the sharpness of the predictive distribution, while guaranteeing the predictive distribution encapsulates all plausible outcomes.

Diffusion models have proven successful in various research domains, tackling complex tasks like image synthesis (Dhariwal & Nichol, 2021), audio synthesis (Kong et al., 2020), and video generation (Voleti et al., 2022; Höppe et al., 2022; Ho et al., 2022). Their desirable properties make them an effective tool for achieving reliable probabilistic forecasts with informative forecast uncertainty estimates. In this study, we propose an advanced diffusion model of nowcasting and verify with the subset of well-established UK MetOffice radar dataset.

2 Methods

2.1 **Probabilistic modeling the Precipitation nowcasting**

Consider a sequence of precipitation field data $\boldsymbol{R} = [r_1, r_2, \ldots r_{\mathrm{M}}]$, the nowcasting task is to predict future precipitation field trajectories (N fields) based on a given past trajectory of observations (M fields). Here, we formulate this problem as a probabilistic machine learning task. Using an extensive dataset of sequences of precipitation field

data, we learn conditional probability model of $P_\theta(\boldsymbol{R}_{\mathrm{M+1:M+N}}|\boldsymbol{R}_{\mathrm{1:M}})$, thus

$$\hat{r}_{M+1}, \ldots \hat{r}_{M+N} \sim P_\theta(\boldsymbol{R}_{M+1:M+N}|\boldsymbol{R}_{1:M}). \quad (3)$$

This learning process is facilitated by a conditional diffusion model. A common strategy for approximating this target distribution is learning a mapping between the target and a tractable latent distribution, such as a standard Gaussian. Then we can deduce the target distribution via a procedure termed ancestral sampling, describe by

$$P_\theta(\boldsymbol{R}_{\mathrm{M+1:M+N}}|\boldsymbol{R}_{\mathrm{1:M}}) = \int P(\boldsymbol{R}_{\mathrm{M+1:M+N}}|Z, \boldsymbol{R}_{\mathrm{1:M}}, \theta)P(Z|\boldsymbol{R}_{\mathrm{1:M}})dZ \quad (4)$$

In the following sections, we demonstrate how this is accomplished in diffusion models. Mathematical details are given in Supporting Information.

2.2 Basic diffusion

Diffusion model approximates a target distribution by sequentially reversing a stochastic process, using a series of neural network models. Let $P(X_0)$ be the targe distribution. We define the following discrete time Gaussian process:

$$q(X_t|X_{t-1}) = N\left(X_t; \sqrt{1-\beta_t}X_{t-1}, \beta_t \boldsymbol{I}\right) \quad (4)$$

Here, $X_{\mathrm{t=[1,T]}}$ are latent variables. $0 < \beta_t < 1$ is diffusion coefficient. Given large enough T, $q(X_T|X_0)$is close to standard Gaussian. Therefore, the forward Gaussian process maps any target distribution $P(X_0)$ to standard Gaussian. To approximate $\mathrm{P}(X_0)$, starting from standard Gaussian, we sequentially reverse the Gaussian process using the following variational distributions:

$$P_\theta(X_{t-1}|X_t) = N(X_{t-1}; \mu_\theta(X_t), \Sigma_\theta(X_t)) \quad (5)$$

a common objective function for learning these variational distributions is the following evidence lower bound L_{VLB} defined over $X_{\mathrm{1:T}}$,

$$L_{VLB} = \mathbb{E}_q[D_{KL}(q(X_T|X_0)||p_\theta(X_T)) + \sum_{t=2}^{T} D_{KL}(q(X_{t-1}|X_t, X_0)||p_\theta(X_{t-1}|X_t)) - logp_\theta(X_0|X_1) \quad (6)$$

under certain simplification, this evidence lower bound can be simplified to a remarkably short expression in terms of fisher divergence:

$$L_{simple} = \mathbb{E}_{t\sim[1,T], X_0\sim q(X_0), \epsilon\sim N(0,I)}\left[||\nabla log\, P(X_t) - \epsilon_\theta(X_t, t)||^2\right] \quad (7)$$

Here $\epsilon_\theta(X_t, t)$ is a neural network parameterization of $\nabla log\, P(X_t)$, which is called score function. By learning the score function of the true data distribution, we can generate samples by starting at $X_T \sim N(0, I)$, and iteratively following the score function until a mode (X_0) is reached.

2.3 Conditional diffusion

Our objective is to approximate the conditional distribution of $P(X_t|y)$. Begin with the score-based formulation of a diffusion model, the goal is to learn $\nabla log\, P(X_t|y)$, by Bayes rules, we can get the equivalent:

$$\nabla log\, P(X_t|y) = \nabla log\left(\frac{P(y|X_t)\, P(X_t)}{P(y)}\right) \quad (8)$$

$$= \nabla log\, P(X_t) + \nabla log\, P(y|X_t) - \nabla log\, P(y) \quad (9)$$

$$= \underbrace{\nabla log\, P(X_t)}_{unconditional\ score} + \underbrace{\nabla log\, P(y|X_t)}_{conditional\ score} \quad (10)$$

To better control the conditional information, a hyperparameter γ is introduced to scale the gradient of the conditioning information. The score function can then be summarized as:

$$\nabla \log P(X_t|y) = \nabla \log P(X_t) + \gamma \nabla \log P(y|X_t). \quad (11)$$

Intuitively speaking, the γ = 0 the diffusion model can ignore the conditional information entirely, while a large γ value would cause the model to heavily incorporate the conditional information during sampling. In order to implement effective control over the conditional information, we use classifier-free guidance (Ho & Salimans, 2021). To get the score function under Classifier-Free Guidance, we can rearrange:

$$\nabla \log P(y|X_t) = \nabla \log P(X_t|y) - \nabla \log P(X_t). \quad (12)$$

Substituting equation (12) into equation (11) then we get:

$$\nabla \log P(X_t|y) = \nabla \log P(X_t) + \gamma(\nabla \log P(X_t|y) - \nabla \log P(X_t)). \quad (13)$$

$$= \underbrace{(1-\gamma)\nabla \log P(X_t)}_{unconditional\ score} + \underbrace{\gamma \nabla \log P(X_t|y)}_{conditional\ score} \quad (14)$$

In this paper, we model the conditional distribution of precipitation frames in the future given the past precipitation frames $\boldsymbol{R} = [p_1, p_2, \ldots p_M]$, we learn two sets of neural networks, $\epsilon_\theta(X_t, t)$ and $\epsilon_\theta(X_t, t, R)$, to approximate the unconditional and conditional score functions $\nabla \log P(X_t)$ and $\nabla \log P(X_t|y)$,our conditional diffusion loss function is:

$$L_{condition} = \mathbb{E}_{t\sim[1,T], X_0\sim q(X_0), \epsilon\sim N(0,I)}\left[||\nabla \log P(X_t|y) - \epsilon_\theta(X_t, t, R)||^2\right] \quad (15)$$

3 Data

We utilized the publicly available UK MetOffice radar network dataset, which was obtained from DeepMind (Ravuri et al., 2021). The dataset provides radar echo data with a temporal resolution of 5 minutes and a spatial resolution of 1 km for the entire UK region from 2015 to 2019. Each data point in the dataset consists of 24 time steps and covers an area of 256 km x 256 km.

Due to computational resource limitations, we employed a subset comprising 11,000 radar samples, partitioned into three subsets: training (8,000 samples), validation (2,000 samples), and testing (1,000 samples). The principal objective of this investigation is to assess the efficacy and reliability of diffusion-based and GAN-based models for precipitation nowcasting. To optimize resource usage, we exclusively evaluated these models for 30-minute precipitation predictions. Consequently, we performed random 80x80 sub-sample extractions from the original 256x256-sized data to speed up training.

4 Model Evaluation

4.1 Baseline models

Generative models of radar (DGMR) holds the current state of the art in precipitation nowcasting. We utilized Google-Colab to load the pre-trained DGMR model and evaluate its performance using the first 30 minutes of forecasted results (Ravuri et al., 2021). UNet serves as the baseline for deterministic forecasting using deep learning (Ayzel et al., 2020). PySTEPS is a widely used precipitation nowcasting system based on ensembles (Pulkkinen et al., 2019). We adopt PySTEPS as a non-

machine learning baseline. More details of the baseline can be found in the support information.

4.2 Evaluation strategy

We employ various metrics to assess the performance of both the baseline and diffusion models on the test set. We evaluate the deterministic skill of the ensemble mean using the mean absolute error (MAE), and we provide versions of MAE that consider extreme value prediction accuracy under different precipitation intensities. The accuracy of spatial prediction is evaluated using the Critical Success Index (CSI) at different precipitation thresholds. We use the Pearson correlation coefficient to evaluate the spital pattern of predictions at different resolutions. Furthermore, the calibration and sharpness of the ensemble together is evaluated using Continuous Ranked Probability Score (CRPS) at different spatial scales. As a measure of the reliability of the ensemble, we examine the spread-skill ratio (Spread/RMSE). For details of these metrics, see support information.

5 Results and discussion

5.1 Model performance for heavy precipitation forecasts

We employ a case study of heavy precipitation to compare the performance of our model with the three baseline models. Figure 1 shows the ground truth and predicted precipitation fields. In this case, our model has consistently demonstrated superior performance across various evaluation metrics.

PySTEPS tends to underestimate the temporal changes in precipitation intensity, and falls short in adequately capturing the entire precipitation field. As lead time increases, the UNet model provides only coarse estimates of the precipitation field, resulting in highly blurred predictions that lack accuracy in predicting precipitation intensity and small-scale spatial features.

GAN-based models (DGMR) can indeed address blurred predictions. However, it is more difficult to capture the precipitation pattern, results in poor probabilistic forecasting performance, which is evident on larger CRPS, higher ensemble-averaged MAE and worse CSI compared to diffusion models.

By comparison, our model ensures accurate and comprehensive coverage of precipitation fields and shows an enhanced ability in predicting precipitation intensity and small-scale spatial features, making its predictions more informative and valuable.

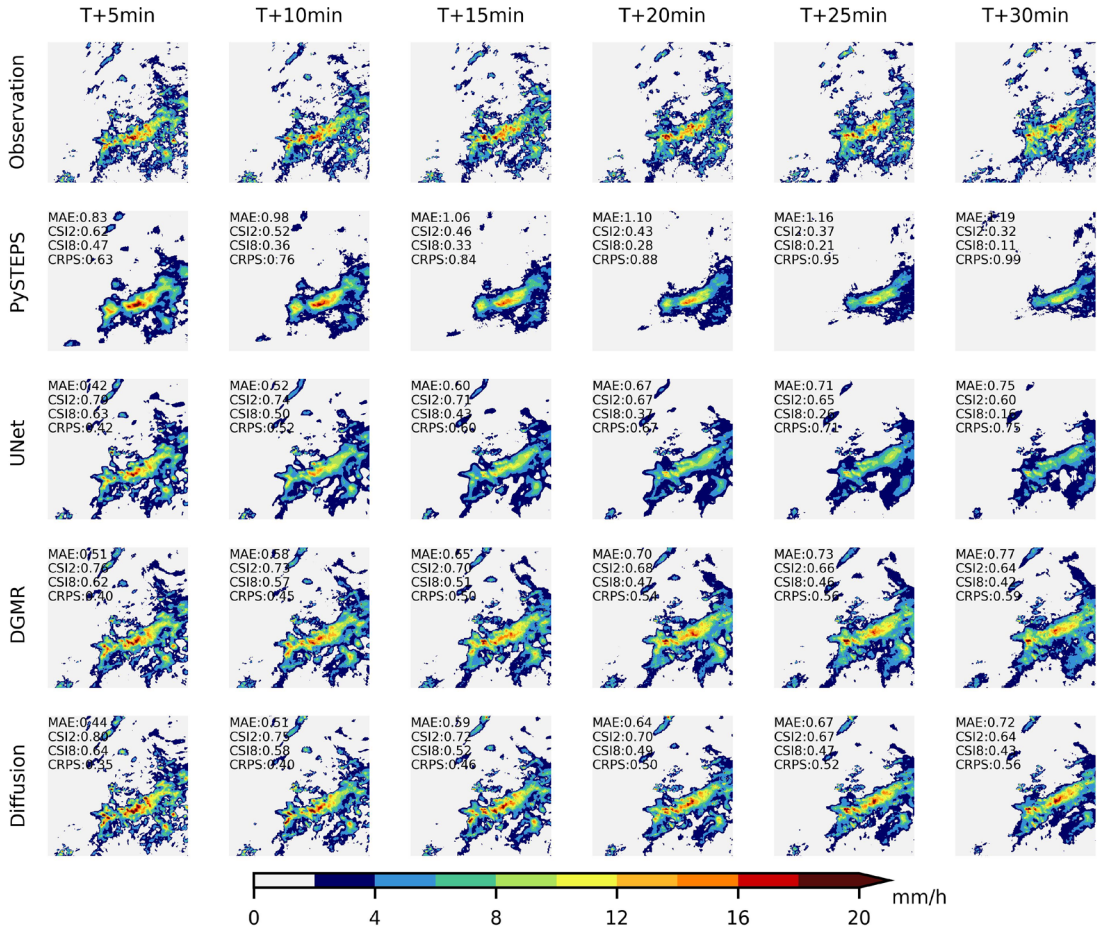


Figure 1. The performance of different baselines in heavy precipitation scenarios. The predictions for 6time steps from T+5min to T+30min were evaluated. CSI at thresholds 2 (mm/h) and 8 (mm/h), MAE and CRPS for an ensemble of 8 samples displayed in the top left corner of each time step prediction.

5.2 Forecast skill evaluation

Machine learning methods are superior to PySTEPS indicated by all metrics except for CSI (thresholds at 8 mm/h) where PySTEPS outperforms UNet. For the sake of clarity, Figure 2 will not display the metrics for PySTEPS, the complete forecast skill evaluation can be found in the support information.

Figure 2a (all scenarios) shows that the performance of UNet is slightly better than that of DGMR and diffusion on MAE. It is because that deterministic models are optimized for the mean of all precipitation scenarios, and therefore, the ensemble mean is expected to exhibit slightly lower performance in deterministic metrics like correlation and MAE compared to UNet. Unet's performance noticeably declines for heavy precipitation due to its tendency to generate blurred precipitation forecasts and our model (diffusion) performing better on heavy precipitation. Figure 2b evaluates the spatial correlation at different resolutions. Our model performs similarly to UNET and outperforms DGMR at resolutions of 1 km and 4 km. Figure 2c proves the superiority of our model over other baseline models in terms of location accuracy, as measured

across varying CSI threshold values. Unet deteriorates significantly with increased lead time and CSI threshold due to its inherent theoretical constraints in addressing this challenge. At both grid scales (1km) and 4km spatial resolutions, our model surpasses other baseline models in terms of CRPS (Figure 2d). With a spatial resolution of 16km, our model performance aligns with that of DGMR.

Despite being trained against a limited dataset, our model shows significant competitiveness. Within its 30-minute training period, our model consistently surpasses DGMR and other baselines in CSI and CRPS metrics. On average, across all forecasted time steps, our model exhibits an improvement of 3.7% in the CRPS (at a resolution of 1km) and an enhancement of 2.6%, 5.2%, 3.5% in CSI at an intensity threshold of 1.0, 4.0, and 8.0 mm/h, compared to the DGMR.

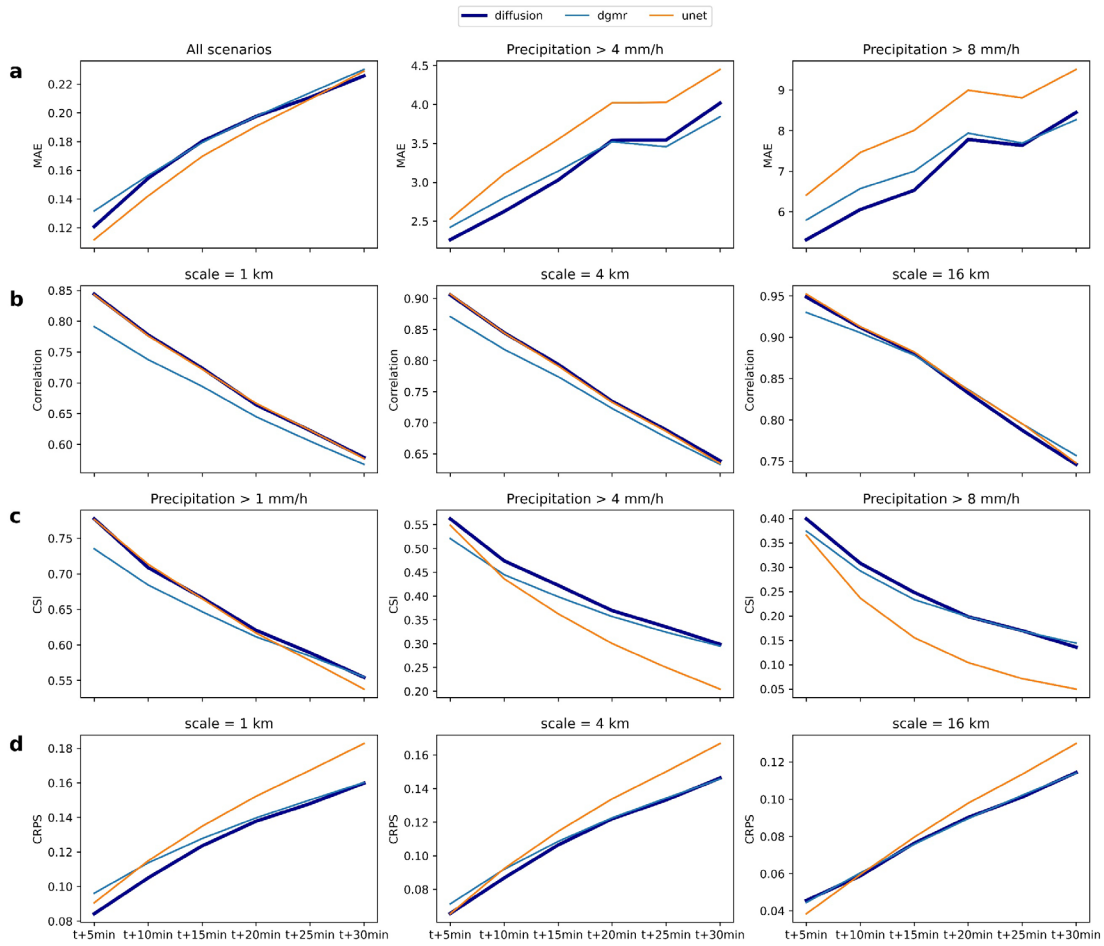


Figure 2. Evaluation metrics for the test-dataset. The probability forecast is generated using 8 ensemble members, while the Unet model is used for a single deterministic forecast. **a**, shows the MAE under different precipitation intensity conditions. MAE across all precipitation conditions (left); MAE considering observed precipitation greater than 4 mm/h (middle), MAE considering observed precipitation greater than 8 mm/h (right). **b**, correlation at different resolution**. c**, CSI for precipitation thresholds at 1 mm/h (left), 4 mm/h (middle) and 8 mm/h (right). **d**, CRPS score at different spital resolution. Grid resolution (1km) (left), average pooled 4km resolution (middle), average pooled 16km resolution (right). For MAE and CRPS, lower is better. For CSI and correlation, closer to 1 is better.

5.3 Reliability quantification

The incorporation of reliability estimation is crucial for decision-making processes and risk assessment. We assess forecast reliability for the diffusion model and DGMR, presenting ensemble members, ensemble mean, standard deviation, and absolute error maps in comparison with observations at the thirtieth minute (Figure 3). Reliability evaluations for alternative scenarios are available in the support information. The standard deviation represents the spread of the ensemble predictions, serving as a proxy of uncertainty within the precipitation forecasts. The spatial map of absolute error provides insights into the areas where the model may struggle to predict. Therefore, it is desirable for the model to provide a higher level of uncertainty in regions where its performance is poor. A balance between calibration and spread must be achieved.

Figure 3 illustrates that DGMR achieve a smaller standard deviation compared to diffusion model, which is also reflected in a high degree of similarity among ensemble members. DGMR enhanced ensemble sharpness, but fell short in terms of calibration, evident in the larger mean absolute error. It failed to establish a spatial consistency between forecast skill and forecast spread. For example, DGMR's predictions fail to reflect uncertainty at the left boundary. This means DGMR may generate overconfident predictions. Reliability is quantified using the spread-skill ratio (SSR), where an ideal ensemble model yields an SSR of 1.0. Here, the diffusion model attains an SSR of 0.96, surpassing DGMR's 0.48, establishing its superior reliability. Additionally, diffusion model exhibits superior probabilistic and deterministic forecast skills.

We also calculated SSR over the test dataset, displayed at the bottom left of Figure 3. For DGMR, the SSR values are 0.745, 0.561, 0.534, 0.522, 0.523, and 0.511, with an average of 0.56. In contrast, diffusion yields SSR values 0.845, 0.885, 0.983, 0.978, 0.988, and 0.970, with an average of 0.94. Diffusion model achieves a 68% gain in the spread-skill ratio, underscoring its ability to provide more reliable forecasts.

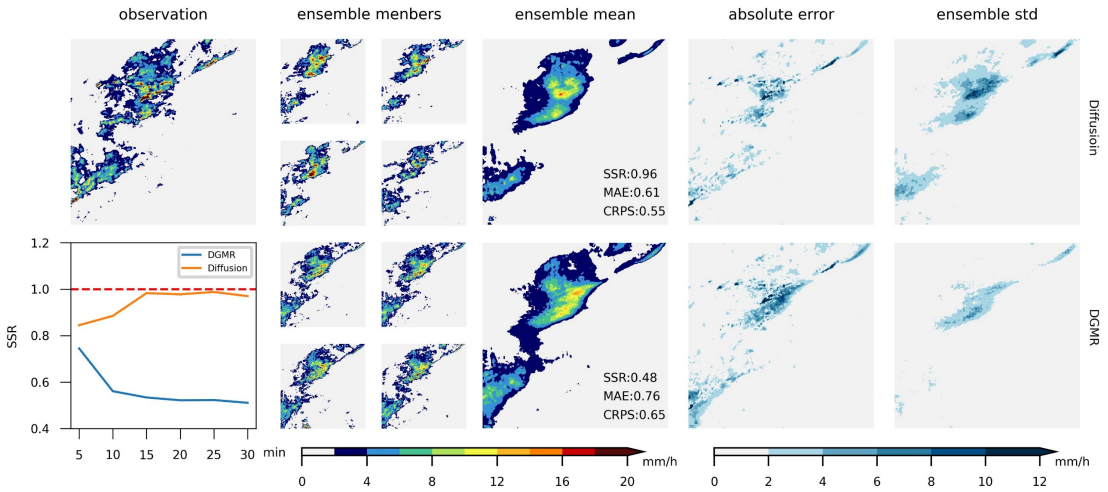


Figure 3. The example of ensemble forecasts provided by DGMR and Diffusion at the thirtieth minute. From left to right: four randomly selected ensemble members, the ensemble mean, the absolute error map comparing the ensemble mean to observations, and the ensemble standard deviation. The bottom-left panel displays the reliability

quantification SSR (Spread-Skill Ratio) calculated using the entire test dataset for forecasts.

6 Conclusions

Predicting when and where precipitation is likely to occur with high accuracy in the short term remains a difficult task. Such forecasts are essentially probabilistic: as we do not have comprehensive initial weather state estimate, and cannot fully resolve the weather dynamics, we should provide a range of possible outcomes along with their likelihood estimates, instead of a single deterministic prediction.

Data-driven methods have proven highly advantageous for precipitation nowcasting, due to their flexibility in utilizing detailed initial hydrometeor observations, and their capability to approximate meteorological dynamics effectively. State-of-the-art data-driven precipitation nowcasting approaches take advantage of deep generative models to yield probabilistic forecast. However, these methods, mostly based on generative adversarial nets (Goodfellow et al. 2014), are often faced with severe approximation/optimization errors, rendering their predictions and associated uncertainty estimates unreliable.

In this study, we present a probabilistic diffusion model-based methodology for precipitation nowcasting. The model learns predictive distributions by explicitly maximizing the data likelihood. It achieves advantageous sample fidelity, distribution diversity, and control flexibility by applying a principled, iterative way for generative modeling tasks.

Our diffusion model provides significantly improved probabilistic forecasts and consistently outperforms benchmark models over a thirty-minute forecast period, as indicated by well-established probabilistic CRPS and SSR skill scores. In terms of deterministic metrics, including MAE CSI and correlation, our model performs on par with the deterministic model UNet and probabilistic model DGMR but particularly excels Unet for heavy rainfall forecasts. More importantly, the diffusion model provides a more informative assessment of the uncertainty associated with its forecasts, making its prediction more reliable.

However, there remain some challenges to be addressed for our probabilistic nowcasting model. Its high computational resource requirement restricts the input size and limits our prediction horizon to 30 minutes. Nevertheless, this constraint may potentially be addressed by employing a latent diffusion model (Robin et al., 2021). Furthermore, we could explore the use of 3D convolutions and the development of temporal attention modules to improve temporal continuity.

In conclusion, despite these constraints, our model has demonstrated superior predictive accuracy and reliability. These qualities make our model a promising tool for precipitation nowcasting, capable of delivering more accurate and reliable forecasts.

Data availability

All data used in this study are available from Ravuri et al. (2021).

Acknowledgements

This research was supported by the Third Xinjiang Scientific Expedition Program (Grant No. 2021xjkk0806), the National Natural Science Foundation of China (42271032, U2243226, 42288101), NSFC-DFG mobility (M-0468), Chinese Academy of Science Light of the West Interdisciplinary Research Grant (grant no. xbzg-zdsys-202104) and National Key R&D Program of China 2021YFA0718000.

Reference

Ayzel, G., Heistermann, M., and Winterrath, T (2019). Optical flow models as an open benchmark for radar-based precipitation nowcasting (rainymotion v0.1), Geosci. Model Dev., 12, 1387–1402, https://doi.org/10.5194/gmd-12-1387-2019

Ayzel, G., Scheffer, T., and Heistermann, M (2020). RainNet v1.0: a convolutional neural network for radar-based precipitation nowcasting, Geosci. Model Dev., 13, 2631–2644, https://doi.org/10.5194/gmd-13-2631-2020

Ayzel, G., Scheffer, T., and Heistermann, M. (2020). A convolutional neural network for radar-based precipitation nowcasting, Geosci. Model Dev., 13, 2631–2644, https://doi.org/10.5194/gmd-13-2631-2020

Cheung, P., & Yeung, H. Y. (2012). Application of optical-flow technique to significant convection nowcast for terminal areas in Hong Kong.

Gneiting, T., Balabdaoui, F., & Raftery, A. E. (2007). Probabilistic forecasts, calibration and sharpness. Journal of the Royal Statistical Society Series B: Statistical Methodology, 69(2), 243-268. https://doi.org/10.1111/j.1467-9868.2007.00587.x

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. Advances in neural information processing systems, 27. https://doi.org/10.48550/arXiv.1406.2661

H. Sakaino, (2013). Spatio-Temporal Image Pattern Prediction Method Based on a Physical Model With Time-Varying Optical Flow, in IEEE Transactions on Geoscience and Remote Sensing, vol. 51, no. 5, pp. 3023-3036, http://doi.org/10.1109/TGRS.2012.2212201

Jonathan Ho and Tim Salimans (2021). Classifier-free diffusion guidance. In NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications

Jonathan Ho, Ajay Jain, and Pieter Abbeel (2020). Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems 33, 6840-6851. https://doi.org/10.48550/arXiv.2006.11239

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi and David J. Fleet (2022). Video Diffusion Models. https://doi.org/10.48550/arXiv.2204.03458

Pulkkinen, S., Nerini, D., Pérez Hortal, A. A., Velasco-Forero, C., Seed, A., Germann, U., & Foresti, L (2019). Pysteps: an open-source Python library for probabilistic precipitation nowcasting (v1.0), Geosci. Model Dev., 12, 4185–4219, https://doi.org/10.5194/gmd-12-4185-2019.

Ravuri, S., Lenc, K., Willson, M., Kangin, D., Lam, R., Mirowski, P., Fitzsimons, M., Athanassiadou, M., Kashem, S., Madge, S., Prudden, R., Mandhane, A., Clark, A., Brock, A., Simonyan, K., Hadsell, R., Robinson, N., Clancy, E., Arribas, A., and Mohamed, S (2021). Skilful precipitation nowcasting using deep generative models of radar, Nature, 597, 672–677, https://doi.org/10.1038/s41586-021-03854-z

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 10684-10695). https://doi.org/10.48550/arXiv.2112.10752

Shi, Xingjian, Chen, Zhourong, Wang, Hao, Yeung, Dit-Yan, Wong, Wai Kin & WOO, Wang-chun. (2015). Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting.

Sohl-Dickstein, J., Weiss, E. A., Maheswaranathan, N., and Ganguli, S (2015). Deep unsupervised learning using nonequilibrium thermodynamics. https://doi.org/10.48550/arXiv.1503.03585

Song, J., Meng, C., and Ermon, S (2021). Denoising Diffusion Implicit Models. In International Conference on Learning Representations. https://doi.org/10.48550/arXiv.2010.02502

Song, Y. and Ermon, S (2020). Generative modeling by estimating gradients of the data distribution. https://doi.org/10.48550/arXiv.1907.05600

Tobias Höppe, Arash Mehrjou, Stefan Bauer, Didrik Nielsen and Andrea Dittadi (2022). Diffusion Models for Video Prediction and Infilling. https://doi.org/10.48550/arXiv.2206.07696

Vikram Voleti, Alexia Jolicoeur-Martineau and Christopher Pal (2022). MCVD: Masked Conditional Video Diffusion for Prediction, Generation, and Interpolation. https://doi.org/10.48550/arXiv.2205.09853

Xingjian Shi, Zhihan Gao, Leonard Lausen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun Woo. (2017). Deep learning for precipitation nowcasting: a benchmark and a new model. In Advances in Neural Information Processing Systems vol. 30, 5622–5632.

Zhang, Y., Long, M., Chen, K. et al. Skilful nowcasting of extreme precipitation with NowcastNet. Nature 619, 526–532 (2023). https://doi.org/10.1038/s41586-023-06184-4

Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao and Bryan Catanzaro (2020). DiffWave: A Versatile Diffusion Model for Audio Synthesis. https://doi.org/10.48550/arXiv.2009.09761

Supporting Information for "Reliable precipitation nowcasting using probabilistic diffusion model"

Congyi Nai[1,3], Baoxiang Pan[2], Jiarui Hai[4], Xi Chen[2], Qiuhong Tang[1,3],Guangheng Ni[4], Qingyun Duan[5], Bo Lu[6], Ziniu Xiao[2], Xingcai Liu[1,3,*]

[1] Key Laboratory of Water Cycle and Related Land Surface Processes, Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences, Beijing, China

[2] Institute of Atmospheric physics, Chinese Academy of Sciences, Beijing, China

[3] University of Chinese Academy of Sciences, Beijing, China

[4] State Key Laboratory of Hydro-science and Engineering, Department of Hydraulic Engineering, Tsinghua University, Beijing 100084, China

[5]The National Key Laboratory of Water Disaster Prevention, Hohai University, Nanjing, China

[6]Laboratory for Climate Studies and CMA-NJU Joint Laboratory for Climate Prediction Studies, National Climate Center, China Meteorological Administration, Beijing, China

*Corresponding author. E-mail address: xingcailiu@igsnrr.ac.cn

1 Details of diffusion model

1.1 Basic diffusion

Let X_0 be a sample from the data distribution $q(X_0)$, and defines a sequence of increasingly noisy versions of X which we call the latent variables X_t $(t = 1 \dots T)$ through the forward diffusion process, described by

$$q(X_t|X_{t-1}) = N(X_t; \sqrt{1-\beta_t}X_{t-1}, \beta_t \boldsymbol{I}) \quad (1)$$

Then, the form of $q(X_t|X_0)$ can be recursively derived through repeated applications of the reparameterization trick, suppose we have $\{\epsilon_t, \bar{\epsilon}_t\}_{t=0}^{T} \sim N(0, I)$, Then, for an arbitrary sample $X_t \sim q(X_t|X_0)$, we can rewrite it as:

$$\begin{aligned} x_t &= \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}\epsilon_{t-1} \\ &= \sqrt{\alpha_t}(\sqrt{\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_{t-1}}\epsilon_{t-2}) + \sqrt{1-\alpha_t}\epsilon_{t-1} \\ &= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \sqrt{\alpha_t - \alpha_t\alpha_{t-1}}\epsilon_{t-2} + \sqrt{1-\alpha_t}\epsilon_{t-1} \\ &= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\bar{\epsilon}_{t-2} \\ &= \cdots \\ &= \sqrt{\textstyle\prod_{i=1}^{t}\alpha_i}\, x_0 + \sqrt{1-\textstyle\prod_{i=1}^{t}\alpha_i}\,\bar{\epsilon}_0 \\ &= \sqrt{\bar{\alpha}_t}X_0 + \sqrt{1-\bar{\alpha}_t}\bar{\epsilon}_0, \textit{ where } \bar{\alpha}_t = \textstyle\prod_{i=1}^{t}\alpha_i \end{aligned} \quad (2)$$

In equation 3, we have leveraged the property that the sum of two independent Gaussian random variables retains a Gaussian distribution, with the mean being the sum of the two individual means and the variance being the sum of their variances. $\sqrt{\alpha_t - \alpha_t\alpha_{t-1}}\epsilon_{t-2}$ is a sample from Gaussian $N(0, (\alpha_t - \alpha_t\alpha_{t-1})\boldsymbol{I})$, $\sqrt{1-\alpha_t}\epsilon_{t-1}$ is a sample from Gaussian $N(0, (1-\alpha_t)\boldsymbol{I})$, we can then treat their sum as a random variable sampled from Gaussian $N(0, (1-\alpha_t+\alpha_t-\alpha_t\alpha_{t-1})\boldsymbol{I})$. Hence, the X_t can be sampled directly from X_0, the transition kernel is

$$q(X_t|X_0) = N(X_t; \sqrt{\bar{\alpha}_t}X_0, \sqrt{1-\bar{\alpha}_t}\boldsymbol{I}), where\ \alpha_t = 1-\beta_t, \bar{\alpha}_t = \textstyle\prod_{i=1}^{t}\alpha_i. \quad (3)$$

Given X_0 and a Gaussian vector $\epsilon \sim N(0, I)$ and applying the transformation

$$X_t = \sqrt{\bar{\alpha}_t} X_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon. \quad (4)$$

When the $\bar{\alpha}_T \to 0$, X_T is well approximated by Gaussian distribution. During the forward process, noise is gradually added to the data until it loses its original spatial structure characteristics and becomes pure noise. If we can solve the reverse process $P(X_{t-1}|X_t)$, we can sample $X_T \sim N(0, I)$ then a sequence of neural networks is employed to gradually reduce the noise in a series of steps $X_T, X_{T-1} \dots X_0$. These properties suggest learning a learnable Markov chain model $P_\theta(X_{t-1}|X_t)$ to approximate the true reverse process:

$$P_\theta(X_{t-1}|X_t) = N(X_{t-1}; \mu_\theta(X_t), \Sigma_\theta(X_t)), \quad (5)$$

Therefore, in a diffusion model, we are only interested in learning conditionals $P_\theta(X_{t-1}|X_t)$, the diffusion model can be optimized by maximizing the variational lower bound (VLB) of the log-likelihood of the data X_0,

$$\begin{aligned}
\mathbb{E}_{q(X_0)}(-logP_\theta(X_0)) &\le \mathbb{E}_{q(X_0)}\left[-logP_\theta(X_0) + D_{KL}\left(q(X_{1:T}|X_0)\middle||P_\theta(X_{1:T}|X_0)\right)\right] \\
&= \mathbb{E}_{q(X_0)}\left[-logP_\theta(X_0) + \int q(X_{1:T}|X_0) log \frac{q(X_{1:T}|X_0)}{P_\theta(X_{0:T})/P_\theta(X_0)} d\, X_{1:T}\right] \\
&= \mathbb{E}_{q(X_0)}\left[-logP_\theta(X_0) + \int q(X_{1:T}|X_0) log \frac{q(X_{1:T}|X_0)}{P_\theta(X_{0:T})} d\, X_{1:T} + logP_\theta(X_0)\right] \\
&= \mathbb{E}_{q(X_{0:T})} log \frac{q(X_{1:T}|X_0)}{P_\theta(X_{0:T})} = L_{VLB} \quad (6)
\end{aligned}$$

We can rewrite variational lower bound (VLB) as,

$$\begin{aligned}
L_{VLB} &= \mathbb{E}_{q(x_0 T)}[log\ \tfrac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] \\
&= \mathbb{E}_q[log\ \tfrac{\prod_{t=1}^{T} q(x_t|x_{t-1})}{p_\theta(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)}] \\
&= \mathbb{E}_q[-log\ p_\theta(x_T) + \sum_{t=1}^{T} log\ \tfrac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)}] \\
&= \mathbb{E}_q[-log\ p_\theta(x_T) + \sum_{t=2}^{T} log\ \tfrac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)} + log\ \tfrac{q(x_1|x_0)}{p_\theta(x_0|x_1)}] \\
&= \mathbb{E}_q[-log\ p_\theta(x_T) + \sum_{t=2}^{T} log\ (\tfrac{q(x_{t-1}|x_t,x_0)}{p_\theta(x_{t-1}|x_t)} \cdot \tfrac{q(x_t|x_0)}{q(x_{t-1}|x_0)}) + log\ \tfrac{q(x_1|x_0)}{p_\theta(x_0|x_1)}] \\
&= \mathbb{E}_q[-log\ p_\theta(x_T) + \sum_{t=2}^{T} log\ \tfrac{q(x_{t-1}|x_t,x_0)}{p_\theta(x_{t-1}|x_t)} + \sum_{t=2}^{T} log\ \tfrac{q(x_t|x_0)}{q(x_{t-1}|x_0)} + log\ \tfrac{q(x_1|x_0)}{p_\theta(x_0|x_1)}] \\
&= \mathbb{E}_q[-log\ p_\theta(x_T) + \sum_{t=2}^{T} log\ \tfrac{q(x_{t-1}|x_t,x_0)}{p_\theta(x_{t-1}|x_t)} + log\ \tfrac{q(x_T|x_0)}{q(x_1|x_0)} + log\ \tfrac{q(x_1|x_0)}{p_\theta(x_0|x_1)}] \\
&= \mathbb{E}_q[log\ \tfrac{q(x_T|x_0)}{p_\theta(x_T)} + \sum_{t=2}^{T} log\ \tfrac{q(x_{t-1}|x_t,x_0)}{p_\theta(x_{t-1}|x_t)} - logp_\theta(X_0|X_1)] \\
&= \mathbb{E}_q[D_{KL}(q(X_T|X_0)||p_\theta(X_T)) + \sum_{t=2}^{T} D_{KL}(q(X_{t-1}|X_t, X_0)||p_\theta(X_{t-1}|X_t) - logp_\theta(X_0|X_1)] \quad (7)
\end{aligned}$$

This formulation also has an elegant interpretation, which is revealed when inspecting each individual term:

1. $L_0 = \mathbb{E}_q[logp_\theta(X_0|X_1)]$ can be interpreted as a reconstruction term.
2. $L_T = \mathbb{E}_q[D_{KL}(q(X_T|X_0)||p_\theta(X_T))]$ represents how close the distribution of the final noisified input is to the standard Gaussian prior, is equal to zero under our

assumptions.

3. $L_t = \mathbb{E}_q[\sum_{t=2}^{T} D_{KL}(q(X_{t-1}|X_t, X_0)||p_\theta(X_{t-1}|X_t)]$ is a denoising matching term. The $q(X_{t-1}|X_t, X_0)$ acts as a ground-truth signal and $p_\theta(X_{t-1}|X_t)$ is our desired denoising transition step. This term is therefore minimized when the two denoising steps match as closely as possible. It is the primary optimization objective.

If we have knowledge of X_0, we can obtain $q(X_{t-1}|X_t, X_0)$ through the Bayes' theorem,

$$q(X_{t-1}|X_t, X_0) = q(X_t|X_{t-1}, X_0)\frac{q(X_{t-1}|X_0)}{q(X_t|X_0)}$$

$$\propto exp\left(-\frac{1}{2}\left(\frac{\left(X_t-\sqrt{\alpha_t}X_{t-1}\right)^2}{\beta_t}+\frac{\left(X_{t-1}-\sqrt{\bar{\alpha}_{t-1}}X_0\right)^2}{1-\bar{\alpha}_{t-1}}-\frac{\left(X_t-\sqrt{\bar{\alpha}_t}X_0\right)^2}{1-\bar{\alpha}_t}\right)\right)$$

$$= exp\left(-\frac{1}{2}\left(\left(\frac{\alpha_t}{\beta_t}+\frac{1}{1-\bar{\alpha}_{t-1}}\right)X_{t-1}^2+\left(\frac{2\sqrt{\alpha_t}}{\beta_t}X_t+\frac{2\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}X_0\right)X_{t-1}+C(X_t,X_0)\right)\right)$$

$$= N(X_{t-1}; \tilde{\mu}(X_t, X_0), \tilde{\beta}(t)I) \tag{8}$$

Recall equation 8 and equation 5, we can obtain,

$$\tilde{\mu}_\theta(X_t, X_0) = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(X_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(X_t, t)\right) \tag{9}$$

Let us consider the $L_t = D_{KL}(q(X_{t-1}|X_t, X_0)||p_\theta(X_{t-1}|X_t))$, given equation 6 and equation 8, we can get the loss function,

$$L_t = \mathbb{E}_{x_0,\epsilon}\left[\frac{1}{2\|\Sigma_\theta(x_t,t)\|_2^2}\| \tilde{\mu}_t(x_t, x_0) - \mu_\theta(x_t, t)\|^2\right]$$

$$= \mathbb{E}_{x_0,\epsilon}\left[\frac{1}{2\|\Sigma_\theta\|_2^2}\| \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_t\right) - \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\right)\|^2\right]$$

$$= \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)\|\boldsymbol{\Sigma}_\theta\|_2^2}\| \boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\|^2\right]$$

$$= \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)\|\boldsymbol{\Sigma}_\theta\|_2^2}\| \boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t, t)\|^2\right] \tag{10}$$

Ho et al. (2020) propose to reweight various terms in L_{VLB} for better sample quality, to compute this objective, we generate samples $X_t \sim q(X_t|X_0)$, then train a model ϵ_θ to predict the added noise using a standard mean-squared error loss:

$$L_{simple} = \mathbb{E}_{t\sim[1,T], X_0\sim q(X_0), \epsilon\sim N(0,I)}[||\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}\mathrm{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t, t)||^2]. \tag{11}$$

2.3 Conditional diffusion

So far, we have focused on modeling the data distribution p(x). However, we are often also interested in the conditional distribution of $\mathrm{P}(X_\mathrm{t}|\mathrm{y})$, as it enables us to better investigate how different conditional information influences the generation of variable X. Begin with the score-based formulation of a diffusion model, the goal is to learn $\nabla \log \mathrm{P}(X_\mathrm{t}|\mathrm{y})$, by Bayes rules, we can get the equivalent:

$$\nabla \log \mathrm{P}(X_\mathrm{t}|\mathrm{y}) = \nabla log\left(\frac{P(y|X_t)\,P(X_t)}{P(y)}\right) \tag{12}$$

$$= \nabla log\, P(X_t) + \nabla log\, P(y|X_t) - \nabla log\, P(y) \tag{13}$$

$$= \underbrace{\nabla \log P(X_\mathrm{t})}_{unconditional\ score} + \underbrace{\nabla \log \mathrm{P}(\mathrm{y}|X_\mathrm{t})}_{conditional\ score} \tag{14}$$

To better control the conditional information, a hyperparameter γ is introduced to

scale the gradient of the conditioning information. The score function can then be summarized as:

$$\nabla \log P(X_t|y) = \nabla \log P(X_t) + \gamma \nabla \log P(y|X_t). \quad (15)$$

Intuitively speaking, the $\gamma = 0$ the diffusion model can ignore the conditional information entirely, while a large γ value would cause the model to heavily incorporate the conditional information during sampling. In order to implement effective control over the conditional information, we use classifier-free guidance (Ho & Salimans, 2021). To get the score function under Classifier-Free Guidance, we can rearrange:

$$\nabla \log P(y|X_t) = \nabla \log P(X_t|y) - \nabla \log P(X_t). \quad (16)$$

Substituting equation (16) into equation (15) then we get:

$$\nabla \log P(X_t|y) = \nabla \log P(X_t) + \gamma(\nabla \log P(X_t|y) - \nabla \log P(X_t)). \quad (17)$$

$$= \underbrace{(1-\gamma)\nabla \log P(X_t)}_{unconditional\ score} + \underbrace{\gamma \nabla \log P(X_t|y)}_{conditional\ score} \quad (18)$$

From Tweedie's formula and equation 5, we can get,

$$\nabla \log p(x_t) = -\frac{1}{\sqrt{1-\bar{\alpha}_t}}\epsilon \quad (19)$$

The equation 19 means that estimating ϵ is equivalent to estimating a scaled version of the score function. So, in this paper, we model the conditional distribution of precipitation frames in the future given the past precipitation frames $\boldsymbol{P} = [p_1, p_2, \ldots p_M]$, we learn two sets of neural networks, $\epsilon_\theta(X_t, t)$ and $\epsilon_\theta(X_t, t, P)$, to approximate the unconditional and conditional score functions $\nabla \log P(X_t)$ and $\nabla \log P(X_t|y)$, our conditional diffusion loss function is:

$$L_{condition} = \mathbb{E}_{t\sim[1,T], X_0\sim q(X_0), \epsilon\sim N(0,I)}[||\epsilon - \epsilon_\theta(X_t, t, P)||^2]. \quad (19)$$

2 Details of baseline model

2.1 Generative models of radar

DGMR holds the current state of the art in precipitation nowcasting, the generator is built with convolutional and convolutional GRU layers and it was trained with two adversarial loss and a regularization loss. The first loss is defined by a spital discriminator, which ensures spital consistency. The second loss is defined by a temporal discriminator, which is a 3D convolutional neural network that aims to impose temporal consistency. The regularization term encourages the prediction's mean precipitation fields to match the mean of past precipitation amount.

We utilized Google-Colab to load the saved DGMR model and pconducted inference on our test dataset, see https://github.com/deepmind/deepmind-research/tree/master/nowcasting. DGMR exhibits the capability to generate forecasts up to 90 minutes. However, for the purpose of comparison, we only evaluated its performance using the first 30 minutes of forecasted results, calculating relevant metrics.

2.2 U-Net

We use a U-Net encoder–decoder model as baseline similarly to how it was used in related studies (Ayzel et al., 2020). This type of model first employs an encoder that reduces the spatial resolution using pooling and convolutional layers, while the decoder then increases the resolution by applying up-sampling and convolutional layers to the

learned patterns. To prevent gradient vanishing and share the low-level patterns of the precipitation fields, skip connections are used from the encoder to the decoder (Srivastava et al., 2015). In this paper, U-Net serves as the baseline for deterministic forecasting using deep learning.

2.3 PySTEPS

PySTEPS is an open-source Python library designed for radar precipitation forecasting and analysis, it is available at https://github.com/pySTEPS/pysteps. It offers a comprehensive range of algorithms, among which STEPS is a widely used precipitation nowcasting system based on ensembles, considered to be state-of-the-art of non-ML-based method. In this study, we adopt PySTEPS as a non-machine learning baseline.

3 Details of metrics

we use the M to denote number of the ensemble members, and f_m to denote the ensemble member, so the ensemble mean can be written as,

$$\bar{f} = \frac{1}{M}\sum_{m=1}^{M} f_m \tag{20}$$

3.1 MAE

The (spatial) mean-absolute-error (MAE) at forecast time step t between ensemble means $\bar{f}$ and observation f_{obser} is defined as,

$$MAE_t(\bar{f}, f_{obser}) = \frac{1}{P}\sum_{p=1}^{P}\left|\bar{f} - f_{obser}\right| \tag{21}$$

where p indexes all the geospatial locations. And we can consider extreme value prediction accuracy under different precipitation intensities, we use an intensity mask $[f_{obser} > 4]$ and $[f_{obser} > 8]$ to get the masked prediction and observation $\bar{f}_m, f_{m\_obser}$

$$MAE_{t,mid}(\bar{f}_m, f_{m\_obser}) = \frac{1}{P}\sum_{p=1}^{P}\left|\bar{f}_m - f_{m\_obser}\right| \tag{22}$$

3.2 Correlation

The spital correlation between ensemble mean and observation is defined as.

$$Corr_t(\bar{f}, f_{obser}) = \frac{\sum_p(\bar{f}_p - \overline{\bar{f}_p})(f_{obser,p} - \overline{f_{obser,p}})}{\sqrt{\sum_p(\bar{f}_p - \overline{\bar{f}_p})^2}\sqrt{\sum_p(f_{obser,p} - \overline{f_{obser,p}})^2}} \tag{23}$$

where $\overline{\bar{f}_p}$ means to average in space. In deployment, we flatten the prediction and observation then use the *corrcoef function* from the *NumPy* library.

3.3 Critical Success Index

The Critical Success Index (CSI) is a statistical measure that quantifies the accuracy of spatial prediction by evaluating the correct identification of specific events or outcomes.

The CSI is defined as the ratio of true positives (TP) to the sum of true positives, false positives (FP), and false negatives (FN). Mathematically, it is expressed as,

$$CSI = \frac{TP}{TP+FP+FN} \quad (24)$$

- TP represents the number of true positive outcomes, which signifies the accurate prediction of events or occurrences.
- FP corresponds to false positives, indicating instances where the event was predicted, but did not materialize.
- FN denotes false negatives, signifying cases where the event occurred but was not correctly predicted.

The CSI values range between 0 and 1, where a CSI of 1 indicates perfect spatial accuracy in prediction, implying that all positive outcomes were correctly forecasted without any false alarms. Conversely, a CSI of 0 suggests that none of the events were accurately predicted.

3.4 Continuous Ranked Probability Score

CRPS is used to evaluate the calibration and sharpness. It quantifies the discrepancy between the forecasted cumulative distribution function (CDF) and the observed CDF, defined as,

$$CRPS = \int_{-\infty}^{+\infty}[F(f_m) - 1(t \leq z)]^2 dz \quad (25)$$

where F denotes the CDF of the prediction distribution and 1(t ≤ z) is an indicator function that is 1 if t ≤ z and 0 otherwise. In the case of a deterministic forecast (like Unet) the CRPS reduces to the mean absolute error (MAE).

3.5 Spread-skill ratio

The SSR evaluates the reliability of the ensemble. It is a ratio that quantifies the balance between calibration and sharpness, providing insights into the trade-off between these two critical aspects of predictive modeling.

$$SSR = \frac{Spread}{RMSE} \quad (26)$$

where the spread is defined as,

$$Spread = \sqrt{\frac{1}{P}\sum_{p=1}^{P} Var(f_{m,p})} \quad (27)$$

and the RMSE is defined as,

$$RMSE = \sqrt{\frac{1}{P}\sum_{p=1}^{P}(\bar{f} - f_{obser})^2} \quad (28)$$

4 Additional results

4.1 Skill evaluation

Figure S1 includes PySTEPS metrics calculated over the entire test dataset. Due to UNet's blurred predictions, it falls short of PySTEPS in terms of CSI8.

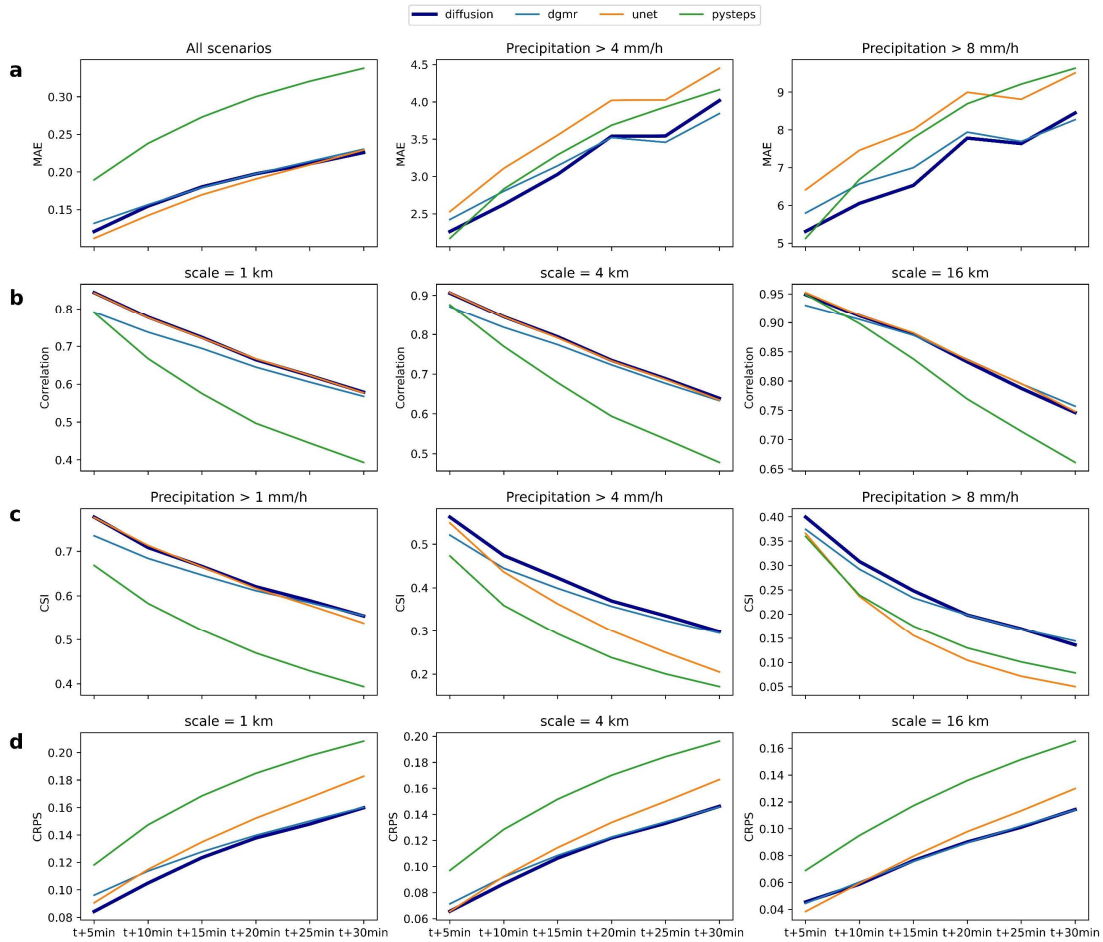


Figure S1. The full version of Figure 2, incorporating metrics for PySTEPS.

4.2 Additional case

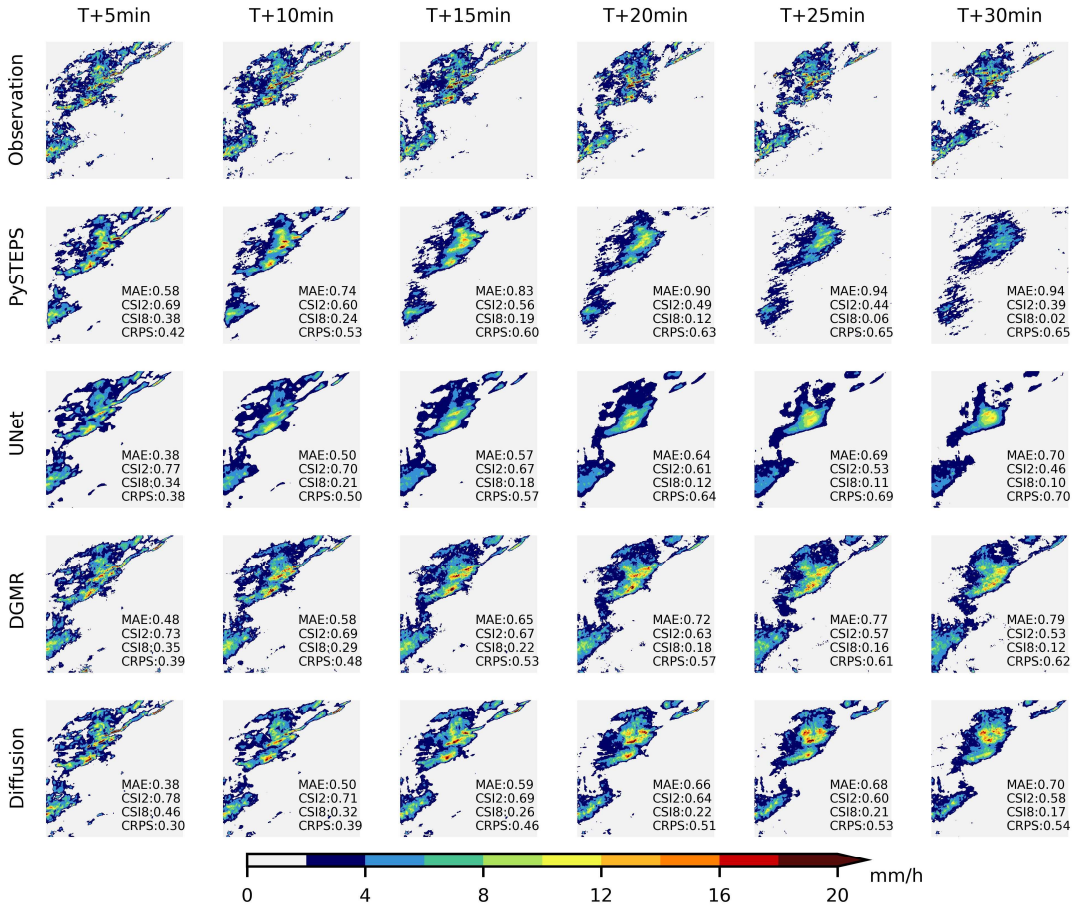


Figure S2. An additional case in Section 5.1

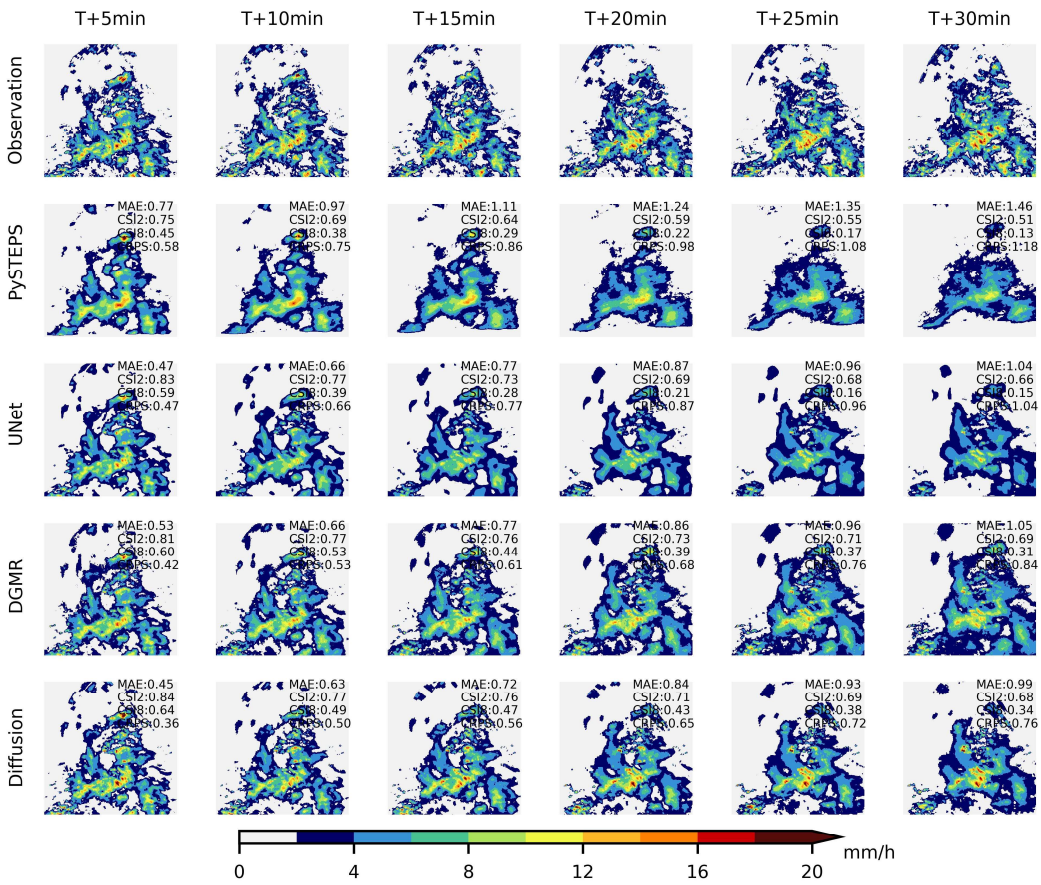


Figure S3. An additional case in Section 5.1

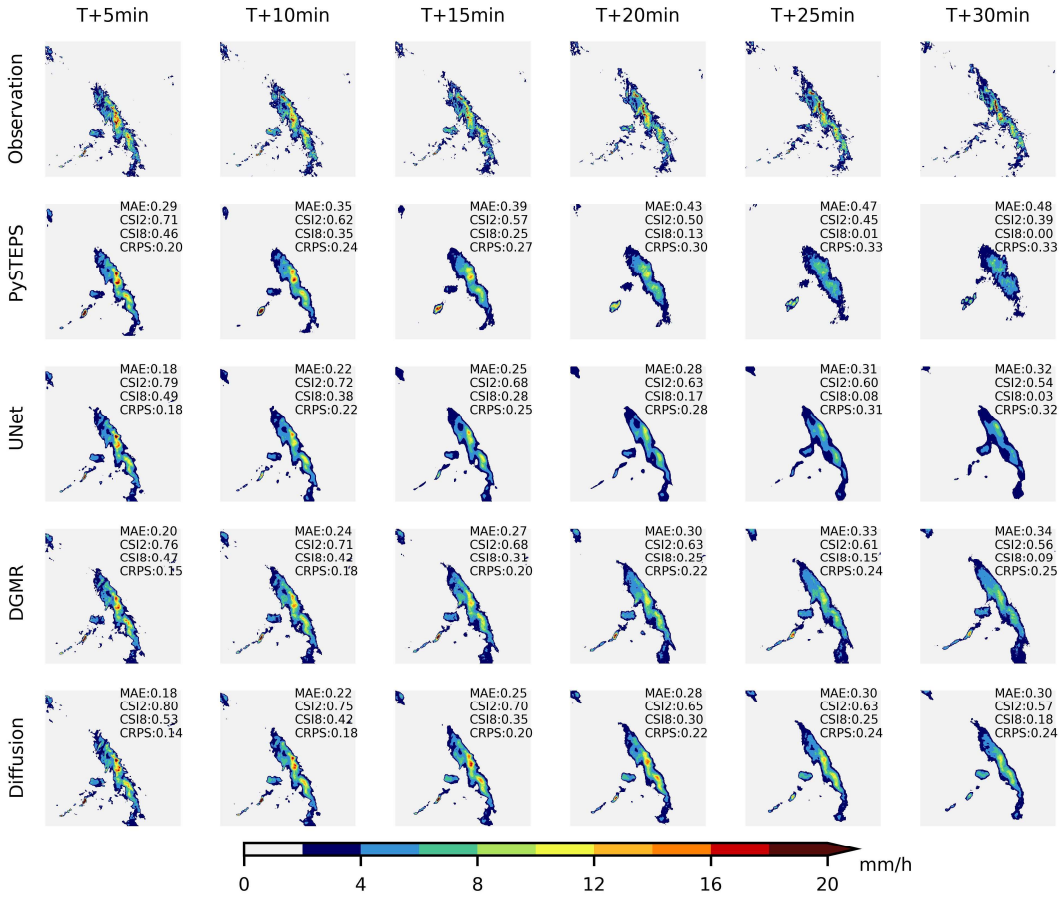


Figure S4. An additional case in Section 5.1

4.3 Reliability cases

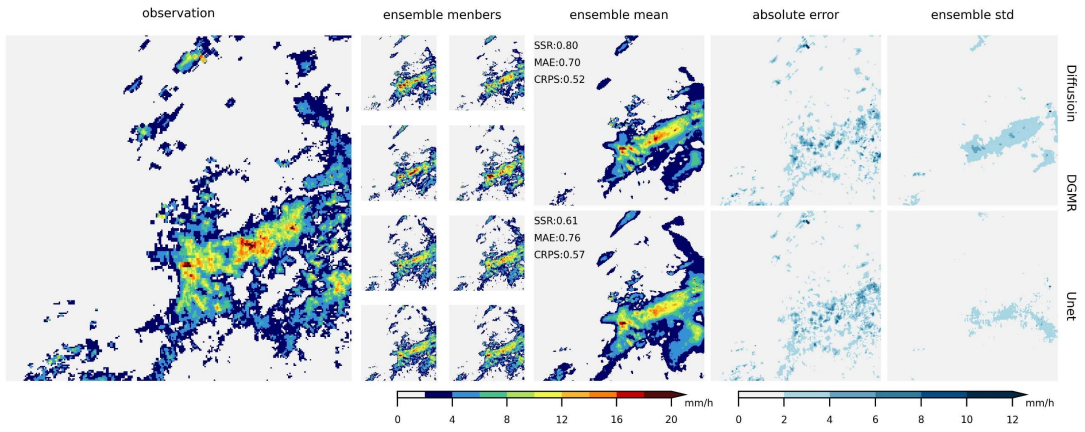


Figure S5. An additional case in Section 5.3

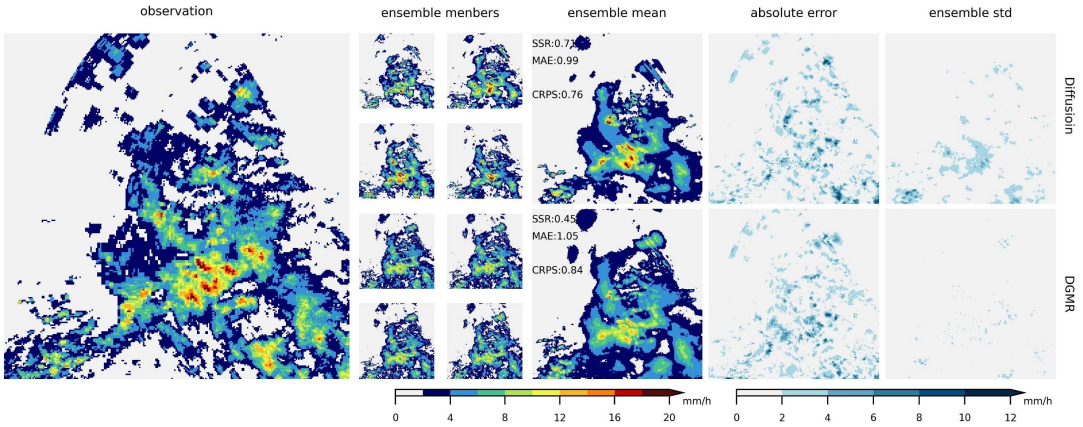


Figure S6. An additional case in Section 5.3

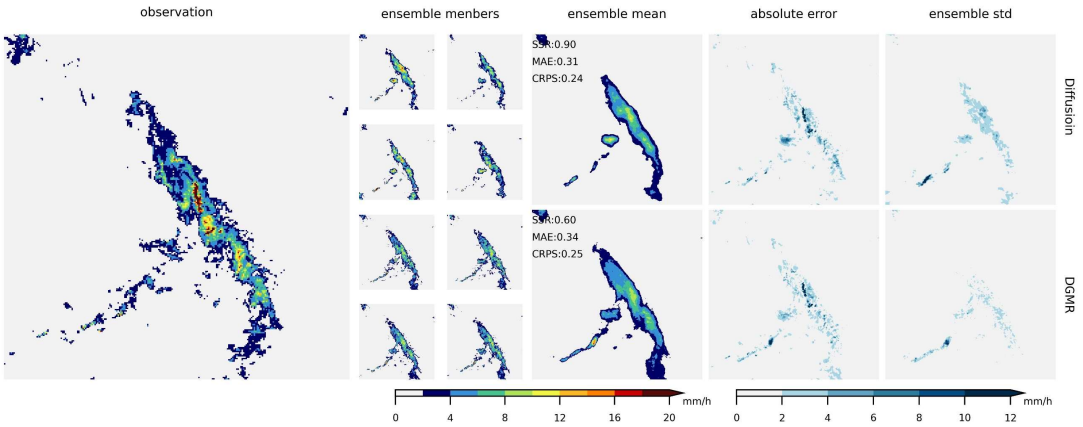


Figure S7. An additional case in Section 5.3

Reference

Vikram Voleti, Alexia Jolicoeur-Martineau and Christopher Pal (2022). MCVD: Masked Conditional Video Diffusion for Prediction, Generation, and Interpolation. https://doi.org/10.48550/arXiv.2205.09853

Understanding diffusion models: A unified perspective." arXiv preprint arXiv:2208.11970 (2022). https://doi.org/10.48550/arXiv.2208.11970

Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., ... & Yang, M. H. (2022). Diffusion models: A comprehensive survey of methods and applications. ACM Computing Surveys. https://doi.org/10.48550/arXiv.2209.00796