Predicting the Day of Cherry Blossom using LSTM

Nitul Singha, M.Sc., Achraf Cohen, PhD
Mathematics and Statistics Department, Hal Marcus College of Science and Engineering
University of West Florida

INTRODUCTION

Supervised and unsupervised machine learning techniques, particularly neural networks like ANN and RNN, have been developed to handle large-scale and nonlinear data. RNN, especially LSTM, has shown superior performance in time series prediction due to its ability to capture timing characteristics (Boné, Assaad, and Crucianu 2003).

LSTM is a type of RNN that effectively handles long-term dependencies in sequential data, making it suitable for time series prediction. It can store both short-term and long-term memory, allowing it to bridge time intervals even in noisy and non-stationary data. LSTM has been widely used in various applications, including speech processing, non-Markovian control, time series analysis, and music composition (Karevan and Suykens 2020).

AIM

The aim of this research project is to predict the day of Cherry blossom using Long Short-Term Memory (LSTM) neural networks,

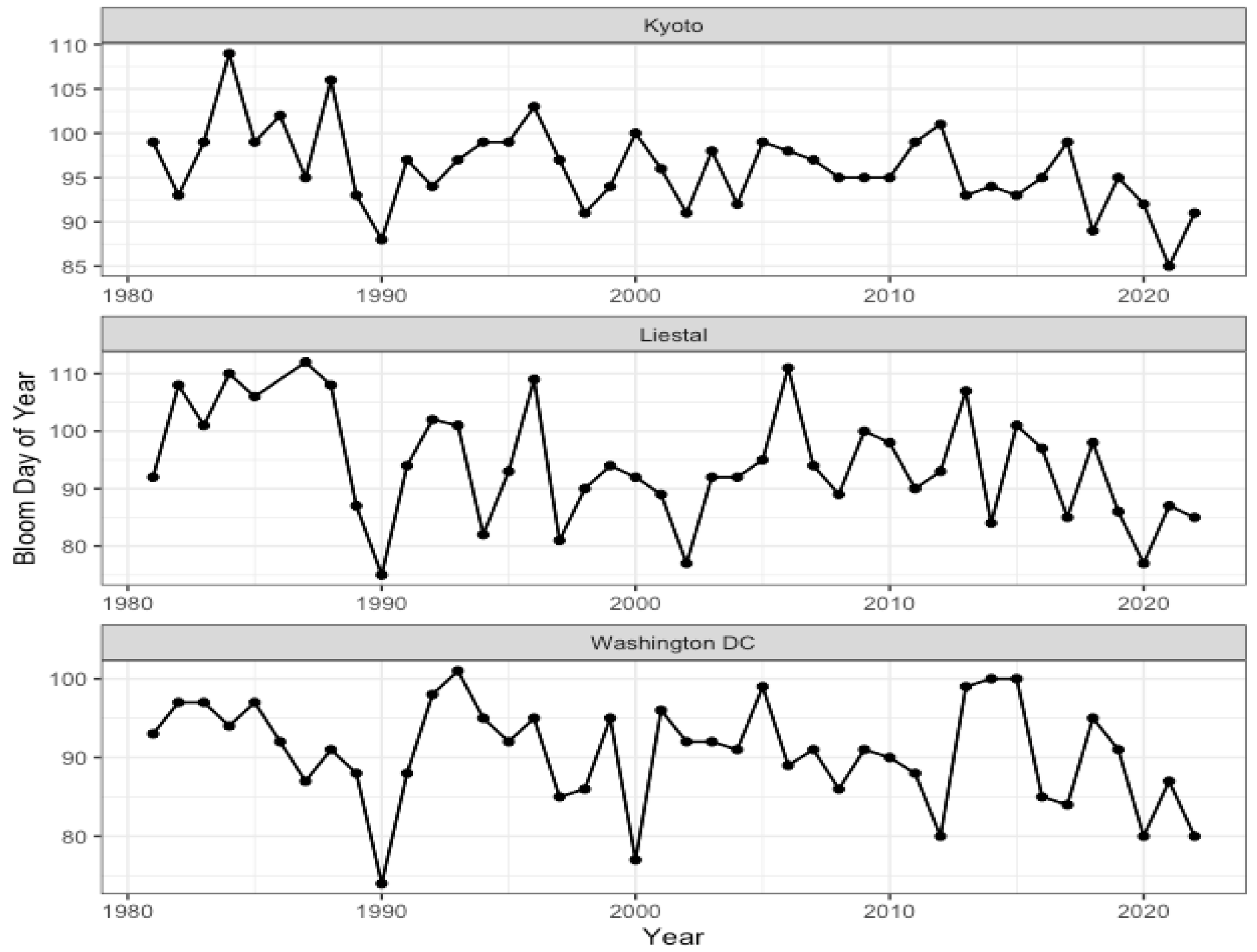
Leveraging historical weather data as input. By training the LSTM model on the weather time series data, we intend to capture the complex temporal relationships between weather variables and the timing of Cherry blossom.

This predictive model can be valuable for various applications, including agriculture, tourism, and environmental monitoring, by providing essential insights and facilitating decision-making related to Cherry blossom events.

DATA

The data available dates back to 1981, allowing to explore patterns in cherry blossom phenology over the past few decades.

Variable	Description	Type
Location	Location	Categorical
Year	year of blossom	Numeric
Month	month of blossom	Numeric
Temp (F)	Avg. temperature from Dec to Feb	Numeric
Dew	Avg. dew from Dec to Feb	Numeric
Humidity	Avg. humidity from Dec to Feb	Numeric
Precip	Avg. precipitation from Dec to Feb	Numeric
Windspeed	Avg. windspeed from Dec to Feb	Numeric
Moon phase	Avg. monophase from Dec to Feb	Numeric
Daytime	Avg. daytime (in sec) from Dec to Feb	Numeric



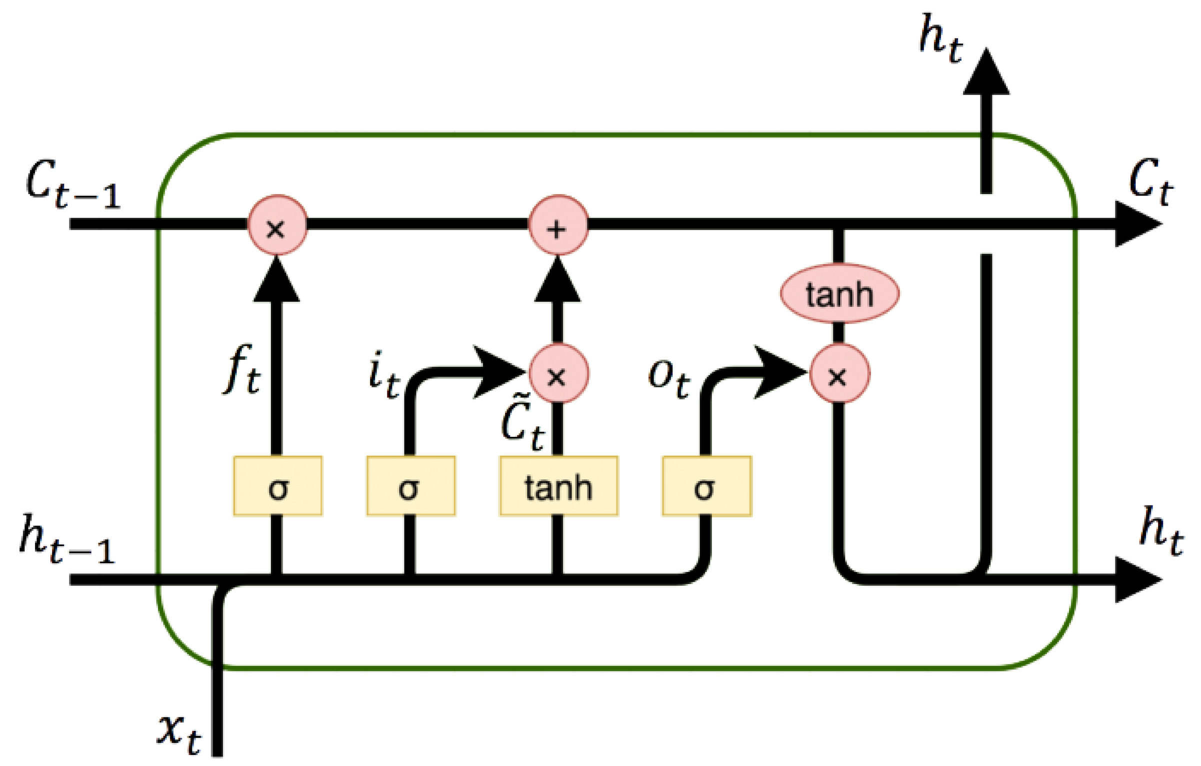
METHODOLOGY

Long Short-Term Memory (LSTM) is a powerful methodology for time series prediction that incorporates tanh gates to effectively manage long-term dependencies and short-term memory in sequential data.

In most conventional Recurrent Neural Networks (RNNs), the hidden layer function H is typically achieved through an element-wise application of a sigmoid function. However, the standard RNN architecture struggles to capture and exploit long-range dependencies in sequential data. This limitation is primarily due to the **vanishing gradient problem**, where gradients become very small during backpropagation, hindering the network's ability to learn from distant time steps. To address these limitations, the Long Short-Term Memory (LSTM) architecture was introduced.

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \qquad \text{(i)}$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \qquad \text{(ii)}$$
$$c_t = f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \qquad \text{(iii)}$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \qquad \text{(iv)}$$
$$h_t = o_t \tanh(c_t) \qquad \text{(v)}$$

The input gate (i) determines which parts of the current input should be stored in the memory cell. The forget gate (f) decides which information from the previous time step should be discarded from the memory cell. The output gate (o) regulates the flow of information from the memory cell to the output of the LSTM cell. The cell (c) represents the memory state of the LSTM cell, and the cell input activation vector (c_tilde) combines new input information with the previous cell state.

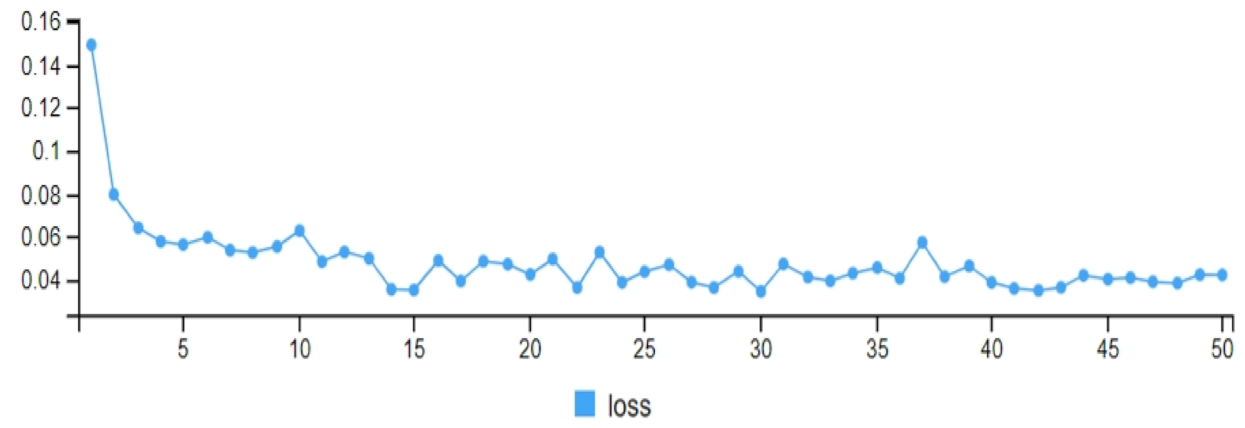


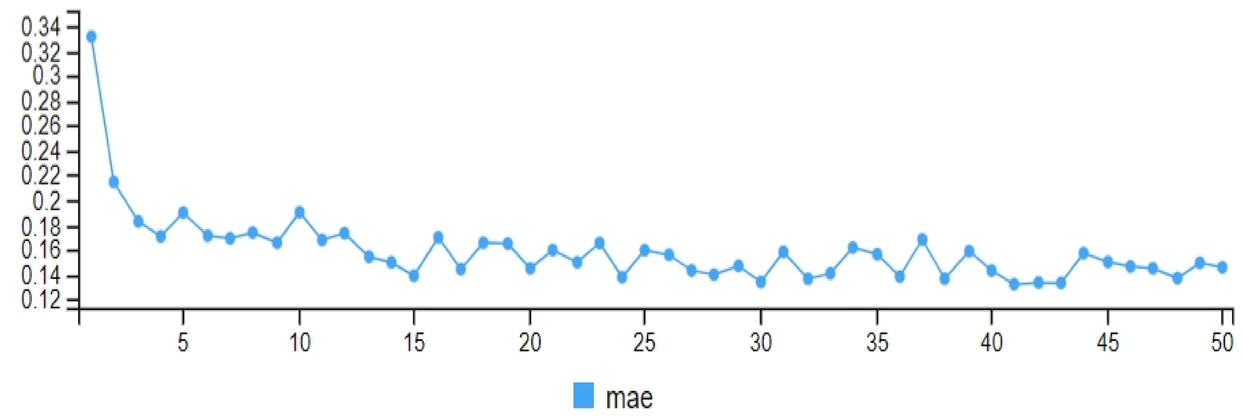
The above Figure illustrates a single LSTM memory cell. In this architecture, the hidden layer function H is implemented using a composite function that involves several components, such as **input** gates, **forget** gates, **output** gates, cells, and cell input activation vectors.

MODELING & RESULTS

The model utilizes the Y time series data with a ***lag of 1***, and three regressor variables: ***dew***, ***wind speed***, and ***temperature***, each with a ***lag of 1***

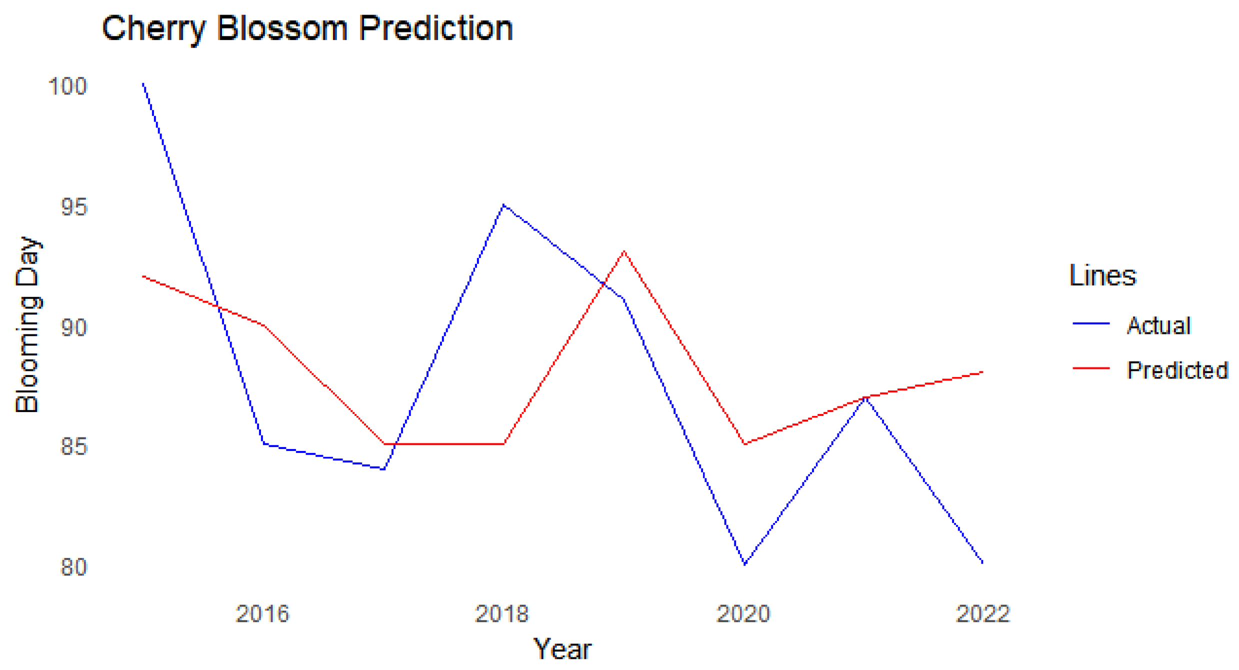
To prepare the data for the model, we utilize **Min-Max** transformation. The model consists of **one LSTM layer** to capture sequential dependencies and a **Dense layer** as the output layer.





The **TSLSTM** function is used to create a Time Series LSTM model in R, using the given parameters and data.

Tuning parameters:
The LSTM model consists of 200 units and incorporates a dropout rate of 0.1 to prevent overfitting. The model is trained for 50 epochs using mean squared error (mse) as the loss function and mean absolute error (mae) as the evaluation metric. The tanh activation function is applied, and the data is split into training (80%) and validation (20%) sets.



<u>During training</u>, the model achieved an **RMSE of 0.1871**, indicating a close fit to the training data. However, the MAPE value being Inf suggests potential issues, possibly due to outliers or division by zero in the data.
In the <u>testing phase</u>, the model performed well, with an **RMSE of 0.1985** and a reasonable **MAPE** of **0.318**, indicating its ability to generalize to new data

CONCLUSION

The Time Series LSTM model demonstrates promising performance in forecasting based on the provided accuracy matrix. Further investigation into the MAPE issue is recommended to enhance the model's accuracy and reliability.

REFERENCES

•Karevan, A., & Suykens, J. A. (2020). A novel evolutionary attention-based LSTM training with competitive random search for multivariate time series prediction. Neurocomputing, 387, 104-116.
•Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
•Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. arXiv preprint arXiv:1808.03314.
•Chimmula, V. K., & Zhang, L. (2020). Time series forecasting using deep learning: A survey. arXiv preprint arXiv:2004.13491.
•Boné, A. M., Assaad, M., & Crucianu, M. (2003). An automatic process for classical arabic music composition. In Proceedings of the 14th International Conference on Multimedia (pp. 31-38).
•Yamak, S., Yujian, L., & Gadosey, D. (2019). Time series forecasting. arXiv preprint arXiv:1910.07454.
•Zahra Karevan, Johan A.K. Suykens,
•Transductive LSTM for time-series prediction: An application to weather forecasting, Neural Networks, Volume 125,2020,Pages 1-9,ISSN 0893-6080,
Paul, Ranjit Kumar, and Sandip Garai. 2021. "Performance Comparison of Wavelets-Based Machine Learning Technique for Forecasting Agricultural Commodity Prices." *Soft Computing* 25 (20): 12857–73.