

Efficient Spatio-temporal Weather Forecasting Using U-Net

Akshay Punjabi, Pablo Izquierdo-Ayala

Abstract

Weather forecast plays an essential role in multiple aspects of the daily life of human beings. Currently, physics based numerical weather prediction is used to predict the weather and requires enormous amount of computational resources. In recent years, deep learning based models have seen wide success in many weather-prediction related tasks. In this paper we describe our experiments for the Weather4cast 2021 Challenge, where 8 hours of spatio-temporal weather data is predicted based on an initial one hour of spatio-temporal data. We focus on SmaAt-UNet, an efficient U-Net based autoencoder. With this model we achieve competent results whilst maintaining low computational resources. Furthermore, several approaches and possible future work is discussed at the end of the paper.

Keywords

weather4cast 2021, weather forecast, deep learning, neural networks, CNN, U-Net, SmaAt-UNet

1. Introduction

Weather prediction is an art that can be traced back to Ancient History. Around the year 650 B.C, the Babylonians were already using clouds and haloes to predict short-term weather variations. 2600 years later, weather forecasting has changed substantially but it still plays an active role in the development of our society, becoming a valuable asset in many situations, such as the creation of warnings prior to a severe storm [1].

Most of these predictions are now generated through Numerical Weather Prediction models (NWP) that provide estimates by means of various physical variables, such as atmospheric pressure, temperature, etc. While accurate, these models are often slow and require vast amounts of computational power, making them inaccessible to the public and impractical when attempting short-term forecasts [2].

In recent years, with the outbreak of Machine Learning and the growing volume of increasing higher-resolution information available, deep learning models have found major success in this domain and have managed to even rival the original NWP-based approaches [3][4]. These deep learning models do not rely in the current physical state of the atmosphere but instead utilize historical weather data to generate a future prediction.

In this paper we focus on a Convolutional Neural Network (CNN) approach.

Convolutional Neural Networks, such as U-Net[5], are a type of Artificial Neural Network (ANN) that is commonly used to process image data. They are based on convolutions, a kernel operation that allows the model to capture local invariant features in a given image. These networks are used in a wide range of tasks, especially in

object detection[6] and image classification[7] tasks.

The model employed in this work is a variant of U-Net defined as SmaAt-UNet[8]. Both model and architecture are further described throughout the text.

2. Weather4cast 2021 Challenge

Weather4cast 2021 Challenge [9] is a competition held by the Institute of Advanced Research in Artificial Intelligence (IARAI) [10] with the goal of generating a short-term prediction of selected weather products based on meteorological satellite data-products from different regions of Europe. These data-products range from February 2019 to February 2021 and are obtained in collaboration with AEMET [11] / NWC SAF [12]. This challenge presents weather forecast as a video frame prediction task, similarly to the Traffic4cast competitions at NeurIPS in 2019 [13] and 2020 [14], hosted by the same institute.

The data consists of four target weather variables: temperature (on accessible surface: top cloud or earth), convective rainfall rate, probability of occurrence of tropopause folding and cloud mask. The weather products are encoded as separate channels in the weather images. Each weather image contains 256 x 256 pixels of a particular region, in which each pixel corresponds to an area of about 4 km x 4 km. The images are recorded at 15 minute intervals throughout a year.

The goal is to predict the next 32 weather images (8 hours in 15 minute intervals) given 4 images (1 hour) of each of the regions provided.

3. Methods

There are several ways to approach this challenge, such as with ConvLSTMs [4], Graph Neural Networks (GNN) [15] and U-Nets [16]. In other similar competitions of spatio-temporal data, U-Net type architectures have shown the



Model	Backbone	Parameters
U-Net with DSC		4 Millions
U-Net with CBAM and DSC (SmaAt-UNet)		4.1 Millions
U-Net++	Efficientnet-b0 [19]	6 Millions
U-Net++	Efficientnet-b1	9.1 Millions
U-Net++	Efficientnet-b2	10.4 Millions
U-Net++	Efficientnet-b3	13.6 Millions
U-Net		17.3 Millions
U-Net with CBAM		17.4 Millions
U-Net++	Efficientnet-b4	20.8 Millions
U-Net++	Efficientnet-b5	31.9 Millions
U-Net++	SE-Resnext50 32x4d	51 Millions

Table 1

Model sizes and parameters sorted by increasing complexity. It is worth noting that transforming the original convolutions in U-Net into depthwise-separable convolutions reduces the size of the model from 17.4 Million parameters to 4.1 Million parameters and how the addition of the CBAM component only increases the amount of parameters by 0.1 Million

best results. For that reason we mainly base our work on U-nets, specially on efficient U-Nets. The neural network architecture used in our work is a recent state of the art model called SmaAt-UNet [8] (See Section 3.1). Some preliminary tests were done on the U-Net++ [17], a U-Net based model with nested dense convolutional blocks, and different backbones. Table 1 shows the size and the number of parameters of each model. These larger autoencoders were not used as they reported virtually the same results while requiring larger training time. In contrast, SmaAt-UNet is a much smaller and efficient model. As a result, all further experiments were done with the SmaAt-UNet model.

3.1. SmaAt-UNet

SmaAt-UNet is a novel model that extends the original encoder-decoder structure proposed in the U-Net architecture [5]. The architecture can be seen in Figure 1. There are two major differences when compared to its forerunner:

Firstly, the encoder contains a Convolutional Block Attention Module (CBAM) [18]. This module combines a channel attention module and a spatial attention module that enhance a given feature map.

Secondly, all the regular convolutions present in the original U-Net version are replaced by Depthwise-Separable Convolutions (DSC), allowing the model to reduce significantly the number of parameters, hence making it lightweight in comparison to the original version.

This combination improves the performance of U-Net while significantly reducing the computational cost of the model (≈ 17 Million parameters of U-Net versus the ≈ 4 Million parameters of its SmaAt counterpart, see Table 1), allowing us to obtain reasonable results in our resource-restricted environment.

4. Experiments and Results

Following the objective of the Weather4cast Core Competition, we trained and experimented with our models on regions R1 (Nile region), R2 (Eastern Europe) and R3 (South West Europe) to obtain an efficient and competent model for spatio-temporal weather forecast.

4.1. Data

We employed the four data elements defined in Section 2 (temperature, convective rainfall rate, probability of occurrence of tropopause folding and cloud mask) and 3 additional static variables (latitude, longitude and elevation) provided by the organiser, adding up to 7 dimensions.

We also modified the data structure. The original models would generate one single prediction given 4 input variables and a lead-time component. This lead-time component would then be used as an index to extract the 32 individual images from the output prediction. Our models avoid using a lead-time component and instead generate the 32 individual predictions directly from the 4 input variables.

4.2. Experimental Settings

Models were trained for 10 epochs using MSE loss and Adam [20] optimizer, with a learning rate of 0.001 and Cosine Annealing with Warm Restarts schedule. [21].

The experiments were run through a Colab Pro subscription, which provides a single restricted Tesla P100 or restricted Tesla V100, and Pytorch v1.9 [22]. This platform limits its usage to a 24h time frame, after which any running code is abruptly terminated. This time frame is reduced if overused, which caused many disruptions in our training pipeline and required active monitoring.

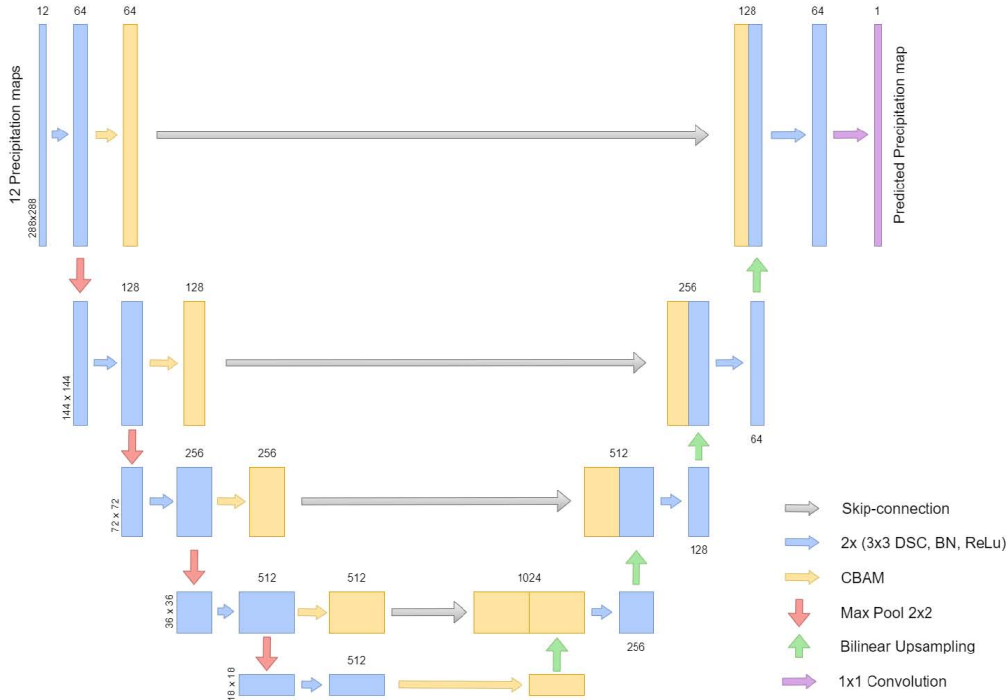


Figure 1: An example of an input fed through our proposed SmaAt-UNet (best viewed in color). Each bar represents a multi-channel feature map. The numbers above each bar display the amount of channels; the vertical numbers on the left side correspond to the x-y-size. Extracted from [8].

We also used 16-bit precision operations for a faster training speed instead of the default 32-bit precision operations.

Code and experiments are publicly available and can be found in our GitHub repository.¹

4.3. Quantitative results

An extract of our results can be seen in Table 2, with several baseline models that we used to compare our findings:

The Persistence model uses the last image of the sequence as the prediction image under the assumption that the weather will not vary significantly from a given time point t to $t+1$. By running this approach we obtain a baseline MSE of 1.0. The U-Net model is a pretrained model provided by IARAI. This model performs with an MSE of 0.669. Next is a single SmaAt-UNet, that already reduces the U-Net MSE down to 0.612, demonstrating the power of this lightweight architecture. By adding a Cosine Annealing scheduler with Warm Restarts[21], the model performs considerably better in comparison to the scheduler-free version. Finally, the best result was

¹github.com/Dauriel/weather4cast2021/

Model	MSE
Persistence	1.000
U-Net	0.669
SmaAt-UNet	0.612
SmaAt-UNet with CAWRS	0.597
Best Ensemble SmaAt-UNet	0.572

Table 2

Test MSE. In this table we indicate the MSE obtained on the test leaderboard of the competition. CAWRS stands for Cosine Annealing with Warm Restarts schedule

obtained through an ensemble of several SmaAt-UNet models, obtaining an MSE of 0.572 over the testing set.

Our methods obtain a significantly lower MSE than the baseline models while keeping a low resource demand.

4.4. Qualitative results

In Figure 3 we visualize a prediction of cloud coverage obtained from one of the test sets, in particular for March 16th 2020. Due to the uncertainty of the future, the model does not really predict future positions of cloud coverage and instead regresses to the mean for all the possible

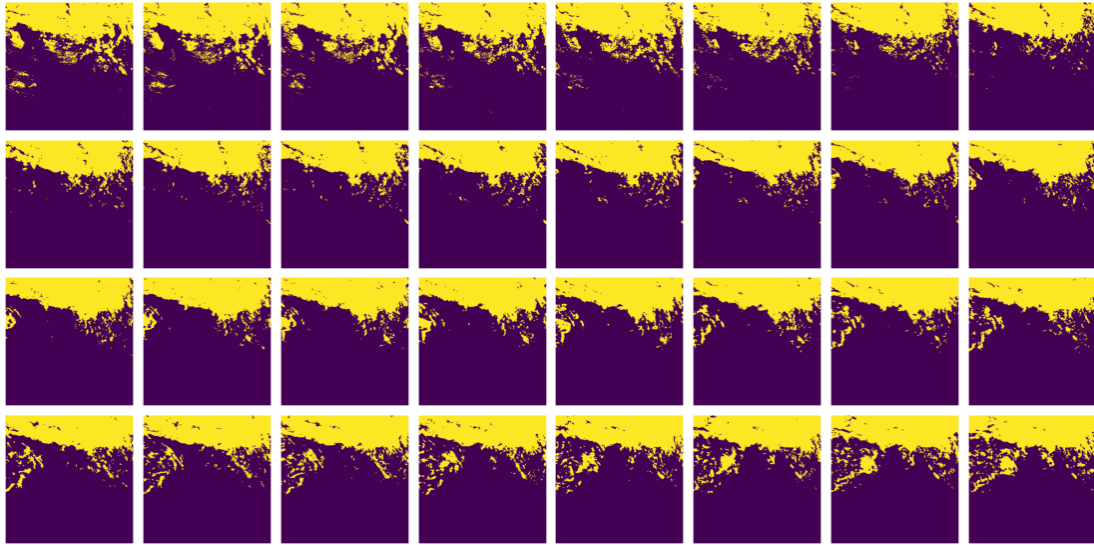


Figure 2: Ground truth of December 25th 2019 on Region R1 (Nile Region)



Figure 3: Predicted results from our best performing model for December 25th on Region R1 (Nile Region)

values.

5. Future Work

In this section, we discuss some important considerations to be taken into account towards future work.

5.1. U-Net

As we have seen in our experiments, and in other similar competitions of spatio-temporal data, U-Net type architectures have shown the best results when dealing with this type of datasets. This is due to the capability of U-Net to model spatial characteristics of the data. However temporal characteristics are not captured correctly by this architecture (See Figure 2 vs Figure 3). Over an increasing time frame, U-Net is not able to capture the

temporality of the data and predictions become considerably homogeneous in comparison to the ground truth. This condition is present in all of our predictions.

Including some kind of "memory", that is, the use of Recurrent Neural Networks (LSTM [23], ConvLSTM [24], etc) could allow the model to handle these temporal characteristics, improving the results substantially at the expense of a considerable increase in the computational resources required.

5.2. MSE loss

Another problem is that of the use of the MSE loss. The MSE loss computes the average of the pixel values so that the error is minimized for any possible real prediction value. For this specific task, a better loss function that does not result on averaging possible pixel values would perform significantly better. Some researchers have tried addressing this problem by including new loss functions like the adversarial loss and the perceptual loss [25], which works well for images (e.g. ImageNet). However, these losses would probably perform poorly for these spatio-temporal physical variables. Moreover, modifying these loss functions comes at the expense of higher and more expensive training times.

5.3. Invertible Neural Networks

Given our main focus of creating efficient low resource neural networks, we also studied the realm of *Invertible Neural Networks* (INN) [26].

INNs enable memory-efficient training by re-computing intermediate activations in the backward pass, rather than storing them in memory during the forward pass [27]. This enables efficient large-scale generative modeling [28] and high-resolution medical image analysis [29].

However, these were proven to be difficult to train and showed very notable checkerboard artifacts yielding very bad predictions. These results are inline with other papers about INN in literature [30].

5.4. Wind data as an optical flow

Optical flow models are gaining a lot of interest in recent video based tasks, such as video object detection [31] and video action recognition [32]. In fact, they are used in some of the state of the art models for video action recognition [33].

An approach could be the computation of the optical flows between each time step of the spatio-temporal images using these optical flow neural networks. However, the wind speed magnitude and wind direction of the provided data could already be considered optical flow, removing the need to artificially compute it. Using

these two variables as an optical flow could boost the prediction score significantly and should be considered in further competitions.

6. Conclusion

In this paper we display the findings obtained during our participation in the Weather4cast 2021 Competition. Our experiments show that the SmaAt-UNet model is a better alternative than the classical U-Net, as it improves the quality of the prediction and requires less resources to train than the original architecture. We achieved the best results by generating an ensemble prediction of several training checkpoints. We also discuss various improvements in the Future Work Section (see Section 5). These ideas will be further developed for future competitions.

References

- [1] P. D. Polger, B. S. Goldsmith, R. C. Przywarty, J. R. Bocchieri, National weather service warning performance based on the wsr-88d, *Bulletin of the American Meteorological Society* 75 (1994) 203–214.
- [2] S. S. Soman, H. Zareipour, O. Malik, P. Mandal, A review of wind power and wind speed forecasting methods with different time horizons, in: *North American Power Symposium 2010, IEEE, 2010*, pp. 1–8.
- [3] Q.-K. Tran, S.-k. Song, Multi-channel weather radar echo extrapolation with convolutional recurrent neural networks, *Remote Sensing* 11 (2019) 2303.
- [4] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, W.-c. Woo, Convolutional lstm network: A machine learning approach for precipitation nowcasting, in: *Advances in neural information processing systems, 2015*, pp. 802–810.
- [5] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: *International Conference on Medical image computing and computer-assisted intervention, Springer, 2015*, pp. 234–241.
- [6] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, *Advances in neural information processing systems* 28 (2015) 91–99.
- [7] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems* 25 (2012) 1097–1105.
- [8] K. Trebing, T. Staczyk, S. Mehrkanoon, Smaat-unet: Precipitation nowcasting using a small attention-unet architecture, *Pattern Recognition Letters* (2021). URL: <https://www.sciencedirect.com/science/article/pii/S0167865521000556>.

- doi:<https://doi.org/10.1016/j.patrec.2021.01.036>.
- [9] Weather4cast 2021 Challenge, <https://www.iarai.ac.at/weather4cast/>, 2021. [Online; accessed 29-July-2021].
 - [10] Institute of Advanced Research in Artificial Intelligence (IARAI), <https://www.iarai.ac.at>, 2021. [Online; accessed 29-July-2021].
 - [11] AEMET, <http://www.aemet.es/>, 2021. [Online; accessed 29-July-2021].
 - [12] NWC SAF, <https://www.nwcsaf.org/>, 2021. [Online; accessed 29-July-2021].
 - [13] D. P. Kreil, M. K. Kopp, D. Jonietz, M. Neun, A. Gruca, P. Herruzo, H. Martin, A. Soleymani, S. Hochreiter, The surprising efficiency of framing geo-spatial time series forecasting as a video prediction task – insights from the iarai 4c competition at neurips 2019, in: H. J. Escalante, R. Hadsell (Eds.), *Proceedings of the NeurIPS 2019 Competition and Demonstration Track*, volume 123 of *Proceedings of Machine Learning Research*, PMLR, 2020, pp. 232–241. URL: <http://proceedings.mlr.press/v123/kreil20a.html>.
 - [14] Traffic4cast 2020 Challenge, <https://www.iarai.ac.at/traffic4cast/2020-competition/challenge/#challenge>, 2021. [Online; accessed 29-July-2021].
 - [15] Q. Qi, P. H. Kwok, Traffic4cast 2020–graph ensemble net and the importance of feature and loss function design for traffic prediction, arXiv preprint arXiv:2012.02115 (2020).
 - [16] S. Choi, Utilizing unet for the future traffic map prediction task traffic4cast challenge 2020, arXiv preprint arXiv:2012.00125 (2020).
 - [17] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, J. Liang, Unet++: Redesigning skip connections to exploit multiscale features in image segmentation, *IEEE Transactions on Medical Imaging* (2019).
 - [18] S. Woo, J. Park, J.-Y. Lee, I. S. Kweon, Cbam: Convolutional block attention module, in: *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.
 - [19] M. Tan, Q. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, in: *International Conference on Machine Learning*, PMLR, 2019, pp. 6105–6114.
 - [20] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, CoRR abs/1412.6980 (2015).
 - [21] I. Loshchilov, F. Hutter, Sgdr: Stochastic gradient descent with warm restarts, arXiv: Learning (2017).
 - [22] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimeshain, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, Pytorch: An imperative style, high-performance deep learning library, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, R. Garnett (Eds.), *Advances in Neural Information Processing Systems 32*, Curran Associates, Inc., 2019, pp. 8024–8035. URL: <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>.
 - [23] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural computation* 9 (1997) 1735–80. doi:10.1162/neco.1997.9.8.1735.
 - [24] X. SHI, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, W.-c. WOO, Convolutional lstm network: A machine learning approach for precipitation nowcasting, in: C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, volume 28, Curran Associates, Inc., 2015. URL: <https://proceedings.neurips.cc/paper/2015/file/07563a3fe3bbe7e3ba84431ad9d055af-Paper.pdf>.
 - [25] M. Mathieu, C. Couprie, Y. LeCun, Deep multi-scale video prediction beyond mean square error, CoRR abs/1511.05440 (2016).
 - [26] I. Kobyzev, S. Prince, M. A. Brubaker, Normalizing flows: Introduction and ideas, ArXiv abs/1908.09257 (2019).
 - [27] A. N. Gomez, M. Ren, R. Urtasun, R. B. Grosse, The reversible residual network: Backpropagation without storing activations, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, volume 30, Curran Associates, Inc., 2017. URL: <https://proceedings.neurips.cc/paper/2017/file/f9be311e65d81a9ad8150a60844bb94c-Paper.pdf>.
 - [28] J. Donahue, K. Simonyan, Large scale adversarial representation learning, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, volume 32, Curran Associates, Inc., 2019. URL: <https://proceedings.neurips.cc/paper/2019/file/18cdf49ea54eec029238fcc95f76ce41-Paper.pdf>.
 - [29] C. Etmann, R. Ke, C.-B. Schönlieb, iunets: Learnable invertible up- and downsampling for large-scale inverse problems, in: *2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP)*, 2020, pp. 1–6. doi:10.1109/MLSP49062.2020.9231874.
 - [30] J. Behrmann, P. Vicol, K.-C. Wang, R. B. Grosse, J. Jacobsen, Understanding and mitigating exploding inverses in invertible neural networks, in: *AISTATS*, 2021.
 - [31] H. Zhu, H. Wei, B. Li, X. Yuan, N. Kehtarnavaz, A review of video object detection: Datasets, metrics and methods, *Applied Sciences* 10 (2020).

URL: <https://www.mdpi.com/2076-3417/10/21/7834>.
doi:10.3390/app10217834.

- [32] I. Rodríguez-Moreno, J. M. Martínez-Otzeta, B. Sierra, I. Rodríguez Rodríguez, E. Jauregi Iztueta, Video activity recognition: State-of-the-art, *Sensors* 19 (2019) 3160. doi:10.3390/s19143160.
- [33] State of the art on Video Action Recognition, <https://paperswithcode.com/task/action-recognition-in-videos>, 2021. [Online; accessed 29-July-2021].