JAMES | Journal of Advances in Modeling Earth Systems

RESEARCH ARTICLE
10.1029/2019MS001705

Can Machines Learn to Predict Weather? Using Deep Learning to Predict Gridded 500-hPa Geopotential Height From Historical Weather Data

Jonathan A. Weyn[1], **Dale R. Durran**[1], **and Rich Caruana**[2]

[1]Department of Atmospheric Sciences, University of Washington, Seattle, WA, USA, [2]Microsoft Research, Redmond, WA, USA

Key Points:
- Deep convolutional neural networks trained to predict gridded weather from historical reanalysis significantly outperform basic benchmarks
- Unlike the dynamical barotropic vorticity model, the neural networks can predict amplification and decay of weather systems
- The neural networks produce realistic 14-day weather forecasts despite having no explicit knowledge of atmospheric physics

Correspondence to:
J. A. Weyn,
jweyn@uw.edu

Citation:
Weyn, J. A., Durran, D. R., & Caruana, R. (2019). Can machines learn to predict weather? Using deep learning to predict gridded 500-hPa geopotential height from historical weather data. *Journal of Advances in Modeling Earth Systems*, *11*, 2680–2693. https://doi.org/10.1029/2019MS001705

Received 29 MAR 2019
Accepted 11 JUL 2019
Published online 17 AUG 2019

Abstract We develop elementary weather prediction models using deep convolutional neural networks (CNNs) trained on past weather data to forecast one or two fundamental meteorological fields on a Northern Hemisphere grid with no explicit knowledge about physical processes. At forecast lead times up to 3 days, CNNs trained to predict only 500-hPa geopotential height easily outperform persistence, climatology, and the dynamics-based barotropic vorticity model, but do not beat an operational full-physics weather prediction model. These CNNs are capable of forecasting significant changes in the intensity of weather systems, which is notable because this is beyond the capability of the fundamental dynamical equation that relies solely on 500-hPa data, the barotropic vorticity equation. Modest improvements to the CNN forecasts can be made by adding 700- to 300-hPa thickness to the input data. Our best performing CNN does a good job of capturing the climatology and annual variability of 500-hPa heights and is capable of forecasting realistic atmospheric states at lead times of 14 days. Although our simple models do not perform better than an operational weather model, machine learning warrants further exploration as a weather forecasting tool; in particular, the potential efficiency of CNNs might make them attractive for ensemble forecasting.

1. Introduction

At present, numerical weather prediction (NWP) is essential for weather forecasting, providing vital information about severe weather threats and risk management. General circulation models (GCMs) used for NWP combine state-of-the-art numerical representation of atmospheric fluid dynamics and other physical processes such as radiative transfer and cloud processes. GCMs still have important shortcomings, including imperfect physical parameterizations, insufficient resolution to resolve fine-scale weather, and very high computational cost for operational forecasting. Increasingly, the NWP community has turned to advanced machine learning (ML) techniques to address some of these shortcomings. ML has improved interpretation and performance of GCMs in many ways. For example, Rasp and Lerch (2018) used neural networks (NNs) to successfully improve postprocessing of GCM forecasts to surface stations, while Rodrigues et al. (2018) demonstrated the ability of deep NNs to downscale GCM output to higher horizontal resolution. Deep NNs have also been used to identify extreme weather and climate patterns in observed and modeled atmospheric states (Kurth et al., 2018; Lagerquist et al., 2019; Liu et al., 2016), improve parameterizations in GCMs (e.g., Brenowitz & Bretherton, 2018; Rasp et al., 2018), and predict uncertainty in weather forecasts (Scher & Messori, 2018). Larraondo et al. (2019) demonstrated the ability of deep NNs to extract spatial patterns in precipitation from gridded atmospheric fields. Finally, ML algorithms have also been used to predict extreme weather events (e.g., Herman & Schumacher, 2018) and provide operational guidance and risk assessment for severe weather (McGovern et al., 2017).

Yet a basic question remains: Can ML produce purely statistical weather forecasts by learning from past weather observations *with no explicit information about atmospheric physics whatsoever*, thereby learning to effectively emulate the physical processes of the atmosphere? With several decades of reliable weather data from satellite observations, widely available open-source software, and efficient graphics processing unit computing, weather prediction using ML, and specifically deep NNs (LeCun et al., 2015), is becoming increasingly feasible. A few previous studies have begun investigating this problem. Dueben and Bauer (2018) used NNs trained on several years of reanalysis data to predict 500-hPa geopotential height on the

globe at 6° resolution. Their "global" model, which simultaneously predicts weather at all points on the globe, only marginally beat persistence at early forecast lead times; NNs trained to predict one grid point at a time from a stencil of local neighboring points and subsequently applied over the entire globe (their "local" model) fared much better. Nevertheless, their experiment is somewhat limited in its use of relatively little historical data (only about 7 years of atmospheric samples) and coarse horizontal resolution. Scher (2018) used a different approach, instead training NNs on the output of a highly idealized GCM to learn to emulate the dynamics of the GCM by mapping the atmospheric state at one time to the state at another later time. Using convolutional NNs (CNNs), which are widely used in image recognition and processing problems, they were able to significantly outperform baseline metrics and effectively represent the simplified GCM dynamics. However, the "weather" generated by their GCM was idealized in comparison to the real world, because processes like chaotic upscale error growth and factors like seasonality were not included. An extension of this work, which applied CNNs to GCM output including seasons and at higher horizontal resolution, showed a more complicated story (Scher & Messori, 2019). The CNNs performed slightly worse on model simulations including seasons, while their performance was more severely degraded on higher-resolution input, albeit more due to the complexity of the resolved weather than the computational cost of increasing the number of grid points. Nevertheless, these results are encouraging for proceeding forward with ML-based global weather forecasts.

Here, we approach ML weather prediction head-on by developing models that use CNNs to learn to predict 500-hPa geopotential heights and 700- to 300-hPa thickness from 24 years of atmospheric reanalysis on a 2.5° grid over the Northern Hemisphere. We improve upon the prior work of Dueben and Bauer (2018) by using more advanced NN architectures, more years of reanalysis data, and higher resolution (albeit over the Northern Hemisphere only), while in contrast to Scher (2018) and Scher and Messori (2019), we predict observed weather instead of idealized GCM forecast states. Not surprisingly, our ML models do not compare in forecast accuracy to current operational NWP models, which have been refined by decades of research, operate at much higher resolution, and use far more data to describe the initial condition for each forecast. Nevertheless, our ML models significantly outperform persistence and climatology benchmarks, as well as a basic dynamical model that computes numerical solutions to the barotropic vorticity equation, which was the type of model used in the earliest years of NWP. In section 2 we provide details about the model structure and the data used for training. Section 3 presents the results from our ML weather predictions. Finally, conclusions and discussion are provided in section 4.

2. The DLWP Model

Our Deep Learning Weather Prediction (DLWP) model uses deep CNNs for globally gridded weather prediction. A global weather prediction model must be given an initial multidimensional atmospheric state $\mathbf{u}(t)$ and yield the state of the atmosphere at a future time, $\mathbf{u}(t + \Delta t)$. To step the model forward in time, the predicted state must include all of the features of the input state. Dynamical models of the atmosphere compute tendencies of physical variables determined by equations of motion and physical parameterizations and then integrate forward in time. Our DLWP CNNs directly map $\mathbf{u}(t)$ to its future state $\mathbf{u}(t + \Delta t)$ by learning from historical observations of the weather, with Δt set to 6 hr. This direct mapping is notably different from the approach of Dueben and Bauer (2018), where their NN is used to predict tendencies, which are then used in an explicit time-stepping scheme to integrate forward. By feeding the predicted atmospheric state back as inputs to the model, DLWP algorithms can be iteratively propagated forward without explicitly using a numerical time-stepping scheme.

A state $\mathbf{u}$ for input to a DLWP model will be defined as a four-dimensional array with dimensions equal to the number of time steps, meteorological variables times vertical levels, latitudes, and longitudes. The "time steps" dimension indicates how many time steps in the past are provided in the data inputs to the model; for example, if a model initialized at 0 UTC 1 January has two input time steps at 6-hourly intervals, it would have data from 18 UTC 31 December and 0 UTC 1 January as inputs and predict the next two time steps, 6 UTC 1 January and 12 UTC 1 January. In early testing, using two time steps yielded significantly better results than just one, and therefore, all results herein use a time step dimension of 2. Using more time steps may further improve the forecasts but is left for future work. Individual atmospheric variables (temperature, humidity, pressure, etc.) on individual vertical levels are simply treated as "channels" similar to the RGB channels of color images. This allows us to use powerful and efficient image-processing algorithms

such as two-dimensional convolutions. We note that using data on a globe presents unique challenges to convolutions due to grid distortion; some of these challenges are discussed further in sections 2.2 and 4.

Careful consideration is required when selecting the specific atmospheric variables to use when building a DLWP model. Increasing the number of variables greatly increases the number of individual features the model must predict as well as the number of degrees of freedom within hidden convolutional layers. While deep CNNs generally benefit from predicting multiple correlated output variables, more complex networks require increased computation and training iterations, or epochs (for a review of multitask learning, see Ruder, 2017). In this paper, we will present several DLWP model variants, with some trained only on 500-hPa geopotential height (Z_{500}) and others trained on both Z_{500} and 700- to 300-hPa thickness ($\tau_{700-300}$). DLWP models might be expected to predict Z_{500} reasonably well from pattern recognition, but adding thickness provides a measure of baroclinicity that should allow them to better predict the amplification and decay of weather systems.

The DLWP CNNs are built using the open-source Keras library for Python (Chollet, 2015) with Google's TensorFlow backend (Abadi et al., 2015). More details about the NN algorithms are provided in section 2.2. The code for this project is available at the website github.com/jweyn/DLWP .

2.1. Data

The NNs are trained on gridded reanalysis data from the Climate Forecast System (CFS) Reanalysis (available at https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/climate-forecast-system-version2-cfsv2 website; Saha et al., 2010). We use the 2.5° product with 6-hourly time resolution in the period from 1 January 1979 to 31 December 2010. We chose this particular data set because its low resolution is ideal for computationally efficient testing and because a CFS reforecast product generated with the operational CFSv2 model is available for benchmarking. To further increase computational efficiency, we subset only the northern hemisphere, as cross-equatorial transport is usually not significant on time scales of a few days.

Data from 2007–2010 were set aside for the test set used in final model performance evaluation. We used the time periods from 1979–2002 for model training and 2003–2006 for model validation. Distinct periods for training, validation, and testing were selected to avoid including in the evaluation data times that have high correlation with neighboring times in the training data. While there is some concern that the distribution of data may shift during the time frame of data availability, we assume that such climatological shifts in geopotential heights are insignificant.

Each variable at each vertical level is scaled by removing its Northern Hemisphere climatological mean and dividing by its mean climatological standard deviation. Scaled variables are denoted with a hat, for example, $\hat{Z}_{500}$. By scaling with the spatially averaged values, we retain local spatial differences in variability, ensuring that the NN loss function appropriately weights regions of high variability. No further input or output feature selection, scaling, or other modification is performed on the data before it is passed to the NNs. Since the NNs predict scaled variables, an inverse scaling is applied to the NN output to recover dimensional atmospheric variables.

2.2. Algorithms

The core of our DLWP model is an auto-encoder-like CNN (Baldi, 2012) with six convolutional layers. This architecture is similar to that used by Scher (2018) and Larraondo et al. (2019). Each of the first two convolutional layers in our CNN is immediately followed by a 2×2 max-pooling dimensionality reduction, while the next two are followed by mirrored 2×2 upsampling dimensionality increases. Lastly, two more convolutional layers with no spatial dimensionality changes map the filters back to the output variables. To avoid dimensionality loss caused by applying filters to the edges of the images, the convolutional layers are all preceded by an appropriate padding that appends zeros in the latitudinal direction and periodic boundary conditions in the longitudinal direction. The latter solves the apparent lateral boundary issue in the CNN forecasts made by Scher (2018, see their Figure 2a). All convolutional layers use the tanh activation function, except for the final ("output") layer, which has linear activation as required for a regression problem. The CNN architecture is summarized in Table 1. The reader is referred to Appendix A for more details on the convolution operations.

Advantages of the fully CNN architecture include a low number of trainable parameters (the most complex model has about 200 K parameters; see Table 1) and a requirement that interactions remain local: That

Table 1
Neural Network Architecture for DLWP as a Sequence of Operation Layers

Layer	Filters	Filter size	Dilation	Output shape[a]	Trainable params[b]
input				(vt, 36, 144)	
ConvLSTM2D[c]	$2v$	3×3	2	(t, $4v$, 36, 144)[d]	736
Conv2D	32	3×3	2	(32, 36, 144)	608
MaxPooling2D		2×2		(32, 18, 72)	
Conv2D	64	3×3	1	(64, 18, 72)	18,496
MaxPooling2D		2×2		(64, 9, 36)	
Conv2D	128	3×3	1	(128, 9, 36)	73,856
UpSampling2D		2×2		(128, 18, 72)	
Conv2D	64	3×3	1	(64, 18, 72)	73,792
UpSampling2D		2×2		(64, 36, 144)	
Conv2D	32	3×3	2	(32, 36, 144)	18,464
Conv2D	v	5×5	1	(vt, 36, 144)	1602

Note. The parameter v represents the number of distinct variable/level pairs, while t represents the size of the time dimension. The layer names correspond to the names in the Keras library. DLWP = Deep Learning Weather Prediction; LSTM = long short-term memory.
[a]Output shape is (channels, y, x). [b]Approximate number of learned parameters for $t = 2$, $v = 1$. [c]Only the LSTM variants include the ConvLSTM2D layer. [d]The ConvLSTM2D layer has a separate time dimension which is subsequently reshaped into $4vt$ output channels. For variants with the ConvLSTM2D layer the inputs and outputs are also reshaped to (t, v, 36, 144).

is, individual grid points in the output layer are only determined by the neighboring grid points within the convolutional stencil, similar to local advection in the atmosphere. Nevertheless, because a unique set of filters is learned for the entire spatial domain, global processes are inherently accounted for, and the depth of the network, along with the spatial dimensionality reduction from pooling, also enables it to learn larger-scale interactions. It is unclear whether this combination of global filters and local interactions is aptly suited for representing multiscale atmospheric phenomena. In particular, it might be better to avoid using a single global set of output filters to learn the widely different meteorology in the tropics and the midlatitudes. To investigate this possibility, we trained variants of DLWP where the output convolutional layer is replaced with a specialized convolutional layer that learns a unique set of filters for every row (latitude band) of the images; these variants are denoted with "ROW" (see section 2.3).

We also train DLWP variants which include a convolutional long short-term memory (LSTM) layer (Gers et al., 2000; Hochreiter & Schmidhuber, 1997); these variants are denoted with "LSTM" (see section 2.3). LSTM units contain a cell state, which is propagated forward in time, modulated by gates (learned weights), which leverage how much to "forget" from the cell state at the previous time step and how much new information to "input" to the cell from the data at the current time step. This enables them to make short-term modulations to their state while maintaining long-term behavior. Thus, LSTMs are ideally suited to problems with complex time-dependent structure, including weather prediction. They have been successfully applied to several weather-related problems, including by Vlachas et al. (2018) to the idealized Lorenz 96 model (Lorenz, 1996) and a barotropic vorticity model, and by Shi et al. (2015) to short-term precipitation forecasting.

All of the DLWP CNNs are trained using the efficient Adam version of stochastic gradient descent optimization (Kingma & Ba, 2014), with a default learning rate of 1×10^{-3}, and using mean-squared-error loss. To ensure that a suitable loss minimization is obtained, we train for a minimum of 200 epochs followed by early stopping conditioned on the validation set loss. If no new validation loss minimum is observed within 50 epochs, training stops and the model weights which yielded the validation loss minimum are restored.

Different NN architectures, including some using densely connected output layers, were tested in the early stages of this project. The architecture described here performed second best in our testing only to a NN with a densely-connected output layer. However, the memory requirement for a matrix which maps every input feature vector element to every output feature vector element is untenably large and increases as n^2 for n input features, or $(1/\Delta x)^4$ for decreases in horizontal grid spacing. Therefore we eliminated such NNs

Table 2
List of Presented DLWP Model Variants

DLWP variant	Z_{500}	$\tau_{700-300}$	LSTM	Latitude dependent
Z	✓	—	—	—
τ	✓	✓	—	—
τ ROW	✓	✓	—	✓
Z LSTM	✓	—	✓	—
τ LSTM	✓	✓	✓	—
τ LSTM ROW	✓	✓	✓	✓

Note. The columns labeled Z_{500} and $\tau_{700-300}$ indicate whether the DLWP inputs/outputs include 500-hPa height and 700- to 300-hPa thickness, respectively. Those labeled "LSTM" and "latitude-dependent" specify whether the DLWP neural network includes an LSTM layer or latitude-dependent output layer filters, respectively. DLWP = Deep Learning Weather Prediction; LSTM = long short-term memory.

in favor of the fully convolutional architecture. Furthermore the success of this type of architecture applied to gridded atmospheric fields (Larraondo et al., 2019; Scher, 2018; Scher & Messori, 2019) further validates its use. We performed limited validation of the CNNs using varying numbers of convolutional layers and convolutional filter stencil sizes (and dilation) before obtaining this architecture. As in nearly every deep learning problem, there is no guarantee that this architecture is optimal.

2.3. Summary of Presented DLWP Variants

Table 2 summarizes the DLWP variants selected for presentation in section 3. We vary the input variables (e.g., the addition of thickness) or CNN architectures (e.g., the addition of an LSTM layer). The most basic model, "Z," is trained only on 500-hPa geopotential height with the CNN architecture described in section 2.2, while the most complex, "τ LSTM ROW," adds 700–300 hPa thickness to the input data, an LSTM layer, and latitudinally dependent output layer convolutional filters.

DLWP variants with thickness included in the input and output should have an advantage given sufficient training data and training epochs because of information about baroclinic growth and decay in the atmosphere. Variants with an LSTM layer are able to interpret time sequences in the input data and should better represent chaotic behavior, while those with latitudinally dependent output layer convolutional filters should be better able to represent latitudinal variations in weather patterns.

2.4. Benchmarks

Finally, benchmarks are necessary to help us contextualize the performance of DLWP. Our benchmarks are (1) an operational full-physics forecast model; (2) the barotropic vorticity model, which solves the fundamental dynamical equation for the evolution of the 500-hPa height field (Z_{500}) based solely on its values at some initial time; (3) climatology; and (4) persistence. Details about these benchmarks are provided below.

1. The CFS reforecast is produced with the full-physics, operational CFSv2 model (Saha et al., 2014) run at T126 spherical harmonic truncation. These forecasts are initialized with the CFS reanalysis. The CFS reforecast represents the skill of typical operational models at 1° latitude-longitude resolution. It should perform far better than our DLWP models because it makes predictions using many variables, many vertical levels, and higher horizontal resolution (1° in contrast to 2.5° for the DWLP models). It includes parameterizations of physical processes including convective precipitation, the planetary boundary layer and radiative transfer.
2. The barotropic vorticity equation may be written as

$$\frac{D}{Dt}(\zeta + 2\Omega \sin \phi) = 0, \tag{1}$$

where ζ is the vorticity of the nondivergent component of the 500-hPa wind, Ω is the angular speed of rotation of the Earth, ϕ is the latitude, and D/Dt is the material derivative following the flow. The barotropic dynamics embodied in (1) conserve absolute vorticity along fluid-parcel trajectories. The first generation of NWP models predicted 500-hPa heights by numerically integrating the barotropic vorticity equation (Charney et al., 1950; Silberman, 1954). Our implementation of this model is global, with initial conditions given by the CFS reanalysis. We retain all available spherical wavenumbers represented on the 2.5°

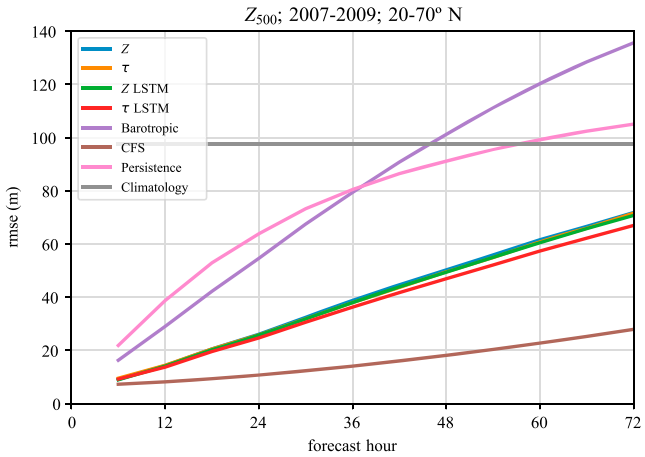

Figure 1. Forecast root-mean-square error (rmse) of 500-hPa geopotential height averaged over the test period 2007–2009 for several Deep Learning Weather Prediction variants and the climatology, persistence, barotropic, and CFS benchmarks. The error is spatially averaged over all longitudes, and over latitudes of 20–70°N. CFS = Climate Forecast System; LSTM = long short-term memory.

regular grid and run the model with a time step of 30 min. Additional details about this model are provided in Appendix B. Because the only input data required for the barotropic vorticity model is the initial Z_{500} field, it provides an apples-to-apples comparison between a dynamical model and our DLWP variants that use only Z_{500}.

3. Climatology is calculated relative to monthly means in the entire CFS reanalysis data set (1979–2010).
4. Persistence is the initial Z_{500} field held constant with time. Forecasts with verification-time errors exceeding climatology or persistence are considered to have no skill.

3. Results

We begin by presenting the spatially and temporally averaged error in DLWP and benchmark forecasts as a function of forecast lead time. To better understand the physical states produced by DLWP, we then present two example forecasts and a zonally averaged forecast state climatology.

3.1. Average Forecast Error

Figure 1 shows the 72-hr evolution of the spatially averaged forecast root-mean-square error (rmse) in 500-hPa geopotential height, averaged over the latitude band 20–70°N and the testing period from 2007 to 2009, for four key DLWP variants and all of the benchmarks. The rmse in Z_{500} is one of the most basic measures of forecast accuracy and is used to assess the performance of state-of-the-art operational NWP models (Palmer, 2018). We immediately note that all the DLWP variants perform much better than the simple climatology and persistence benchmarks. The DLWP schemes also do much better than the barotropic model, which in turn, outperforms persistence through a lead time of 36 hr. Not surprisingly, the operational CFS performs significantly better than DLWP and the other benchmarks, especially at longer forecast lead times.

Among the DLWP variants, "τ LSTM" stands out from the rest with the lowest forecast rmse at all times. Its rmse is about 5% lower than that of the other variants at all lead times past 12 hr. It is encouraging that the CNN with the best treatment of chaotic time-dependent behavior (an LSTM layer) and with information about baroclinic growth and decay (inclusion of thickness) performs best. This suggests that with sufficient training the CNN is able to learn more from additional atmospheric state variables. However, it is also interesting to note that preliminary tests using CNNs trained on geopotential height at three pressure levels (300, 500, and 700 hPa), but no thickness (not shown), did not perform any better than the single-level "Z LSTM" variant. As in many ML problems, appropriately informed selection of the input and output features is key to improving model performance. In this case, the extra levels of geopotential height were of limited use because the model had no information about the vertical structure of the input variables.

The robustness of the preceding results was tested by training ensembles of each of the four DLWP variants using 12 different initial random seeds without any other changes in the ML architecture. The best performing CNN of each variant, as measured by the forecast rmse out to 72 hr on the validation set (not shown),

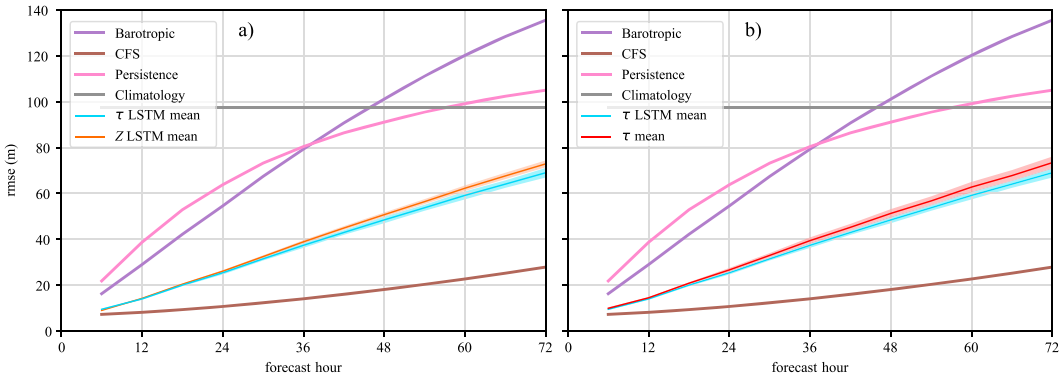

Figure 2. (a) As in Figure 1 but with the curves for the Deep Learning Weather Prediction variants replaced by the ensemble mean and one-standard-deviation spread, for the "τ LSTM" ensemble (light blue line and shading) and the "τ" ensemble (orange line and shading). (b) As in (a) except with the "Z LSTM" ensemble (red line and shading) replacing the "τ" ensemble. CFS = Climate Forecast System; LSTM = long short-term memory; rmse = root-mean-square error.

was selected for the results in Figure 1. Figure 2a compares the mean and standard deviation in the forecast rmse for the "Z LSTM" and "τ LSTM" DLWP variants, while Figure 2b compares those of the "τ" and "τ LSTM" variants. The performance of the 12-member ensemble means is very similar to that of the single members shown in Figure 1, with "τ LSTM" again having lower error compared to both "Z LSTM" and "τ" There is about 25% more spread in the τ ensemble than in the others, indicating that this variant has the least consistent results during training of the CNNs. The improvement produced by using an LSTM layer is statistically significant at the 99% confidence level when the ensemble means from the "τ LSTM" and the "τ" variants are compared using Welch's unequal variance t test (Welch, 1947). Applying the same test to the "τ LSTM" and "Z LSTM" ensemble means shows the improvement generated by adding thickness data

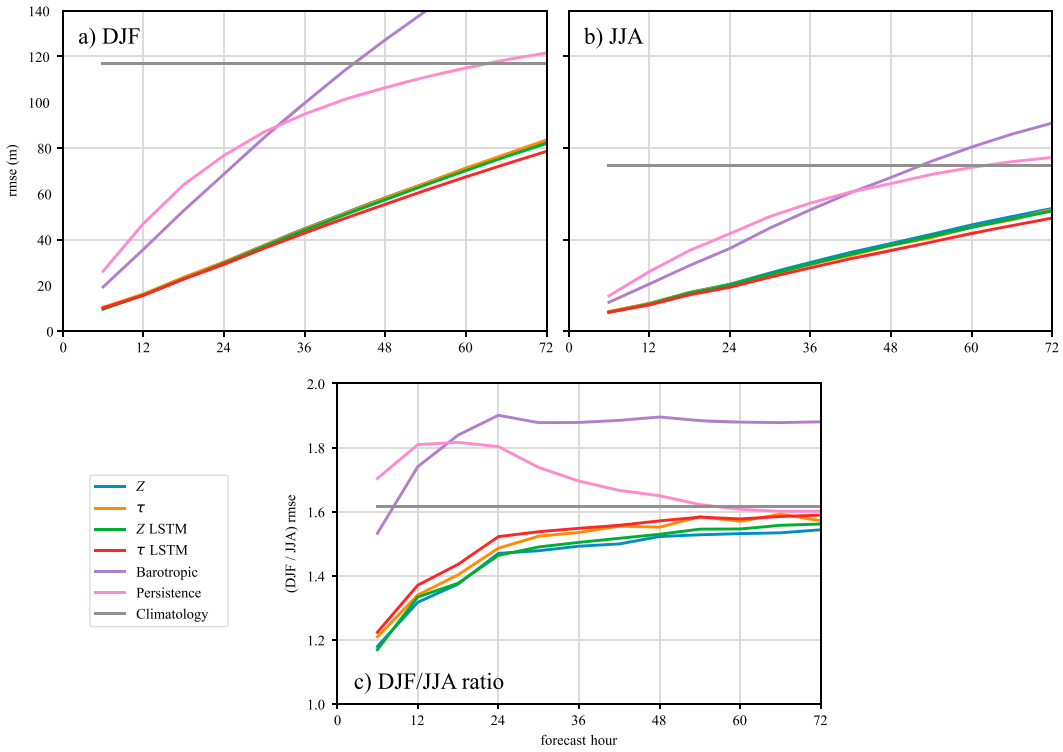

Figure 3. (a, b) As in Figure 1 but for (a) only the DJF season and (b) only the JJA season. (c) Ratio of forecast rmse in the DJF season divided by forecast rmse in the JJA season. DJF = December, January, and February; JJA = June, July, and August; LSTM = long short-term memory; rmse = root-mean-square error.

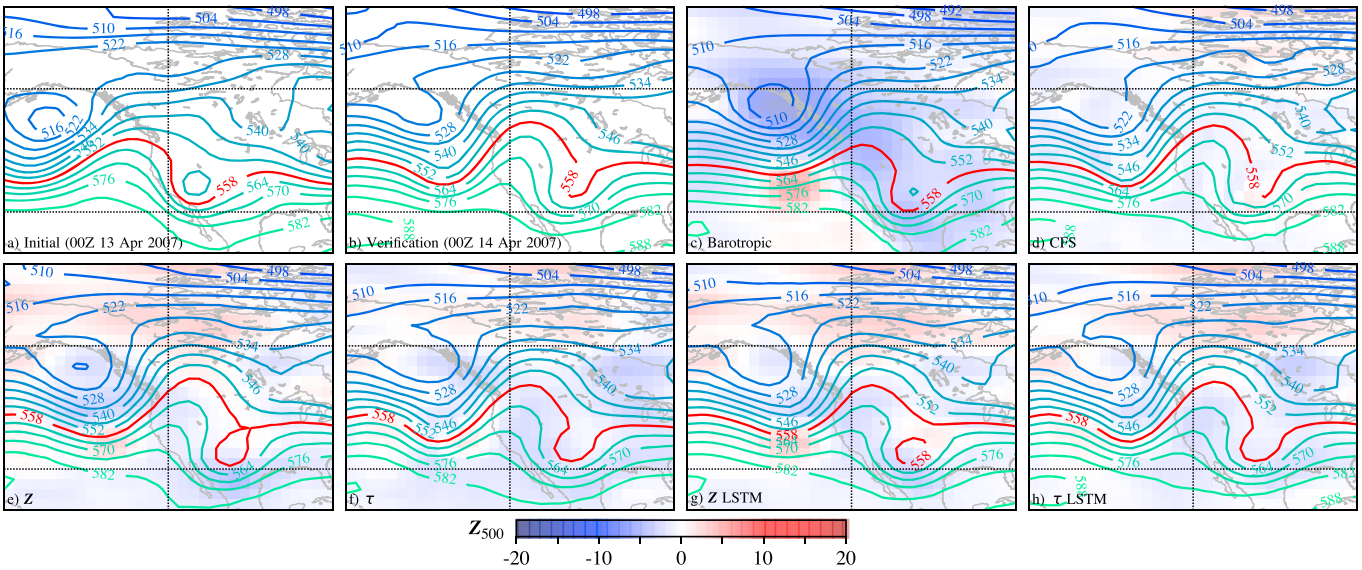

Figure 4. 24-hr forecasts of Z_{500} (contours, dkm) initialized at 0 UTC 13 April 2007, for a subset of the forecast domain over North America. (a) Initialization state from reanalysis. (b) The 24-hr verification from reanalysis. (c) Forecast from the barotropic vorticity model. (d) Forecast from the CFS reforecast. (e–h) Forecasts from Deep Learning Weather Prediction variants, as labeled. Colored shading indicates the difference between the forecast and verification (b) states, in dkm. The 558 dkm contour is highlighted red for emphasis. CFS = Climate Forecast System; LSTM = long short-term memory.

is significant at the 95% confidence level at lead times of 18 and 24 hr, and at the 99% level for lead times of 30 hr or more.

While Figure 1 shows annual averages of rmse, it is also instructive to examine the performance of the models for the winter and summer seasons separately because in the midlatitudes, there is much greater variability in Z_{500} in winter than in summer. Figure 3a shows the rmse for December, January, and February (DJF), while Figure 3b shows the rmse for June, July, and August (JJA), averaged in both cases over the years 2007–2009. Forecast rmse in DJF is much higher than in JJA, as expected, and in both seasons the DLWP variants continue to easily outperform the climatology, persistence, and barotropic-vorticity-model benchmarks. Examining the ratio of DJF rmse to JJA rmse (Figure 3c), we find that, over the first 12 hr, the DJF/JJA ratios for the DLWP variants are closest to unity, whereas persistence and climatology show a much larger seasonal dependence in their errors. From 24 to 72 hr, the DLWP variants have a nearly constant DJF/JJA ratio of about 1.5, while that of the dynamical barotropic vorticity model is near 1.9. The DLWP models do not suffer from the same degradation of performance in the winter season as the dynamical barotropic vorticity model does, especially at earlier forecast lead times, indicating that the CNNs learned valuable information about all the seasons from the historical training data. In comparison to the other variants, the best performing DLWP variant, "τ LSTM", has the largest DJF/JJA ratio owing to better performance in the summer months rather than degradation in the winter months.

3.2. Example Forecasts

The DLWP models perform well in terms of spatially and temporally averaged error metrics, but how well do they forecast individual weather events? To address this question, we compare forecasts generated by the DLWP variants and the benchmark barotropic and CFS models for two initial times, 0 UTC 13 April 2007 and 0 UTC 15 April 2007, that are associated with a significant late-season snowstorm in the eastern United States. Figures 4 and 5 show contours of the 24-hr forecast Z_{500} field produced by the models initialized at these two times, respectively, along with the difference between the observed height field and each forecast (color fill in panels c–h), for a subset of the model domain covering North America.

The initial condition for the first case shows cutoff lows over the southwestern United States and the Gulf of Alaska (Figure 4a), both of which decay into open-wave troughs as they propagate eastward over the next 24 hr (Figure 4b). The barotropic model conserves absolute vorticity along the trajectories traced by the geostrophic wind. Since the relative vorticity is proportional to the Laplacian of the Z_{500} height field, the barotropic model cannot capture major changes in the strength of troughs or ridges. As a consequence, it fails to capture the weakening of these systems and surprisingly, it intensifies the cutoff low in the Gulf of

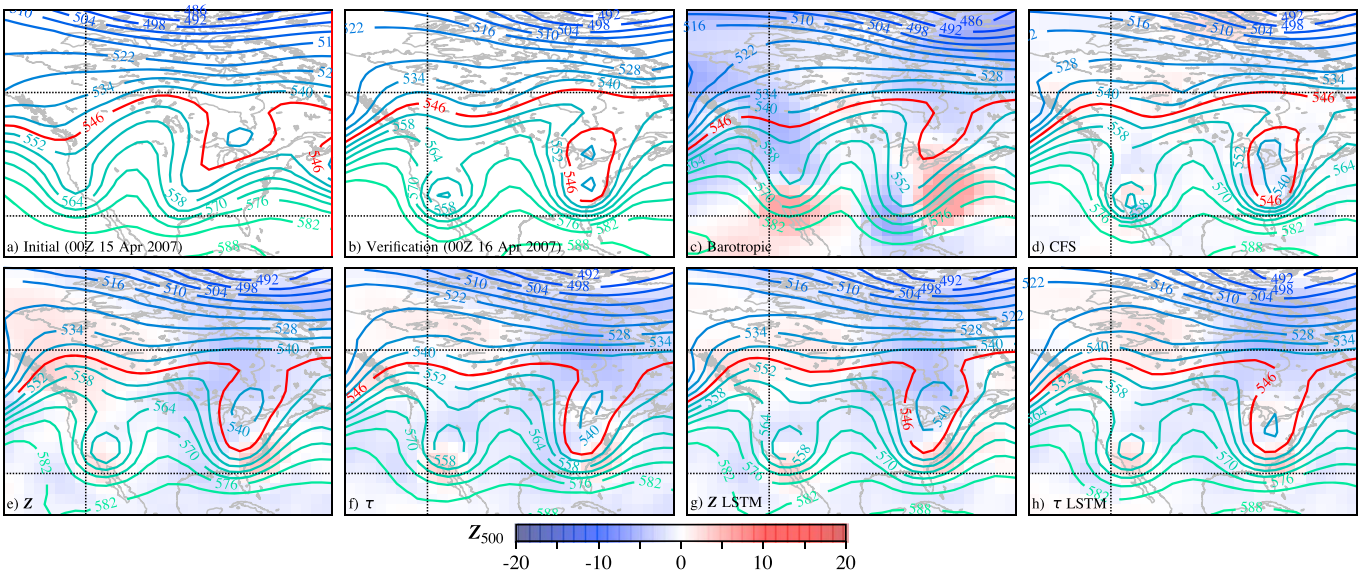

Figure 5. As in Figure 4 except for forecasts initialized at 0 UTC 15 April 2007. The 546 dkm contour is highlighted in red.

Alaska (Figure 4c; The horizontal scale of this cutoff low increases, which will act to decrease $\nabla^2 Z_{500}$ and thereby tend to preserve a constant value for the vorticity as the low modestly deepens). As expected, the operational CFS model forecast (Figure 4d) is very good, with only minor differences between it and the verification field. The DLWP variants (Figures 4e–4h) generally capture the evolution of these troughs well, but those that do not include information about the 700- to 300-hPa thickness allow both lows to remain too deep. Nevertheless, the DLWP forecasts are far better than that from the barotropic model.

In contrast to the decaying systems present in the first case, 2 days later the troughs initially present along the west coast and over the Midwest United States (Figure 5a) deepen dramatically. Over the next 24 hr, the trough over the West Coast evolves into a cutoff low and the trough in the Midwest digs strongly to the southeast (Figure 5b). As expected, based on its limited dynamics, the barotropic model fails to correctly amplify both systems (Figure 5c). Not surprisingly, the operational CFS model again performs quite well (Figure 5d). The DLWP variants all do reasonably well; those that include information about thickness better capture the amplitude and shape of the eastern trough and actually produce less error than the CFS model near the center of the cutoff low over the Great Lakes.

In both of these cases, the DLWP models appear to suffer from large-scale errors in Z_{500} at high latitudes. In the first case the heights in much of the northern part of the domain are overestimated; in the second case they are underestimated. This may be evidence of the inability of the CNN architecture to accurately capture multiscale atmospheric interactions, as described in section 2.2, particularly near the poles, where the correct cross-pole connections are not accounted for in the two-dimensional convolution operations on the regular grid. Nevertheless, these examples show that DLWP produces very realistic short-range weather forecasts. It is particularly impressive that the two DWLP variants trained only on information about the geopotential height at a single level can learn to predict changes in amplitude, since this is beyond the capabilities of dynamical model forecasts based on the same input data.

3.3. Extended Forecasts

In contrast to climate models, where confidence that the model is accurately representing the physics of the atmosphere is crucial for its utility, our focus is on weather prediction, where daily opportunities for forecast verification allow one to readily assess the performance and practical utility of a system relying on ML. Despite this focus on weather prediction, it is useful to examine the behavior of the preceding DLWP models at longer forecast lead times. Forecasts from almost identical initial states gradually diverge such that the root-mean-square difference between them approaches a factor of $\sqrt{2}$ the difference between either state and climatology (Leith, 1974). For synoptic-scale midlatitude weather systems, such error saturation and loss of predictability occurs after roughly 2 weeks (Selz, 2019; Zhang et al., 2019). Our DWLP models

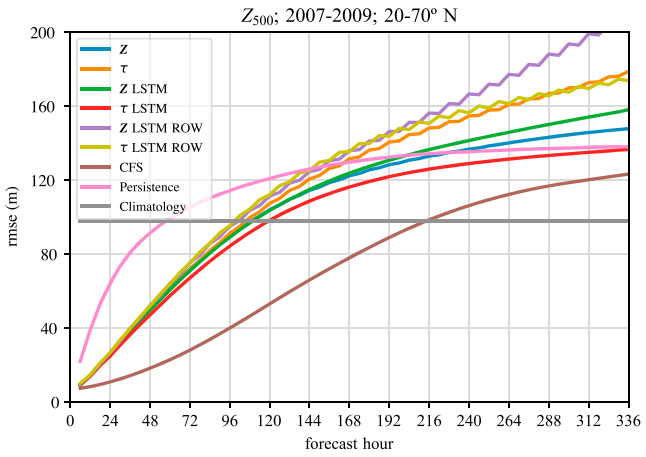

Figure 6. As in Figure 1 except including all Deep Learning Weather Prediction variants integrated out 14 days (336 hr.)

were trained to minimize the error in a 6-hr time step; do they continue to predict realistic atmospheric states on 2-week time scales?

The rmse for all of the DLWP variants and benchmarks integrated out 14 days (336 hr) are shown in Figure 6. As in Figure 1, these data are averaged over the latitude band 20–70°N and the years 2007–2009. As expected the rmse in the persistence forecast asymptotes to $\sqrt{2}$ times the error in a climatological forecast. Not surprisingly, the operational CFS model exhibits the lowest rmse and its errors approach the expected $\sqrt{2}$ times climatology more slowly than that of the other models. Among the DWLP variants, the "τ LSTM" variant is clearly superior: its rmse is the smallest and appears closest to correctly approaching $\sqrt{2}$ times the climatological error. Clearly evident in Figure 6 is the unexpectedly poor performance of "τ" model at later forecasts times compared to the "Z" and "Z LSTM" variants, which have no thickness information. Recalling that the "τ" variant shows the largest ensemble spread among identical model architectures trained with different random seeds (Figure 2), it may be that the "τ" variant is not as suited for this particular combination of data variables, or that more training data is needed. The "τ" variant's forecasts also exhibit a clear $2\Delta t$ oscillation not seen in the "Z," "Z LSTM," and "τ LSTM" DLWP variants. Due to the configuration of the problem whereby two time steps are used in the inputs and outputs, this oscillation is a result of the CNN not accurately correlating predicted time steps and giving an inferior forecast for the second step. Also shown in Figure 6 are a pair of DLWP variants with latitude-dependent output layer filters. These variants are also prone to the $2\Delta t$ oscillation and unfortunately do not perform better than the variants with standard output convolutional filters. The increased CNN complexity for these "ROW" variants may require more training data to learn the additional parameters.

At longer forecast lead times, DLWP becomes particularly sensitive to nonoptimal weights learned during training. If the model forecast gradually diverges from the set of realistic atmospheric states, the CNNs can be confronted with input that is too physically dissimilar from that seen in the training data to allow an accurate estimate of the future state, thereby reinforcing any tendency of the model to diverge from reality. This may explain the disappointing results from the "τ" variant and the two "ROW" variants apparent in Figure 6.

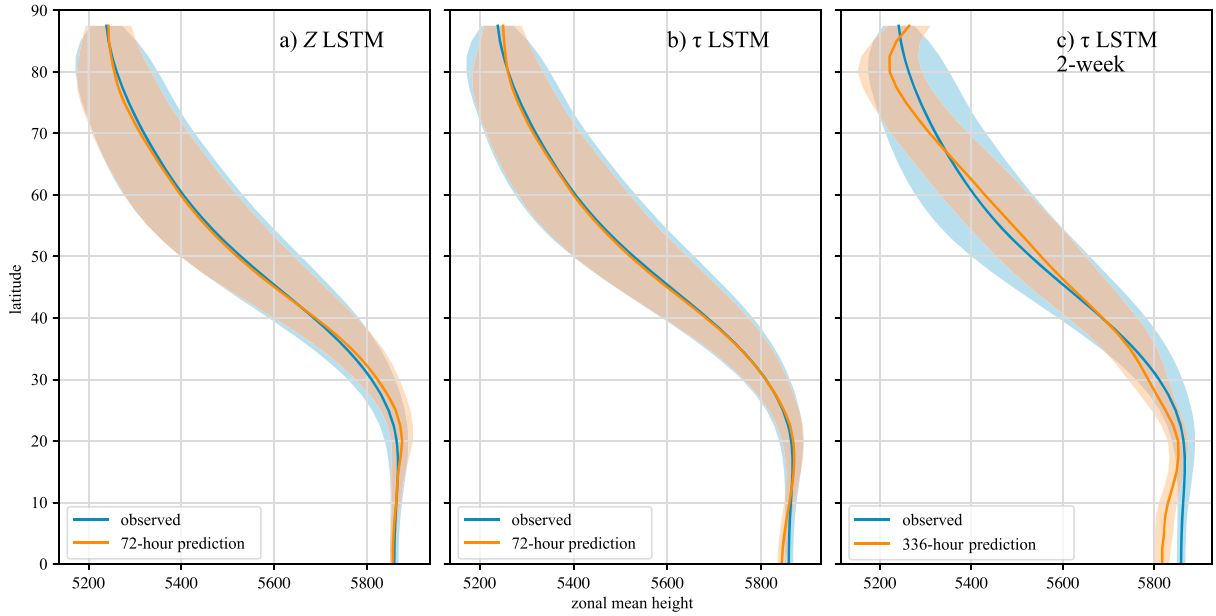

Figure 7. Zonal mean and standard deviation of Z_{500} from reanalysis (blue line) and the (a) "Z LSTM" forecasts and (b) "τ LSTM" forecasts (orange lines) at lead times of 72 hr for the period from 2007–2009. In (c), "τ LSTM" is extended out to 14 days (336 hr). The blue (orange) shading represents ±1 standard deviation from the zonal mean of the reanalysis (forecasts). LSTM = long short-term memory.

Nevertheless, it is impressive that, at least over a 2-week forecast, the magnitude of the rmse in the "τ LSTM" model stays within the correct bounds for the difference between two randomly selected atmospheric states.

As a final assessment of the physical realism in the DLWP forecasts, we compare the zonal mean and standard deviation of the 72-hr forecasts of Z_{500} with the reanalysis data at each verification time for the full Northern Hemisphere and the period 2007–2009. The meridional dependence of the zonal-mean heights measures how well DLWP reproduces the mean state of the atmosphere, while the zonal standard deviation measures DLWP's ability to retain realistic atmospheric variability, such as that produced by midlatitude storm systems. As shown in Figure 7b, the forecasts from the "τ LSTM" model follow the observed zonal mean nearly exactly except for very slight errors near the equator and the pole. The standard deviation about the zonal means is also captured almost exactly. Despite having no information about thickness, the "Z LSTM" model also retains nearly correct zonal mean and standard deviation (Figure 7a), only exhibiting minor overprediction at latitudes of about 15–30°N.

Figure 7c is identical to Figure 7b, except that we extend the "τ LSTM" forecasts out to 14 days. Even at the 2-week forecast lead time, the model does a good job of maintaining physically reasonable states. There are modest low biases of about 50 m in $\hat{Z}_{500}$ near the pole and the equator, both of which are likely due to inappropriate boundary conditions. The zonal variance in the midlatitudes is reduced in the DLWP prediction, indicating a reduction in the storm track amplitudes, but it is slightly increased near the equator, again likely a result of the incorrect southern boundary condition. These results suggest that the "τ LSTM" DLWP model is maintaining realistic representations of mean and variability of the atmosphere over time scales relevant to deterministic weather forecasting, despite having no explicit knowledge of atmospheric physics.

4. Conclusions

In this study, we trained CNNs to predict the most fundamental atmospheric field, 500-hPa geopotential height, using historical gridded reanalysis data. Our DLWP model significantly outperforms persistence, climatology, and the most directly comparable dynamical model based on the barotropic vorticity equation. The CNNs produce realistic representations of evolving weather patterns without the provision of explicit information about the equations governing atmospheric motions. (The gridded reanalysis data used for the initial conditions in our forecasts, and for training, testing, and verification of each DWLP variant, are produced using state-of-the-art weather prediction models as part of the data assimilation procedure. In this limited respect our DLWP models still rely on explicit information about atmospheric physics and dynamics.) The DLWP variant with a recurrent LSTM layer and 700- to 300-hPa thickness included in its input performs best, indicating that the CNN is learning useful information about the time dependence from the LSTM layer and baroclinic effects from the inclusion of thickness. Even DLWP variants trained only on 500-hPa geopotential height data were capable of forecasting the growth and decay of midlatitude systems, which is remarkable in the sense that the dynamical model that relies solely on information about 500-hPa height field, the barotropic vorticity model, cannot capture such changes in intensity. This superiority provides an example of the potential for DLWP models to improve forecasting of atmospheric phenomena for which we do not have completely accurate governing equations.

More accurate governing equations are, of course, used for all operational NWP, and our best DLWP model, the "τ LSTM" variant, is much less accurate than current state-of-the-art NWP models, including the CFS benchmark. This is not surprising because our models are initialized with far less information about the current atmospheric state (one or two fields at a single vertical level) than every operational NWP model. One avenue for potential improvement of these DLWP models is extend them to include more input variables (temperature, winds, and moisture) and data at more vertical levels. Adding more inputs generally requires more training data, and it remains to be seen how much improvement can be obtained when adding additional fields while training with reliable reanalysis data sets that only go back about 60 years. One indication that this data record could be enough for substantial progress is provided by Scher and Messori (2019) who suggested that there is little benefit to using more than 100 years of GCM data to train CNNs to forecast the GCM analog atmospheres.

Important improvements might also be obtained by refining the DLWP architecture. Larraondo et al. (2019), who used deep CNNs to predict precipitation fields from geopotential height, experimented with multiple CNN architectures and found that the popular U-net (Ronneberger et al., 2015) performed best. The U-net approach can better capture scale interactions by extracting features from the first convolutional layers,

which have higher spatial resolution, and appending them at the end of the network. Off-the-shelf CNN algorithms are not optimally suited for processing global data mapped from a spherical surface, particularly near the poles. We took some steps to address these deficiencies by adding periodic boundary conditions and testing CNNs with latitude-aware convolutional filters. We also tested a weighted optimization loss function that scaled the grid points by area, but this made no significant difference to the model performance. Recently, progress has been made toward developing efficient algorithms that perform convolutions on the sphere in native spherical harmonics (Cohen et al., 2018). These spherical convolutions have shown much promise for three-dimensional rotation-invariant image processing, and we are investigating their application to global ML weather prediction.

One particularly intriguing property of our DLWP models is their extreme computational efficiency. On a consumer-grade graphics processor, training the DLWP models in this study required a one-time computation cost of 6–8 hr. Running a 72-hr forecast takes about one hundredth of a second. It is difficult to estimate how the computation required for a more complex DLWP model with many more input variables at higher resolution would compare to operational NWP models that compute time tendencies from the dynamical forcing and physical parameterizations for a large set of governing equations and integrate those equations forward over a large number of comparatively small time steps. Nevertheless, if major computational efficiencies remain after scaling up to more accurate and more complex DLWP models, those models could be particularly useful in ensemble weather forecasting. We are currently investigating whether CNNs can indeed serve as a viable ensemble weather forecasting tool.

In the early 1900s, Lewis Richardson attempted the first numerical weather forecast by manually integrating a numerical approximation to the dynamical and physical equations governing atmospheric motion (Richardson, 1922). Richardson was not successful, and the first useful weather forecasts had to await the development of electronic computers and a more sophisticated understanding of numerical methods. Those first successful weather forecasts occurred in the early 1950s and were based on the barotropic model, which has served as a benchmark in this paper (Charney et al., 1950). In a similar way, the development of ML methods capable of forecasting the weather based solely on historical weather data have had to await the development of sufficiently powerful computers and advances in deep learning. As demonstrated in this paper, DLWP models are now capable of producing weather forecasts that are far superior to those of the early 1950s. The prospects for the development of much more accurate deep learning weather forecast models, based on much more information about the initial atmospheric state, appears excellent.

Appendix A: Convolutional Algorithms

Deep NNs perform very well in image analysis tasks because of convolutional operations, which interpret spatial grids as congruent images rather than uncorrelated points. A convolutional layer learns weights for a specified number of filters with a specified grid point size and a depth equivalent to the number of input channels. These filters are then translated over the entire images to produce a new spatial grid, and generally a nonlinear activation function, such as tanh or the rectified linear unit (ReLU), is subsequently applied to the new grid.

With no special treatment of the edges of images, the application of the convolutional filters requires the edges to be cropped, resulting in a small dimensionality reduction. In image recognition tasks, this is not a problem, but when reduction is not desired, such as when applying convolutions on an entire globe to yield a new grid over the entire globe, a common remedy is to pad the images with zeroes. We apply zero padding at the pole and equator, but in the longitudinal direction, we pad with periodic boundary conditions.

Hyperparameters, which can be tuned to optimize a convolutional layer, include the number of filters, the filter size, and the dilation of the filters. We only briefly experimented with adding more filters (with no significant effect) but did try a number of combinations of filter size and dilation. The latter in particular made important performance improvements in DLWP. In a dilated filter, as shown in Figure A1, the effective size of the filter is increased without increasing the number of trainable parameters by skipping neighboring points and only using further points. For example, a 3×3 filter dilated by a factor of 2 becomes a 5×5 filter with zeros everywhere except at the center, corners, and edges (Figure A1). In DLWP, 3×3 dilated filters applied over the whole globe resulted in better performance than the full 5×5 filters. This suggests that the CNNs learned more effectively when not tasked with learning weights for extra highly correlated neighboring points.

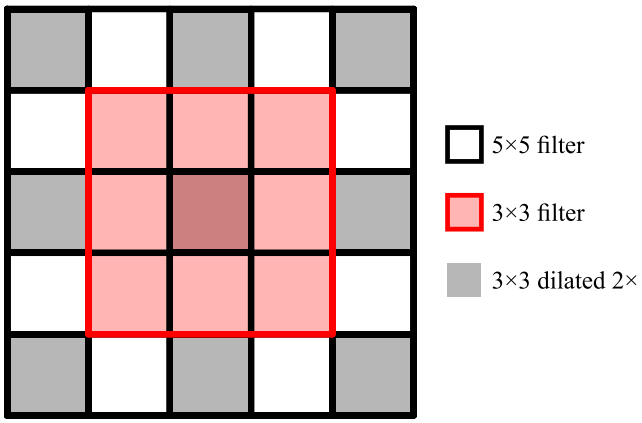

Figure A1. Illustration of the convolutional filter stencils used in Deep Learning Weather Prediction. The black outline corresponds to a 5×5 filter, while the red outline and shading corresponds to a 3×3 filter. The shaded gray squares compose a 3×3 filter dilated by a factor of 2.

Lastly, convolutional layers in image recognition often also use a technique called maximum pooling to further reduce the dimensionality of the images. This operation pools together 2×2 grids of the image into a single maximum value, thus reducing the dimensions of the image by a factor of 2 in both the height and width. The "encoder" part of the autoencoder CNN of Baldi (2012) applies this technique. To return to a fully global grid for our weather forecasting task, we apply the inverse operation: upsampling, which duplicates each value in a grid into a 2×2 box, thus increasing the dimensions of the image by a factor of 2. This is used in the "decoder" part of the autoencoder CNN.

Appendix B: The Barotropic Model

Following Charney et al. (1950), we approximate the flow at 500 hPa as nondivergent. We define $\psi(\lambda, \mu, t)$ as the streamfunction for that flow, where λ is longitude, ϕ is latitude, and $\mu = \sin\phi$. The barotropic vorticity equation (1) on the surface of a sphere of radius a can be expressed entirely in terms of ψ as

$$\frac{\partial \nabla^2 \psi}{\partial t} = \frac{1}{a^2}\left[\frac{\partial \psi}{\partial \mu}\frac{\partial \nabla^2 \psi}{\partial \lambda} - \frac{\partial \psi}{\partial \lambda}\frac{\partial \nabla^2 \psi}{\partial \mu}\right] - \frac{2\Omega}{a^2}\frac{\partial \psi}{\partial \lambda}. \tag{B1}$$

As discussed in Holton & Hakim (2013, p. 468), the preceding can be solved very efficiently if, at any time t_0, $\psi(\lambda, \mu, t_0)$ is approximated as the sum of spherical harmonic basis functions, because the computation required to compute ψ from $\nabla^2\psi$ is trivial. Our numerical solution to (B1) uses a spherical harmonic expansion truncated at T72, leapfrog time differencing with a 30-min time step and includes a scale selective ∇^4 smoother implemented with trapezoidal time differencing over a $2\Delta t$ time step. We interpolate the initial conditions from a 2.5° latitude-longitude mesh to the spherical harmonic basis at the initial time and interpolate back to the latitude-longitude mesh at output times.

Acronyms

CFS Climate Forecast System
CNN Convolutional neural network
DLWP Deep Learning Weather Prediction
GCM General circulation model
LSTM Long short-term memory
ML Machine learning
MSE Mean-squared error
NN Neural network
NWP Numerical weather prediction
rmse Root-mean-square error

Acknowledgments
J. A. Weyn and D. R. Durran's contributions to this research were funded by Grant N00014-17-1-2660 from the Office of Naval Research (ONR). J. A. Weyn was also supported by a National Defense Science and Engineering Graduate (NDSEG) fellowship from the Department of Defense (DoD). This paper was substantially improved thanks to comments by Peter Dueben and another anonymous reviewer. The CFSv2 reanalysis and reforecast data are publicly available at the link in the text. The code for this work is available at the website github.com/jweyn/DLWP. Computations were performed using Microsoft Azure computing resources granted by Microsoft Corporation's AI for Earth program.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Brevdo, Z., Citro, C., et al. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Retrieved from https://www.tensorflow.org/ (Software available from tensorflow.org).

Baldi, P. (2012). Autoencoders, unsupervised learning, and deep architectures. Proceedings of ICML Workshop on Unsupervised and Transfer Learning, 37–50.

Brenowitz, N. D., & Bretherton, C. S. (2018). Prognostic validation of a neural network unified physics parameterization. *Geophysical Research Letters*, *45*, 6289–6298. https://doi.org/10.1029/2018GL078510

Charney, J. G., Fjörtoft, R., & Neumann, J. V. (1950). Numerical integration of the barotropic vorticity equation. *Tellus A*, *2*(4), 237–254.

Chollet, F. (2015). Keras. https://keras.io

Cohen, T. S., Geiger, M., Koehler, J., & Welling, M. (2018). Spherical CNNs. ArXiv. http://arxiv.org/abs/1801.10130

Dueben, P. D., & Bauer, P. (2018). Challenges and design choices for global weather and climate models based on machine learning. *Geoscientific Model Development*, *11*(10), 3999–4009.

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. *Neural Computation*, *12*(10), 2451–2471.

Herman, G. R., & Schumacher, R. S. (2018). Money doesn't grow on trees, but forecasts do: Forecasting extreme precipitation with random forests. *Monthly Weather Review*, *146*(5), 1571–1600. https://doi.org/10.1175/MWR-D-17-0250.1

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Holton, J. R., & Hakim, G. J. (2013). *An Introduction to Dynamic Meteorology* (5th ed.). Waltham, MA: Academic Press. https://doi.org/10.1016/C2009-0-63394-8

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. ArXiv. http://arxiv.org/abs/1412.6980

Kurth, T., Treichler, S., Romero, J., Mudigonda, M., Luehr, N., Phillips, E., et al. (2018). Exascale deep learning for climate analytics. In *Proceedings of the international conference for high performance computing, networking, storage, and analysis*, IEEE Press, Dallas, Texas, pp. 51.

Lagerquist, R., McGovern, A., & Gagne, D. J. (2019). Deep learning for spatially explicit prediction of synoptic-scale fronts. *Weather and Forecasting*. https://doi.org/10.1175/WAF-D-18-0183.1

Larraondo, P. R., Renzullo, L. J., Inza, I., & Lozano, J. A. (2019). A data-driven approach to precipitation parameterizations using convolutional encoder-decoder neural networks. ArXiv, http://arxiv.org/abs/1903.10274

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436.

Leith, C. E. (1974). Theoretical skill of Monte Carlo forecasts. *Monthly Weather Review*, *102*(6), 409–418. https://doi.org/10.1175/1520-0493(1974)102h0409:tsomcfi2.0.co;2

Liu, Y., Racah, E., Prabhat, Correa, J., Khosrowshahi, A., Lavers, D., et al. (2016). Application of deep convolutional neural networks for detecting extreme weather in climate datasets. ArXiv. http://arxiv.org/abs/1605.01156

Lorenz, E. N. (1996). Predictability: A problem partly solved. In *Proceedings of a Seminar held at ECMWF on Predictability* (pp. 1–18). Reading, UK.

McGovern, A., Elmore, K. L., Gagne, D. J., Haupt, S. E., Karstens, C. D., Lagerquist, R., et al. (2017). Using artificial intelligence to improve real-time decision-making for high-impact weather. *Bulletin of the American Meteorological Society*, *98*(10), 2073–2090.

Palmer, T. (2018). The ECMWF ensemble prediction system: Looking back (more than) 25 years and projecting forward 25 years. *Quarterly Journal of the Royal Meteorological Society*. https://doi.org/10.1002/qj.3383

Rasp, S., & Lerch, S. (2018). Neural networks for postprocessing ensemble weather forecasts. *Monthly Weather Review*, *146*(11), 3885–3900.

Rasp, S., Pritchard, M. S., & Gentine, P. (2018). Deep learning to represent subgrid processes in climate models. *Proceedings of the National Academy of Sciences of the United States of America*, *115*(39), 9684–9689.

Richardson, L. F. (1922). *Weather Prediction by Numerical Process*. Cambridge: Cambridge University Press.

Rodrigues, E. R., Oliveira, I., Cunha, R., & Netto, M. (2018). DeepDownscale: A deep learning strategy for high-resolution weather forecast, *2018 IEEE 14th International Conference on e-Science* (pp. 415–422). Amsterdam, Netherlands: IEEE. https://doi.org/10.1109/eScience.2018.00130

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation, *International conference on medical image computing and computer-assisted intervention* (pp. 234–241). Cham: Springer.

Ruder, S. (2017). An overview of multi-task learning in deep neural networks. ArXiv. http://arxiv.org/abs/1706.05098

Saha, S., Moorthi, S., Pan, H.-L., Wu, X., Wang, J., Nadiga, S., et al. (2010). The NCEP climate forecast system reanalysis. *Bulletin of the American Meteorological Society*, *91*(8), 1015–1058.

Saha, S., Moorthi, S., Wu, X., Wang, J., Nadiga, S., Tripp, P., et al. (2014). The NCEP climate forecast system version 2. *Journal of Climate*, *27*(6), 2185–2208.

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. https://doi.org/10.1029/2018GL080704

Scher, S., & Messori, G. (2018). Predicting weather forecast uncertainty with machine learning. *Quarterly Journal of the Royal Meteorological Society*, *144*(717), 2830–2841. https://doi.org/10.1002/qj.3410

Scher, S., & Messori, G. (2019). Weather and climate forecasting with neural networks: Using GCMs with different complexity as study-ground. *Geoscientific Model Development Discussions*, *12*, 2797–2809.

Selz, T. (2019). Estimating the intrinsic limit of predictability using a stochastic convection scheme. *Journal of the Atmospheric Sciences*, *76*(3), 757–765. https://doi.org/10.1175/JAS-D-17-0373.1

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-k., & Woo, W.-c. (2015). Convolutional LSTM network: A machine learning approach for precipitation nowcasting, *Advances in neural information processing systems* (pp. 802–810). MA, USA: MIT Press.

Silberman, I. (1954). Planetary waves in the atmosphere. *Journal of Atmospheric Sciences*, *11*(1), 27–34.

Vlachas, P. R., Byeon, W., Wan, Z. Y., Sapsis, T. P., & Koumoutsakos, P. (2018). Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, *474*(2213), 20170844.

Welch, B. L. (1947). The generalization of "student's" problem when several different population variances are involved. *Biometrika*, *34*(1/2), 28–35.

Zhang, F., Sun, Y. Q., Magnusson, L., Buizza, R., Lin, S.-J., Chen, J.-H., & Emanuel, K. (2019). What is the predictability limit of midlatitude weather? *Journal of the Atmospheric Sciences*, *76*(4), 1077–1091.