UJM 13 (2) 2024
UNNES Journal of Mathematics
http://journal.unnes.ac.id/sju/index.php/ujm

Stacking Ensemble Modeling of Bidirectional LSTM and Bidirectional GRU for Air Temperature Prediction in Ngawi

Nike Yustina Oktaviani *, Iqbal Kharisudin

Department of Mathematics, Faculty of Mathematics and Natural Sciences,
Universitas Negeri Semarang, Semarang, 50229, Indonesia

Article Info

Article History:
Received: 07-02-2025
Accepted:
Published:

Keywords:
Stacking, Bidirectional Long Short-Term Memory (BiLSTM), Bidirectional Gated Recurrent Unit (BiGRU), temperature

Abstract

Artificial Neural Networks (ANN) have rapidly developed and are used in forecasting, classification, and regression by mimicking how the human brain processes data. Recurrent Neural Networks (RNN), such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), are effective in processing sequential data and handling long-term dependencies. Bidirectional LSTM (BiLSTM) and Bidirectional GRU (BiGRU) process data in both directions to enhance accuracy. This study evaluates the implementation of the Stacking Ensemble method using BiLSTM and BiGRU as base learners and Random Forest as the meta learner to predict air temperature in Ngawi Regency. Air temperature prediction is crucial as it affects agriculture, health, and energy sectors. The data used comprises 2282 records from January 1, 2028, to March 31, 2024, processed using Google Colab. The results show that the Stacking BiLSTM-BiGRU model with Random Forest provides the best performance with a Mean Squared Error of 0.0005, Root Mean Squared Error of 0.0233, Mean Absolute Error of 0.0179, and R-squared of 0.9832, outperforming other individual models. This study confirms that the Stacking Ensemble method with BiLSTM and BiGRU significantly improves air temperature prediction accuracy.

How to cite:

Oktaviani, Nike Yustina, & Kharisudin, Iqbal. (2024). Stacking Ensemble Modeling of Bidirectional LSTM and Bidirectional GRU for Air Temperature Prediction in Ngawi. *UNNES Journal of Mathematics*. 13 (2): 1-8

*Correspondence Address
E-mail: nikeyustina12.o@gmail.com

p-ISSN 2252-6943
e-ISSN 2460-5859

1. Introduction

Artificial Neural Network (ANN) has significantly advanced in recent decades and has been applied in various fields such as forecasting, classification, and regression. ANN operates by mimicking the way the human brain processes data for tasks like object detection, speech recognition, and decision-making (Mijwel, 2021). One of the key developments in ANN is the Recurrent Neural Network (RNN), which is designed to process sequential data and is effective in handling long-term dependencies by remembering previous information and using it for forecasting or prediction.

RNN has evolved into Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) to address the vanishing gradient problem. These methods can store information for longer periods and more stably, making them highly effective in predicting complex sequential data. LSTM and GRU have further been developed into Bidirectional LSTM (BiLSTM) and Bidirectional GRU (BiGRU), which process data in two directions, enhancing prediction accuracy. Research by Liang *et al*. (2021) demonstrates that BiLSTM can predict air temperature better than LSTM, GRU, and RNN.

Additionally, Ensemble Learning methods, particularly Stacking Ensemble, can improve air temperature prediction accuracy. Stacking Ensemble combines predictions from multiple base learners to produce better predictions than individual models. In this study, Stacking Ensemble BiLSTM and BiGRU are used to predict air temperature changes as an impact of climate change. Research by Ye *et al*. (2022) and Gu *et al*. (2022) shows that Stacking Ensemble methods provide better prediction results compared to the individual base models.

Developing countries like Indonesia are vulnerable to the impacts of climate change due to their heavy reliance on climate for development, especially in the agricultural sector (Nyasulu *et al*., 2022). Ngawi, one of the largest rice-producing regions in Indonesia, relies heavily on stable air temperatures for crop growth and harvest periods. Unexpected or extreme air temperature changes can seriously impact agricultural production. Therefore, a good understanding of air temperature patterns in Ngawi will help farmers anticipate and manage risks associated with climate fluctuations.

This study aims to model air temperature predictions in Ngawi Regency using Stacking Ensemble BiLSTM and BiGRU methods. The dataset is trained, validated, and tested to measure the accuracy of these methods compared to the base learner methods used in the study. Based on the above descriptions, the authors conducted research on Stacking Ensemble Bidirectional LSTM and Bidirectional GRU modeling for air temperature prediction in Ngawi Regency.

2. Method

2.1. Data dan Variabel Penelitian

The data used in this study is the daily air temperature data in ngawi regency at longitude: 111.332197 and latitude: -7.460987, from january 1, 2018, to march 31, 2024, comprising 2282 data points obtained from the NASA Langley Research Center's POWER Project (https://power.larc.nasa.gov/data-access-viewer/). The variable analyzed in this study is the daily air temperature at a height of 2 meters above the ground. The data was cleaned, normalized, and split into training data (90%) and testing data (10%).

2.2. Long Short-Term Memory (LSTM)

LSTM (Long Short-Term Memory) is an enhancement of Recurrent Neural Network (RNN) designed to address long-term dependency issues and the vanishing gradient problem (Seabe *et al*., 2023). LSTM can store past information and use it in predictions without losing information (Brownlee, 2017). The LSTM model consists of memory cells and three gates: the forget gate, input gate, and output gate. The forget gate determines what information needs to be erased from the memory cell, the input gate controls what new information is to be stored, and the output gate decides what information will be used for the next output. The LSTM model operates through a series of gates and a cell state to manage and preserve information over long sequences. The mechanism involves four main steps:

1. Forget Gate: The forget gate decides which information from the previous cell state should be discarded. This is achieved through a sigmoid function that outputs a value between 0 and 1. The formula is:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{1}$$

2. Input Gate: The input gate controls the new information that is added to the cell state. It involves two parts: a sigmoid layer to decide which values to update and a tanh layer to create new candidate values. The formulas are:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\widetilde{C_t} = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{3}$$

3. Update Cell State: The cell state is updated by combining the previous cell state, scaled by the forget gate, and the new candidate values, scaled by the input gate. The formula is:

$$C_t = f_t \odot C_{t-1} + i_t \odot \widetilde{C_t} \tag{4}$$

4. Output Gate: The output gate determines the next hidden state, which is used for predictions and as input to the next time step. It uses a sigmoid function to decide what parts of the cell state to output and a tanh function to scale the output. The formulas are:

$$\boldsymbol{o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)} \tag{5}$$

$$\boldsymbol{h_t = o_t \odot \tanh\,(C_t)} \tag{6}$$

2.3. Gated Recurrent Unit (GRU)

Gated Recurrent Unit (GRU), introduced in 2014, is designed to handle sequences of varying lengths and retain past information (Chung *et al*., 2015). Unlike LSTM, which uses multiple gates and memory cells, GRU simplifies the architecture by using only an update gate and a reset gate (Song & Choi, 2023). This makes GRU simpler and easier to train while achieving similar performance. GRU combines LSTM's forget and input gates into a single update gate and uses fewer tensor operations, leading to faster training (Song & Choi, 2023). The GRU model operates through a series of gates to manage and preserve information over long sequences. The mechanism involves four main steps:

1. Reset Gate: The reset gate controls the influence of the previous hidden state. It determines which part of the previous hidden state to forget, using the sigmoid function to produce values between 0 and 1. The formula is:

$$\boldsymbol{r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r)} \tag{7}$$

2. Update Gate: The update gate controls the amount of the new input to be added to the cell state. It decides how much of the past information needs to be passed along to the future. The formula is:

$$\boldsymbol{z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z)} \tag{8}$$

3. Candidate Memory Vector: The candidate memory vector is updated based on the reset gate's activation and the current input. It determines the new candidate values for the cell state, using a tanh activation function. The formula is:

$$\boldsymbol{\widetilde{h_t} = \tanh(W \cdot [r_t \odot h_{t-1}, x_t])} \tag{9}$$

4. Final Output: The final output is obtained by combining the candidate memory vector with the previous hidden state, controlled by the update gate. The formula is:

$$\boldsymbol{h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \widetilde{h_t}} \tag{10}$$

2.4. Bidirectional LSTM & Bidirectional GRU

Bidirectional LSTM (BiLSTM) and Bidirectional GRU (BiGRU) are variants of RNNs that process sequential data in both forward and backward directions. This allows the networks to leverage information from both previous and future data points during prediction or classification. This feature is particularly useful in tasks where the current context depends on both past and future events (Soares & Franco, 2021). In BiLSTM or BiGRU, two layers of LSTM or GRU cells are combined: one layer processes the data forward, while the other processes it backward. This bidirectional approach enhances the model's ability to understand and interpret complex sequential dependencies.

2.5. Random Forest

Random Forest is a machine learning algorithm that uses a combination of several decision trees to make predictions that are more accurate and stable (Ao *et al*., 2019). This algorithm falls under ensemble learning, which combines multiple learning models to enhance overall performance. Each decision tree in a random forest is built based on values from random vectors that are independently drawn and distributed similarly.

Random Forest is a well-known implementation of Bagging. In Bagging, multiple data subsets are created through bootstrap sampling from the original dataset, allowing some data points to appear more than once in one subset and not at all in others. Different models (decision trees) are then built using each bootstrap subset. Since each decision tree is trained on a different data subset, the results from all individual models are combined through majority voting for classification tasks or averaging for regression tasks (James *et al.*, 2023).

2.6. Stacking Ensemble

Stacking is a model that uses base learners to generate metadata for a problem dataset and then employs another learner, known as a meta learner, to process this metadata (Kyriakides & Margaritis, 2019). Base learners act as level 0 learners, while the meta learner functions as a level 1 learner. Essentially, the meta learner is stacked on top of the base learners, hence the term "stacking." In a stacking ensemble, the predictions from each base learner are typically combined using a meta learner or averaged (Zhang *et al*., 2023). The Stacking Ensemble model consists of two levels: the first level includes two RNNs; sub-model 1 is a BiLSTM (Bidirectional LSTM), and sub-model 2 is a BiGRU (Bidirectional GRU).

The dataset is divided into two parts: training data (90%) and test data (10%). The training data is used to train the level 1 sub-models: BiLSTM and BiGRU. After the initial training phase, the trained level 1 models are used to generate predictions that will be used by the level 2 model. The test data is then used for final predictions and accuracy calculations.

First, the training data is used to train sub-model 1: the BiLSTM model, which has a single LSTM layer processing data in both directions, with 50 neurons in that layer. We apply a dropout rate of 0.2 and train the model for 100 epochs. Next, we train sub-model 2: the BiGRU model, which also has a single GRU layer processing data in both directions, with

50 neurons in that layer, dropout of 0.2, and 100 epochs. After training the BiLSTM and BiGRU models, we use the training data to generate predictions from both the BiLSTM and BiGRU models.

The predictions from BiLSTM and BiGRU are combined into a new training dataset in the form of p × m (where p represents the number of predictions and m represents the number of models). This new dataset is used to train the meta-learner, which is a Random Forest model with 100 decision trees. The meta-learner integrates predictions from the sub-models. After training the meta-learner, the test data is fed back into the sub-models to generate intermediate test data, which the meta-learner uses to make the final predictions.

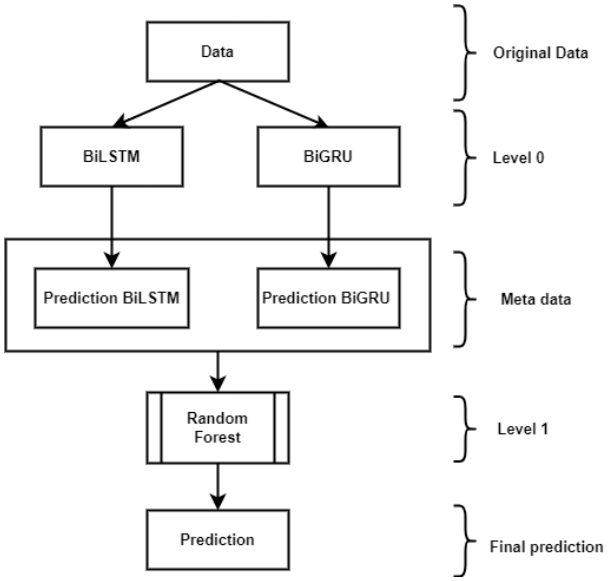


Figure 1 Architecture overview

3. Results and discussions

In this section, we present the experimental results of the proposed models using daily air temperature data. The performance of these models is evaluated using several key metrics, namely Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2). These metrics assess how well the models predict daily air temperature and facilitate the comparison of different models. Additionally, the models were tested with various window sizes 7 days, 15 days, and 30 days to evaluate the impact of window size on model performance.

The dataset was divided into training data (90%) and test data (10%). The benchmark models compared include RNN, LSTM, BiLSTM (Bidirectional LSTM), GRU, BiGRU (Bidirectional GRU), and Stacking LSTM-GRU. All these models were trained with the following parameters to ensure a fair comparison: 1 layer, 50 neurons per layer, 100 epochs, a learning rate of 0.0005, a batch size of 32, the MSE loss function, and the Adam optimizer. This ensures that the training settings for each model are consistent with those used for the Stacking BiLSTM-BiGRU model.Benchmark Models.

Below is the comparison table of the Stacking BiLSTM-BiGRU model with the benchmark models:

Table 1 Evaluation results

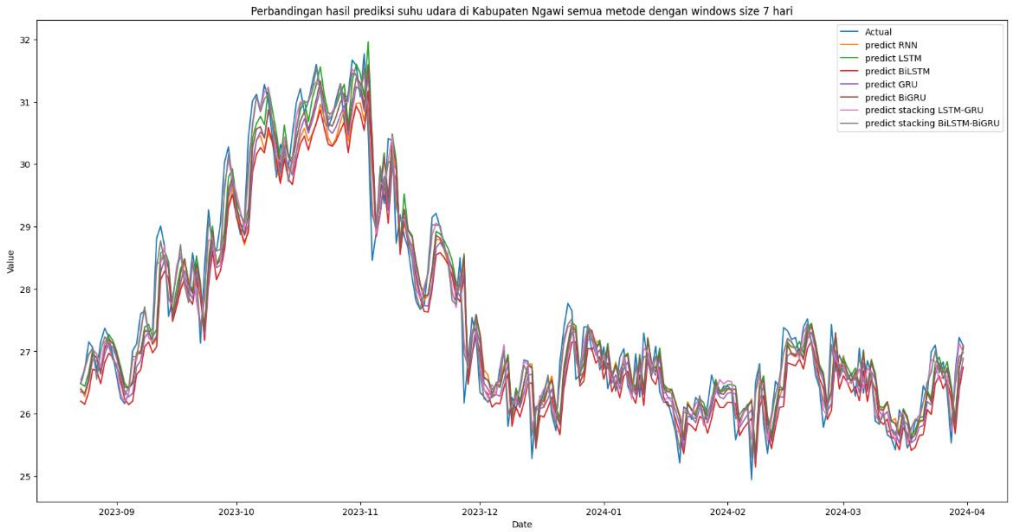
Evaluation Metrics	method	Window size		
		7 days	15 days	30 days
MSE	RNN	0.0038	0.0040	0.0038
	LSTM	0.0035	0.0039	0.0042
	BiLSTM	0.0044	0.0039	0.0037
	GRU	0.0037	0.0045	0.0041
	BiGRU	0.0035	0.0037	0.0043
	Stacking LSTM-GRU	0.0006	0.0006	0.0007
	Stacking BiLSTM-BiGRU	**0.0005**	0.0006	0.0006
RMSE	RNN	0.0616	0.0630	0.0620
	LSTM	0.0588	0.0621	0.0646

	BiLSTM	0.0666	0.0625	0.0608
	GRU	0.0611	0.0674	0.0642
	BiGRU	0.0595	0.0605	0.0659
	Stacking LSTM-GRU	0.0253	0.0246	0.0269
	Stacking BiLSTM-BiGRU	**0.0233**	0.0239	0.0247
MAE	RNN	0.0477	0.0487	0.0476
	LSTM	0.0453	0.0480	0.0500
	BiLSTM	0.0526	0.0480	0.0471
	GRU	0.0472	0.0523	0.0497
	BiGRU	0.0460	0.0468	0.0508
	Stacking LSTM-GRU	0.0193	0.0188	0.0208
	Stacking BiLSTM-BiGRU	**0.0179**	0.0187	0.0190
R^2	RNN	0.8885	0.8870	0.8963
	LSTM	0.8987	0.8902	0.8873
	BiLSTM	0.8697	0.8888	0.9002
	GRU	0.8904	0.8708	0.8886
	BiGRU	0.8961	0.8959	0.8827
	Stacking LSTM-GRU	0.9803	0.9819	0.9796
	Stacking BiLSTM-BiGRU	0.9832	0.9830	0.9829

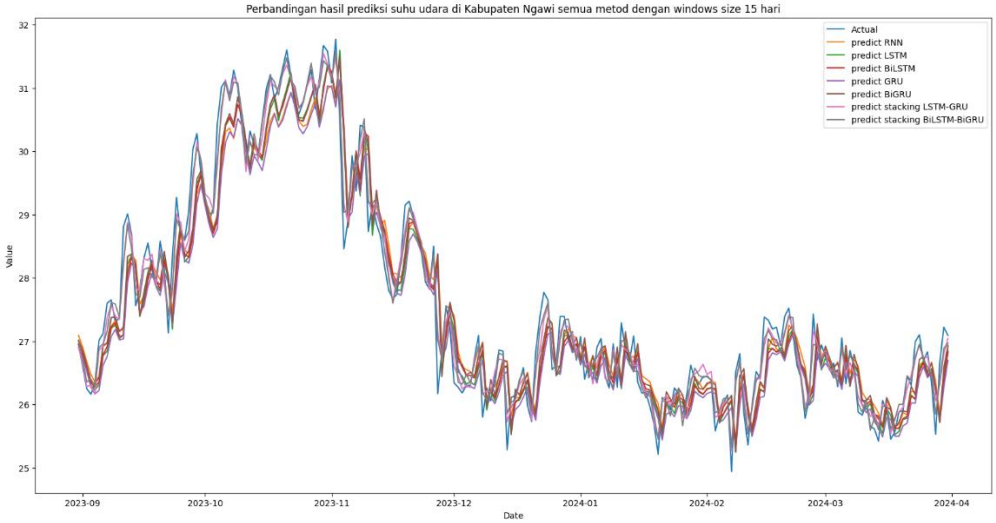
Consistent with the research of Ye *et al.* (2022) and Gu *et al*. (2022), this study also found that Stacking Ensembles produced better predictions than individual base models In contrast to the research of Liang *et al.* (2021) which only compares BiLSTM with LSTM, GRU, and RNN. In this study, Stacking BiLSTM-BiGRU was compared with Stacking LSTM-GRU, BiGRU, GRU, BiLSTM, LSTM, and RNN. It was found that Stacking BiLSTM-BiGRU produced the best predictions, as shown in Table 1. It demonstrated the lowest errors in terms of MSE, RMSE, and MAE, signaling more accurate and stable predictions. The R^2 value close to 1 further suggests that the Stacking BiLSTM-BiGRU model effectively explains most of the variability in the daily air temperature data.

The results also reveal that window size has a significant impact on model performance. The window size determines how much historical data is used to make predictions for the next day. Smaller window sizes, such as 7 days, capture short-term trends, while larger window sizes, such as 30 days, capture long-term trends. The Stacking BiLSTM-BiGRU model exhibited stability across different window sizes, maintaining high accuracy regardless of the window size. In contrast, models such as RNN, LSTM, and GRU showed considerable performance variations with changes in window size. For example, the performance of the RNN model fluctuated notably, with increased errors and lower R^2 values as the window size grew. LSTM and GRU models also displayed variability, indicating that their ability to generalize was influenced by the window size.

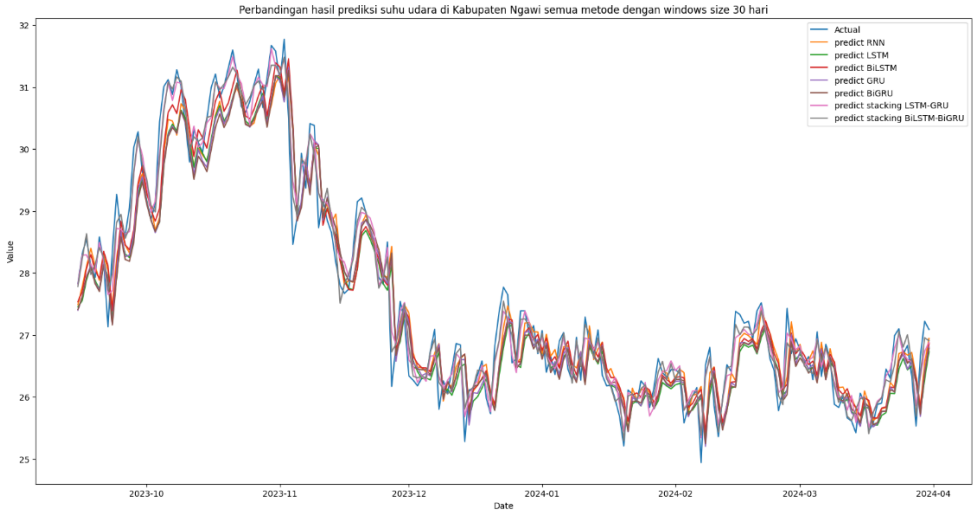
For a more detailed visualization of these comparison results, Figure 2 presents graphs comparing evaluation metrics for various window sizes. This figure provides an intuitive view of each model's performance under different conditions, helping to identify the most optimal model for predicting daily air temperature.



(a) Window size 7



(b) Window size 15



(c) Window size 30

Figure 2 Comparison plot of air temperature predictions in Ngawi

4. Conclusion

This study evaluated the performance of various machine learning models, including RNN, LSTM, BiLSTM, GRU, BiGRU, Stacking LSTM-GRU, and Stacking BiLSTM-BiGRU, for daily air temperature prediction tasks in Ngawi Regency. The Stacking BiLSTM-BiGRU model emerged as the most effective, consistently delivering the lowest errors in terms of MSE, RMSE, and MAE. Its R^2 value close to 1 underscores its strong ability to explain the variability in daily air temperature data, making it a highly reliable choice for accurate and stable predictions.

Additionally, the study revealed the significant impact of window size on model performance. Smaller window sizes, such as 7 days, were effective at capturing short-term trends and rapid changes in temperature data. In contrast, larger window sizes, such as 30 days, were better at capturing long-term trends and more stable patterns. The Stacking BiLSTM-BiGRU model exhibited exceptional stability across various window sizes, maintaining high performance regardless of the window size used. This demonstrates its superiority in handling variations in historical data.

Conversely, models such as RNN, LSTM, and GRU showed significant performance fluctuations with changes in window size, indicating that their effectiveness is more sensitive to the choice of window size. For instance, the RNN model experienced increased errors and decreased R^2 values as the window size grew, while LSTM and GRU models also displayed variability in their generalization capabilities.

Overall, these findings highlight the effectiveness of the Stacking BiLSTM-BiGRU model in providing accurate temperature predictions and its robustness against window size variations. The results suggest that Stacking BiLSTM-BiGRU is a superior model for daily air temperature prediction with high precision. Future work could explore further optimizations and test additional configurations to enhance model performance even more, such as adjusting model architectures, exploring a wider range of window sizes, and integrating advanced data preprocessing techniques..

References

Ao, Y., Li, H., Zhu, L., Ali, S., & Yang, Z. (2019). The linear random forest algorithm and its advantages in machine learning assisted logging regression modeling. *Journal of Petroleum Science and Engineering*, *174*, 776-789. Brownlee, J. (2017). Long Short-Term Memory Networks With Python. In *Machine Learning Mastery With Python* (Vol. 1, Issue 1).

Brownlee, J. (2017). Long Short-Term Memory Networks With Python. In *Machine Learning Mastery With Python* (Vol. 1, Issue 1).

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2015, June). Gated feedback recurrent neural networks. In *International conference on machine learning* (pp. 2067-2075). PMLR. Gu, J., Liu, S., Zhou, Z., Chalov, S. R., & Zhuang, Q. (2022). A Stacking Ensemble Learning Model for Monthly Rainfall Prediction in the Taihu Basin, China. *Water (Switzerland)*, *14*(3), 1–20. https://doi.org/10.3390/w14030492

Gu, J., Liu, S., Zhou, Z., Chalov, S. R., & Zhuang, Q. (2022). A Stacking Ensemble Learning Model for Monthly Rainfall Prediction in the Taihu Basin, China. *Water (Switzerland)*, *14*(3), 1–20. https://doi.org/10.3390/w14030492

James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). Tree-based methods. In *An Introduction to Statistical Learning: with Applications in Python* (pp. 331-366). Cham: Springer International Publishing.

Kyriakides, G., & Margaritis, K. G. (2019). *Hands-on Ensemble Learning with Python*.

Liang, S., Wang, D., Wu, J., Wang, R., & Wang, R. (2021). Method of bidirectional lstm modelling for the atmospheric temperature. *Intelligent Automation and Soft Computing*, *30*(2), 701–714. https://doi.org/10.32604/iasc.2021.020010

Mijwel, M. M. (2021). Artificial neural networks advantages and disadvantages. *Mesopotamian Journal of Big Data*, *2021*, 29-31.Nyasulu, C., Diattara, A., Traore, A., Deme, A., & Ba, C. (2022). Towards Resilient Agriculture to Hostile Climate Change in the Sahel Region: A Case Study of Machine Learning-Based Weather Prediction in Senegal. *Agriculture (Switzerland)*, *12*(9). https://doi.org/10.3390/agriculture12091473

Nyasulu, C., Diattara, A., Traore, A., Deme, A., & Ba, C. (2022). Towards Resilient Agriculture to Hostile Climate Change in the Sahel Region: A Case Study of Machine Learning-Based Weather Prediction in Senegal. *Agriculture (Switzerland)*, *12*(9). https://doi.org/10.3390/agriculture12091473

Seabe, P. L., Moutsinga, C. R. B., & Pindza, E. (2023). Forecasting Cryptocurrency Prices Using LSTM, GRU, and Bi-Directional LSTM: A Deep Learning Approach. *Fractal and Fractional*, 1–18. https://doi.org/https://doi.org/10.3390/fractalfract7020203

Soares, L. D., & Franco, E. M. C. (2021). BiGRU-CNN neural network applied to short-term electric load forecasting. *Production*, *32*, e20210087. Seabe, P. L., Moutsinga, C. R. B., & Pindza, E. (2023). Forecasting

Cryptocurrency Prices Using LSTM, GRU, and Bi-Directional LSTM: A Deep Learning Approach. *Fractal and Fractional*, 1–18. https://doi.org/https://doi.org/10.3390/fractalfract7020203

Song, H., & Choi, H. (2023). Forecasting Stock Market Indices Using the Recurrent Neural Network Based Hybrid Models: CNN-LSTM, GRU-CNN, and Ensemble Models. *Applied Sciences (Switzerland)*, *13*(7). https://doi.org/10.3390/app13074644

Ye, Z., Wu, Y., Chen, H., Pan, Y., & Jiang, Q. (2022). A Stacking Ensemble Deep Learning Model for Bitcoin Price Prediction Using Twitter Comments on Bitcoin. *Mathematics*, *10*(8). https://doi.org/10.3390/math10081307

Zhang, C., Sjarif, N. N. A., & Ibrahim, R. (2023). Deep learning models for price forecasting of financial time series: A review of recent advancements: 2020–2022. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1–33. https://doi.org/10.1002/widm.1519