International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 22 Issue 12, DEC, 2022

HPP-LSTM: A Novel Hurricane Path Prediction Method using Long Short-Term Memory

Vijayakumar Polepally
Department of Computer Science & Engineering
Kakatiya Institute of Technology & Science
Warangal, Telangana
vijay.cse@kitsw.ac.in

Jagannadha Rao D.B.
Department of Computer Science & Engineering
Malla Reddy University
Hyderabad, Telangana
jagandb@gmail.com

***Abstract*— Hurricanes, which are powerful tropical cyclones originating in the Atlantic basin, pose significant threats to both life and infrastructure upon landfall. Anticipating their trajectory is essential to mitigate associated risks. Traditional predictive techniques often fall short due to the inherent complexity and variability of storm paths. In this research, we employ Long Short-Term Memory (LSTM) networks, a class of deep learning models known for handling sequential data, to forecast hurricane trajectories. Given their ability to retain long-term dependencies, LSTMs are well-suited for sequence modeling tasks such as this. The proposed model is trained on relevant meteorological parameters to learn the patterns influencing storm movements and deliver more accurate path predictions.**

***Keywords*— Hurricane, deep learning, LSTM, RNN.**

I. INTRODUCTION

Hurricanes are tropical cyclones which originate in Atlantic basin that can cause severe damages when the storms make landfall. There are many factors like wind speed, seal level pressure, intensity change, humidity, temperature, etc., that affect hurricane path. These hurricanes can cause landfall resulting in huge life loss and property loss. The time between June 1 and November 30 is called North Atlantic Hurricane season during which the warm waters in Atlantic cause tropical cyclones. Few of those cyclones cause many casualties and property loss. For example, Hurricane Harvey caused US$ 96.9 billion loss and Hurricane Maria caused US$ 69.4 billion loss. Widely used models in predicting storm tracks are data intensive numerical models which are computationally expensive. Generally, traditional data mining and machine learning models are used for hurricane path prediction, which are less effective than deep learning models. Building a deep learning model which can predict the path of hurricane by taking different factors like wind speed, seal level pressure, intensity change, humidity, temperature, etc., as input. Building a model using proper framework and data. Collect the SHIPS data from the sources and preprocess the data. Build an LSTM model on the preprocessed data. Validate the model using validation set. Predict the hurricane's path for hurricanes in test data set. This project is used to predict path of the hurricane given the external environment factors. It can be used by meteorological department and anyone who is interested in study of hurricanes.

II. LITERATURE SURVEY

Hurricanes are tropical cyclones which originate in Atlantic basin that can cause severe damages when the storms make landfall. Conditions essential for hurricane formation are water temperature greater than or equal to 26deg C, moist warm air, slight wind shift, Coriolis effect. The rotation of wind is provided by thunderstorms above or below equator which is called Coriolis effect. This results in anti-clockwise rotation of hurricanes above equator and clockwise rotation below equator.

The water vapor above sea rises and cools in the process. As it cools the water vapor condenses which results in the release of latent heat and thus thunderstorms clouds are formed. As this continues columns of intense thunderstorm clouds are formed. As the air continuously rises, at the base of hurricane an area with low pressure is created. Due to low pressure air is pulled in and rotated during this process which results in hurricane. Hurricane prediction models helps to save life loss and property loss. There exist many models which are used in real life for prediction of hurricane paths. These existing models are complex in structure. There are four types of models:

A. *Dynamic models or numerical models*

These models solve complex physical equations governing atmospheric motion, incorporating sub-models for land-ocean interactions, radiation processes, cloud dynamics, and surface interactions. Although highly accurate under ideal conditions, they are computationally intensive and often require high-performance computing infrastructure to simulate and predict storm behavior.

B. *Statistical models*

Statistical approaches leverage historical storm data and apply regression or probabilistic models to forecast future paths. These methods are relatively lightweight and rely on features such as geographical location, date, and prior trajectory data. They are computationally efficient but can underperform in capturing dynamic atmospheric variations.

C. *Statistical-Dynamical models*

These hybrid models aim to combine the strengths of both statistical and physical modeling. A notable example involves predicting hurricane paths using sea surface temperatures and vertical wind shear as key variables. Such models attempt to balance accuracy with computational feasibility.

D. *Ensemble or Consensus model*

Ensemble models' aggregate predictions from multiple forecasting systems to improve reliability and reduce uncertainty. By initializing each model with slightly different parameters or conditions, these systems can capture a range of potential outcomes and improve average accuracy over any single model.

E. *Hurricane Path Prediction using data mining*

Data mining is extraction of unknown information from known data and observing patterns in the known data. In this data mining there is sequential mining which is used on time – series data. Sequential mining builds association rules from time – series data and using these association rules are used to make predictions. Some of the earlier works used Apriori algorithm and Weighted Apriori Algorithms to predict hurricane paths. 57.5% and 65.0% of accuracy is achieved using these models respectively.

F. *Hurricane Path Prediction Deep Learning*

Deep learning is a subset of machine learning, which is a neural network with multiple layers. These neural networks try to resemble human brain. Neural networks contain several nodes and each node acts as a neuron. Generally, it contains three layers. They are input, hidden and output layers. Between these layers weights are used to estimate outputs for given inputs. Determination of these weights by minimizing the loss is main objective.

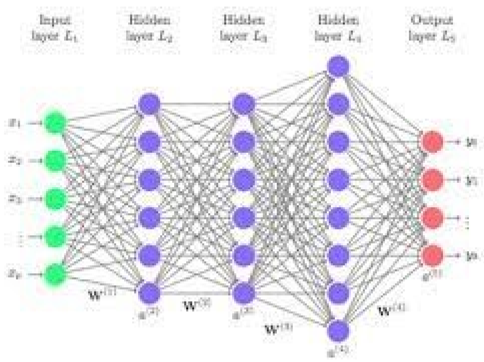

Figure 1 Deep Learning

III. PRELIMINARIES

A. Recurrent Neural Network

Recurrent neural network is a neural network in which there are multiple time steps. The output of one time-step is given as input to the next step. The inputs and outputs of RNN are dependent on each other unlike in traditional neural networks. In time-series data to predict next output some information needed to be remembered by the model but traditional neural network models have no such feature but RNN solved this problem using hidden layer. Hidden layer remembers information about the sequence so that output can be predicted. This hidden layer added a new feature to neural network called "memory". Each hidden layer has its own set of weights which makes them independent.

In the below figure, which shows three hidden layers, each hidden layer has different weights w1, w2, w3 respectively.

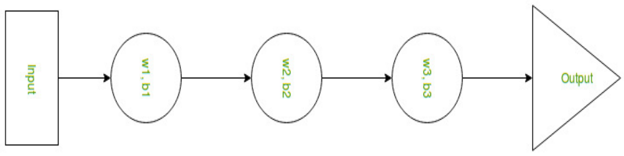

Figure 2 RNN

RNN converts the independent activations into dependent activations by providing same weights to all layers due to which complexity of increasing number of parameters and memorizing previous outputs and feeding them to next hidden layer.

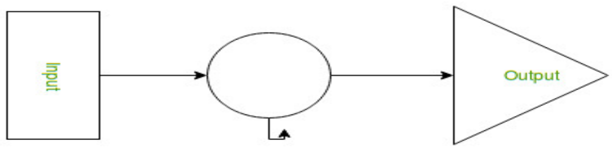

Figure 3 RNN Cell

Current state: $\boldsymbol{h_t = f(h_{t-1}, x_t)}$

Current state is $\boldsymbol{h_t}$

Previous state is $\boldsymbol{h_{t-1}}$

Input state is $\boldsymbol{x_t}$

The activation function formula is $\boldsymbol{h_t = tanh(W_{hh}h_{t-1} + W_{xh}X_t)}$

Weight at recurrent neuron $\boldsymbol{W_{hh}}$

Weight at input neuron $\boldsymbol{W_{xh}}$

Output: $\boldsymbol{y_t = W_{hy}h_t}$

Output is $\boldsymbol{y_t}$

Weight at output layer $\boldsymbol{W_{hy}}$

Initially a single time step of input is provided to the network. Then it calculates current state using the given input and then calculates previous input. So, the current state $\boldsymbol{h_t}$ becomes $\boldsymbol{h_{t-1}}$ for next time step. Based on problem statement one can choose any number of time steps. After completion of all timesteps final output is calculated. By comparing original output to the calculated output error is calculated. This error is backpropagated to network to calculate and update the weights again. The main problems due to RNN are Gradient vanishing and gradient exploding problems. This problem is solved by LSTM (Long Short -Term Memory).

B. Long- Short term Memory

LSTM networks are an extension of RNNs which handle situations that RNNs fail. RNNs work on current input by considering feedback i. e. previous output and store in its memory for a short period of time. Firstly, for the situations where the information from a long time ago is required, RNNs fail. Secondly, there are no constraints over which information from the previous steps needs to be stored and which information is no longer required. Drawbacks of RNN's also include exploding and vanishing gradient problem that will be faced while training a network through backtracking. LSTM is designed to almost remove vanishing gradient problem. LSTMs handle noise, distributed representations and continuous values. LSTMs have many parameters like learning rates, input and output biases. O(1) is the complexity to update weight in LSTM.

Exploding and vanishing gradients: When training data is sent through a network, the main aim is to optimize the loss which is error or cost. Loss with respect to a particular set of weights is calculated, weights are updated and the process is repeated until a particular set of weights are obtained for which loss is minimum. This is backtracking and sometimes the gradient almost becomes zero. Gradient of a layer depends on certain components in successive layers. If the values are smaller, then the gradients will be much smaller when multiplied with the learning that ranges between 0.1 and 0.001. The decrease in the gradient is called scaling effect. This is called as vanishing gradient problem. Similarly, if the weights will be updated with values greater than optimal value.

This is called as exploding gradients problem. To remove this scaling effect, scaling factor was fixed to one and neural network was re-built. LSTM is the enriched version of this cell with several gating units.

Architecture: The main difference between RNN cell and LSTM cell is that the hidden layer in LSTM is a gated unit. RNN has only single neural net layer of tanh. Whereas, LSTM has three logistic sigmoid gates and a tan h layer. These gates determine which information is to be passed to next and which shouldn't be. Four layers interact with each other to produce output of that cell and to produce cell state which are passed onto next hidden layer. The output ranges between 0 and 1 where 0 implies 'reject' and 1 implies 'include'.

Every LSTM cell contain three inputs $\boldsymbol{h_{t-1}}$, $\boldsymbol{C_{t-1}}$ and $\boldsymbol{x_t}$ and each LSTM cell has two outputs $\boldsymbol{h_t}$ and $\boldsymbol{C_t}$ where, $\boldsymbol{h_t}$ is hidden state $\boldsymbol{C_t}$ is cell state and $\boldsymbol{x_t}$ is input at a given time t. First layer is fed with input $\boldsymbol{h_{t-1}}$ and $\boldsymbol{x_t}$, $\boldsymbol{h_{t-1}}$ is hidden state of the previous cell, only selected amount of information from this layer is remembered so it is called as forget gate and the output ranges from [0,1] that will be multiplied with the previous cell state i. e, $\boldsymbol{C_{t-1}}$.

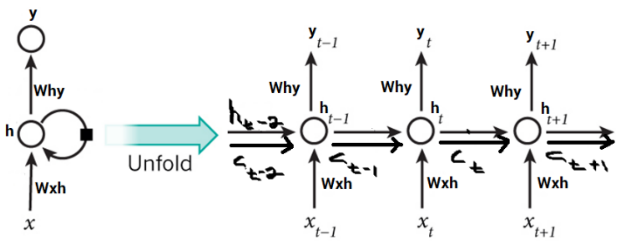

Figure 4 LSTM

IV. PROPOSED SOLUTION

SHIPS (Statistical Hurricane Intensity Prediction Scheme) is an intensity model run by National Hurricane Centre (NHC). Dataset contains information about hurricanes from 1982 to 1999. There were 716823 data lines in the dataset considered.

Many factors influence the path of hurricanes. Some of them are Beta drift, Vertical wind shift, Terrain, Global trade winds and location of high-pressure systems like Bermuda triangle. Maximum Surface Wind, Minimum Sea Level Pressure, Strom Type, Latitude, Longitude, Climatological depth of 20deg C isotherm, Distance to nearest major landmass (km) vs time, Ocean Age, Reynolds SST (deg C*10) vs time, 200 hPa zonal wind (kt*10) vs time, etc., are some of the features considered for training of the model.

Stochastic Gradient is used as optimizer. Optimizer modifies the attributes like weights and learning rate which helps in decreasing the overall loss and increase the accuracy. A function which outputs the partial derivative of a set of parameters of its inputs is called a gradient descent. It is run iteratively to find a particular set of parameters which optimizes the cost function. 'Stochastic' is related to probability. A stochastic gradient descent selects a few samples randomly instead of a whole dataset for each iteration.

Algorithm for Stochastic Gradient Descent:

$$\text{for } i \text{ in range } (m):$$
$$\theta_j = \theta_j - \alpha\left(\hat{y}^i - y^i\right) x_j^i$$

Generally, whole dataset is considered for iteration. But it is troublesome as the size of the dataset increases. Whereas, SGD uses a single sample for each iteration. It reaches the solution in shorter time.

For methods using ground truth from a previous time step as input teacher forcing can be used to train them efficiently and quickly. It is used for many applications such as image captioning, machine translation and text summarization. It helps in preventing gradient explosion. By using teacher forcing convergence can be achieved faster and accuracy of the model will be improved. Instead of giving output from the previous step as input to current timestep ground truth is given as input. It doesn't use the output from the network, instead it uses expected/actual output from training dataset.

Back propagation through time (BPTT) is used to reduce the error of the outputs from the network in comparison with expected output by updating the weights of the neural network. A training input is taken and propagated through the network and an output is obtained. The output and expected output are compared and error is calculated. The derivatives of the error with respect to weights of the network are calculated. Then, the weights are adjusted so the errors are minimized. This process is repeated. Truncated backpropagation through time is a different version of back propagation through time. In this algorithm similar to back propagation through time. But in this algorithm sequence is processed one step at a time periodically let us say k1 timesteps but updates to another fixed number of timesteps say k2, k1 influences the training speed.

Using Basemap API in python a map showing North Atlantic basin with actual and predicted paths for test hurricanes.

V. RESULTS

After training, validating and testing LSTM network with SHIPS dataset, a loss of 0.004968 was obtained and the mean distance error was 235.23km.

A graph between number of epochs and loss function is plotted. Training curve and validation curve both coinicided which represents that model is not over fitted.

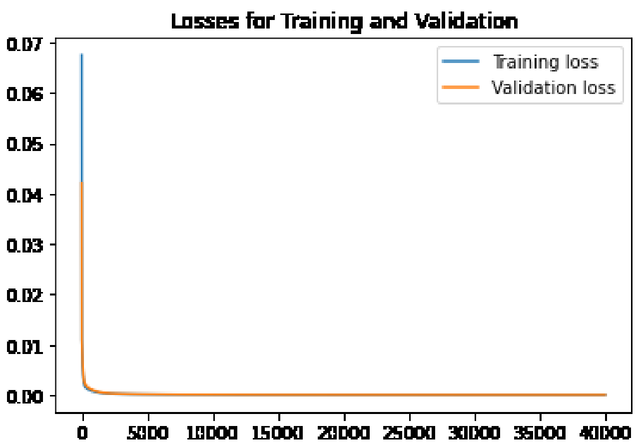

Figure 5 Losses for training and Validation

A graph of mean distance error per every hundred epochs was plotted.

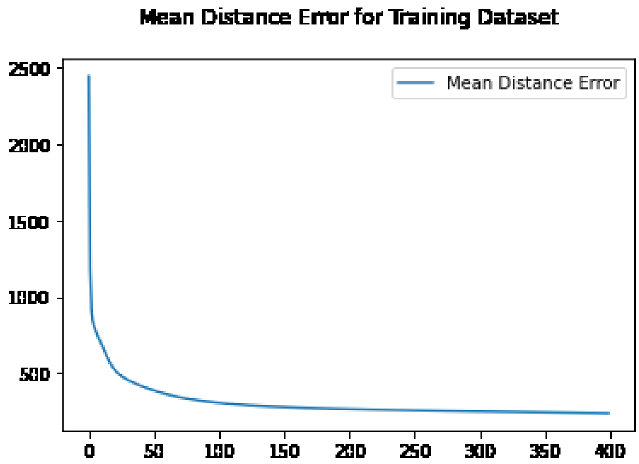

Figure 6 Mean distance error for Training Dataset

Hurricane paths for testing dataset were displayed using Basemap API. Below is a sample output.

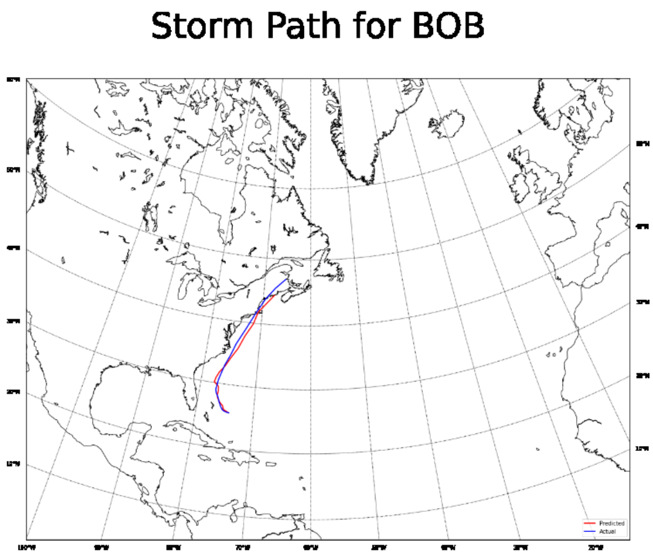

Figure 7 Storm path for BOB hurricane

VI. CONCLUSION

Hurricanes are a specific type of tropical cyclone exclusive to the Atlantic basin that can cause severe damages when the storms make landfall. Hurricane prediction models helps to save life loss and property loss. In this project, we used LSTM model to predict hurricane path. The minimum mean distance error we achieved is 235.23km, which is less compared to the huge north Atlantic region. Our future work is to improve this model and ensemble it with other deep learning models for better performance.

VII. REFERENCES

[1] Taylor S. Cox, Calvin S.H. Hoi, Carson K. Leung*, and Caden R. Marofke, "An Accurate Model for Hurricane Trajectory Prediction", IEEE 2018.

[2] Sophie Giffard-Roisin, Mo Yang, Balázs Kégl, Claire Monteleoni, Guillaume Charpiat "Deep Learning for Hurricane Track Forecasting from Aligned Spatio-temporal Climate Datasets", NIPS Conference paper 2018.

[3] Mudigonda, M., Kim, S., Mahesh, A., Kahou, S., Kashinath, K., Williams, D., Michalski, V., O'Brien, T., and Prabhat, M. (2017). "Segmenting and tracking extreme climate events using neural networks". In Deep Learning for Physical Sciences (DLPS) Workshop, held with NIPS Conference.

[4] Moradi Kordmahalleh, M., Gorji Sefidmazgi, M., and Homaifar, A. (2016). "A sparse recurrent neural network for trajectory prediction of atlantic hurricanes". In Proceedings of the Genetic and Evolutionary Computation Conference 2016.

[5] Bai, S., Kolter, J. Z., and Koltun, V. (2018). "An empirical evaluation of generic convolutional and recurrent networks for sequence modelling".

[6] de Bezenac, M., Pajot, A., and Gallinari, P. (2017). "Deep learning for physical processes: Incorporating prior scientific knowledge".

[7] Giffard-Roisin, S., Yang, M., Charpiat, G., Kégl, B., and Monteleoni, C. (2018). "Fused deep learning for hurricane track forecast from reanalysis data". In Climate Informatics Workshop Proceedings 2018, Boulder, United States.

[8] Sheila Alemany, Jonathan Beltran, Adrian Perez, Sam Ganzfried, "Predicting Hurricane Trajectories using a RNN", 2018.

[9] Witten, I. H.; Frank, E.; Hall, M. A.; andPal, C. J. 2016. "Data Mining: Practical machine learning tools and techniques". Morgan Kaufmann.

[10] P. Chandrasekar, K. Qian, H. Shahriar, & P. Bhattacharya, "Improving the prediction accuracy of decision tree mining with data pre-processing," Proc. IEEE COMPSAC 2017.

[11] Hurricane Path Prediction using Deep Learning, Kamban Parasuram https://medium.com/@kap923/hurricane-path-prediction-using-deep-learning-2f9fbb390f18, 2019.