MAchinE Learning for Scalable meTeoROlogy and climate

MAELSTROM

Report on a survey of MAELSTROM applications and ML tools and architectures

www.maelstrom-eurohpc.eu

D1.2 Report on a survey of MAELSTROM applications and ML tools and architectures

Author(s): Bing Gong (FZJ), Michael Langguth (FZJ), Peter Dueben (ECMWF), Matthew Chantry (ECMWF), Thomas Nipen (MetNor), Markus Abel (4cast), Tal Ben-Nun (ETH)

Dissemination Level:	Public
Date:	30/09/2021
Version:	1.0
Contractual Delivery Date:	30/09/2021
Work Package/ Task:	WP1/ T1.2
Document Owner:	Organisation
Contributors:	FZJ, ECMWF, MetNor, 4cast, ETH
Status:	Final

MAELSTROM

Machine Learning for Scalable Meteorology and Climate

Research and Innovation Action (RIA)
H2020-JTI-EuroHPC-2019-1: Towards Extreme Scale Technologies and Applications

Project Coordinator: Dr Peter Dueben (ECMWF)
Project Start Date: 01/04/2021
Project Duration: 36 months

Published by the MAELSTROM Consortium

Contact:
ECMWF, Shinfield Park, Reading, RG2 9AX, United Kingdom
Peter.Dueben@ecmwf.int

The MAELSTROM project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 955513. The JU receives support from the European Union's Horizon 2020 research and innovation programme and United Kingdom, Germany, Italy, Luxembourg, Switzerland, Norway

Contents

Figures

Tables

1 Executive Summary

This document provides the results of a survey of the machine learning (ML) tools and architectures of the six MAELSTROM application. The traditional ML approaches and state-of-the-art deep learning architectures are reviewed. The loss function and evaluation criteria that will be used in the MAELSTROM project were also surveyed for this report. Furthermore, the literature that is relevant for the six MAELSTROM applications was reviewed thoroughly and a summary is provided. The sizes of datasets and ML models are estimated and the ML software and parallelization strategy used for the six applications are summarised in the end of this report.

2 Introduction

2.1 About MAELSTROM

To develop Europe's computer architecture of the future, MAELSTROM will co-design bespoke compute system designs for optimal application performance and energy efficiency, a software framework to optimise usability and training efficiency for machine learning at scale, and large-scale machine learning applications for the domain of weather and climate science.

The MAELSTROM compute system designs will benchmark the applications across a range of computing systems regarding energy consumption, time-to-solution, numerical precision and solution accuracy. Customised compute systems will be designed that are optimised for application needs to strengthen Europe's high-performance computing portfolio and to pull recent hardware developments, driven by general machine learning applications, toward needs of weather and climate applications.

The MAELSTROM software framework will enable scientists to apply and compare machine learning tools and libraries efficiently across a wide range of computer systems. A user interface will link application developers with computer system designers, and automated benchmarking and error detection of machine learning solutions will be performed during the development phase. Tools will be published as open source.

The MAELSTROM machine learning applications will cover all important components of the workflow of weather and climate predictions including the processing of observations, the assimilation of observations to generate initial and reference conditions, model simulations, as well as post-processing of model data and the development of forecast products. For each application, benchmark datasets with up to 10 terabytes of data will be published online for training and machine learning tool-developments at the scale of the fastest supercomputers in the world. MAELSTROM machine learning solutions will serve as a blueprint for a wide range of machine learning applications on supercomputers in the future.

2.2 Scope of this deliverable

2.2.1 Objectives of this deliverable

To analyse the design of several ML options and loss functions. To identify the requirements for ML architectures, loss functions, ML software and tools, expected data sizes, and scalability limits for the six MAELSTROM ML applications.

Deliverable 1.2 is one of four MAELSTROM deliverables that survey the state-of-the-art in terms of methods, tools and developments in machine learning at the beginning of the project and aim to build additional links between the three work packages that are involved in the MAELSTROM co-design cycle. Deliverable 1.2 is a survey of machine learning methods and tools that are currently used for weather and climate applications. Deliverable 2.1 is a survey of existing machine learning workflow tools and a summary of the MAELSTROM protocol and machine learning requirements. Deliverables

3.1 and 3.2 provide a systematic analysis of the hardware requirements for the MAELSTROM applications and a roadmap analysis of hardware that will be relevant for machine learning in MAELSTROM.

2.2.2 Work performed in this deliverable

The potential ML architectures and loss functions are identified and analysed. The state-of-the-art ML solutions in the Weather and Climate (W&C) domain are thoroughly reviewed to demonstrate how the ML solutions have been adopted in the W&C applications. In addition, the ML architectures, software and tools that are going to be used in the subsequent studies for the six applications are surveyed in this report. The expected data sizes and the scalability limits of the six applications are summarized.

2.2.3 Deviations and counter measures

There are no significant deviations from the planned contributions of the deliverable.

3 Introduction to the background of the six applications

Modern weather prediction relies on numerical weather prediction (NWP), whose operational workflow is illustrated in Figure 1. Continuously collected observations constitute the starting point in this workflow. These observations are pre-processed and assimilated to generate a gridded initial state of the Earth system. Given the estimated initial conditions, a numerical model is used to simulate the future evolution of the atmospheric state which then provides a forecast for each grid point in the NWP model. However, the predictions contain systematic errors and are not necessarily representative for a specific location due to the truncated spatial-temporal resolution of the numerical model. To correct for these biases, but also to extract relevant information for a bespoke forecast product, the model output undergoes a final post-processing step.

In light of the successful application of machine learning (ML)/ Deep learning (DL) in domains such as healthcare (Esteva et al. 2019), autonomous driving (Hu et al. 2020) and fraud detection (Awoyemi et al., 2017), the W&C community is increasingly aware of ML techniques and starts to explore their application across the NWP workflow to further improve weather predictions. For observations, these include weather data monitoring, real-time quality control for observational data, anomaly interpretation, guided quality assignment and decision making, data fusion from different observation sources as well as the correction of systematic errors. For data assimilation, ML techniques can replace the traditional statistical approach and be deployed to learn and correct model and observational errors, to build faster and more accurate observation operations and linearised models for variational data assimilation. For numerical weather forecasts, these include to emulate model components, develop improved parameterization schemes to represent sub-grid-scale features, build better error models, learn the underlying equations of motion, or develop low-complexity models. ML is also used to postprocess model output, including feature detection (e.g., tropical cyclones or atmospheric rivers), uncertainty quantification, or error corrections, for example for seasonal predictions. For product generation, these include down-scaling to improve local predictions, real-time adjustments of forecast products, bespoke products for business usage and many more. If fully implemented, ML offers both more accurate forecasts and much advanced efficiency for computing and data handling, especially for very large computational problems at peta or exascale.

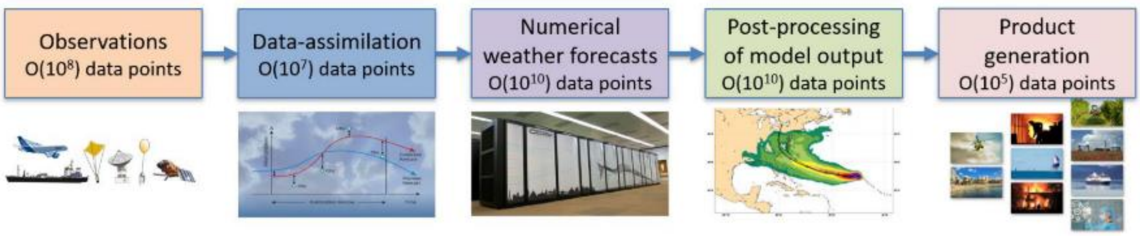

Figure 1: The workflow of current numerical weather prediction

MAELSTROM will cover the entire W&C workflow with six selected ML applications A1-A6 to explore the capability of ML in the context of NWP, in detail:

A1: Blend citizen observations and numerical weather forecasts: Public weather forecast providers strive to deliver forecasts that are accurate and tailored to the specific locations of end users. To produce high resolution forecasts, machine learning models need high resolution target fields for training. The target field in the A1 dataset is constructed from measurements of Netatmo's network of citizen weather stations. The network density is roughly two orders of magnitude higher than the density of the network of conventional stations operated by MET Norway. The target of A1 is to provide post-processed NWP forecasts on a 1 km resolution grid of size 2321x1796 and 60 lead hours. The objective of AP1 is to train neural network models on such grids with TBs of training data. The machine learning application aims to improve MET Norway's operational short-range forecasts of temperature and precipitation for the Nordic countries.

A2: Incorporate social media data into the prediction framework: The application A2 is studying the analysis of social media data (Twitter) to extract weather information and to make prediction. Twitter messages (Tweets) may contain messages about weather events like "rain", or "sun", however these words are context-dependent and must be distinguished from non-weather-related messages. Tweets most often contain location information, depending on user consent, like geolocation or browser information. The Twitter API allows full access for academic research. Application A2 aims at incorporating the Twitter information to enhance weather prediction. The steps to process Tweets include a first evaluation of the probability that the information is credible, a second evaluation if the content is for classification or contains numerical data, a third evaluation to provide the information as a feature stream to weather prediction schemes, and a fourth evaluation to use the data in a learning task together with NWP data.

A3: Build neural network emulators to speed-up weather forecast models and data assimilation: Due to limited resolution, not all important processes can be represented explicitly within weather and climate models. Processes that cannot be resolved explicitly are represented by so-called parameterization schemes that mimic sub-grid-scale processes based on the physical fields that are resolved. Application A3 will explore the emulation of physical parameterization schemes using neural networks. We will build datasets from the IFS operational weather forecasting model to enable researchers to build emulators for components of the physical parametrisation schemes. These emulators will map the inputs of a conventional parametrisation scheme to the outputs of the schemes, while taking less time than the conventional scheme. Crucially, final testing will need to couple emulators to the IFS model to test stability and accuracy. These emulators would be useful in both the forecasting and data assimilation tasks.

A4: Improve ensemble predictions in forecast post-processing: To be useful, weather forecasts are not only required to predict the most likely future scenario, but also need to produce the probabilities for certain predictions of weather events. Methods such as Ensemble Model Output Statistics (EMOS) (Gneiting et al. 2005) and Bayesian Model Averaging (BMA) (Raftery et al., 2005) currently allow for improvements of the raw ensemble forecast skill. Hamill and Whitaker (2007) show initial explorations of those techniques on re-forecast datasets, also used in our work, for temperature at 850 hPa (T850) and geopotential at 500 hPa (Z500). Advances in neural networks have only recently reached the field of ensemble models in weather forecasting, focusing on its application to specific weather stations. A4 will expand on this work by applying deep neural networks to improve the forecast skill for global ensemble predictions as a post-processing step. If the quality for ensemble predictions can be improved, less ensemble members can be run in ensemble predictions, and computational cost can be reduced.

A5: Improve local weather predictions in forecast post-processing: Contemporary NWP models have reached a remarkable quality in forecast accuracy (see e.g. Bauer et al., 2015). However, there are enormous computational costs of running these models on a high spatial resolution and shortcomings in the numerical model itself, e.g. due to the applicability of physical parameterizations. Thus, statistical downscaling is an appealing approach to circumvent the burden of operating costly numerical models at ever higher spatial resolution. Inspired by the success of deep neural networks for generating super-resolution images in computer vision (e.g. Mahapatra et al. 2019, Wang et al. 2021), first studies have started to transfer these techniques to the meteorological domain (e.g. Sha et al., 2020, Leinonen et al., 2020). AP5 will follow up these first approaches aiming to establish a robust downscaling neural network that allows handling different meteorological variables over complex terrain.

A6: Provide bespoke weather forecasts to support energy production in Europe: One of the most important challenges to mitigate climate change is to increase the generation of renewable energies. Accurate forecasts for renewable energy generation rely heavily on weather predictions, local measurements, and real-time production data from wind turbines and solar panels. A4 aims to significantly improve predictions of power production from renewable energy sources in order to optimise the production of renewable energy. Possible users of the improved predictions are power producers, trading companies, grid operators and even whole countries. Machine learning will be used to fuse the information of local conditions (measurements of local energy production and weather) and numerical weather predictions to learn to predict the energy production at local sites in the future.

4 Survey of Machine learning solutions

4.1 Review of ML architectures

ML is a highly interdisciplinary field that builds upon various fields such as artificial intelligence, information theory, optimization theory, statistics, and many other disciplines of sciences (Qiu et al., 2016). There is a long history of using statistical methods in W&C applications. Generalized linear models show potential capability for simulating realistic sequences of daily rainfall (Yang et al. 2005) and rainfall occurrence forecast (Little et al. 2009).

Even though the traditional statistics and machine learning can be used for prediction and inference, there are still differences between them. In principle, the statistical approach focuses on the inference through creating and fitting a probability model, the assumption is verified for the given specified model. In contrast, ML is widely used for prediction through discovering the underlying patterns of the data as noted in Bzdok et al. 2018.

ML has been employed in many W&C applications. Multilayer Perceptron (ANN) were explored for downscaling of 2m temperature and precipitation (Schoof and Pryor, 2001) and for predicting global radiation by the study of Voyant et al. (2012). Support Vector Machines with various types of kernel functions were employed for solar intensity forecasting (Sharma et al., 2011). Tree-based models, decision trees and random forests, were also applied successfully to predict mesoscale convective areas (Gagne et al., 2009) and the probability of severe hail (Ahijevych et al., 2016).

Unsupervised learning such as cluster approaches are widely adopted to reduce the input dimensions, simplify the data analysis and for better visualization in meteorology. Kumpf et al. (2018) introduce the cluster approach Principle Component Analysis (PCA) to reduce input dimensionality and apply K-mean clusters to investigate the ensemble behaviour and determination of physical processes that caused the uncertainty for toopical cyclone Karl in 2016.

More recently, DL has proven success in the Computer Vision domain on how to learn the spatial-temporal patterns from data for better prediction and has started to be employed in W&C applications. DL can be considered a subset of ML. In contrast to traditional ML techniques that typically adopt architecture with “shallow“ structure, DL consists of deeper architectures that include more hidden layers to automatically learn the hierarchical representation from the data, and therefore, increase the prediction skills. Convolutional neural networks (CNN) are considered to be the most popular DL for W&C applications. Gagne et al. (2019) have applied CNN to NWP data to identify storms most likely to develop severe hail, and the corresponding features that were learned via convolution layers were contributing most to the hail prediction. (Lagerquist et al., 2019) used CNN to predict the movement of weather fronts. Shi et al. (2015) extended the CNN with Long-short term memory (LSTM) cells to predict radar images for precipitation nowcasting. GANs with dropout techniques for ensemble weather forecasts were tried in the study by Bihlo (2020).

In the following part, we will briefly review and introduce the traditional machine learning techniques and then highlight some advanced deep learning techniques, which will potentially be used in the six MAELSTROM’s applications.

4.1.1 Traditional machine learning techniques

The traditional ML methods are mathematical models with specified structure that learn to perform the task from the data. Traditional ML includes algorithms such as feedforward artificial neural network, support vector machine, decision trees etc. These methods can potentially learn complex and underlying patterns and relationships from the data, and are easy to interpret.

Feedforward Artificial Neural Network (ANNs; Figure 2). The first neural model was introduced by McCulloch and Pitts (McCulloch and Pitts, 1943). The model was inspired by animals' central nervous systems (in particular the brain). An increasing number of different models considered as ANNs are continuously developed. Multiplayer layer Perception (MLP) is one of the most popular feedforward artificial neural networks. The weight of the neurons that are interconnected across different layers of ANNs are adjusted at each iteration by minimizing the cost function, which is computed from the loss functions computed by predicted and real outputs. The back-propagation algorithm is used to update individual weights during the training period and is building the foundation of supervised learning networks.

Support Vector Machine (SVM; Figure 3): SVM has been developed in the framework of statistical learning theory by Vapnik (1999). Afterwards, they became one of the most robust classification algorithms in various fields of applications. Due to its strong generalization capability with optimal solution and discriminative power, the SVM is one of the most popular machine learning tools in recent years (Cervantes et al., 2020). The idea of SVM is to find an optimal separating hyperplane (defined by equation $wx + b = 0$ in Figure 3) with maximum margin $||\frac{1}{w}||$ in a high-dimensional space between classes in the training dataset. The simplest SVM is to separate data linearly in the feature space. However, for the cases that cannot be separated linearly, non-linear kernel functions, such as radial basis function, polynomial and sigmoidal functions are employed in SVM. These kernels can project the input to a higher feature space to make it linear separable to enhance the generalization ability of the SVM (Cervantes et al., 2020).

Decision tree method: Decision trees use information gain to determine the structure of the tree. As one of the most important decision tree algorithms, the **Classification and Regression Tree (CART)** can be considered as a binary recursive partitioning (Breiman et al., 1984). The term 'binary' implies that the data (represented by the parent node) are split into subsets (child nodes) based on evaluation functions (e.g., Gini diversity index, the entropy, or the error index). The binary partition will be processed repeatedly on child nodes to yield additional children until only a few samples remain in the terminal subset. Normally, each node corresponds to a variable. In such a case, the importance score for a variable, when it is used as a splitter, can be determined by summing the improvement scores with respect to the used evaluation function of all nodes. In this regard, tree-based algorithms are always employed for variable importance rank and variable selection. **Random Forests,** which was first introduced by Breiman (2001)., are based on the CART (Figure 4), It is an ensemble algorithm that applies bagging methods on CART. Different from CART, RF selects only a few features randomly during training instead of using all the inputs, which allows to reduce the runtime.

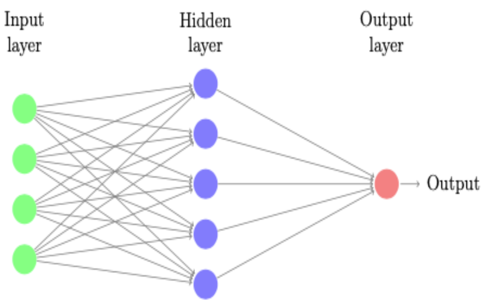

Figure 2: MLP architecture consists of three layers: inputs, hidden, and output layer, as well as the neurons and their connections.

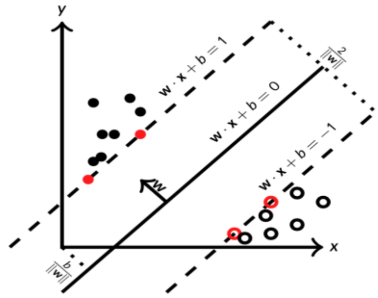

Figure 3: SVM architecture to find the optical hyperplane to maximize the margin between classes.

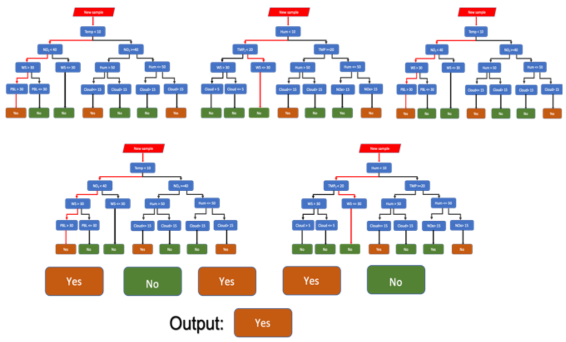

Figure 4: The RF architecture is composed of several individual CART

4.1.2 Advanced deep learning techniques

To gain better performance, traditional ML requires large data sets and extensive hand-crafted feature engineering processes. While for DL, the features of the data can be learned through DL and require less feature engineering process. Although various DL architectures have been and are continuously invented nowadays, this report will focus on architectures which are used in the MAELSTROM project, i.e. convolutional neural network (CNN), recurrent neural network (RNN), Generative Adversarial Nets (GAN), Dense nets, Inception Nets, Transformer, and Symbolic regression.

CNN: Convolution layers (LeCun et al. 1989) in a neural network connect a neighbourhood of pixels to a shared neuron. The weights governing this connection are reused for each neighbourhood of pixels. CNN layers are very popular, particularly in image-based tasks, due to their small number of trainable parameters when compared with fully-connected layers. A convolutional neural network is constructed by a series of convolutional layers, often alongside pooling layers where the task requires aggregation of information, e.g. classification of an image.

U-net: The U-net structure was introduced for biomedical image segmentation (Ronneberger et al., 2015) which contains a contracting path and an expansive path as well as several copy and crop paths (shown in Figure 5). The contracting path can be seen as an encoder which down-scales the high-resolution images into coarse ones using convolutional and pooling layers. Oppositely, the expansive path, acting as a decoder, applies ‘up-convolution’ layers for up-scaling to generate high-resolution prediction. To prevent losing resolution during up-scaling, the copy and crop paths are built to concatenate the same-level features between encoding and decoding. The concatenation is performed through skip-connections approach, which is used to skip some layers of the neural network and feed the output of one layer as input for other layers. The U-net architecture allows the network to propagate spatial information after down-scaling back to higher resolution layers and shows great success in image segmentation tasks. The switch between different resolutions within the network can potentially be used to represent scale-interactions that are very important for the dynamics of both atmosphere and ocean.

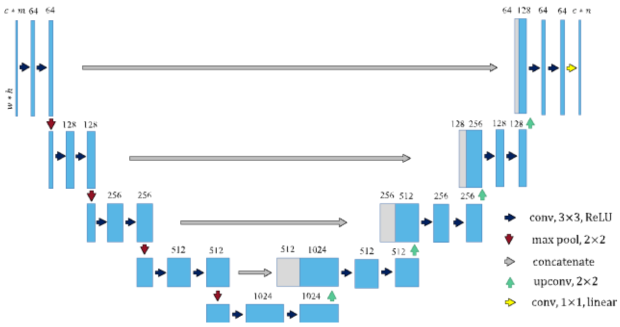

Figure 5 U-Net architecture consists of encoder and decoder layers, which can be connected through skip-connections

Inception modules: Szegedy et al. (2015) introduced the GoogLeNet deep neural network, which consists of a series of Inception modules (Figure 6). Instead of direct convolutions, each inception module contains several convolutions of different sizes applied on the same features, and concatenated together at the end of the module. In preliminary research of application A4 on weather and climate data, we observe benefits for Inception module variants in convolutional neural networks.

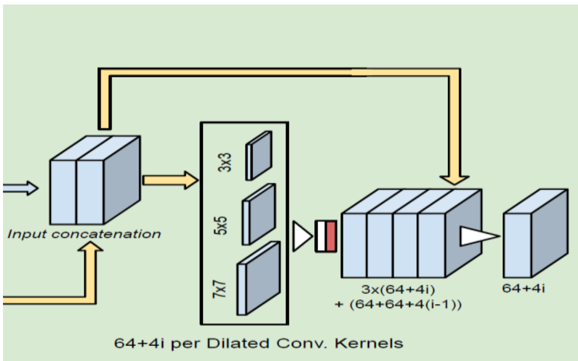

Figure 6 Inception-style module

Recurrent neural network (RNN): This architecture represents a directed acyclic of a cyclic graph (for feedbacking systems) where, most often, time is used to define an input vector. In addition to the temporal ordering RNNs may have an internal state to process variable length sequences of inputs. The most prominent realizations of RNNs are Long-Short-Term-Memory (LSTM) and Gated Recurrent units (GRU), cf. Figure 7. The unit is “enrolled” as a tensor for timeslices *x(t),... x(t-N)* which serves as input to the neural network. RNNs have a huge impact in natural language processing applications and appear very attractive for weather applications, if adequate feature engineering has determined parameters, as e.g. the length of the timeslice.

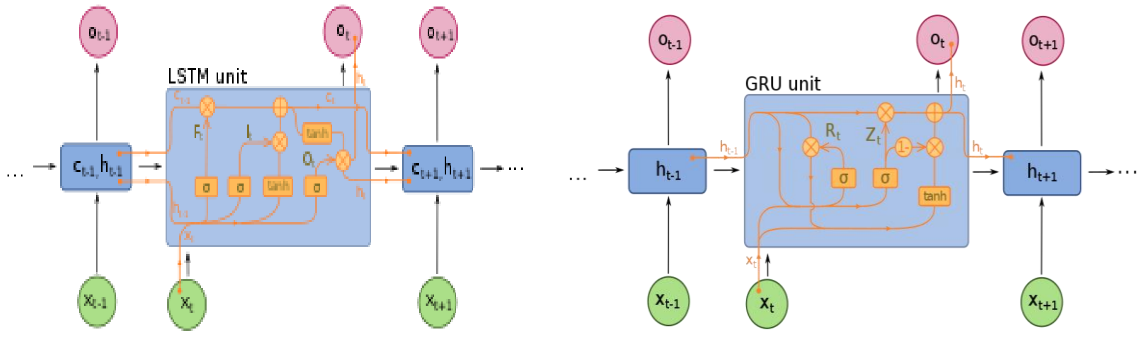

Figure 7 LSTM (left) and GRU (right) architecture (from odelling.org: https://en.wikipedia.org/wiki/File:Gated_Recurrent_Unit.svg)

<u>Convolutional LSTM (ConvLSTM):</u> ConvLSTM was introduced in the study of Shi et al. (2015; Figure 8). It is built upon the architectures of CNN and LSTM. Their model architecture extends the fully connected LSTM encoder-decoder-predictor model introduced in Srivastava et al. (2015) to incorporate spatial patterns with the help of convolutional layers in both the input-to-state and the state-to-state transition. With that, the ConvLSTM network can capture spatial-temporal correlations in the data which is known to be crucial for atmospheric processes. ConvLSTM so far has been applied in the problem of images-to-images, also in the W&C prediction studies as baseline models to compare with other advanced architectures (Kim et al. 2017; Weyn et al. 2019) .

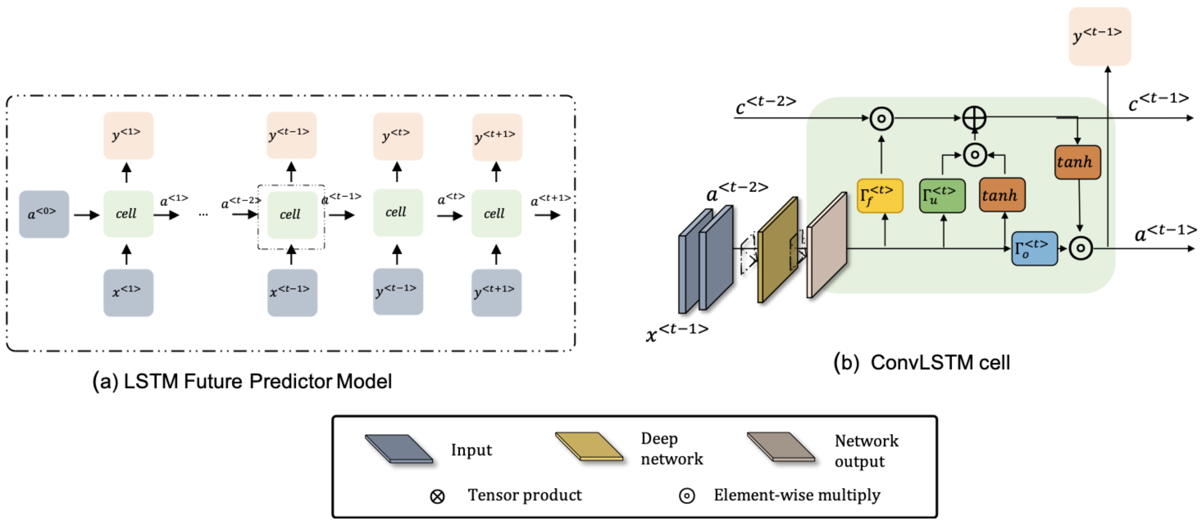

Figure 8: Convoluational LSTM architecture: convolutional layers couple with Long-short term memory architecture

<u>Generative Adversarial Nets:</u> GANs were first proposed by Goodfellow et al. (2014; Figure 19). The GAN architecture constitutes a composite model architecture which consists of a generator and a discriminator model. The discriminator is trained to distinguish between real and artificially generated video sequences. Conversely, the generator gets optimized to fool the discriminator, i.e. it aims to produce video sequences that cannot be differentiated from real ones by the discriminator. By training both models adversarially, the generator must learn the statistical properties of the underlying data and thereby become capable of generating images that look natural.

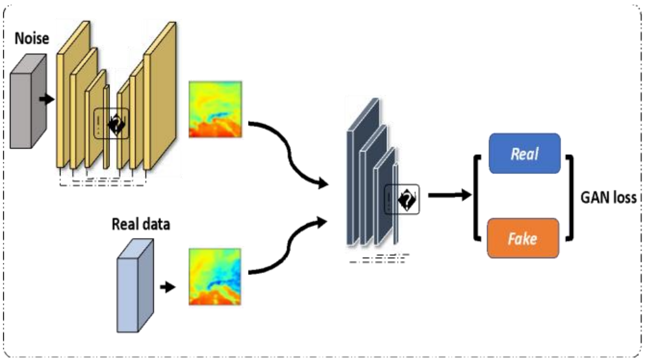

Figure 9: Generative adversarial nets architecture

DenseNet: Huang et al. (2017) report that convolutional networks can become deeper and more accurate if they exhibit shorter paths from the input image to the output. In their paper, Huang et al. propose DenseNets (Figure 9) - a convolutional network in which convolutional blocks are densely connected to each other. That is, for each block, the features of all the preceding blocks are connected directly as inputs. Such densely-connected blocks could be used, in conjunction with, e.g., Inception-like modules as blocks, in order to improve accuracy and reduce the memory footprint of parameters in neural networks for weather and climate.

Transformers: Transformers (Vaswani et al., 2017) are sequence transducer networks, which are based on the mechanism of attention. Instead of their recurrent neural network predecessors, these networks observe the entire sequence at once (subject to masks that induce temporal causality) and can learn, for each element in the sequence, how to "attend" to any of the other elements in the sequence via learnable weights. Transformers are built out of a series of encoder blocks followed by decoder blocks (either of which could be omitted for different purposes), each of which containing a multi-head attention layer, capable of attending to multiple criteria at the same time; and a fully-connected network. These blocks can be adapted to work with other types of data rather than sequences of textual elements, e.g., image patches. Transformers and their variants are widely used today for translation, image recognition, and text generation tasks, and have shown promise on weather models.

Symbolic regression: Symbolic regression (Quade et al., 2016) searches the space of mathematical expressions applied to the target and input variables such that an optimal expression is found fulfilling $y = F(\boldsymbol{x})$, where y is a target variable, possibly of many dimensions, and x is the vector of input variables. The function F may be complex, and the input variables may be subject or prior transformation, e.g. to compute derivatives. A simple subclass of general symbolic regression is the conventional linear regression or generalized linear regression schemes. In machine learning, genetic programming (Koza 1992) is used to obtain very general representations of the equations, whereby the search in function space may involve sophisticated tuning of the search parameters.

4.2 Loss functions

The network is optimized by the loss function that typically measures the difference between predictions and the truth. The choice of the loss function depends on the problem of the applications and the statistical properties of the data. For instance, distance-based loss functions such as Mean Square Error (MSE) and **Mean Absolute Error (**MAE) are widely used for regression problems. Loss functions such as the Continuous Rank Probability Score (CRPS) are adopted for ensemble predictions. GAN architectures try to learn the probability distribution of the data. The adversarial loss of GAN reflects the distance difference between the distribution of real data and the distribution of generated data. In the following, we introduce the loss and evaluation functions in the six applications of the project.

MAE and MSE: The MAE and MSE is computed as follows:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{N}\left|\hat{Y}_i - Y_i\right| \tag{1}$$

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{N}(\hat{Y}_i - Y_i)^2 \tag{2}$$

Where $\hat{Y}_i$ is the prediction and Y_i are the target values, n is the number of total samples.

Latitude-weighted MSE: This metric adds a reweighting scheme to the MSE loss. The values are given as the mean squared error with a multiplicative weighting function:

$$MSE_{lw} = \frac{1}{n}\sum_{i=1}^{N}(\hat{Y}_i - Y_i)^2\, L(i), \tag{3}$$

where the function L weights values at latitudes closer to the equator more heavily than values further away from the equator. For a vector *lat* that contains the latitudes at each point i (in radians), we can define $L(i) = \frac{\cos lat_i}{\sum_{j=1}^{N}\cos lat_j}$.

Quantile Score: Quantile score is used for evaluating the probabilistic forecasts (Gneiting and Raftery, 2007; Koenker and Bassett Jr, 1978), which is given by:

$$S_\tau(u) = \begin{cases} u(\tau - 1), & u < 0 \\ u\,\tau, & u \geq 0 \end{cases} \tag{4}$$

where τ is quantile level and *u* is the difference between the quantile forecast and the target. The quantile scores for the three predicted quantiles (10%, 50%, and 90%) are added together to a final score.

SSIM: Structural Similarity Index (SSIM), which is commonly applied to access the perceptual similarity between images, measures the similarity between two images based on three features (Wang et al., 2004): Luminance $[l(\hat{x},x)]^{\alpha}$., contrast $[c(x,x)]^{\beta}$ and structure $[s(\hat{x},x)]^{\gamma}$ in Eq. 5. luminance is measured by averaging over all the pixel values for a given image x; contrast taking the standard deviation given by Eq. (9) over all pixel values and is defined by Eq. (8); structure features compare the predicted and true images given via Eq . (10).

$$SSIM\,(\hat{y},y) = [l(\hat{x},x)]^{\alpha}.\,[c(\hat{x},x)]^{\beta}.\,[s(\hat{x},x)]^{\gamma} \tag{5}$$

$$\mu_y = \frac{1}{N}\sum_{i=1}^{N} x \quad (6)$$

$$l(\hat{x}, x) = \frac{2\mu_x\mu_{\hat{x}+} C_1}{\mu_x^2 + \mu_{\hat{x}}^2 + C_1} \quad (7)$$

$$c(\hat{x}, x) = \frac{2\sigma_x\sigma_{\hat{x}+} C_2}{\sigma_x^2 + \sigma_{\hat{x}}^2 + C_2} \quad (8)$$

$$\sigma_x = \left[\frac{1}{N-1}\sum_{i=1}^{N}((x_i - u_x)^2)\right]^{\frac{1}{2}} \quad (9)$$

$$s(\hat{x}, x) = \frac{\sigma_{x\hat{x}} + C_3}{\sigma_x\sigma_{\hat{x}+} C_3} \quad (10)$$

$$\sigma_{x\hat{x}} = \frac{1}{N-1}\sum_{i=1}^{N}((x_i - u_x)\ (\hat{x} - \mu_{\hat{x}}\ ) \quad (11)$$

Where Y_i are the target values and $\hat{Y}_i$ is the prediction, and $\alpha > 0, \beta > 0, \gamma > 0$ are parameters to control the relative importance of features.

CRPS: The Ranked Probability Score (CRPS) is an integral of the square of the difference between the Cumulative Distribution Function of the probabilistic predictions F and the ground truth y.

$$\text{CRPS}(F, y) = \int_{-\infty}^{\infty} \left[F(x) - 1_{x>y}\right]^2 dx \quad (12)$$

Adversarial loss: the adversarial loss function in the vanilla GAN architecture consists of two loss functions: generator loss and discriminator loss (Goodfellow et al. 2014). GAN uses a minimax optimization strategy: to minimize the following equation for generator and maximize for discriminator.

$$\min_G \max_D (D, G) = E_{x\sim p_{data}(x)}\left[\log\left(D(x)\right)\right] + E_{z\sim p_z(z)}\left[\log\left(1 - D(G_Z)\right)\right] \quad (13)$$

Where $D(x)$ is the estimated probability by discriminator that the sample from real data x is real; $E_{x\sim p_{data}(x)}$ is the expected values of all the real data; $G(z)$ is the generated (fake) data given noise z; $D(G(z))$ is the estimated probability by the discriminator that the generated (fake) sample $G(z)$ is real.

Earth Mover distance (Wasserstein GAN): Instead of predicting a probability to distinguish real data from the data created by the generator, the Wasserstein distance can be used to optimize a GAN model. Mathematically, it constitutes the greatest lower bound to convert the generated data distribution P_g to the real data distribution P_r :

$$W(P_r, P_g) = E_{(x,y)}\ [|x - y|] \quad (14)$$

To make the objective tractable, the Kantorovich-Rubinstein duality is exploited

$$W(P_r, P_g) = E_{(x\sim p_r)}[f(x)] - E_{(y\sim p_g)}[f(y) \quad (15)$$

where f(x) constitutes the learned approximation of a 1-Lipschitz function with the help of a neural network. To fulfil Lipschitz continuity, different techniques are applied. While (Arjovsky et al., 2017) suggest clipping the weight resulting from the loss function, (Gulrajani et al., 2017) apply an additional gradient penalty that enforces a norm close to 1. Irrespective of the latter technique, the Wasserstein Loss carries out a conceptual switch from a discriminator network (classifier for real or generated data) to a critic (score the supplied data).

4.3 ML in Weather and Climate applications

Machine Learning for observations and product generation: Observations can have different character: string-valued tweets are rather for classification, whereas real-valued sensor measurements are suited for regression. Counting tweets may be translated to numeric values, sensor measurement may be transformed into so-called features or observations, which in turn may serve to improve weather-based products. Such products may be used to make predictions for the finance, insurance, mobile, or energy sectors. Typically, these products transform NWP together with observations to location-based, time-dependent information, e.g. a forecast of energy production for very short-term (intraday), short term (day-ahead), mid-term (weeks ahead) scales. Typical methods involve gradient boosting, RNN and CNN (Giebel and Kariniotakis 2017). The development of further products involving weather time-scales is a vibrant area of research and subject of future exploitation work.

Machine learning for data assimilation: Data assimilation is the process of finding the best model state that matches recent observations. It is a costly task with many possible areas of interest for the application of machine learning. 4D-var is a popular algorithm for data assimilation, with many similarities to machine learning (Geer 2021). Within the 4D-var framework researchers used a neural network to learn the error correction term within weak-constrained 4D-var or used neural networks to learn the model error that was diagnosed within data assimilation offline (Bonavita & Laloyaux 2020). The incremental 4D-var algorithm requires tangent-linear and adjoint versions of the forecasting model to propagate gradients to the initial condition. Hatfield et al. 2021 showed that by emulating a nonlinear component using a neural network they could simply derive and use the tangent linear and adjoint versions of the neural network within the 4D-var system.

Machine Learning for parameterization within numerical weather forecasts: Ongoing work has shown that parameterization schemes can be emulated by neural networks and that the neural network emulators are often much faster when compared to the conventional parameterization scheme (for example Chevalier et al., 1998, O'Gorman and Dwyer, 2018, and Chantry et al., 2020). To date, these have typically made simplifications to the modelling framework, e.g. simulating an aqua-planet or using coarse spatial resolution. Beyond learning existing parameterization schemes several groups have attempted to learn new parameterization schemes, using coarse-grained high-resolution simulations as a truth (Brenowitz et al., 2018, Rasp et al., 2018, Matsuoka et al. 2021). This approach promises greater possible benefits, but there are challenges in producing new parameterization that can run stably within a numerical weathering model (Brenowitz et al., 2018). An alternate approach seeks to nudge lower-resolution simulations towards higher-resolution to correct for incomplete or inaccurate parameterization (Watt-Meyer 2021).

Machine Learning for Post-processing model output: Methods such as Ensemble Model Output Statistics (EMOS) (Gneiting et al., 2005) and Bayesian Model Averaging (BMA) (Raftery et al., 2005) currently allow for improvements of the raw ensemble forecast skill. Hamill and Whitaker (2007) show initial explorations of those techniques on re-forecast datasets for temperature at 850 hPa (T850) and geopotential at 500 hPa (Z500). Advances in neural networks have only recently reached the field of ensemble models in weather forecasting, focusing on its application to specific weather stations (Rasp and Lerch, 2018).

Downscaling: Over the last few years, the meteorological domain has started to exploit sophisticated deep learning techniques to enhance the spatio-temporal resolution of weather predictions. Inspired by the success of sophisticated neural networks for generating super-resolution images in computer vision (Mahapatra et al. 2019; Wang, 2021), first studies try to transfer these techniques to the meteorological domain.

For instance, Sha et al. (2020a) applied a U-net architecture to downscale the daily minimum and maximum 2m temperature over complex terrain and showed that this approach may outperform classical ones. In their follow-up study Sha et al. (2020b), a variant of the U-net was trained to generate highly resolved precipitation patterns which also yields better results than a classical method based on bias correction and spatial disaggregation.

Leinonen et al. (2020) suggest a GAN equipped with a Convolutional Gated Recurrent Unit (ConvGRU) for downscaling time-evolving atmospheric fields. Their model architecture even allows them to generate ensembles which are considered to be beneficial for highly complex, non-linear atmospheric processes such as cloud dynamics and precipitation formation. A more comprehensive comparison study by Baño-Medina et al. (2020) further manifests that deep learning techniques constitute a promising tool to downscale meteorological fields.

5 Machine Learning solutions and tools used for the six applications

ML tools have been toughly surveyed in the previous section. The ML solutions used for the six applications in the project are summarized in Table 1. CNN, the CNN variant U-Net, and GAN are the main architectures that will be explored. The estimated maximum trainable parameters achieve 10 orders of magnitudes. Most of the applications employ distance-based loss/evaluation metrics, i.e., MSE, MAE. Some loss function is particular used for a special problem, i.e. CRPS is for ensemble forecasts in A4. TensorFlow 2.X and PyTorch are the two main DL frameworks, which will be explained with more details in Deliverable 2.1. Data parallelization strategy will be tried in the coming studies of the six applications with the assistance of Horovod Framework. A customized model parallelization strategy will be implemented in AP4. For application A2 we could, however, not yet collect sufficient data in the form of Twitter feeds that are selected around the topic of weather and climate predictions, as we are still waiting on a response from our application to Twitter to receive such data.

Table 1: Survey of the ML solutions and tools used for the six applications

AP	ML solutions	Estimated trainable parameters	Estimated size of Tier 2 Dataset	Loss function	ML tools	Distributed Training
1	U-Net, GAN	$O(10^4)$-$O(10^8)$	~ 10 TB	Quantile score	TensorFlow, Python	Data parallelization
2	-	-	~ 10 TB	-	TensorFlow2.X, PyTorch,	-
3	CNN, DenseNet	$O(10^5$-$10^6)$	~ 10 TB	MSE	TensorFlow 2.X	Data parallelization
4	CNN, 3D-Unet, DenseNet Inception module Transformers	$O(10^6)$-$O(10^{10})$	~ 3 TB	CRPS, Latitude-weighted MSE variants, SSIM	PyTorch 1.8 with or without PyTorch Lightning	Model /Data parallelization
5	U-Net, GAN, and RNN	$O(10^7)$- $O(10^8)$	~ TB	MAE, MSE, Adversarial loss, Earth Mover distance	TensorFlow v2.3.1 or v2.5 with Keras API	Data parallelization
6	RNN, GAN, CNN, Symbolic regression	$O(10^3)$- $O(10^5)$	~ GB - ~TB	MAE	TensorFlow2 or PyTorch	Data parallelization

6 Conclusion

MAELSTROM deliverable 1.2 provides the survey of requirements of ML solution and tools for the six applications. Importantly, the comprehensive review of ML and DL architectures and ML in W&C applications are given in this report. The tools used for the six applications are provided, which are essential information for the interaction between work package 1, 2 and 3 within this project.

7 References

Ahijevych D, Pinto JO, Williams JK, Steiner M. Probabilistic forecasts of mesoscale convective system initiation using the random forest data mining technique. Weather and Forecasting. 2016;31(2):581–99. https://doi.org/10.1175/WAF-D-15-0113.1

Arjovsky M, Chintala S, Bottou L. Wasserstein GAN. 2017; Available from: http://arxiv.org/abs/1701.07875

Awoyemi, J.O., Adetunmbi, A.O. and Oluwadare, S.A., 2017, October. Credit card fraud detection using machine learning techniques: A comparative analysis. In 2017 International Conference on Computing Networking and Informatics (ICCNI) (pp. 1-9). IEEE. 10.1109/ICCNI.2017.8123782

Baño-Medina J, Manzanas R, Gutierrez JM. Configuration and intercomparison of deep learning neural models for statistical downscaling. Geoscientific Model Development. 2020;13(4):2109–24. https://doi.org/10.5194/gmd-13-2109-2020

Ben-Nun T, Hoefler T. Demystifying parallel and distributed deep learning: An in-depth concurrency analysis. ACM Computing Surveys. 2019;52(4). https://doi.org/10.1145/3320060

Bonavita M, Laloyaux P. Machine learning for model error inference and correction. Journal of Advances in Modeling Earth Systems. 2020 Dec;12(12):e2020MS002232. https://doi.org/10.1029/2020MS002232

Breiman L. Random Forests. Machine Learning [Internet]. 2001 Oct;45(1):5–32. Available from: http://dx.doi.org/10.1023/A:1010933404324

Breiman L; Friedman J; Olshen R; Stone C (1984). Classification and Regression Trees, Chapman & Hall, page 95

Bauer, P., Thorpe, A. & Brunet, G. The quiet revolution of numerical weather prediction. Nature 525, 47–55 (2015). https://doi.org/10.1038/nature14956

Brenowitz ND, Bretherton CS. Prognostic validation of a neural network unified physics parameterization. Geophysical Research Letters. 2018 Jun 28;45(12):6289-98. https://doi.org/10.1029/2018GL078510

Chantry M, Hatfield S, Dueben P, Polichtchouk I, Palmer T. Machine learning emulation of gravity wave drag in numerical weather forecasting. Journal of Advances in Modeling Earth Systems. 2021 Jul;13(7):e2021MS002477. https://doi.org/10.1029/2021MS002477

Chevallier F, Chéruy F, Scott NA, Chédin A. A neural network approach for a fast and accurate computation of a longwave radiative budget. Journal of applied meteorology. 1998 Nov;37(11):1385-97. https://doi.org/10.1175/1520-0450(1998)037<1385:ANNAFA>2.0.CO;2

Bzdok, Danilo, Naomi Altman, and Martin Krzywinski. 2018. “Points of Significance: Statistics versus Machine Learning.” Nature Methods 15(4): 233–34. http://dx.doi.org/10.1038/nmeth.4642.

Cervantes J, Garcia-Lamont F, Rodríguez-Mazahua L, Lopez A. A comprehensive survey on support vector machine classification: Applications, challenges and trends. Neurocomputing. 2020; 408:189–215. https://doi.org/10.1016/j.neucom.2019.10.118

Esteva, A., Robicquet, A., Ramsundar, B., Kuleshov, V., DePristo, M., Chou, K., Cui, C., Corrado, G., Thrun, S. and Dean, J., 2019. A guide to deep learning in healthcare. Nature medicine, 25(1), pp.24-29. doi: 10.1038/s41591-018-0316-z

Gagne, David John, Amy McGovern, and Jerry Brotzge. 2009. "Classification of Convective Areas Using Decision Trees." Journal of Atmospheric and Oceanic Technology 26(7): 1341–53. https://doi.org/10.1175/2008JTECHA1205.1

Geer AJ. Learning earth system models from observations: machine learning or data assimilation?. Philosophical Transactions of the Royal Society A. 2021 Apr 5;379(2194):20200089. https://doi.org/10.1098/rsta.2020.0089

Giebel J, Kariniotakis G, Wind power forecasting—a review of the state of the art, in Woodhead Publishing Series in Energy,Renewable Energy Forecasting, Woodhead Publishing, 2017:59-109. https://doi.org/10.1016/B978-0-08-100504-0.00003-2

Gneiting T, Raftery AE. Strictly proper scoring rules, prediction, and estimation. Journal of the American statistical Association. Taylor & Francis; 2007;102(477):359–78. https://doi.org/10.1198/016214506000001437

Gneiting T, Raftery AE, Westveld AH, Goldman T. Calibrated probabilistic forecasting using ensemble model output statistics and minimum CRPS estimation. Monthly Weather Review. 2005;133(5):1098–118. https://doi.org/10.1175/MWR2904.1

Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative Adversarial Nets. Advances in neural information processing systems. 2014;2672–80. https://papers.nips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf

Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville A. Improved training of odellingn GANs. Advances in Neural Information Processing Systems. 2017;2017-Decem:5768–78. https://arxiv.org/abs/1704.00028

Gagne, David John, Sue Ellen Haupt, Douglas W. Nychka, and Gregory Thompson. 2019. "Interpretable Deep Learning for Spatial Analysis of Severe Hailstorms." Monthly Weather Review 147(8): 2827–45. https://doi.org/10.1175/MWR-D-18-0316.1

Hamill TM, Whitaker JS. Ensemble calibration of 500-hPa geopotential height and 850-hPa and 2-m temperatures using reforecasts. Monthly Weather Review. 2007;135(9):3273–80. https://doi.org/10.1175/MWR3468.1

Hatfield S, Chantry M, Dueben P, Lopez P, Geer A, Palmer T. Building tangent-linear and adjoint models for data assimilation with neural networks. Journal of Advances in Modeling Earth Systems. 2021 Feb 26:e2021MS002521. https://doi.org/10.1029/2021MS002521

Hu A, Cotter F, Mohan N, Gurau C, Kendall A. Probabilistic future prediction for video scene understanding. European Conference on Computer Vision. 2020. p. 767–85. DOI: 10.1007/978-3-030-58517-4_45

Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. Proceedings of the IEEE conference on computer vision and pattern recognition. 2017;4700--4708. https://openaccess.thecvf.com/content_cvpr_2017/papers/Huang_Densely_Connected_Convolutional_CVPR_2017_paper.pdf

Koenker R, Bassett Jr G. Regression quantiles. Econometrica: journal of the Econometric Society. JSTOR; 1978;33–50. https://doi.org/10.2307/1913643

Kumpf, Alexander et al. 2018. “Visualizing Confidence in Cluster-Based Ensemble Weather Forecast Analyses.” IEEE Transactions on Visualization and Computer Graphics 24(1): 109–19. 10.1109/TVCG.2017.2745178

Koza, J.R. (1992). Genetic Programming: On the Programming of Computers by Means of Natural Selection, MIT Press. https://doi.org/10.1007/BF00175355

LeCun, Yann, Bernhard Boser, John S. Denker, Donnie Henderson, Richard E. Howard, Wayne Hubbard, and Lawrence D. Jackel. “Backpropagation applied to handwritten zip code recognition.” Neural computation 1, no. 4 (1989): 541-551. 10.1162/neco.1989.1.4.541

Little, M.A., McSharry, P.E. and Taylor, J.W., 2009. Generalized linear models for site-specific density forecasting of UK daily rainfall. Monthly Weather Review, 137(3), pp.1029-1045. https://doi.org/10.1175/2008MWR2614.1

Leinonen J, Nerini D, Berne A. Stochastic Super-Resolution for Downscaling Time-Evolving Atmospheric Fields with a Generative Adversarial Network. arXiv. 2020;1–14. 10.1109/TGRS.2020.3032790

Lewis RJ. An introduction to classification and regression tree (CART) analysis. Annual meeting of the society for academic emergency medicine in San Francisco, California. 2000. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.4103&rep=rep1&type=pdf

Lagerquist, Ryan, A. M.Y. McGovern, and David John Gagne. 2019. “Deep Learning for Spatially Explicit Prediction of Synoptic-Scale Fronts.” Weather and Forecasting 34(4): 1137–60. https://doi.org/10.1175/WAF-D-18-0183.1

O’Gorman PA, Dwyer JG. Using machine learning to parameterize moist convection: Potential for odelling of climate, climate change, and extreme events. Journal of Advances in Modeling Earth Systems. 2018 Oct;10(10):2548-63. https://doi.org/10.1029/2018MS001351

Mahapatra D, Bozorgtabar B, Garnavi R. Image super-resolution using progressive generative adversarial networks for medical image analysis. Computerized Medical Imaging and Graphics [Internet]. 2019;71:30–9.https://www.sciencedirect.com/science/article/pii/S0895611118305871

Matsuoka, Daisuke, Shingo Watanabe, Kaoru Sato, Sho Kawazoe, Wei Yu, and S. Easterbrook. "Application of deep learning to estimate atmospheric gravity wave parameters in reanalysis data sets." Geophysical Research Letters 47, no. 19 (2020): https://doi.org/10.1029/2020GL089436

McCulloch WarrenS, Pitts W. A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics [Internet]. 1943;5(4):115–33. Available from: http://dx.doi.org/10.1007/BF02478259

Qiu, J., Wu, Q., Ding, G., Xu, Y. and Feng, S., 2016. A survey of machine learning for big data processing. EURASIP Journal on Advances in Signal Processing, 2016(1), pp.1-16. https://doi.org/10.1186/s13634-016-0355-x

Quade, Markus and Abel, Markus and Shafi, Kamran and Niven, Robert K. and Noack, Bernd R. "Prediction of dynamical systems by symbolic regression". Phys. Rev. E, no. 94 (1): 012214 (2016). https://link.aps.org/doi/10.1103/PhysRevE.94.012214

Rasp S and Lerch S. Neural Networks for Postprocessing Ensemble Weather Forecast. Monthly Weather Review. 2018;146(11). 10.1162/neco.1989.1.4.541

Rasp S, Pritchard MS, Gentine P. Deep learning to represent subgrid processes in climate models. Proceedings of the National Academy of Sciences. 2018 Sep 25;115(39):9684-9. https://arxiv.org/abs/1806.04731

Raftery AE, Gneiting T, Balabdaoui F, Polakowski M. Using Bayesian model averaging to calibrate forecast ensembles. Monthly Weather Review. 2005;133(5):1155–74. https://doi.org/10.1175/MWR2906.1

Schoof JT, Pryor SC. Downscaling temperature and precipitation: A comparison of regression-based methods and artificial neural networks. International Journal of Climatology. 2001;21(7):773–90. https://doi.org/10.1002/joc.655

Sha Y, Gagne DJ, West G, Stull R. Deep-learning-based gridded downscaling of surface meteorological variables in complex terrain. Part i: Daily maximum and minimum 2-m temperature. Journal of Applied Meteorology and Climatology. 2020a;59(12):2057–73. https://doi.org/10.1175/JAMC-D-20-0057.1

Sha Y, Gagne DJ, West G, Stull R. Deep-learning-based gridded downscaling of surface meteorological variables in complex terrain. Part II: Daily precipitation. Journal of Applied Meteorology and Climatology. 2020b;59(12):2075–92. https://doi.org/10.1175/JAMC-D-20-0057.1

Sharma N, Sharma P, Irwin D, Shenoy P. Predicting solar generation from weather forecasts using machine learning. 2011 IEEE international conference on smart grid communications (SmartGridComm). 2011. p. 528–33. DOI: 10.1109/SmartGridComm.2011.6102379

Shi, Xingjian et al. 2015. "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting." In Advances in {{Neural Information Processing Systems}} 28, eds. C Cortes et al. Curran Associates, Inc., 802–10. https://proceedings.neurips.cc/paper/2015/file/07563a3fe3bbe7e3ba84431ad9d055af-Paper.pdf

Xingjian S H I, Chen Z, Wang H, et al. Convolutional LSTM network: A machine learning approach for precipitation nowcasting[C]//Advances in neural information processing systems. 2015: 802-810.Srivastava N, Mansimov E, Salakhutdinov R. Unsupervised Learning of Video Representations Using LSTMs. International conference on machine learning. 2015;843–52. https://papers.nips.cc/paper/2015/file/07563a3fe3bbe7e3ba84431ad9d055af-Paper.pdf

Ronneberger, O., Fischer, P. and Brox, T., 2015, October. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham. https://doi.org/10.1007/978-3-319-24574-4_28

Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition. 2015;1--9. https://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Szegedy_Going_Deeper_With_2015_CVPR_paper.pdf

Kim S, Hong S, Joh M, et al. Deeprain: Convlstm network for precipitation prediction using multichannel radar data[J]. arXiv preprint arXiv:1711.02316, 2017. https://arxiv.org/abs/1711.02316

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. Advances in neural information processing systems. 2017;5998--6008. https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

Vapnik VN. The Nature of Statistical Learning Theory. Springer; 1999;

Voyant C, Muselli M, Paoli C, Nivet M-L. Numerical weather prediction (NWP) and hybrid ARMA/ANN model to predict global radiation. Energy [Internet]. 2012;39(1):341–55. https://www.sciencedirect.com/science/article/pii/S0360544212000114

Wang Y. Single Image Super-Resolution with U-Net Generative Adversarial Networks. IEEE; 2021;2021:1835–40. DOI: 10.1109/IMCEC51613.2021.9482317

Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing. IEEE; 2004;13(4):600–12. DOI: 10.1109/TIP.2003.819861

Yang, C., Chandler, R.E., Isham, V.S. and Wheater, H.S., 2005. Spatial-temporal rainfall simulation using generalized linear models. Water Resources Research, 41(11).

Watt-Meyer O, Brenowitz ND, Clark SK, Henn B, Kwa A, McGibbon J, Perkins WA, Bretherton CS. Correcting weather and climate models by machine learning nudged historical simulations. Geophysical Research Letters. 2021 Aug 16;48(15):e2021GL092555. https://doi.org/10.1029/2021GL092555

Document History

Version	Author(s)	Date	Changes
0.1	Bing, Gong (JSC)	16/09/2021	Initial version
1.0	Bing, Gong (JSC)	30/09/2021	Final version

Internal Review History

Internal Reviewers	Date	Comments
Peter Dueben (ECMWF)	17/09/2021	Accepted with minor comments
Greta Denisenko (4cast)	21/09/2021	Review passed with minor edits

Estimated Effort Contribution per Partner

Partner	Effort
JSC	1
ECMWF	0.2
Total	**1.2**