MCIT Modeling, control & information technologies

Integrating CNN-BiLSTM Architecture for Predicting Precipitation and Meteorological Patterns

https://doi.org/10.31713/MCIT.2024.084

Thea Camille-Maxime
National University of Water and Environmental Engineering
Rivne, Ukraine
c.m.thea@nuwm.edu.ua

Olena Belozerova
National University of Water and Environmental Engineering
Rivne, Ukraine
ol.d.kozhushko@nuwm.edu.ua

Abstract – **Accurate forecasting of precipitation is essential for various sectors, including agriculture, disaster management, and water resource planning. This paper presents a deep learning architecture that combines Convolutional Neural Networks (CNN) and Bidirectional Long Short-Term Memory (BiLSTM) layers to predict precipitation using a set of weather parameters, including temperature, station level pressure, dew point, and calculated relative humidity. The proposed architecture leverages CNN for feature extraction and BiLSTM for capturing temporal dependencies, offering insights into the model's efficiency and the challenges associated with predicting precipitation using calculated inputs.**

Keywords – meteorology data, weather forecasting, convolutional neural network, LSTM.

I. Introduction

Forecasting precipitation is a complex task due to the chaotic nature of weather systems and the multitude of influencing factors. Accurate short-term predictions of precipitation events like rain or snow are vital for decision-making in agriculture, transportation, and disaster preparedness [1]. Traditional numerical weather prediction models often struggle with fine-scale phenomena, prompting the exploration of machine-learning techniques that learn from historical data patterns [2].

Previous studies have demonstrated the effectiveness of CNN-LSTM architectures in predicting continuous variables such as temperature and wind speed. In our earlier work [3], we successfully applied a CNN-BiLSTM model for gap-filling in temperature time series data, addressing the issue of missing meteorological measurements. However, predicting a binary precipitation event using similar techniques poses significant challenges. Furthermore, precipitation quantities should be adequately estimated for rainy days.

This paper aims to analyze a CNN-BiLSTM model architecture explicitly designed to predict precipitation using weather parameters, including calculated relative humidity. We focus on the roles of custom functions for binary outputs, the integration of CNN and BiLSTM layers, preprocessing steps such as scaling and transformation, and the challenges involved in handling calculated inputs and binary targets.

II. Materials and Methods

A. Calculation of Relative Humidity

Since relative humidity is a critical parameter for predicting precipitation but is not directly available in our data, we calculate it from temperature and dew point temperature using a formula derived from the Clausius-Clapeyron equation [4]:

$$\mathrm{RH} = 100 \times \exp\left(\frac{17.625 \times T_d}{243.04 + T_d} - \frac{17.625 \times T}{243.04 + T}\right).$$

B. Data acquisition and preprocessing

Daily weather records for model training and validation are sourced from the NCDC Climate Data Online – an open data service that aggregates historical records from thousands of weather stations across the world [5]. We selected Ukrainian stations with the least missing data frequency for analysis: Zhuliany, Rivne, Poltava, Vinnytsya, and Sumy. Downloaded meteorology records for 2014 – 2024 from these stations were separated into train and test datasets by 70:30 rule.

Min-Max feature scaling is applied to quantitative meteorology variables: average daily temperature, relative humidity, air pressure, precipitation quantities. For the binary target variable (precipitation events), we apply a logarithmic transformation using the natural logarithm of one plus the value to handle skewness in the data distribution:

$$X_{transformed} = \log(1 + X).$$

C. Model Architecture

The proposed model is designed to predict precipitation using multiple weather parameters as input features. Its architecture is schematically visualized in Figure 1.

Each weather parameter, namely temperature, dew point, station level pressure, and relative humidity, has its input layer corresponding to the historical sequence length. Each input feature is processed through its convolutional layer, applying filters to extract local temporal patterns. After convolution, MaxPooling layers reduce dimensionality by selecting the maximum value

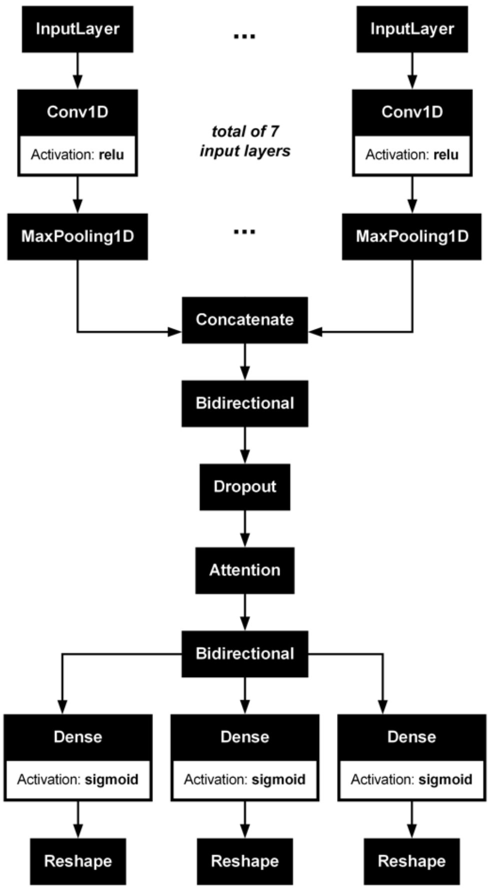

Figure 1. Architecture of the CNN-BiLSTM model

within a pooling window, summarizing the most prominent features.

Outputs from the feature-specific convolutional and pooling layers are concatenated to combine the extracted features from all inputs, forming a unified representation. The merged features are then passed through two Bidirectional LSTM layers, which process the data forward and backward to capture temporal dependencies. The BiLSTM captures information from past and future contexts by processing sequences in both directions. An attention layer is applied to focus on relevant time steps within the sequence, allowing the model to weigh the importance of different time steps when making predictions [6]. We also add dropout layers that randomly set a fraction of input units to zero during training after the LSTM and attention layers to prevent overfitting.

The network has three output layers for predicted parameters: binary rain and snow events, and precipitation quantity. A custom output layer is applied for the binary targets using a sigmoid activation function to output probabilities between 0 and 1:

$$\hat{y} = \sigma(W_{\text{out}} \cdot h + b_{\text{out}}),$$

where σ is the sigmoid function, W_{out} and b_{out} are the weights and biases, and h is the output from the previous layer [7].

Handling binary targets like precipitation events requires specialized activation functions and loss functions. The sigmoid activation function maps the output to a range between 0 and 1, suitable for representing probabilities. The binary cross-entropy loss function measures the discrepancy between the predicted probabilities and the actual binary labels:

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)],$$

where y_i is the true label, $\hat{y_i}$ is the predicted probability, and N is the number of samples.

D. Training the Model

The model is trained using binary cross entropy loss function, which encourages the model to output probabilities close to the true labels. An adaptive optimizer like Adam is used with a learning rate of $\alpha = 0.0001$ and gradient clipping with a norm threshold of 1 to prevent exploding gradients.

While binary cross-entropy is used as the loss function during training, we evaluate the model's performance using Mean Squared Error (MSE) and Huber loss to measure the regression aspects of the predictions, especially when dealing with continuous representations or probabilities. The Huber loss combines the best properties of MSE and Mean Absolute Error (MAE), quadratic for minor and linear for significant errors, thus being more robust to outliers.

It should be noted that precipitation events are significantly less frequent compared to dry day events for temperate continental climate. This leads to class imbalance that can bias the model toward predicting the majority class. The proposed model currently does not address this issue; however, the recommended strategy in this case would be to oversample the minority class using Synthetic Minority Over-sampling Technique (SMOTE) [2] or other similar approaches.

III. Results and discussion

From the validation data set, 40 intervals are selected randomly, and the precipitation forecast for 31 days ahead is made. The calculated series are concatenated and evaluated with the following metrics: mean absolute error (MAE), root mean squared error (RMSE) and R^2 score. Table 1 presents the results for these validation metrics.

In the parameters column, '*prcp q*' stands for precipitation quantity, in mm, and '*rain*' and '*snow*' for probability of rain and snow events, respectively. For the event probabilities, only the days with significant (above 0.01 mm) precipitation quantities are considered.

The results demonstrate that our model achieves satisfactory accuracy in predicting precipitation and high accuracy in predicting rain and snow events across different locations. The validation metric values for rain and snow are particularly optimistic; however, due to filtering out predicted dry days from validation, these only account for type I classification error.

TABLE I. RESULTING VALIDATION METRICS FOR SELECTED STATIONS.

Station	Parameter	MAE	RMSE	R^2 score
Zhuliany	***prcp q***	1.13	2.57	0.615
	rain	0.048	0.150	0.980
	snow	0.041	0.128	0.963
Rivne	***prcp q***	1.04	2.19	0.505
	rain	0.036	0.066	0.995
	snow	0.019	0.030	0.996
Poltava	***prcp q***	0.98	2.20	0.575
	rain	0.031	0.085	0.985
	snow	0.027	0.045	0.999
Vinnytsia	***prcp q***	0.93	2.22	0.128
	rain	0.077	0.133	0.910
	snow	0.089	0.133	0.807
Sumy	***prcp q***	1.00	2.16	0.420
	rain	0.023	0.043	0.999
	snow	0.039	0.105	0.973
Average	***prcp q***	1.02	2.27	0.449
	rain	0.043	0.095	0.974
	snow	0.043	0.088	0.948

For precipitation prediction, the average MAR and RMSE are around 1 and 2.27 mm, which is a relatively low error for precipitation quantity. However, the R2 scores vary among stations, ranging from 0.128 in Vinnytsia to 0.615 in Zhuliany. While the model performs well in most cases, the lower R2 values in some of the cases indicate that the quality of prediction has certain issues and can further be improved.

Comparing these findings with our previous study on temperature gap-filling, the model demonstrates robust performance across different meteorological variables. The ability to effectively predict multiple binary targets highlights the versatility of the CNN-BiLSTM architecture.

IV. CONCLUSION

This study presents a CNN-BiLSTM model architecture for predicting precipitation using calculated relative humidity and other meteorological parameters. The model leverages feature-specific convolutional layers to extract local temporal patterns, BiLSTM layers to capture long-term dependencies, and an attention mechanism to focus on relevant time steps. Preprocessing steps, including calculating relative humidity and scaling strategies, are critical for model performance. The model achieves high accuracy, as evidenced by the low MAE and RMSE values in predicting precipitation, rain, and snow.

Our findings demonstrate that the proposed architecture effectively handles mixed data types and calculated inputs, making it a valuable tool for meteorological forecasting. This approach contributes to advancing machine learning applications in meteorology, offering a framework that can be adapted and extended for various predictive tasks.

Future work will incorporate additional parameters such as wind speed, atmospheric pressure changes, and cloud cover to enhance prediction accuracy further. We also plan to apply the model to different geographical regions and fine-tune it with local data through transfer learning. Experimenting with alternative architectures like transformer-based models or other attention mechanisms may yield further improvements.

V. REFERENCES

[1] P. Bauer, A. Thorpe, and G. Brunet, “The quiet revolution of numerical weather prediction,” Nature, vol. 525, no. 7567, pp. 47–55, 2015.

[2] N. V. Chawla, K. W. Bowyer, L. O Hall, and W. P Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling Technique,” Journal of Artificial Intelligence Research, vol. 16, pp. 321–357, 2002.

[3] C.-M. Thea and O. Belozerova, “Filling gaps in daily temperature data with a CNN-LSTM model,” Proceedings of the International Scientific Conference Information Technologies and Computer Modelling 2024, May 2024.

[4] M. G. Lawrence, “The Relationship between Relative Humidity and the Dewpoint Temperature in Moist Air: A Simple Conversion and Applications,” Bulletin of the American Meteorological Society, vol. 86, no. 2, pp. 225-235, 2005.

[5] National Centers for Environmental Information, National Oceanic and Atmospheric Administration. “Climate Data Online (CDO) provides free access to NCDC's archive of global historical weather and climate data in addition to station history information”. NOAA.gov. https://www.ncei.noaa.gov/cdo-web/ (retrieved October 5, 2024).

[6] S. Sakib, M. K. Mahadi, S. R. Abir et al., “Attention-Based Models for Multivariate Time Series Forecasting: Multi-step Solar Irradiation Prediction,” Heliyon, vol. 10, no. 6, e27795, 2024.

[7] Y. Qin, D. J. Song, H. F. Chen et al., “A Dual-Stage Attention-Based Recurrent Neural Network for Time Series Prediction,” in Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI'17), AAAI Press, pp. 2627–2633, 2017.