Evaluation of weather forecast with DL

A B M Abir Mahboob
Department of Electronic Engineering
Hochshule Hamm Lippstadt
Lippstadt, Germany
a-b-m-abir.mahboob@stud.hshl.de

***Abstract*—The capabilities of a deep neural network to simulate the dynamics of a straightforward general circulation model is demonstrated. The network can anticipate the full model state many time steps in advance after being trained on the model, conceptually equivalent to issuing weather forecasts in the model world. In addition, the network can produce a run with climate statistics that are close to those of the climate of the global circulation model after being set with any given model state. This is done by continually feeding back forecasts into the network's inputs. Even though no explicit conservation characteristics were included into the network, this network climate run exhibits no lengthy deviation.**

I. Introduction

This section is based on [1]. Due to its impact on human existence worldwide, weather forecasting has drawn the interest among several scholars from different research fields. Many studies have been driven to investigate hidden patterns in the vast dataset for weather prediction because of the increasing prevalence of vast weather observation data nowadays and the development of computer and information technologies in the past ten years. Flight navigation, cultivation, and tourism are just a few of the many possible uses for weather forecasting, making it a captivating research problem. Among the problems of weather forecasting are learning weather interpretation using a vast volume of weather dataset and constructing an effective weather prediction model that utilizes latent patterns and structures in the huge quantity of weather dataset.

Major efforts have been made in weather forecasting in the past ten years, utilizing statistical modeling approaches, particularly machine learning, with favourable outcomes.

Deep learning neural networks (NNs) have more layers than "shallow" ones employed in conventional machine learning models, as indicated by the term "deep" in the phrase. After successfully implementing the layer-wise unsupervised pre-training methodology that is used to effectively address the training challenges, this multi-layer NN has attracted significant research interest. When compared to shallow models, the "deep" architecture is crucially significant since NN with deep architecture can offer superior learning capabilities.

Deep learning is implemented in weather representation and modeling as a result of its effectiveness in several domains, as documented by several researchers. For instance, a new study suggests [2] utilizing deep neural networks (DNN) to learn hierarchical features from a significant amount of weather data to describe the weather. [1]

II. Various Types of Models

This section is based on [3]. Weather prediction and climate science depend heavily on numerical weather prediction (NWP) models and general circulation models (GCMs). They solve stochastic physical equations of the thermodynamics of the atmosphere in order to calculate the development of atmospheric states through time. Many applications of machine learning methods in relation to GCMs and weather prediction models have recently been put forth. All of the methods represented are essential in climate research and meteorology. To extract specific data from models or to add information to other models has been the common goal of all of them, nevertheless. In this case, deep learning is utilized to directly resemble the entire physics and dynamics of a GCM rather than to retrieve data from a climate model or to merge various models. A neural network is developed that uses the full-scale model state of the GCM as input to predict the upcoming model state. On this domain, significant advancements have been reported for highly idealized models for example. the Lorenz63 and Lorenz96 models. There have lately been notable advances in the simplified weather forecasting setting's ability to anticipate geopotential height. [3]

III. Different Types of Algorithms

This section and its subsections are based on [6]. Deep learning is a part of Machine learning. Now, machine learning success is dependent on the algorithms that power it. Without being expressly taught to do so, ML algorithms by adopting sample data, generate a numerical model in order to make predictions or choices. This method sometimes classified as "training data". This can highlight patterns in the data that organizations can utilize to enhance decision-making, maximize productivity, and collect meaningful data at scale. AI solutions that automatically automate processes and resolve data-based business challenges are built on top of machine learning (ML). Companies can use it to supplement or replace some human skills. [6], [7]

Three categories of approaches are used in machine learning.

A. Supervised Learning

Supervised machine learning aims to develop a model that creates predictions in the face of uncertainty based on data. Utilizing a collection of existing input data and recognised responses to the data(output), supervised learning trains a model to generate reliable prediction for the response to inbound data. When the data for the prediction outputs is already available, supervised learning is employed. Classification and regression algorithms are utilized in supervised learning to generate prediction models. Logistic regression is a supervised machine learning method. [6]

Discrete responses are predicted using classification techniques. Some instances include categorizing a tumor as malignant or benign and determining whether an email is real or spam. Models for classification cohorts the incoming data into categories. Examples of typical uses include fraud detection, voice recognition, and medical imaging.

Continuous responses are predicted via regression algorithms. Changes in temperature and fluctuations in power consumption are two examples of when regression techniques can be employed. When operating with a data range or when the nature of a response is a real number, regression techniques are required. Two prominent implementations of supervised learning include algorithmic trading and forecasting electricity load. Convolutional Neural Network (CNN) that we used in this study is an example of supervised learning. [6]

B. Unsupervised Learning

Unsupervised learning detects basic data patterns or underlying patterns. It is designed to draw inferences from datasets that have input data but no labeled responses. In unsupervised learning, clustering continues to be the most utilized algorithm. To uncover hidden patterns or groups in data, it is employed in exploratory data analysis. Market analysis, object identification, and DNA sequence analysis are some usages for cluster analysis. Gaussian mixtures is an example of unsupervised learning. [6]

C. Reinforcement Learning

Algorithms for reinforcement learning interact with their surroundings by taking actions, identifying mistakes, and learning from successes or failures. Trial-and-error learning and delayed rewards are two of reinforcement learning's most important features. With the help of this technique, machines and software proxies can automatically select the best course of action in a given situation to enhance performance. To learn which action is optimal, the agent requires simple reward feedback, often referred as the reinforcement signal. [7]

IV. PUMA Method

This section is based on [3]. The dynamical center of the PLASIM (Planet Simulator) model is the Portable University Model of the Atmosphere (PUMA), a straightforward representation of the global atmosphere. Modern global NWP models, which are the primary tool for medium-range weather predictions, are theoretically very similar to this model since they are based on simple equations. A key instrument in climate research, global circulation models, are also associated to it. Due to the fact that this study serves as a statement of concept, PUMA was selected because of its minimal computing needs, which make it simple to construct lengthy time sequences of its model environment. Furthermore, The volume of data is kept within manageable bounds by the comparatively low resolution. The fairly modest resolution maintains the volume of data in a manageable range. The simplest feasible model arrangement was chosen for the same reasons. The model has 10 vertical levels, a time step of 45 minutes, and a horizontal resolution of T21 (about 625 km, 32 $\times$ 64 grid points when projected on a standard latlon grid). Oceans, orography, the daily cycle, and the seasonal cycles were not included. Nevertheless, the model is intricate and similar to the actual environment. The four state variables (geopotential height, horizontal and meridional wind, and temperature) of the model are resampled from the 10 model levels to the 10 pressure levels for the final data. A common method for presenting model data on pressure level enables for the analysis of fundamental variables, such as the 500-hPa geopotential height, which simplifies the analysis of the results. The state of model is daily stored. All variables are standardized to zero mean and unit variance for each layer individually since neural networks are not scale dependent.

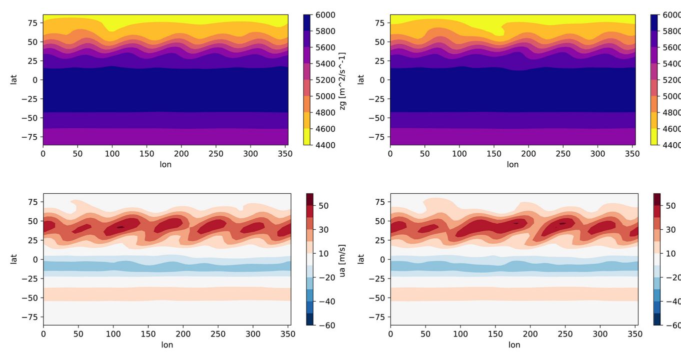

Fig. 1. Two model states of the PUMA model, separated by 5 days (from left to right). The upper row shows geopotential at 500 hPa (zg), and the lower row shows zonal wind at 300 hPa (ua).[3]

Two samples of the model integration are presented in Figure 1 and are separated by five days. In spite of the model's simplicity, the standard eastward spreading waves can be distinctly seen in the winter hemispheres. The real atmosphere is composed with such waves as well. On timeframes of several days, they are also one of the primary characteristics of midlatitude weather variability. Typically, the available data samples are divided into three categories for machine learning tasks. The training set is where the network is actually trained. The development set, also known as the validation set, is employed for tuning. Finally, the test set is utilized for validation. Algorithms for machine learning have a tendency to overfit, hence this is required. This indicates that while they are highly accurate in predicting samples from the training set, they struggle to do so when presented with data that were not present during the training. Following is a breakdown of the PUMA run: 150 years' worth of daily data are left after the initial 30 years are nullified as "spin up years". The data are then divided into three sections: 20 years for development, 30 years for testing, and 100 years for training. The training data were trimmed down to 30 years for the network's tuning in order to maximize computational efficiency. [3]

V. Neural Network Architecture

This section is based on [3]. All nonlinear functions can generally be approximated by artificial neural networks. For image recognization, a specific type of artificial neural networks, or CNNs, are implemented. Each convolutional layer is followed by a nonlinear transformation, and they utilize convolutional kernels to manage input as they move through the layers. On regular grids, atmospheric data is frequently represented. As a result, individual atmospheric variables on a certain height or pressure level correlate to individual image channels in a single time slice of a weather or climate model. CNN applications in relation to atmospheric models are therefore intriguing. CNNs have a significant advantage over other machine learning approaches in that no previous dimensionality reduction is required. Autoencoders served as the foundation for the architecture employed here. A particular kind of neural network called an autoencoder has two components. After each layer in the first half, the dimensionality is reduced. Following each layer in the second section is mirrored upsampling, which increases the dimensionality. The majority of neural network topologies feature either high-dimensional input and low dimensional output or low-dimensional input and output. Our autoencoder-like system has the unique characteristic of having high-dimensional input and output, or, to be more specific, output of the same dimension as input (40channels$\times$2,048gridpoints=81,920). As a result, the network is able to forecast a new complete set of atmospheric fields at a later time using the whole set of atmospheric fields from the GCM as input at one particular time step. This is not always required for a weather forecast scenario. This may be taken further by actually using the network to build a climate run that starts with a single beginning state and then is given time to evolve. It takes 20 CPUs about 90 minutes to train the network. [3]

VI. Results

This section along with the subsections are based on [3]. The results of having the network anticipate the climate model many days ahead are displayed first in this section. The whole test set is averaged to determine all results. The results of utilizing the network to construct a climate run are reported in the second section. The 500-hPa geopotential height, a standard NWP validation parameter, is the focus of this analysis.

A. *Weather Forecasting Mode*

Here, we evaluate the trained network's prowess in weather forecasting using a model (hence, projecting the model state's progression up to 14 days in advance). The model state's correlation between t = 0 and t = 1 day, t = 0 and t = 2 days, and so forth was emulated by the network by training it individually for each lead time.

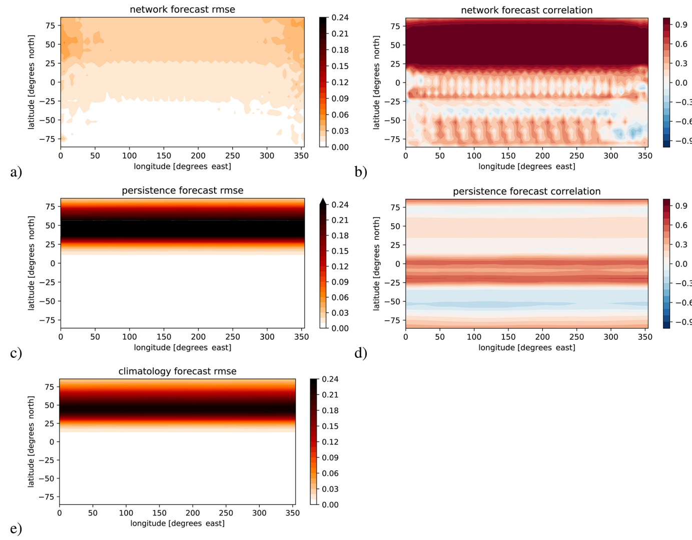

Fig. 2. (a) Root-mean-square error (rmse) and (b) correlation for 6-day 500-hPa geopotential height forecasts of the network. (c, d) Same as a and b but for persistence forecasts. (e) Same as a but for climatology forecasts.[3]

In Figures 2a and 2b, maps of the root-mean-square error (rmse) for 6-day predictions of 500-hPa geopotential height are exhibited in comparison to the GCM's real state. In the Northern (permanent winter) Hemisphere, the predictions exhibit extremely low error and a very significant correlation. At the domain's margins, error rates are higher. The network is not structured to loop around margins, but the climate model has an uninterrupted (circular) domain, that might account for this. The relatively low variation of the model weather throughout the summer is what leads to the low correlation in the Southern (permanent summer) Hemisphere, making correlation as a metric less relevant. The rmse in the summer hemisphere is even lower than in the winter hemisphere, which is also significant in low-variance weather. When compared to climatological and persistence forecasts, the network forecasts are much significant (as long as the model state doesn't change). This is applicable for both the correlation and the rmse forecasts for persistence (Figures 2c and 2d). It is impractical to compute a correlation for climatological forecasts. The persistence forecasts (Fig 2e) and rmse are similar.

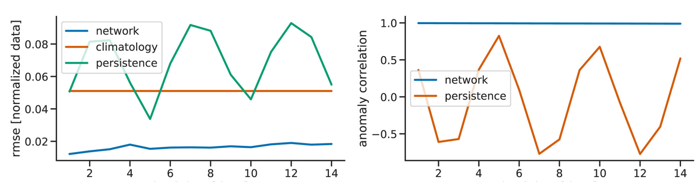

Fig. 3. Anomaly correlation and root-mean-square error of the baseline methods and network predictions for various lead times.[3]

The anomaly correlation coefficient and rmse of network predictions are shown in Figure 3, where the blue line represents the lead times of 1-14 days, the orange line represents the climatological forecasts and the green line represents the persistence forecasts. For every lead time, network projections outperform baselines by a significant margin. For the majority of lead times, persistence predictions outperform climatology. The low rmse and extremely high anomaly correlation (0.99) demonstrate how well the network appears to reflect the dynamics of the model. The results are comparable for various variables and pressure levels (not shown). Thus, we can say that the network is equipped to predict the climate model's "weather" with accuracy. [3]

B. Climate mode

The trained network is now employed to construct a climate run. This is done by making a 1-day forecast using test data from a model state that was randomly selected. This prediction is then utilized as the network's starting point, leading to the generation of a new prediction and so on. This results in the construction of a timeline with 30 years' worth of daily fields.

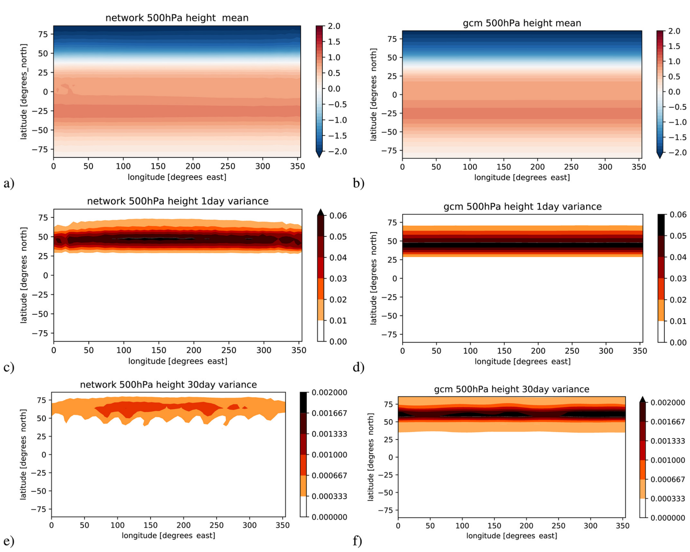

Fig. 4. (a, b) Climatology of normalized 500-hPa geopotential height of (a) the network and (b) the general circulation model (GCM). (c, d) Variance of daily mean 500-hPa geopotential height of (c) the network and (d) the GCM. (e, f) Same as c and d but for variance of 30-day means.[3]

The 500-hPa geopotential height for the network climate is depicted in Figure 4 as a climatology (panel a). The climatology closely resembles the map of the climate model in terms of both regional distribution and size (panel b). Despite the fact that the network overlooks the 30-day variance, it encompasses the variation of daily and 30-day means rather well (panels c–f). Furthermore, the variance at the domain's edges is negligible. This could be the result of the network's inability to wrap around edges, similar to the RMSE in the weather forecasting mode, although the climate model has a complete (circular) domain. The climate model's starting state had no effect on any of the results, which had been evaluated with a variety of randomly selected initial states.

In order to determine whether the network exhibits any drift or misleading long-term fluctuations, we undergo a much lengthier climate run with it. Since our network does not explicitly mandate energy expenditure or anything similar, this cannot be ruled out as a priority and there is nothing that could theoretically stop it from deviating off into wholly implausible states.

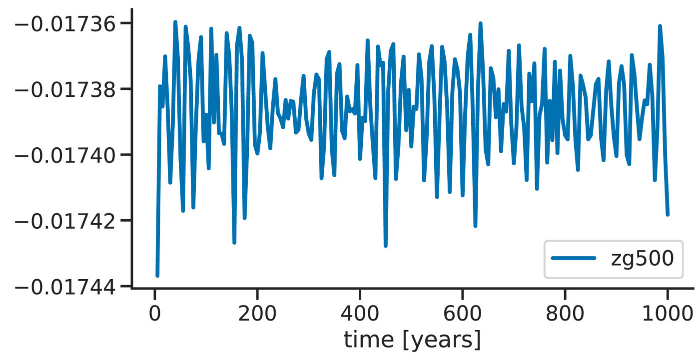

Fig. 5. 500-hPa height five-year mean values across a 1,000-year network climatic run.[3]

A 1,000-year network climate run is featured in Figure 5 together with the evolution of the 5-year averages of the 500-hPa geopotential height. Because they can still be plotted effectively, five-year means were used. It is clear that none of the variables, nor any other variables or levels, have seen any long-term drift. The existing state choice has no impact on all these results. Similar to the 30-day variance, Figure 5's 5-year mean variance shows that the network exhibits lower volatility than the model. [3]

VII. Conclusion

This section along with the subsections are based on [3]. To perfectly simulate a basic GCM, we employed a reasonably common machine learning approach, specifically a deep CNN. In order to synchronize the dynamics of the model, the network is trained using the GCM. After training, the network accurately predicts the paradigm multiple time cycles in advance, conceptually equivalent to issuing weather forecasts in the model environment. An initial random state from the climate model was also used to establish the network and the network's input was continually sent back to the prediction. By doing so, the network is able to generate an entire climate run with climatic statistics that are comparable to those of the climate model. Even over extremely long durations, this network environment remains consistent (1,000 years). This is especially promising since the network still creates a steady climate without drifting even though we haven't explicitly mandated any kind of energy or other conservation qualities.

This study should be viewed as a proof of concept since it demonstrates that, in theory, it is feasible to allow a neural network to learn the time evolution and, consequently, the whole dynamics of a straightforward GCM. The next logical step for future researches would be to explore more complicated and realistic models as the relatively high skill of the neural network in forecasting the model weather may in part be driven by the relative simplicity of the model. When using neural networks to climate research, it is crucial to accurately reflect external forces. Due of the more complicated characteristics in those models and the greater volume of data, the aforementioned will create additional obstacles. The latter is brought on by both the increased resolution of more complicated models and the likelihood that longer training years are required for a complex climate or weather system. However, the idea raises the prospect of a fresh approach to data-driven weather forecasting. Long collections of detailed simulations from climate models might be used to train neural networks. In order to utilize the network to anticipate the real weather many days in advance, one may input the analysis of a weather forecast model into it as a predictor. Furthermore, one may train the network using observations/reanalysis rather than model data and then execute it in climate mode to construct a climate simulation.

Attempting to create a unique convolutional network design that uses the native model grid would be fascinating when it comes to neural network research. One might also examine in depth how the dimensionality minimization of the autoencoder-like network utilized in this study influences the outcomes and whether or not adding recurrent components to the network can enhance predictions. [3]

VIII. ACKNOWLEDGEMNTS

The University of Hamburg offers the PUMA model [4]. The associated Zenodo repository contains the machine learning code and namelist documents for the run performed in this paper [5].

IX. DECLARATION OF ORIGINALITY

I, A B M Abir Mahboob, herewith declare that I have composed the present paper and work by myself and without the use of any other than the cited sources and aids. Sentences or parts of sentences quoted literally are marked as such; other references with regard to the statement and scope are indicated by full details of the publications concerned. The paper and work in the same or similar form have not been submitted to any examination body and have not been published. This paper was not yet, even in part, used in another examination or as a course performance. I agree that my work may be checked by a plagiarism checker.

Abir

11.01.2023 & Hamm - A B M Abir Mahboob

REFERENCES

[1] A. G. Salman, B. Kanigoro and Y. Heryadi, "Weather forecasting using deep learning techniques," 2015 International Conference on Advanced Computer Science and Information Systems (ICACSIS), 2015, pp. 281-285, doi: 10.1109/ICACSIS.2015.7415154.

[2] J. N. K. Liu, Y. Hu, Y. He, P. W. Chan and L. Lai, "Deep Neural Network Modeling for Big Data Weather Forecasting", pp. 389-408, 2015.

[3] Scher, S. (2018). Toward data-driven weather and climate forecasting: Approximating a simple general circulation model with deep learning. Geophysical Research Letters, 45, 12,616– 12,622.

[4] University of Hamburg (Puma Model) https://www.mi.uni-hamburg.de/en/arbeitsgruppen/theoretische-meteorologie/modelle/plasim.html

[5] Code and data for "Towards data-driven weather and climate forecasting: Approximating a simple general circulation model with deep learning" https://doi.org/10.5281/zenodo.1472023

[6] Aery, M., and Ram, C. (2017). A Review on Machine Learning: Trends and Future Prospects. Research Cell: Int. J. Eng. Sci., 25, 89-96.

[7] Selig, J. (2022, April 11). What is machine learning? A definition. Expert.ai.