



A real time prediction methodology for hurricane evolution using LSTM recurrent neural networks

Rikhi Bose¹ · Adam Pintar¹ · Emil Simiu¹

Received: 25 November 2021 / Accepted: 28 April 2022

© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2022

Abstract

Fast and accurate prediction of hurricane evolution from genesis onwards is needed to reduce loss of life and enhance community resilience. In this work, a novel model development methodology for predicting storm trajectory is proposed based on two classes of Recurrent Neural Networks (RNNs). The RNN models are trained on input features available in or derived from the HURDAT2 North Atlantic hurricane database maintained by the National Hurricane Center (NHC). The models use probabilities of storms passing through any location, computed from historical data. A detailed analysis of model forecasting error shows that Many-To-One prediction models are less accurate than Many-To-Many models owing to compounded error accumulation, with the exception of 6-hr predictions, for which the two types of model perform comparably. Application to 75 or more test storms in the North Atlantic basin showed that, for short-term forecasting up to 12 h, the Many-to-Many RNN storm trajectory prediction models presented herein are significantly faster than ensemble models used by the NHC, while leading to errors of comparable magnitude.

Keywords Deep learning · Long short-term memory (LSTM) · Hurricane forecasting · HURDAT2 · North Atlantic hurricanes · Recurrent neural networks (RNN) · Time series forecasting

1 Introduction

Hurricanes and tropical storms are rotating storms originating in the Atlantic basin. Tropical storms whose maximum sustained wind speeds exceed 74 mph (33 ms^{-1}) are called hurricanes. The purpose of this paper is to present a simple Deep Learning (DL) based methodology for forecasting North Atlantic basin hurricane trajectories over one or more 6-hr intervals from genesis.

The National Hurricane Center (NHC) uses several models for forecasting storm tracks and intensity. These models may be characterized as statistical, dynamical and statistical-dynamical. Examples are: the statistical

Climatology and Persistence (CLIPER) model [23]; the National Oceanic and Atmospheric Administration's (NOAA) Geophysical Fluid Dynamics Laboratory (GFDL) hurricane prediction dynamical model [18]; the statistical-dynamical model of the European Centre for Medium-Range Weather Forecasts (ECMWF) ensemble prediction systems (EPSs) [4, 25].

DL methods have recently taken giant strides in using large amounts of data to make accurate predictions in fields such as computer vision [26] that have traditionally been very challenging for conventional statistical models. We propose a DL approach for efficient prediction of hurricane trajectories over several 6-hr time intervals. The target variables in the DL hurricane track prediction problem are the coordinates of a storm's center. The DL models include as inputs storm features available in or derived from the hurricane database maintained by the NHC.

Hurricane forecasting using data-based approaches have mainly been formulated as image processing problems. Giffard-Roisin et al. [9] used atmospheric flow velocity and pressure information at several heights above the Earth's surface to train a model consisting of several convolutional

✉ Rikhi Bose
rikhi.bose@nist.gov

Adam Pintar
adam.pintar@nist.gov

Emil Simiu
emil.simiu@nist.gov

¹ Engineering Laboratory, National Institute of Standards and Technology, 100 Bureau Dr, Gaithersburg 20899, MD, USA

and feed-forward neural network layers; a mean error of 115 km for 24-hr forecasts was obtained for test data over the North Atlantic. A Convolutional Long Short-term memory (ConvLSTM) model, was used by [15] to extract spatio-temporal information from a large database of instantaneous atmospheric conditions recorded as a pixel-level history of storm tracks. The ConvLSTM consists of Convolutional Neural Network (CNN) layers and layers of Long short-term memory (LSTM) [11]. A tensor-based Convolutional Neural Network (TCNN) was used by [6]. Rüttgers et al. [27] utilized the Generative Adversarial Network (GAN) to forecast storm trajectories using satellite images. They used image time-series of typhoons in the Korean peninsula for model training. The GAN model was tested on 10 storms unseen by the model during training. The average prediction error for 6-hr forecasts for the test storms was 95.6 km. More recently, motivated by the work of [3, 9] developed storm forecasting models utilizing reanalysis of data extracted during tropical storms. Several DL models, such as CNNs, GRUs, Transformers [29], as well as ML models, e.g., XGBoost [5], were utilized. They have reported a minimum mean 24-hr forecast error of ~ 110 km for 2016–2019 Atlantic test storms. However, the aforementioned models are data-hungry and in most instances require current satellite images, or in some instances atmospheric post-storm reanalysis data that are not instantaneously available, and may be affected by non-negligible errors themselves. For these reasons, the use of such data is not advantageous in a forecasting context. This was noted by some of the aforementioned researchers themselves. Boussieux et al. [3] in their paper note that ‘... *although our models can compute forecasts in seconds, the dependence on reanalysis maps can be a bottleneck in real-time forecasting.*’

In [17], a flexible sparse RNN architecture was used to model the evolution of North Atlantic hurricane tracks. Available time records of the target storm were initially compared with those of other storms, and a Dynamic Time Warping technique was used to assign similarity scores to each storm available in a historical storm database. Only storms with high similarity scores were chosen for model training. Storm evolution prediction was thus improved by incorporating historical trends. However, to obtain appropriate storms for the model to train on, significant numbers of time records from the target storm are required. In the present work, we account for translation trends of historical storms by using storm displacement probabilities as input features. The storm displacement probabilities take into consideration each 6-hr displacements of all storms from a given computational cell to adjacent cells in a coordinate-transformed domain.

In another attempt to use RNN models, [1] used a grid-based Neural Network approach for hurricane trajectory

forecasting. A Long Short-term Memory (LSTM) RNN model predicted storm-wise scaled grid numbers in the 2-D latitude–longitude domain. The grid-based prediction scheme was based on the fact that, for a historical storm, the distance being traveled is proportional to the number of time records available for it. The model forecasts a storm’s location 3 h in advance, and is claimed to improve upon the forecasting performance of the models developed in [17]. They used storm-wise data scaling, which renders predictions for new storms impossible. Moreover, no results were provided for long-term forecasts. The compounded error accumulation, a most pressing issue related to the application of NNs for storm track prediction, was not considered in their work. In the present effort, we use LSTM-RNNs to provide long-term forecasts for storm evolution up to 30 h in advance. We demonstrate the compounded error accumulation problem and a remedy for it using Many-To-Many prediction-type RNNs.

In a more recent work, CNNs coupled with Gated Recurrent Unit (GRU) RNNs were employed by [22]. The use of a feature selection layer before the NN layers augmented the model’s learning capability from the underlying spatial and temporal structures inherent in trajectories of tropical cyclones. Their model utilized both CNNs and GRUs. They did not segregate the training, validation and test storms; 6108 data sequences were used for testing the model. The model’s performance was compared with the performance of other numerical and statistical weather forecasting models. The NN model used in [22] was more accurate than the statistical model used in [13] and a numerical model used in [7] for long-term forecasts. In particular, the 72-hr track forecast error was less than half the errors reported for traditional track forecasting models. However, they reported a 12-hr forecast error of ≈ 100 km (62 mi) between the predicted and true cyclone-eye locations, which is comparable or marginally less than the errors incurred by the aforementioned statistical and dynamical models. The RNN models presented herein were tested for hundreds of validation and test storms, and the prediction errors were extensively analyzed. Computed over different splits of the HURDAT2 database, mean 6- and 12-hr forecast errors for the prediction models were ~ 33 km and 72 km, respectively. To our knowledge, the models outperform all data-based models for short-term trajectory forecasting up to 12 h.

In spite of the substantial existing work on the topic, there exist opportunities for improvement, specifically in the context of formulating a model usable in real-time that uses readily available information. In this regard, the use of historical storm trajectory information is the most suitable as these are easily available. We find herein that the use of 6-hr displacement probabilities to have predictive power. The storm displacement probabilities embed

historical storm translation trends by considering the 6-hr displacements of all storms from a given location. Results from the RNN-DL models presented herein demonstrate that the use of this simple input variable is able to provide results that are competitive in accuracy with state-of-the-art models using large volumes of atmospheric flow field/satellite image data. The flexible model forecasting scheme can predict trajectories even when only one record is available, i.e., from genesis, unlike most ML/ DL based models developed so far. Furthermore, the current work emphasizes that traditional ML/ DL accuracy measures are not applicable to storm trajectory forecasting, especially for long-term forecasts that are prone to compounded error accumulation. The error analysis of the presented models highlight this pressing issue and suggests the use of Many-To-Many type prediction architectures in RNNs that can defer the compounded error accumulation.

The paper is structured as follows. In Sect. 2, the database used for model development, is discussed from the statistical, the feature engineering, and the model formulation points of view, and the methodology used for the calculation of 6-hr storm displacement probabilities is described. Section 3 discusses the model type, its architecture and implementation, training strategies, and hyperparameter tuning. Section 4 contains the prediction results of the trained models and an extensive analysis of forecasting error, compares the predicted trajectories with trajectories of historical test storms, and discusses limitations of the models developed herein and the scope for future improvements. Conclusions are drawn in Sect. 5.

2 Database and feature engineering

See Fig. 1.

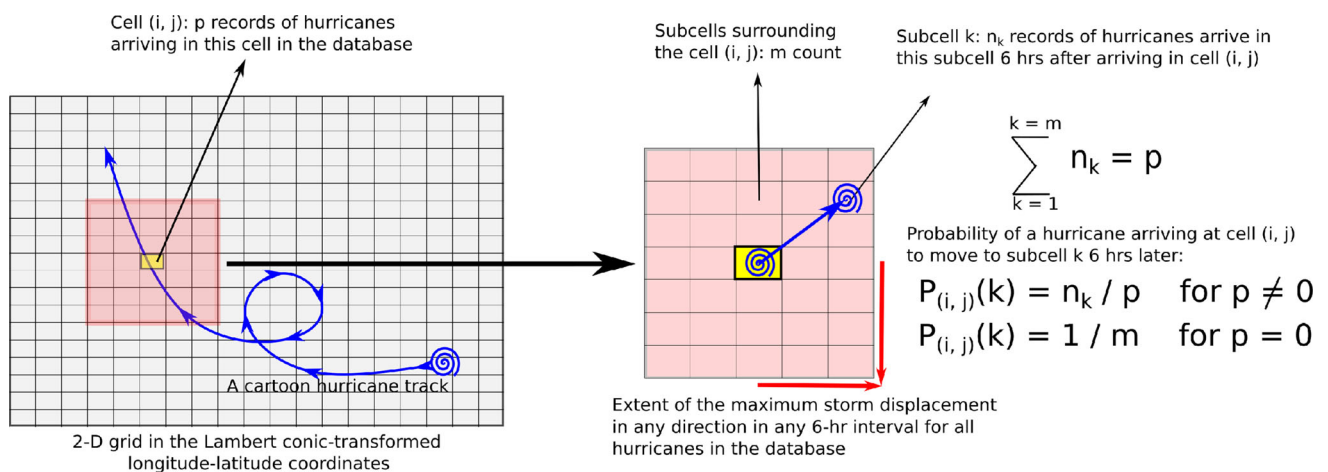


Fig. 1 Schematic depicting the calculation of displacement probabilities associated with each cell in the grid domain

2.1 Feature space

The NHC conducts post-storm analyses and maintains the HURDAT2 database [12, 20] that lists features of Atlantic-basin storms at 6-hr intervals (e.g., date and time, landfall/intensity status, latitude and longitude, maximum 1-min wind speed at 10 m elevation, central pressure). However, only part of the database is useable for DL purposes. The present models use a maximum of five time records as input. Of the 1893 listed storms, 1736 storms have at least seven time records with maximum wind speeds listed for all records. Only 560 of these have both the central pressure and maximum wind speed included for all records. The central pressure is highly correlated with the maximum wind speed; in fact, the observed maximum wind speed varies approximately linearly with the central pressure with a negative slope for all time records of the aforementioned 560 storms in HURDAT2. Therefore, the central pressure was excluded from the input feature space which allowed us to use a much larger volume of data without losing much useful predictive information. We used the latitude (ϕ), longitude (λ) and maximum wind speed (w_m) to construct the input feature space.

To reduce the effect of areal distortion associated with the spherical coordinate system at higher latitudes, a transformed pair of x- and y-coordinates obtained by Lambert’s conic conformal (LCC) projection [19] was used instead of ϕ and λ . LCC projection has superior properties at mid-latitudes. Two standard parallels associated with the conic projection (ϕ ’s where the cone passes through the sphere) used in the current transformation are 33°N and 45°N. Most of the U.S. mainland is contained within these standard parallels. No distortion occurs along the standard parallels but the distortion increases away from them. In addition, we use the storm translation direction and speed as inputs, as these are important for storm intensity

modeling [8, 28]. Six-hour-averaged translation speed (V) and direction (θ) are calculated going backward in time. The Haversine formula gives the distance (d) between storm locations at two consecutive time instants.

$$V_l = \frac{d(\phi_{l-1}, \phi_l, \lambda_{l-1}, \lambda_l)}{\Delta t \equiv 6hrs} \quad (1)$$

$$\theta_l = \tan^{-1} \left(\frac{\phi_l - \phi_{l-1}}{\lambda_l - \lambda_{l-1}} \right) \quad (2)$$

In HURDAT2, w_m is approximated to the nearest 10 kt (5.14 ms^{-1}) between 1851 and 1885 and to the nearest 5 kt (2.57 ms^{-1}) thereafter. Binned w_m is used from the HURDAT2 database. Both w_m and V were expressed in ms^{-1} units.

It is necessary for the input features to contain information associated with trends of past storm motion [17, 24]. Those trends are encapsulated in 6-hr storm displacement probabilities (d.p.) computed from the historical storms, and are included as input features at each time instant. A schematic depicting the calculation of d.p. is shown in Fig. 1. At first, the 2-D domain bounded by the extents of the x - and y - coordinates (the extents are the maximum and minimum of each coordinate corresponding to any storm record in the considered database) was decomposed into rectangular computational cells. Consider all storms for which a record is contained in any one of those computational cells. For those storms, the maximum number of cells traversed by any storm in any 6-hr interval in the x - and y - directions are m_x and m_y , respectively. Therefore, associated with any given cell (i, j) (colored yellow in Fig. 1) is a set of $m = (2m_x + 1)(2m_y + 1)$ cells within which all 6-hr displacements of any historical storm are contained. Such a set is colored red in Fig. 1. Assume, that there are p records of hurricanes that arrived at the (i, j)th cell and then transitioned to the k th associated cell in the next 6 .

If this leads to n_k records being sampled at the k th associated cell, $\sum_1^m n_k = p$. Therefore, the displacement probability of a storm arriving at the (i, j)th cell and transitioning to the k th associated cell in the next 6 h is,

$$p_{(i,j)}(k) = \begin{cases} \frac{n_k}{p}, & \text{if } p \neq 0 \\ \frac{1}{m}, & \text{if } p = 0 \end{cases} \quad (3)$$

For a fine grid the number of cells with $p \simeq 0$ would be larger than for a coarse grid. For a finer grid (and/ or for a very thin upper tail of the distribution of V), m would increase, and n_k would decrease. In addition to an increased number of features at each instant, d.p. from a cell to adjacent cells could be more biased on a specific historical storm's displacement, which is not desirable for the

prediction of new storms. On the other hand, if the grid is too coarse, storm motion trends reflected by the d.p. could be obscured.

2.2 Statistics

The computational domain containing the 1736 storms with at least seven records, including w_m , covers $\phi \in [7.4^\circ\text{N}, 81^\circ\text{N}]$ and $\lambda \in [109.5^\circ\text{W}, 63^\circ\text{E}]$. The storm translation speed, which does not exceed $V = 40.5 \text{ ms}^{-1}$ for any storm in the database, determines the number m of subcells associated with a given computational cell as shown in Fig. 1. Histograms of ϕ , λ and V are shown in Fig. 2. Only for 33 of the 47,880 time records was $\phi > 70^\circ\text{N}$, and only for 26 records was $\lambda > 10^\circ\text{E}$. Only 93 records of storm motion satisfied the condition, $V > 25 \text{ ms}^{-1}$. Therefore, the 71 storms violating these limits were excluded; 1665 storms remained in the final database used for model development. The coordinate limit $\phi = 70^\circ\text{N}$ is chosen because it is even north of the Arctic circle. The limit for longitude, $\lambda = 10^\circ\text{E}$ ensures that storm records from all of the Atlantic ocean in the Northern hemisphere upto the western shores of the continent of Africa are retained in the database. We chose $V \leq 25 \text{ ms}^{-1}$ to limit the magnitude of m .

Storm d.p. were calculated based on the evolution of these 1665 storms. 61 grid points were used in both x and y directions, resulting in 3600 computational cells decomposing the whole 2-D LCC projected domain. Despite application of the three above-mentioned criteria, for 1687 out of the 3600 computational cells, $p = 0$. The maximum 6-hr displacement for any storm in the 1665 storm-database could be captured with $m_x = m_y = 4$. So, the d.p.s were calculated for $m = 9 \times 9 = 81$ surrounding cells for each computational cell. Consequently, the number of input features for each time record was 86 (LCC transformed x and y coordinates, V and θ , w_m , and historical 6-hr d.p. calculated at 81 subcells associated to the cell containing x and y).

3 Model development

Once a database was obtained, we proceeded to the model development. Both classification and regression problems may be formulated. In the current work, only the regression problem is considered for which we used the LSTM RNN [11].

In the supervised regression problem being considered, inputs to the models are the features at the chosen number of input time instants, n ($86 \times n$ for each prediction). The model outputs are the storm position(s) x and y at 6-hr

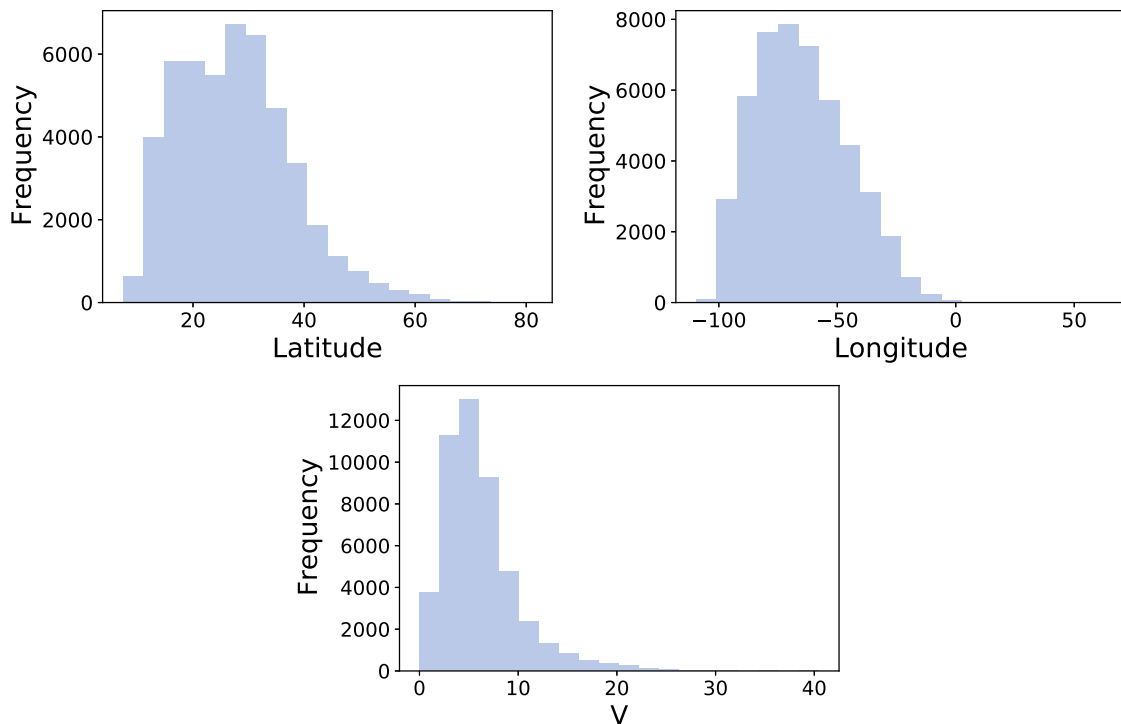


Fig. 2 Histograms of latitude (ϕ) and longitude (λ) in degrees and storm translation speed (V) in ms^{-1} . Positive values of ϕ and λ represent the northern and eastern hemispheres, respectively. 1736 storms from the *HURDAT2* database are used (see text)

intervals for the chosen number of output timesteps. The model performance is measured by the loss function defined as the mean squared error (*m.s.e.*) between predicted and true position(s) of N samples (N is the number of data sequences used from the storms in the database),

$$m.s.e. = \frac{\sum_{i=1}^N (x_{predicted,i} - x_{true,i})^2 + (y_{predicted,i} - y_{true,i})^2}{N}.$$

Based on the gradients of the *m.s.e.*, model weights are updated so that the *m.s.e.* is minimized.

3.1 Model architecture and implementation

RNNs extract pattern and context from sequences. *HURDAT2* contains sequences of storm time records, and therefore, RNNs are applicable. An LSTM-RNN unit is used as the base architecture for the models developed herein (see Appendix 1 for a brief description of the LSTM architecture). Early use of LSTMs in weather forecasting is reported in [10].

Models trained on varying numbers of input time records (n) can predict one or several time steps at once based on their architecture. Both Many-To-One (*M2O*) and Many-To-Many (*M2M*) type prediction algorithms have been used. Upon processing a sequence with n time records, the *M2O* models forecast the storm locations at

one time instant, while the *M2M* models used here output n time records. From here onwards, models are named as *M2On* or *M2Mn* to reflect their architecture. As each timestep in the training database \equiv 6-hr, the *M2On* and *M2Mn* models forecast 6 and 6n h at once, respectively. The features inputted to the models and the outputs at each time step for both *M2On* and *M2Mn* models are tabulated in Table 1.

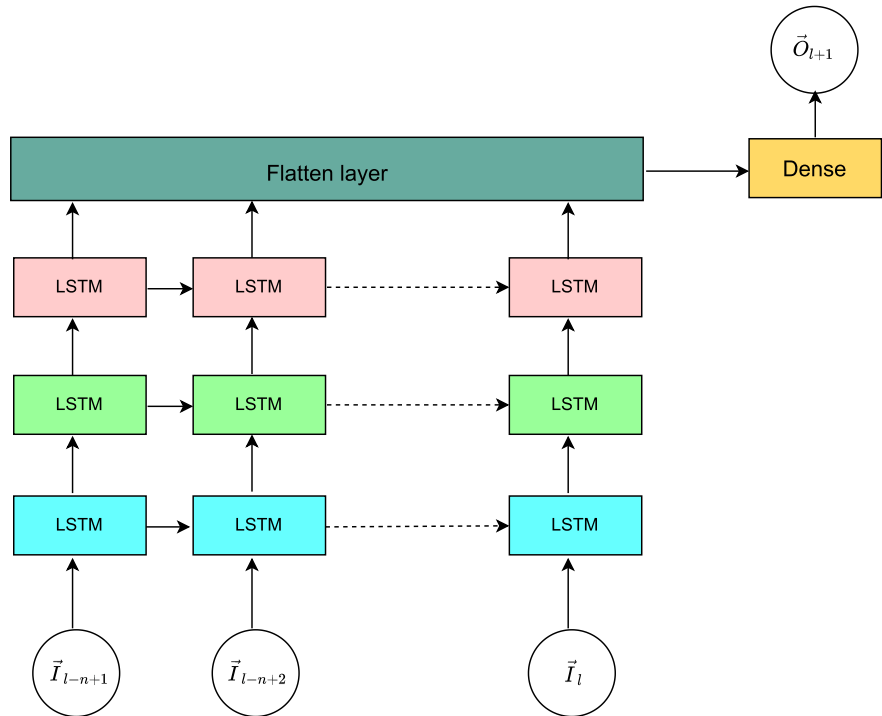
Schematic diagrams of *M2On* and *M2Mn* prediction algorithms are shown in Figs. 3 and 4, respectively. For both sets of models, the output layer is a dense layer. In addition to the input and output layers, the *M2On* models presented here have three layers of LSTM units. The numbers of neurons in the three LSTM layers were 128, 32 and 8, respectively. In preliminary tests, the performance of converged models did not improve with additional LSTM layers. Two bi-directional LSTM layers were used at either side of a repeat-vector layer for the *M2Mn* models (Fig. 4). Each LSTM unit in a bi-directional layer had an output dimension of 64; so 128 neurons were used in each layer. The numbers of trainable parameters for all the tested models were between 3 to 6 times the number of data sequences used for training.

To increase model robustness, dropout is used to reduce a model's bias towards the training data. A dropout value of 0.1 was used for each hidden layer. The Adam algorithm [16] was used for model optimization. Finally, the models

Table 1 Storm features inputted to the models at the l th time step for a storm located in the (i, j) th cell of the 2-D LCC-transformed computational domain

Model	Input features (\vec{I}_l)	Time records	Features per step	Output features (\vec{O}_j)	Output records
$M2On$	x_l (LCC), y_l (LCC), V_l , θ_l , $w_{m,l}$, $p_{(i,j),l}$	$l = 1, 2, \dots, n$	86	x_j (LCC), y_j (LCC)	$j = n + 1$
$M2Mn$	x_l (LCC), y_l (LCC), V_l , θ_l , $w_{m,l}$, $p_{(i,j),l}$	$l = 1, 2, \dots, n$	86	x_j (LCC), y_j (LCC)	$j = n + 1, n + 2, \dots, n + n$

Fig. 3 Schematic of the $M2On$ prediction algorithm. The input and output at the l th time step are \vec{I}_l and \vec{O}_l , respectively



were implemented in the Keras API with TensorFlow as the backend library. Further details on the model architectures used herein, their implementation and additional considerations for their training are provided in [2].

3.2 Number of input time records (n)

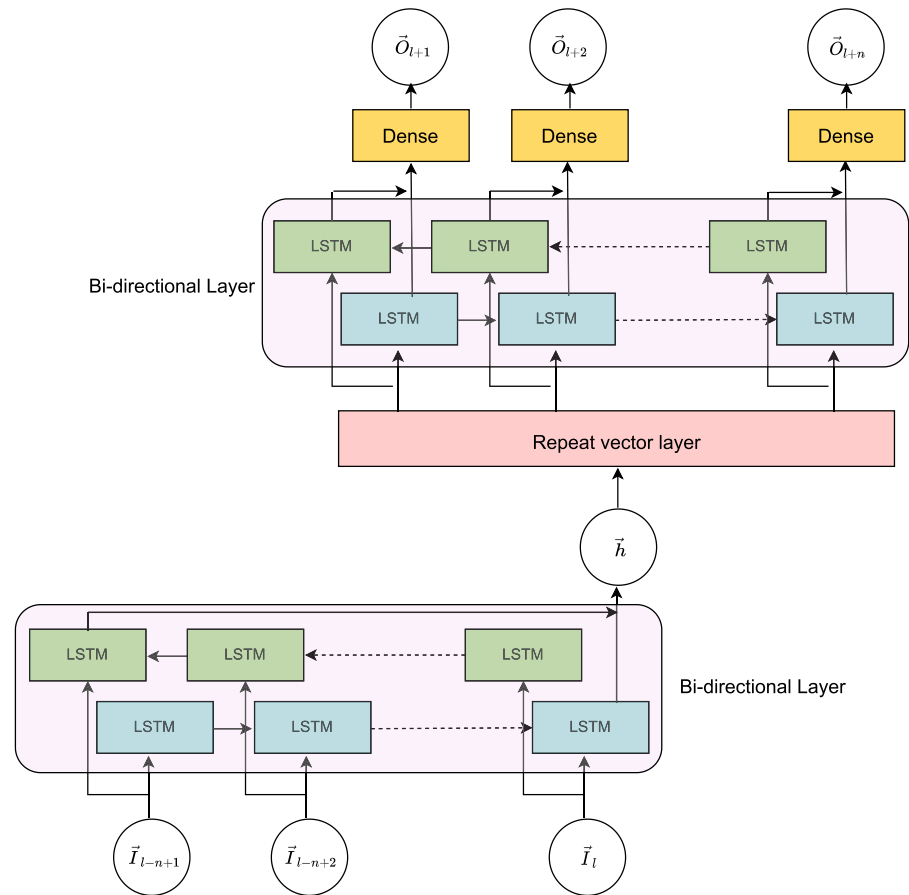
An important consideration is the number of input time instants, n that could minimize the model prediction error. Intuitively, increasing n should make the model predictions more accurate. However, high values of n imply less available time for preparation of a possible landfall. Also, the value of n implicitly determines the number of data sequences available from the storm database for model training/ validation/ testing. Preprocessing, such as zero padding may be used at early stages of a storm’s life span (available time instants $< n$). However, training of a model with zero-padded data is inefficient, because the mode of

the number of records for all storms in the HURDAT2 database is 13, while storms with a very long lifespan, e.g., as long as ~ 80 time records, are relatively rare. If one chose to use zero padding so that each storm has, for example, 50 records, a significant number of time records fed to the network for training would be zeros. On the other hand, requiring 12 time records in each input sequence, for example, renders the zero padding unnecessary as only a few storms have a lifespan shorter than 13 time records in the database (12 input records + 1 output). Both $M2On$ and $M2Mn$ LSTM models were developed for a range of n , between 1 and up to 5 time instants ($n_{max} = 5$).

3.3 Data scaling

NNs perform better when the input data are scaled to the interval $\in [0, 1]$. Normalization of the whole database was preferred over a storm-wise feature scaling approach [1].

Fig. 4 Schematic of the *M2Mn* prediction algorithm. The input and output at the *l*th time step are \vec{I}_l and \vec{O}_l , respectively



The storm-wise data scaling renders prediction of a new storm impossible, because the relevant scaling parameters for a feature from an entirely new storm are unknown a priori. Each feature *f* is scaled using the Min-Max scaler as, $f_s = \frac{f-f_{\min}}{f_{\max}-f_{\min}}$, so that the scaled feature $f_s \in [0, 1]$. This scaling is preferred over standardization, because, the d.p. also belong in this interval. In contrast to a storm-wise scaling approach, feature scaling for an entirely new storm is well defined.

3.4 Sequence generation

To generate a sequence for the *M2M5* model, the life span of a storm must be at least 10 time-record long. 1514 out of the 1665 storm database with at least 10 time records were used to facilitate comparison between the *M2On* and *M2Mn* models, which must be trained, validated and tested on the same set of storms.

To avoid training and testing on the same storm, the 1665 storms were partitioned into training, validation, and testing sets. Test storms (5% of the total) always contained a set of five important historical storms chosen a priori because of their destructiveness upon landfall and of the complexity of their trajectories. The rest of the storms were

chosen randomly from the database without replacement. Of the 1665 storms considered, ~80% were used for training the model while 15% storms were used for validation. Data sequences for training, validation and testing were generated separately from the segregated lists of storms. The training data always contained ~24,000 sequences or more.

3.5 Learning rate and batch size

In Keras, the default learning rate for the Adam optimizer is 0.001. At this value, the validation loss was prone to sudden jumps in and around the optimal valley's minimum. We reduced the learning rate by an order of magnitude every 250 epochs from an initial learning rate of 0.0001 until the models stopped improving.

We used a batch size of 32 which significantly lowered training time in the Graphic Processing Unit (GPU) clusters. A model checkpoint was used after completion of each epoch to compute the validation error and store the weights in case the error obtained is the minimum at that point. At the end of model training, model weights that yield the lowest loss function value for the validation sequences.

4 Results

4.1 Forecast error: compounded error accumulation

The models were tested on 228 validation and 75 test storms to compute the average error in distance between the predicted and the true positions of a storm's eye at each prediction step. To facilitate comparison, for the results presented herein, both $M2On$ and $M2Mn$ models were trained, validated and tested on the same set of storms. Average 6-hr and $6n$ -hr forecast errors for the test sequences are shown in Fig. 5 for all models. The errors for the validation sequences are very similar and are not reported herein. Figure 5a shows the mean 6-hr forecast error plotted for different input time records (n). Overall, the $M2M$ models are more accurate. The $M2M2$ model performs best of all models for 6-hr forecasting; mean error computed for the validation and test storms are 34.2 km and 30 km, respectively. This is in agreement with [8], who noted a near linear statistical relationship between time-rate changes in translation variables at current and previous timesteps implying that the use of two time records may be optimal for 6-hr forecasting. Although the number of trainable parameters for all models is similar, the $M2On$ models only slightly improve with increase in n . It must be noted that the mean errors shown are for only a single split of the storm database. The $M2Mn$ model predictions worsen slightly with increase in $n > 2$. However, a maximum

difference of 5 km in mean 6-hr forecast error between the $M2On$ and $M2Mn$ models is obtained for $n = 2$.

For a given number of available time records for a storm, which model predicts a storm's future trajectory more accurately over longer time intervals? Figure 5b reports the mean forecasting errors for up to 30 h computed for the same test storms as in Fig. 5a. While the $M2Mn$ models can forecast n time records (i.e., $6n$ h) at once, the $M2On$ models predict only one time record. To obtain long-term forecasts with the $M2On$ models, the predicted record must be used as an input for the next time instant and so on. The forecasting errors for up to 5 time steps in advance obtained in this manner are also reported for all $M2O$ models. While the LCC projected x - and y - coordinates, V and θ and the 6-hr storm d.p. are updated at each time step, the true value of w_m is used from the database. This provides a fair comparison between $M2O$ and $M2M$ model accuracies in predicting several future time steps. This prediction scenario is similar to a real-time storm trajectory forecast.

Due to the aforementioned time record updating procedure for the $M2O$ models, the 12, 18, 24 and 30-hr forecasts are prone to compounded error growth. Error incurred at each prediction step worsens the forecasting accuracy at future steps as the predicted coordinates which deviate from the truth are used as input for predicting the future storm locations. Due to the error accumulation, the $M2O$ model forecasts beyond the 1st prediction step worsen for increasing n , as for a higher n more erroneous input time records are fed to the model for forecasting. All $M2M$ models have similar accuracy over each 6-hr interval.

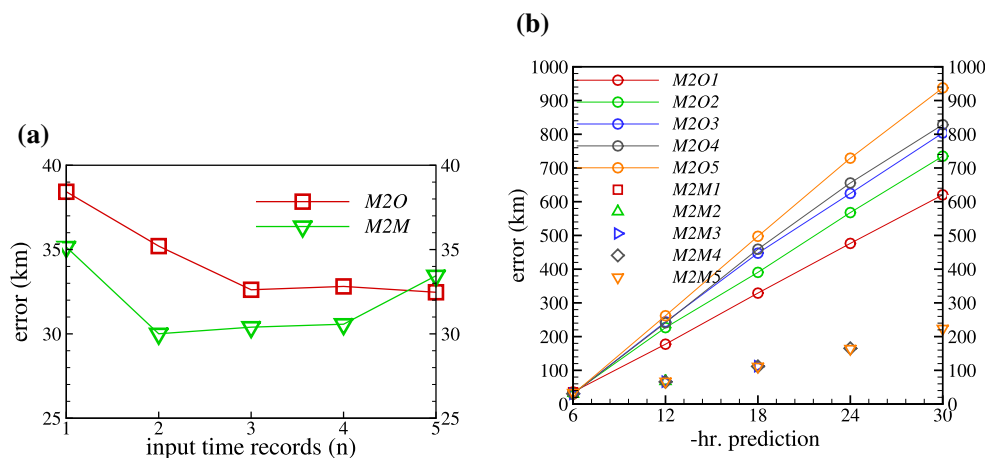


Fig. 5 In the following discussion, models have been named $M2On$ and $M2Mn$, where, $n = 1, 2, 3, 4$ and 5 represents the number of records a model takes in as input. The $M2On$ and $M2Mn$ models forecast 6 and $6n$ h at once, respectively. Mean (a) 6-hr forecast error and (b) 6, 12, 18, 24 and 30 -hr forecast errors in distance computed on test storms for $M2O$ and $M2M$ models trained on different number

of input time records. In (b), errors from the $M2O$ models are represented by circular symbols and the $M2M$ models are represented by other types of symbols; models trained on a given number of time records are represented by symbols of same color. All models were trained, validated and tested on the same set of storms

Therefore, given an initial time sequence to predict a storm's evolution, especially to predict it several hours in advance, *M2M* models are more reliable. The average 12-hr (24-hr) forecasting error for test sequences for the *M2M4* and *M2M5* models are 65.9 km and 66.85 km (165.2 km and 163.9 km), respectively.

4.2 Displacement probabilities

The advantage of using the 6-hr d.p. as input features is demonstrated by reporting mean errors in distance for the *M2M2* model trained on input features including, and excluding, the 6-hr d.p. Three different splits of the database for train, validation and test storms are used which have been also used later herein for benchmarking. The NHC produces forecasting error estimates for specified time periods, e.g., for the periods 2005–2009, 2011–2015, and 2016–2019. The three different test databases were created by including all storms from one of the time periods with at least 10 time records and then selecting the rest of the 75 storms at random. The same sets of training, validation, and test storms are used for training models with/ out d.p. as inputs for comparison. The results are shown in Fig. 6. Use of the d.p. is clearly advantageous and significantly improves model performance. The maximum gain in accuracy is obtained for the period 2011–2015; using d.p., mean 6-hr and 12-hr forecasting errors are reduced by ~ 16 km and 23 km, respectively. The accuracy gain for the chosen test datasets is at least ~ 11 km for 6-hr forecasts and 17 km for 12-hr forecasts.

4.3 Storm trajectory forecasting

Figure 5 shows forecasting errors of the LSTM-RNN models computed over hundreds of validation and test storms. How do these models perform for individual test storms? This section reports trajectory predictions by these models for four extraordinarily powerful test storms: Andrew (1992), Ivan (2004), Sandy (2012) and Harvey (2017). Andrew, Ivan and Harvey were generated in the tropical Atlantic sub-basin; their trajectories exhibit the trend of trajectories of most storms generated in that sub-

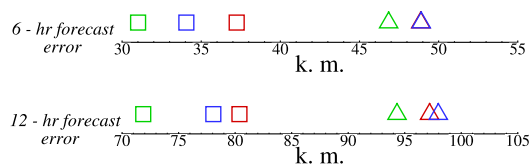


Fig. 6 Mean 6-hr (top) and 12-hr forecast errors for the *M2M2* models trained on input features including 6-hr displacement probabilities (\square) and excluding those (Δ). The test storms are in different time intervals: 2005–2009 (red), 2011–2015 (green), and 2016–2019 (blue)

basin. Hurricane Ivan was chosen because it circled back across its original track near the end of its lifespan. The trajectory of hurricane Harvey has a sharp change toward south-east, opposite to the general trend. Hurricane Sandy was generated in the Caribbean sea. Its overall motion is consistent with the general trajectory trend of storms generated in the Caribbean. However, at the very beginning of its lifespan, it moves southward before turning sharply to travel northward. The aforementioned trajectory trends are complex compared to the general trends of storm motion. Therefore, these storms were chosen for demonstrating the performance of the models.

Figure 7 shows the 6-hr forecasts by the *M2On* models for the four hurricanes. Forecasting error was computed at all prediction instants and the mean forecasting error (per prediction step) was computed over the entire trajectories which are reported for all *M2O* models. Exact features were used as inputs to the model. The *M2O1* performs the worst with average errors of ~ 47 km, ~ 52 km, ~ 50 km, and ~ 34 km, for hurricanes Andrew, Ivan, Sandy, and Harvey, respectively. Model performance significantly improves for $n > 2$. Among these three models, despite their complex trajectories, the maximum of the mean 6-hr forecasting errors computed for each of these storms were ~ 28 km, 41 km, 39 km and 29 km for the hurricanes Andrew, Ivan, Sandy, and Harvey, respectively.

Trajectories of the four aforementioned storms predicted by the *M2Mn* models are shown in Fig. 8. To obtain predictions by the *M2Mn* model for a given storm, exact input features at n instants have been fed to the model to predict the storm locations at next n time instants. So, the input time sequences are as $l = (0 \rightarrow n - 1), (n \rightarrow 2n - 1), (2n \rightarrow 3n - 1) \dots$. Corresponding output sequences are $l = (n \rightarrow 2n - 1), (2n \rightarrow 3n - 1), (3n \rightarrow 4n - 1) \dots$. Therefore, each model predicts a target time record only once. Mean error incurred at each prediction step computed over the whole trajectory is reported for each model. For models with $n > 1$, the reported error for each model is the mean forecasting error per prediction, which is different from the mean $6n$ -hr forecast errors in Fig. 5b. Therefore, although the forecasting error for the *M2Mn* model is highest at prediction step n , the reported error in Fig. 8 takes into account the model's more accurate predictions at the earlier prediction steps. The computed mean error per prediction step is smallest for Harvey and largest for Sandy. Mean errors computed over 12 and 24 h forecasts for the *M2O2* and *M2O4* models for hurricane Harvey are 38.76 and 76.2 km. For hurricane Sandy these are 63.8 and 128.2 km, respectively.

A hybrid approach is proposed for real-time storm evolution forecasting with the current models. When one record is available for a storm, *M2O1/M2M1* is used for prediction. When records of two time instants are available,

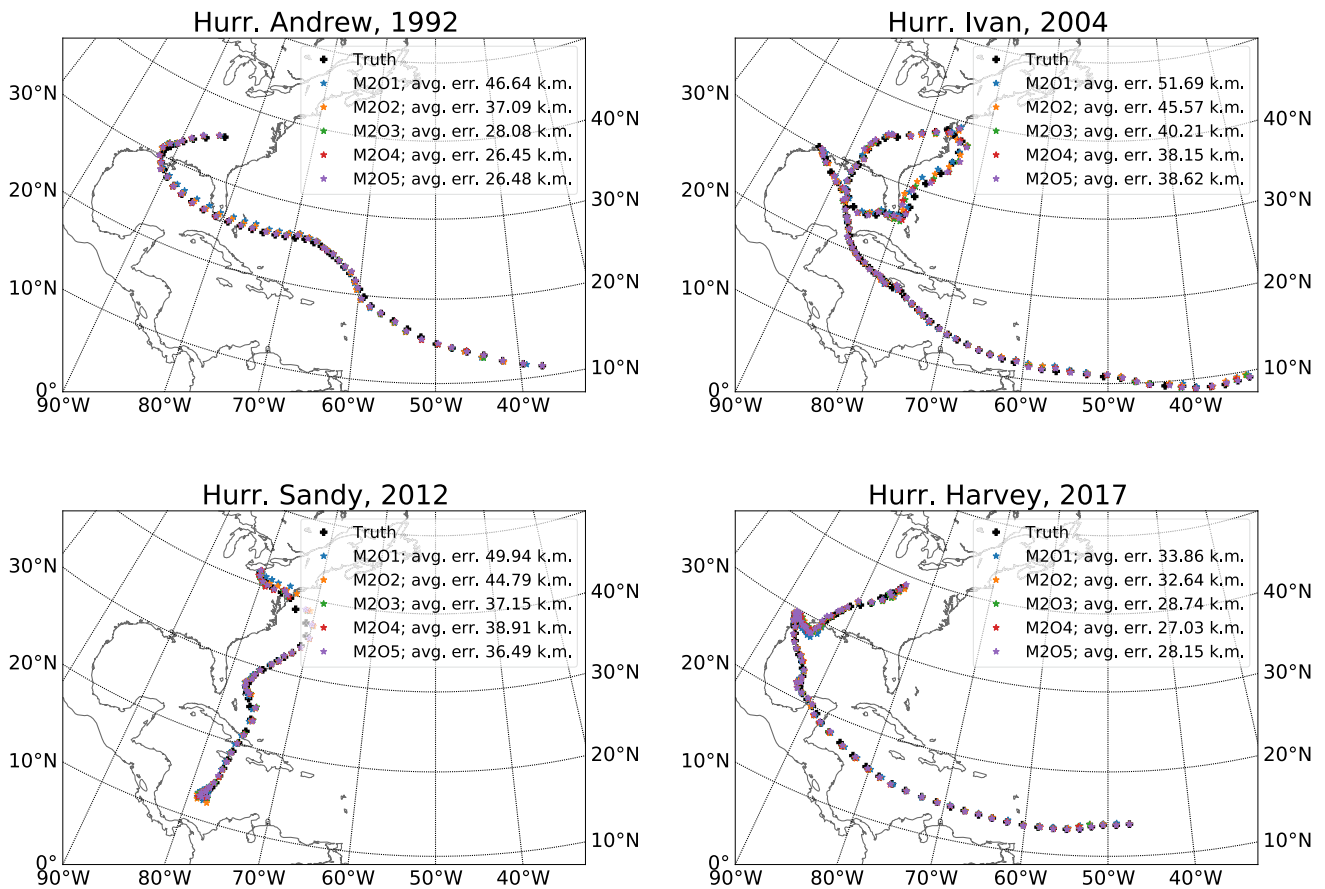


Fig. 7 6-hr forecast of select hurricane trajectories; Black '+' symbols represent the true locations; Colored '*' symbols represent the $M2O_n$ model forecasts. Inputs to the model are the true feature data at each timestep. Average error computed over the whole trajectory is also given

a model trained on $n = 2$ may be used. Similarly, models trained on increasing n can be used, up to the point where 5 time records are available, beyond which the remaining span of the storm's evolution is predicted by the model with $n = 5$.

4.4 Benchmarking

The literature on DL/ ML models applied to this problem does not use a fixed set of storms for comparison/ benchmarking. Rather, the NHC's forecasting error estimates given by the 2/3-probability circle radii for the periods 2005–2009, 2011–2015, and 2016–2019 are used. Models were trained keeping all storm events from these time frames in the test dataset as in Fig. 6. The trained models were evaluated on test sequences generated from storms in those time periods only, and the respective 2/3-probability circle radii were computed (after sorting the errors in ascending order, errors for 2/3 of all sequences are less than the 2/3-probability circle radius). The radii are listed in Table 2. $M2M$ model forecasts are more consistent over different 5-y periods than the NHC forecast errors. In addition, the 2/3-probability circle radii are competitive

with the NHC's 12-hr forecast errors. In view of the small number of features it uses, the performance of the displacement probability approach is rather remarkable. Although the radius for the 24-hr forecasts is significantly larger for the $M2M$ models, interestingly, for the period 2016–2019, the mean and standard deviation of the 24-hr forecast errors for the $M2M5$ model are 198 km and 153 km, respectively. These are similar to the forecasting errors for the CLIPER statistical model used by the NHC (mean 201 km, std 149 km; see Table 4 of [3]).

4.4.1 Limitations

For real-time forecasting of storm trajectories, only the available time records from the storm's current and past status may be used. Previous discussions on the performance of the LSTM-RNN models are based on a Machine/ Deep Learning perspective on the best/ ideal performance of the models for the problem being considered. The models were only fed the true features and storm locations were predicted up to 30 h in advance. However, longer forecasts of storm trajectories are typically necessary for disaster management purposes. To test these models'

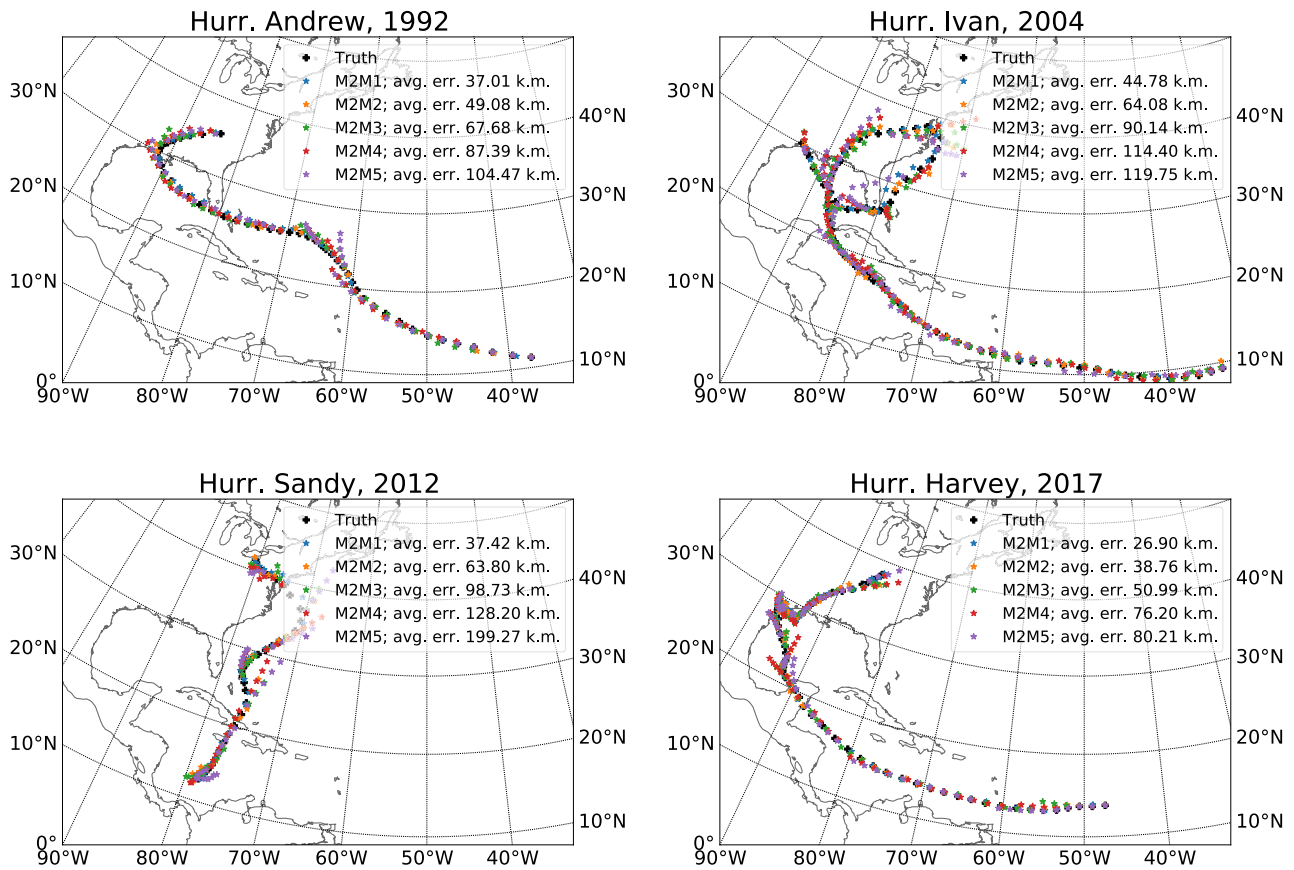


Fig. 8 6n-hr forecasts of select hurricane trajectories; Black '+' symbols represent the true locations; Colored '*' symbols represent the *M2Mn* model forecasts. Inputs to the model are the true feature data at each timestep. Average error computed over the whole trajectory is also given

Table 2 Comparison of the 2/3-probability circle radii corresponding to 12- and 24-hr forecast errors of the current *M2M* models and the NHC in different time intervals

Time range	Forecast (h)	2/3- probability circle radius <i>M2M</i> models (<i>n</i> miles)	2/3- probability circle radius NHC (<i>n</i> miles)
2005–2009	12	47	36
	24	116	62
2011–2015	12	42	8
	24	112	23
2016–2019	12	45	27
	24	115	40

capabilities for this purpose, only the first input sequence of a storm’s trajectory was fed to the model and the whole trajectory was predicted by the *M2Mn* models. Inputs to the models from the second prediction sequence onwards include the predicted storm locations and corresponding derived storm speed, direction and displacement probabilities in previous time instants (only the true value for w_m was used in all sequences). The trajectories for the four chosen test storms are shown in Figs. 9 and 10. This is equivalent to a 228-hr forecast by the *M2M1* model for hurricane Sandy, which had the shortest lifespan among the

four test storms. Deviations of the predicted storm trajectories from the true trajectories in these Figures are basically illustrations of the compounded error growth previously discussed. Computed error in distance at each time step averaged over the whole trajectory is also reported for all the models.

The reported errors for the *M2Mn* models are in most instances in thousands of kilometers for each predicted time step. The predicted trajectories in the initial stages of hurricane evolution are significantly closer to the true trajectories compared to later on in a storm’s lifespan. This is to

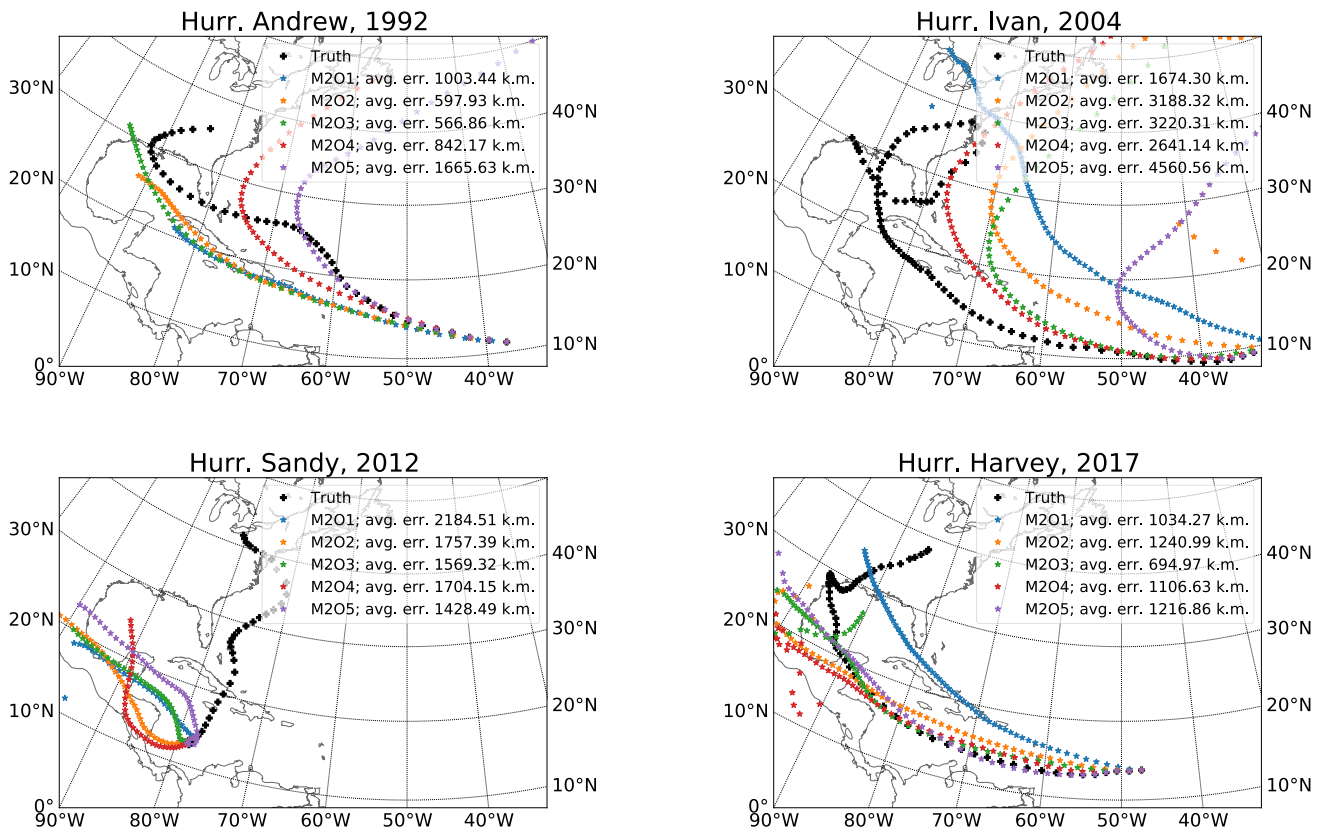


Fig. 9 Forecast of whole trajectories of select hurricanes in advance given only the initial/ first sequence; actual trajectory is shown by the gray '+' symbols; the '*' symbols represent the predicted coordinates

be expected, however, as the compounded errors incurred at each prediction step worsen the forecast at the next prediction step. Consequently, the predictions veer off from the true trajectories. None of the models is able to predict the complex loop in Ivan's trajectory. Similar deviations are obtained for the trajectory of hurricane Sandy. This storm initially travels southward and then turns about 180° to travel northward. Although most models correctly predict the eventual northward motion of the storm at least qualitatively, the models are unable to predict the sudden change in direction of translation. The models perform better in predicting the trajectories of hurricanes Andrew and Harvey, which follow the general trajectory trends for the North Atlantic basin hurricanes. However, the forecast errors are still very large.

The limitation of the LSTM-RNN models' ability to predict a whole storm trajectory from an initial condition is not surprising. Starting from its genesis, a storm's evolution is a complex nonlinear phenomenon dependent on several fluid and thermodynamic effects. This remains one of the most challenging problems in Geophysical Fluid Dynamics. For example, in [21], testing several EPSs for forecasting North Atlantic hurricane trajectories, the minimum 24-hr total track error (same as the error in distance

by the $M2O_n$ models. Inputs to the models are updated based on predictions at previous prediction steps. Average error in miles calculated over the whole trajectory is reported

used in the present work) increased by ≈ 75.64 – 123.92 km for every 24-hr prediction interval for a 120-hr forecast for all storms in the period 2008–2015. Mean 24-hr forecast errors for the test storms are ≈ 160 km for our $M2M4$ and $M2M5$ models (Fig. 5). It should be noted that the EPSs are physics-based models that are guided by accurate fluid dynamic simulations, and therefore use a wealth of information unavailable to the current data-based models. Furthermore, following the current methodology, $M2Mn$ models with $n > 5$ may be developed to further reduce long-term forecast errors due to compounded error accumulation associated with long-term trajectory forecasting.

5 Conclusions

A family of LSTM-RNN models was developed to predict storm trajectories in the North Atlantic basin. Both Many-To-One and Many-To-Many type prediction architectures were used. The 6-hr storm d.p. feed the models with historical storm trajectory trends at a given location, which results in a significant gain in short-term forecasting accuracy (Fig. 6). The proposed models are able to forecast up to 30 h in advance. Mean forecasting errors computed over more

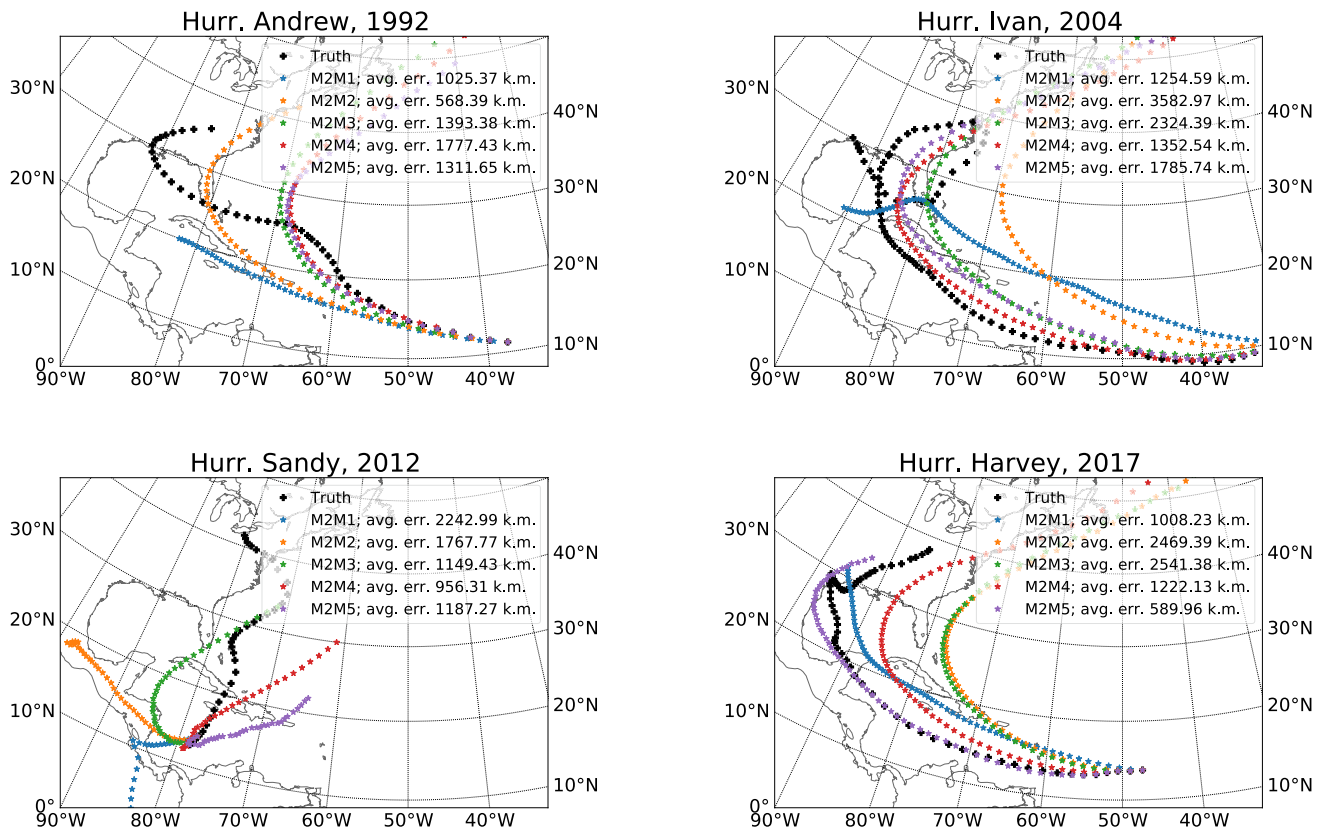


Fig. 10 Forecast of whole trajectories of pre-selected hurricanes given only the initial/ first sequence; Black '+' symbols represent the true locations; Colored '*' symbols represent the $M2M_n$ model

than three hundred validation and test storms demonstrate the efficacy of these models. Overall, the $M2M$ models are more accurate. The minimum mean 6-hr forecasting error of all models was ≈ 30 km. Considering that the storm-eye radii may extend up to 80 km and storm radii may be as large as hundreds of kilometres [30], the model forecasts are reasonably accurate. $M2O$ models are more error-prone due to compounded error accumulation beyond 6-hr predictions; this limit for the $M2M_n$ model is $6n$ h. Therefore, for long-term forecasting, the $M2M$ prediction models are more appropriate; these are more accurate at earlier prediction steps and the error increases linearly between prediction steps. The minimum and maximum mean 6-hr forecast errors for the $M2M$ models in predicting four chosen complex-trajectory test hurricanes were 27 km and 37 km (Fig. 8). The most significant advantage of the proposed models is their fast forecasting capability using modest computational resources. The model development methodology described herein uses readily available data and produces the requisite results in seconds. NHC uses an ensemble of models, including dynamical models that solve the fluid dynamics equations, that require hours of computation on supercomputers. A maximum of 30-hr storm evolution data is required. The models developed in this work are capable of providing

forecasts. 2nd prediction sequence onwards, inputs to the models are based on predictions at previous prediction steps. Average error calculated over the whole trajectory is reported

predictions with as little as only one time record. The models presented herein are reliable and accurate, incurring errors comparable to those of state-of-the-art ensemble models used by the NHC for forecasting at least up to 12 h (Table 2).

Base RNN model: LSTM

Recurrent Neural Networks (RNN) were innovated to extract pattern and context from sequences. RNNs are applied in a wide range of sequence related problems, including modeling and prediction of languages and sentiment, video tagging, a sequence prediction in time. The Long Short-Term Memory (LSTM), an RNN algorithm, was prescribed in [11] to tackle the vanishing gradient problem. Over long sequences, relevant past information may get lost or, equivalently, gradients may vanish while training a model using backpropagation. In an LSTM unit, past information may be retained via a cell state that passes through all LSTM layers. LSTM is the base RNN architecture of the models developed in the present work (Fig. 11).

An LSTM unit/ cell is shown in Fig. 11. The superscript in the vector variables indicates the layer number (this unit belongs to layer q of the model); the subscript denotes the

corresponding input step l . A cell comprises four main components, the cell state (passing through the units, colored red), the forget gate (colored orange), the input gate (colored green) and the output gate (colored blue). The three gates basically apply the three activation functions (in the schematic, σ and \tanh represent sigmoid and hyperbolic tangent activation functions, respectively), each of which has a specific role in information propagation through the model. The cell state ($\vec{C}_l^{(q)}$) is the unique component of an LSTM RNN. The cell state passes through all timesteps $l = 1, 2, \dots$ of a given layer, and is therefore able to preserve information from the past and also accumulate new information with increasing l [14]. The cell shown in the diagram receives an input from the previous layer belonging to the same time step $\vec{h}_l^{(q-1)}$, and also from the previous time step in the same layer, $\vec{h}_{l-1}^{(q)}$. Based on these inputs to the cell, the forget gate dictates the part of the cell state to be discarded at the current unit. On the basis of these same inputs, input gate dictates the information from the present inputs to be added (marked by $+$) to the cell state. Consequently, after these operations, the cell state gets modified in the current LSTM unit ($\vec{C}_{l-1}^{(q)} \rightarrow \vec{C}_l^{(q)}$), which is the cell state received by the LSTM cell to the right, i.e., the next timestep in the same layer. The updated cell state also participates in obtaining the output from the current cell ($\vec{h}_l^{(q)}$) after the sigmoid activation is applied at the output gate.

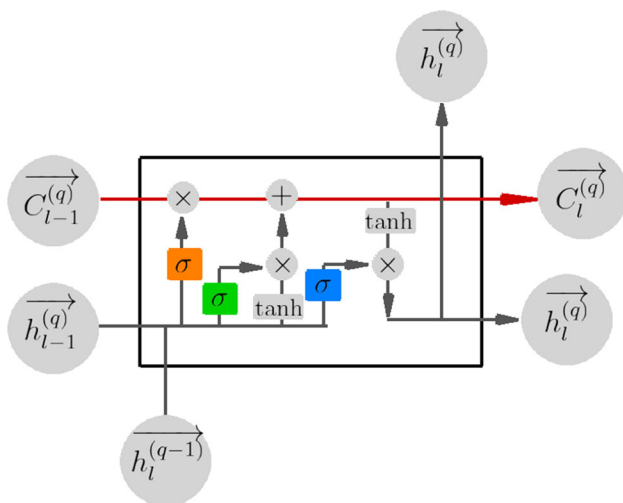


Fig. 11 A representative LSTM unit; the unit belongs to the q th layer of the model. Corresponding input time step is l , i.e., at the input layer ($q = 1$), $\vec{h}_l^0 = \vec{I}_l$ input features at the l th timestep. The symbols \times and $+$ represent pointwise operation

Declarations

Conflict of interest The authors have no conflicts of interest to declare.

References

- Alemamy S, Beltran J, Perez A, et al (2019) Predicting hurricane trajectories using a recurrent neural network. In: Proceedings of the AAAI conference on artificial intelligence, pp 468–475
- Bose R, Pintar A, Simiu E (2021). Forecasting the evolution of north Atlantic hurricanes: a deep learning approach. <https://doi.org/10.6028/NIST.TN.2167>
- Boussiou L, Zeng C, Guénais T, et al (2020) Hurricane forecasting: a novel multimodal machine learning framework. arXiv preprint [arXiv:201106125](https://arxiv.org/abs/201106125)
- Buizza R, Milleer M, Palmer TN (1999) Stochastic representation of model uncertainties in the ecmwf ensemble prediction system. *Q J R Meteorol Soc* 125(560):2887–2908
- Chen T, Guestrin C (2016) Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pp 785–794
- Chen Z, Yu X (2020) A novel tensor network for tropical cyclone intensity estimation. *IEEE Trans Geosci Remote Sensing*
- DeMaria M, Aberson SD, Ooyama KV et al (1992) A nested spectral model for hurricane track forecasting. *Mon Weather Rev* 120(8):1628–1643
- Emanuel K, Ravela S, Vivant E et al (2006) A statistical deterministic approach to hurricane risk assessment. *Bull Am Meteor Soc* 87(3):299–314
- Giffard-Roisin S, Yang M, Charpiat G, et al (2018) Deep learning for hurricane track forecasting from aligned spatio-temporal climate datasets
- Gómez P, Nebot A, Ribeiro S et al (2003) Local maximum ozone concentration prediction using soft computing methodologies. *Syst Anal Model Simul* 43(8):1011–1031
- Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9(8):1735–1780
- Jarvinen BR, Neumann CJ, Davis MA (1984) A tropical cyclone data tape for the North Atlantic basin, 1886–1983: contents, limitations, and uses
- Jeffries RA, Miller RJ (1993) Tropical cyclone forecasters reference guide. 3. Tropical cyclone formation. Tech. rep., NAVAL RESEARCH LAB MONTEREY CA
- Karpathy A, Johnson J, Fei-Fei L (2015) Visualizing and understanding recurrent networks. arXiv preprint [arXiv:150602078](https://arxiv.org/abs/150602078)
- Kim S, Kim H, Lee J, et al (2019) Deep-hurricane-tracker: Tracking and forecasting extreme climate events. In: 2019 IEEE winter conference on applications of computer vision (WACV), IEEE, pp 1761–1769
- Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint [arXiv:14126980](https://arxiv.org/abs/1412.6980)
- Kordmahalleh M, Sefidmazgi M, Homaifar A (2016) A sparse recurrent neural network for trajectory prediction of Atlantic hurricanes. *Proc Genet Evolu Comput Conf* 2016:957–964
- Kurihara Y, Bender MA, Tuleya RE et al (1995) Improvements in the gfdl hurricane prediction system. *Mon Weather Rev* 123(9):2791–2801
- Lambert JH (1972) Notes and Comments on the Composition of Terrestrial and Celestial Maps (1772). 8, Department of Geography, University of Michigan

20. Landsea CW, Franklin JL (2013) Atlantic hurricane database uncertainty and presentation of a new database format. *Mon Weather Rev* 141(10):3576–3592
21. Leonardo NM, Colle BA (2017) Verification of multimodel ensemble forecasts of north Atlantic tropical cyclones. *Weather Forecast* 32(6):2083–2101
22. Lian J, Dong P, Zhang Y et al (2020) A novel data-driven tropical cyclone track prediction model based on cnn and gru with multi-dimensional feature selection. *IEEE Access* 8:97114–97128
23. Neumann CJ (1972) An alternate to the Hurrell (hurricane analog) tropical cyclone forecast system
24. Parasuraman K (2020) Hurricane florence - building a simple storm track prediction model. <https://towardsdatascience.com/hurricane-florence-building-a-simple-storm-track-prediction-model-1e1c404eb045>
25. Richardson DS (2000) Skill and relative economic value of the ecmwf ensemble prediction system. *Q J R Meteorol Soc* 126(563):649–667
26. Roy AM, Bose R, Bhaduri J (2022) A fast accurate fine-grain object detection model based on yolov4 deep neural network. *Neural Comput Applic* 34(5):3895–3921. <https://doi.org/10.1007/s00521-021-06651-x>
27. Rüttgers M, Lee S, Jeon S et al (2019) Prediction of a typhoon track using a generative adversarial network and satellite images. *Sci Rep* 9(1):1–15
28. Schwerdt RW, Ho FP, Watkins RR (1979) Meteorological criteria for standard project hurricane and probable maximum hurricane windfields, Gulf and East Coasts of the United States
29. Vaswani A, Shazeer N, Parmar N, et al (2017) Attention is all you need. In: *Advances in neural information processing systems*, pp 5998–6008
30. Willoughby HE, Darling R, Rahn M (2006) Parametric representation of the primary hurricane vortex. Part ii: A new family of sectionally continuous profiles. *Mon Weather Rev* 134(4):1102–1120

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.