Weather Forecasting and Analysis with LSTM Based on Deep learning

Fengyuan Zhang

Wuhan University of Technology, Wuhan, 430070, China
1807060207@stu.hrbust.edu.cn

Abstract. Weather forecasting is paramount for various sectors, fueling growing interest in leveraging machine learning for predictive weather analysis. The model proposed in this paper represents a significant advancement in this domain by integrating Long Short-Term Memory (LSTM) to augment the capabilities of Recurrent Neural Networks (RNN). By capitalizing on LSTM's unique architectural features, the model excels in processing extended data sequences, thus bolstering learning and prediction accuracy. Empirical results underscore the model's superiority over traditional forecasting methods, characterized by substantial error margins. This study not only underscores LSTM's prowess in sequence prediction but also sheds light on its practical utility, offering valuable insights into LSTM's functioning. Moreover, the findings of this research serve to deepen understanding of sequence prediction methodologies, paving the way for addressing more complex predictive challenges in the future. Ultimately, the integration of LSTM into the RNN framework represents an important step towards improving the reliability of weather forecasting models and refined predictions of forecasts.

Keywords: LSTM, RNN, Sequence Prediction.

1 Introduction

With the development of deep learning technology, many complex geophysical phenomena can be discovered by people. It can be predicted by numerical simulation. This can help to some extent to understand the complex laws of the Earth [1]. The core of weather forecast service is numerical value, and more accurate weather forecast depends on the accuracy of numerical prediction [2]. Over the past 50 years, a great deal of resources has been devoted to the study of how to predict the weather and weather patterns, and observational websites have been developed to provide enough data for predictions to improve their accuracy [3]. Using Recurrent Neural Network (RNN) to forecast the weather can help improve the accuracy of the weather forecast, and better help people to judge and plan.

Meteorological processes usually show complex changes, mainly reflected in time and space. In the physical process, nonlinearity, spatial and temporal scale conflicts, and uncertainty of parameter estimation are also constantly troubling people [4]. As deep learning gained prominence, neural networks became the go-to approach for

Y. Wang (ed.), *Proceedings of the 2024 2nd International Conference on Image, Algorithms and Artificial Intelligence (ICIAAI 2024)*, Advances in Computer Science Research 115,
https://doi.org/10.2991/978-94-6463-540-9_19

forecast tasks. After years of development, neural networks have emerged into many different categories, such as feedforward neural networks, backpropagation neural networks, recurrent neural networks, etc. Because of different logical structures, their prediction methods also have their own characteristics [5]. Traditional sequence prediction methods have obvious shortcomings. Most of that can only predict the sequence by learning the measured numerical information, but cannot process the historical data represented by language for prediction [6]. The final model selection is a neural network model to perform sequence prediction relative to the original data, and a Long Short-Term Memory (LSTM) network layer is used to further improve the problem of long time easy to forget [7]. Through its special gates, LSTM logically forms the forget gate, input gate and output gate to make the model better to learn, delete and save data [8]. Compared with traditional RNNS, LSTMs have more obvious benefits. Compared to the problem of gradient disappearance or explosion that may exist in RNN training, LSTM makes a huge improvement in this aspect and can be effectively improved. All these features above are showing that LSTM, a specialized recurrent neural network architecture, has been utilized to achieve significant advancements in sequential data analysis tasks and can play a role in forecasting. In this study, deep learning technology driven by data is used to make sequential weather prediction [9].

To enhance weather prediction accuracy through historical data analysis, this study integrates the RNN framework with LSTM for traditional series forecasting. Utilizing an open-source weather dataset from Kaggle, the experiment is structured into distinct phases. Beginning with data preprocessing to handle missing values and format adjustments, followed by dataset segmentation, the model construction phase employs RNN and LSTM layers with dropout regularization to mitigate overfitting risks. Model compilation entails the use of Adam optimizer and mean square error loss function. Subsequently, predictions are made and compared with test set outcomes, followed by data visualization and model evaluation. The combined RNN and LSTM approach yields superior accuracy and efficiency compared to conventional methods. Experimental findings demonstrate notable enhancements in temperature and humidity predictions based on past weather data, affirming the model's efficacy in weather forecasting.

2 Methodology

2.1 Dataset Description and Preprocessing

The data set selected for this study comes from Kaggle [10]. Data set contains the date, precipitation, maximum temperature, minimum temperature, wind and weather. Predictable weather conditions are divided into drizzle, rain, sun, snow and fog. The data set contains four years of relevant data sets, which can provide sufficient data support for predictions. The author first divided the dataset according to the number of datasets. For the purpose of model training, the first 800 points of the dataset are used for training the model, the next 200 points are used for validation, and the rest are used as the test set. The data format is reconstructed to meet the specific requirements of LSTM model for input data. To make it easier for the model to understand the dynamic characteristics of the data, the authors reshaped it from a two-dimensional array into a three-dimensional array, which contains three aspects. First, it represents the number of independent samples at a time (sample size), the number of time units that can be viewed before prediction (time step), and the number of features. This value is set to 1 to indicate that the time points in the data are all single variables.

2.2 Proposed Approach

The proposed weather forecasting method in this paper underscores the fusion of RNN and LSTM architectures, leveraging the complementary strengths of each. While RNN excels in processing sequential text, LSTM's specialized memory units tackle the challenge of processing lengthy sequences more effectively. By harnessing the capabilities of both models, this approach aims to overcome the limitations of conventional RNN models, particularly in handling long-term dependencies in weather data. Through this integrated framework, the research endeavors to achieve more accurate and reliable weather predictions based on extensive datasets. The schematic representation provided in Fig. 1. offers a visual depiction of the intricate interplay between RNN and LSTM components within the system architecture.

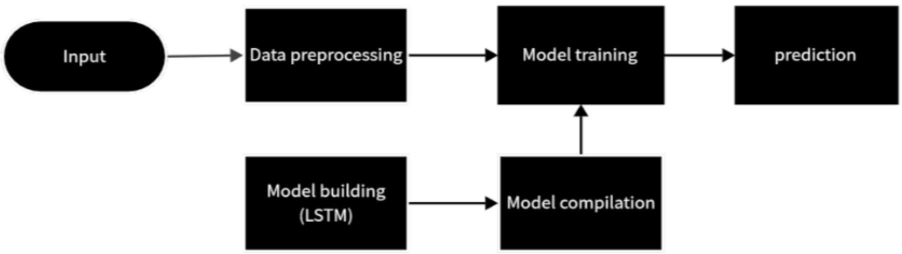

Fig. 1. The pipeline of the model.

The authors first preprocess the data, which will put the data in a form that can be used by the model. In terms of building the model, this study included four layers of LSTMS in the model, and each layer was configured with 50 units which can help the model learn long-term dependencies in the data. At the same time, stacking multiple LSTM layers to form a deep recurrent neural network allows the model to learn in more diverse data, and it can also have better results for abstract time series features, which is a great improvement for RNN. After each LSTM, the authors followed a Dropout layer, with the proportion set to 20%, indicating that 20% of the connections were randomly discarded during the training process, which can effectively reduce overfitting, make the model less dependent on specific data, and effectively improve the generalization ability of the model. After the LSTM layer, the authors add a fully connected output layer (Dense). This was used to generate the predictions of the model. In this study, only one output unit is used because the target to be predicted is a continuous value (temperature). In the part of compiling the model, the authors use the Adam optimizer to optimize the algorithm in order to optimize the parameters during training. The author also uses the mean square error as the loss function, which makes it easier to detect the difference between the model prediction and the original data in the test set. Model validation is performed by feeding the already distinguished training set into the model and using the validation set. The LSTM model trained in this way can update the weights based on the feedback from the loss function in each training epoch, which can help the model to get a great improvement in prediction.

LSTM. LSTM is a commonly used variant of RNN for processing and learning sequences. Compared with the traditional recurrent neural network, LSTM introduces a forget gate, which makes the model more effective in processing time series data in learning the data at an earlier time and predicting the output later. The core components of LSTM are the output gate, update gate, forget gate, and cell state, allowing the LSTM network to selectively remember or forget past information and update its internal memory based on the current input and state when processing time-series data. Fig. 2 shows the logical structure of LSTM. Where C_{t-1} and C_t represent the cellular memory, the former represents the memory at the time of t-1, and the latter shows the memory at the time of t. h_{t-1} and h_t are states, where the former is the state at time t-1 and the latter is the state at time t. The unique forgetting gate of LSTM can make it give up the useless things, keep the useful things, and better learn the useful things. i_t and c_t represent the update gate, which can filter out some irrelevant data during the process of model learning. The tanh represents the output gate, which allows the model to convert its learning and memory capabilities into predictive capabilities, equivalent to applying some of its capabilities. In this study, the use of LSTM enables the output gate to control the data information extracted from the cell state and used for output. These control gate mechanisms make LSTM better able to deal with long sequence data and effectively alleviate the problem of exploding or vanishing gradients. Overall, using LSTM can better process the weather data in the training set and make more accurate predictions.

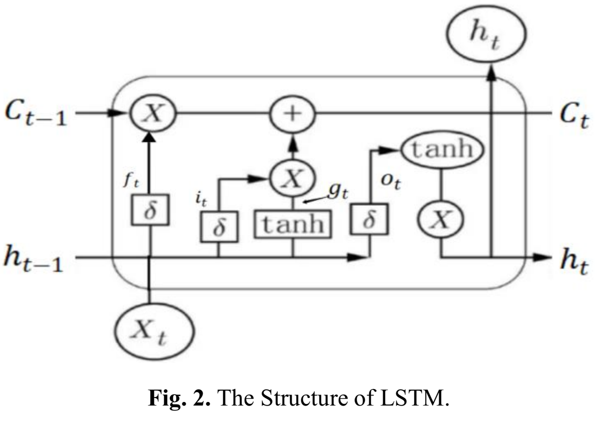

Fig. 2. The Structure of LSTM.

Loss Function. Choosing an appropriate loss function plays a crucial role in model training in deep learning. For sequence prediction tasks, the loss function is an important measure of the difference between the predicted values of the trained model and the original real data. By making the value of the loss function as small as possible, the model can predict the sequence data more accurately. The author chooses the mean square error as the loss function of this experiment. Through this improvement, the results predicted by the model after learning can be better compared with the initial data in the original data set, so that the author can more easily find the difference between the two:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - z_i)^2 \tag{1}$$

The above formula represents the mean squared error calculation. n is the number of input data samples, y_i is the actual value, z_i is the prediction of the output through the learned model. The smaller the MSE, the more accurate the model is.

2.3 Implementation Details

Python version 3.11 was used in this study. The hyperparameters used here include window size, number of LSTM cells, Dropout rate. The window size is used to predict the length of historical data, which is set to 0 in this study, and the number of LSTM units identifies the number of neurons in each layer, which is set to 50 in this study. The Dropout rate is 0.2. The data visualization section mainly uses the 'matplotlib' and 'seaborn' libraries to help understand the data values and make the predictions of the model clearer for everyone to evaluate and show the results clearly to a non-technical audience.

3 Results and Discussion

In this conducted study, a dataset of nearly 1500 days is used and a model is designed based on RNN and LSTM for learning and prediction. By dividing the data set into different parts for operation, and plotting the predicted data as images with the original data for comparative analysis. Fig. 3 shows the comparison of the prediction results between the data that need to be verified after prediction and the predicted data and the original data:

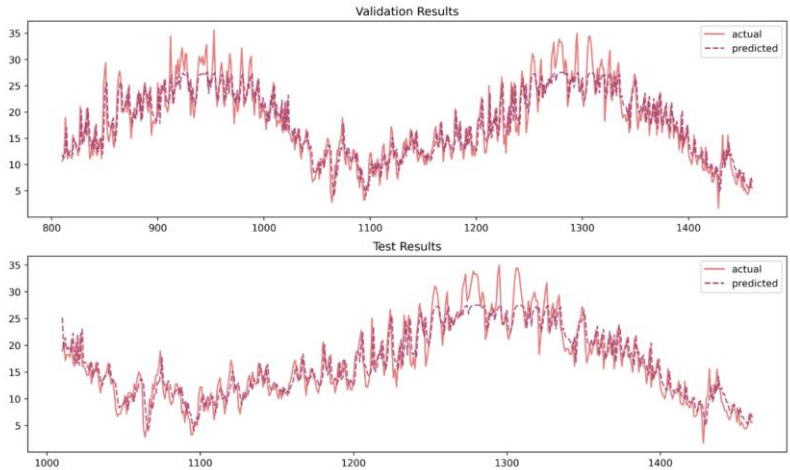

Fig. 3. The result of the validation and test.

As can be seen from Fig. 3, the model using LSTM can have high accuracy in sequence prediction, but there is still high volatility in individual places, which makes the prediction result deviate from the true value by a large margin. In the first subplot, it can be seen that the predicted curve in the validation set can follow the actual temperature change very well, reflecting a high accuracy. This is mainly due to the multiple hidden layers and the unique forgetting gate mechanism of LSTM. The four hidden layers used in this study significantly improve the learning ability and prediction ability of the model. By stacking multiple LSTM layers, the network can better utilize the complex patterns in the sequence, which can help the model better adapt to the complexity of data and improve the accuracy of prediction. However, in some specific periods, there is still a certain deviation between the two, which may be because the model can capture the temperature change law in some specific periods, and may also be caused by some external factors. The second subplot shows the predictions for the test set, which is similar to the validation set, showing a good fit most of the time, but further analysis and improvement are needed to improve the prediction accuracy.

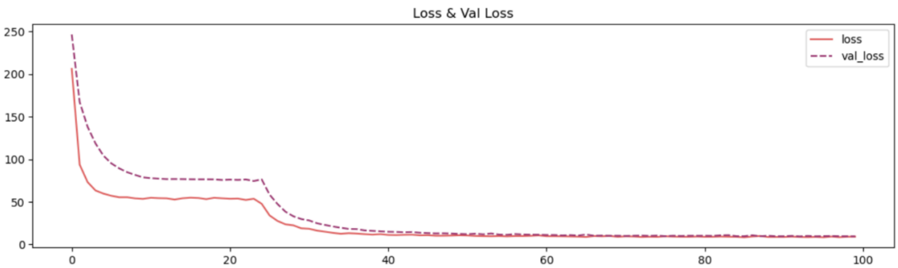

Fig. 4. The curve of the loss and validation.

The loss function is an effective indicator to measure the absorption level of the model for the original data and the prediction accuracy. Fig. 4 shows the loss function image of the proposed model. By observing the bisection plot, the training loss has already been greatly reduced after very little data training, and this trend can be easily detected. After a period of training, it can be found that both loss curves begin to converge, and there is almost no visible observation gap between the two, which can prove that the model performs well and has strong predictive ability. In summary, the use of LSTM can make the model effective for large amounts of data learning and increase the accuracy of sequence prediction. It also shows some areas that can be refined. The integration of multiple different models can be considered in future research, and methods such as model fusion or stacking can be used to further improve the prediction ability. Models can also be used in a wider range of domains, such as finance, energy, and biomedicine.

4 Conclusion

The incorporation of LSTM atop RNN stands as the pivotal contribution of this experiment. Empirical evidence derived from extensive data sets underscores the efficacy of employing LSTM in sequence prediction, manifesting in heightened learning capabilities and enhanced prediction accuracy. Post-training, the model's forecasted temperatures align closely with actual temperature trends, a testament to LSTM's distinctive architecture. The integration of memory and gating units endows the model with superior capacity to extract salient information from prolonged sequences and autonomously discern crucial data features, facilitating adept handling of voluminous data. Furthermore, the stacking of multiple LSTM layers enables the model to adeptly tackle intricate tasks. In contrast to conventional factors, future research should consider a broader spectrum of pertinent factors. Careful consideration of potential limitations is imperative in forthcoming studies. Beyond raw data, attention to statistical data characteristics and external factors is warranted. Additionally, employing data augmentation techniques, such as translation and rotation operations on time series, can bolster the model's predictive prowess in scenarios characterized by limited data or imbalanced samples, enabling adaptability to diverse environments.

References

1. Lynch, P.: The origins of computer weather prediction and climate modeling. Journal of computational physics, 227(7), 3431-3444 (2008).
2. Shuman, F. G.: Numerical weather prediction. Bulletin of the American Meteorological Society, 59(1), 5-17 (1978).
3. Parker, W. S.: Predicting weather and climate: Uncertainty, ensembles and probability. Studies in history and philosophy of science part B: Studies in History and Philosophy of Modern Physics, 41(3), 263-272 (2010).
4. Abraham, A., Khan, M. R., & Maqsood, I.: Weather forecasting models using ensembles of neural networks, 33-42 (2003).

5. Poornima, S., & Pushpalatha, M.: Prediction of rainfall using intensified LSTM based recurrent neural network with weighted linear units. Atmosphere, 10(11), 668 (2019).
6. Aliev, R. A., Aliev, R. R., Fazlollahi, B., & Guirimov, B. Linguistic time series forecasting using fuzzy recurrent neural network. Soft Computing, 12, 183-190 (2008).
7. El Amrani, C., & Zaytar, M. A.: Sequence to sequence weather forecasting with long short-term memory recurrent neural networks. International Journal of Computer Applications, 143(11), 7-11 (2016).
8. Li, J., Li, X., Sun, Q., Wang, J., & Wang, H.: NGCU: A new RNN model for time-series data prediction. Big Data Research, 27, 100296 (2022).
9. Arasu, A. I., Modani, M., & Vadlamani, N. R.: Application of Machine Learning Techniques in Temperature Forecast. In 2022 21st IEEE International Conference on Machine Learning and Applications, pp. 513-518. IEEE (2022).
10. Weather prediction, https://www.kaggle.com/datasets/ananthr1/weather-prediction/data, last accessed 2022/6/14.