International Journal of Artificial Intelligent and Informatics
Vol. 3, No. 2, January 2025, pp. 60~ 66
ISSN: 2622-626X, DOI: http://doi.org/10.33292/ijarlit.v3i1.48

Performance Comparison of Standard LSTM and LSTM with Random Search Optimization for Spark New Zealand Limited Stock Price Prediction

Axia Flavia[1*], Camila Mio[1]
[1]Universidad de Ingeniería y Tecnología, Peru
Email: axiafla@yahoo.com

Article Info

Article history:

Received: January 20, 2025
Revised: March 12, 2025
Accepted: July 10, 2025
Available Online: July 30, 2025

Keywords:

LSTM
Random Search
Hyperparameter Optimization
Stock Price Prediction
Deep Learning

ABSTRACT

This study compares the performance of the standard Long Short-Term Memory (LSTM) model with the LSTM model optimized using the Random Search method to predict the stock price of Spark New Zealand Limited. The data used is historical stock price data from Yahoo Finance for the period 2018-2023. Model evaluation is performed using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), coefficient of determination (R^2), and accuracy metrics. The results showed that the standard LSTM model achieved Test RMSE performance of 0.04, Test MAE of 0.03, Test MAPE of 0.73%, Test R^2 of 0.8571, and Test Accuracy of 99.27%. While the LSTM model with Random Search optimization achieved Test RMSE performance of 0.04, Test MAE of 0.03, Test MAPE of 0.78%, Test R^2 of 0.8302, and Test Accuracy of 99.22%. Although both models performed very well, the standard LSTM model was slightly superior in some evaluation metrics on the test data. This research provides insight into the effectiveness of hyperparameter optimization in the context of stock price prediction.

1. INTRODUCTION

Stock price prediction is a complex and important area of research in computational finance. The ability to accurately predict stock price movements can provide a strategic advantage for investors and capital market analysts[1] . In recent years, deep learning-based approaches, particularly Long Short-Term Memory (LSTM) models, have shown promising performance in predicting financial time series data .[2]

Spark New Zealand Limited is one of the largest telecommunications companies in New Zealand whose shares are actively traded. The volatility of the company's share price presents both challenges and opportunities for investors. Developing an accurate prediction model for Spark New Zealand shares can assist stakeholders in making more informed investment decisions.

LSTM models have been proven effective in capturing both long-term and short-term temporal patterns in time series data[3]–[5] . However, the performance of LSTM models is highly dependent on the selection of appropriate hyperparameters, such as the number of units, dropout rate, learning rate, and batch size[6] . The selection of optimal hyperparameters can significantly improve the prediction accuracy.

Random Search is one of the efficient hyperparameter optimization methods, where hyperparameters are randomly selected from a pre-defined distribution[7] . Several studies have shown that Random Search can produce models with equivalent or even better performance compared to more complex hyperparameter search methods in a shorter computation time .[8]

Although many studies have examined the application of LSTMs for stock price prediction[9], [10] , studies that specifically compare the performance of standard LSTMs with LSTMs optimized using Random Search for Spark New Zealand stocks are limited. This study aims to fill the gap by analyzing the comparison between the two approaches. The problem formulation includes evaluating the performance of the standard LSTM model, the LSTM model with Random Search optimization, and the impact of hyperparameter optimization on prediction accuracy. By using Yahoo Finance historical data for the period 2018-2023, this research is expected to contribute to the development of a more effective stock price prediction model, as well as provide insight for researchers and practitioners in the field of computational finance.

2. METHOD

2.1. Data Collection

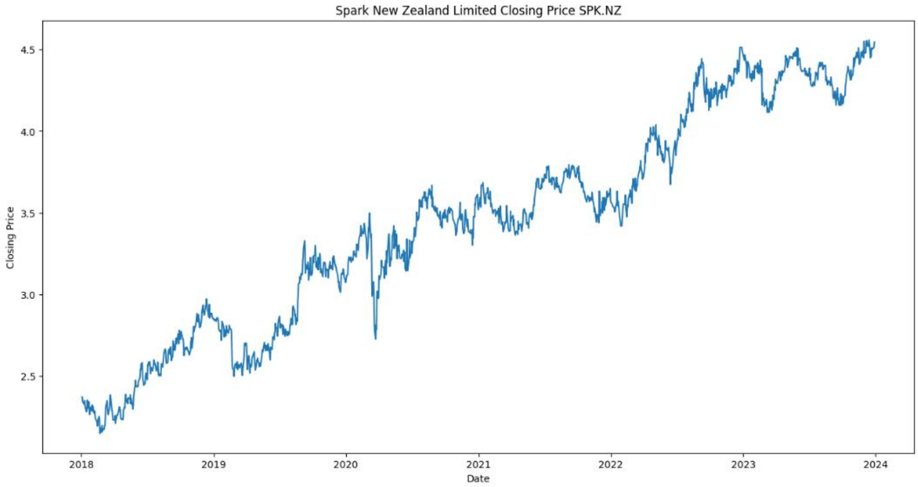

Figure 1. Graph of historical stock movement data of Spark New Zealand Limited

The data used in this study is historical Spark New Zealand Limited stock price data obtained from Yahoo Finance for the period 2018 to 2023. This dataset includes information on the opening price, closing price, highest price, lowest price, and daily trading volume. For the purpose of this research, the main focus is on the closing price of the stock as the prediction target variable.

The data collection process is carried out using the Yahoo Finance API which allows the extraction of historical stock data in a structured manner. The collected data was then saved in CSV format to facilitate further processing and analysis. A total of around 1,260 daily data points were collected over the five-year period.

2.2. Data Pre-processing

Data pre-processing is a crucial step in ensuring that the data used in modeling is of good quality. In this stage, several methods and techniques are applied to handle common problems such as missing values, normalization, and dataset sharing. The data pre-processing stage includes several important steps to ensure the quality of the data to be used in modeling:

1. Handling Missing Data: One method that is often used to handle missing values is linear interpolation. This method is considered effective because it assumes that the change between values is linear, so it can fill in the gaps quite accurately[11]–[14] . Research by Liu et al. demonstrated the application of linear interpolation to impute missing values in CGM (Continuous Glucose Monitoring) data, which validated the success of this method in various fields[12] . In the context of another study, Crafoord et al. used linear interpolation to fill in missing values in measurement instruments, demonstrating the effectiveness of this technique in maintaining data integrity .[15]
1. Data Normalization: Normalization is an important step in data pre-processing, especially for data used in machine learning models. The Min-Max Scaler technique, which transforms data values into a range[16] , has been shown to accelerate model convergence and improve training stability[17]–[19] . Research on data normalization shows that this step can significantly improve model performance when the data used has different scales.
2. Dataset Split: Splitting datasets into training and test data is a standard practice in machine learning, often in a proportion of 80:20. Performing a proper split ensures that the trained model can evaluate its performance on data that has never been seen before[20], [21] . Past research has shown that a proper split helps in reducing overfitting and ensuring model generalization .[14], [22]
3. Sequence Establishment: Given the nature of time series data, defining sequences with a window length of 60 days is important to provide historical context for the predictions to be made by the model. Related research shows that this data format is well suited to LSTM models, which are specifically designed to capture temporal patterns in data .[23]

2.3. LSTM Model Architecture

The LSTM model used in this study adopts a layered (*sequential*) architecture designed to capture temporal dependencies in stock price data. The model consists of two LSTM layers with a configurable number of units, each followed by a *dropout* layer for regularization to prevent *overfitting*. The first LSTM layer is configured with return_sequences=True so that its sequential output can be processed by the next LSTM layer. After the LSTM layer, a *dense* layer with 25 units serves as an intermediate feature extractor, while the final *dense* layer with one unit generates the stock price prediction. In detail, the model architecture is organized as follows:

1. First LSTM layer:
   - Unit: Configurable (optimized via *hyperparameters*)
   - Input Shape: (60, 1) (60 *time step* window with univariate input)
   - return_sequences=True (to allow *stacking* with the next LSTM layer)
2. First Dropout Layer:
   - *Dropout* Rate: Configurable (optimized via *hyperparameters*)
3. Second LSTM layer:
   - Units: Same amount as the first layer
   - return_sequences=False (because this is the last LSTM layer)
4. Second Dropout Layer:
   - *Dropout* Rate: Same as first layer
4. Intermediate Dense Layer:
   - 25 units (for additional feature extraction before final prediction)
5. Output Layer:
   - 1 unit (generates stock price predictions in univariate format)

This architecture was chosen for its ability to model both long- and short-term patterns in time series data, while maintaining model generalization through a *dropout* mechanism. *Hyperparameter* optimizations on the number of LSTM units and the *dropout* rate are performed to improve the predictive performance of the model.

2.3. LSTM Implementation Without Optimization

For the LSTM implementation without optimization, the following hyperparameters are used:

Table 1. Basic hyperparameters of LSTM Model

Parameters	Value
Units	100
Dropout_rate	0.2
Learning rate	0.001
Batch_size	32

The model was trained using Adam's optimizer and Mean Squared Error (MSE) loss function. Early stopping with patience=10 was applied to prevent overfitting and keep the best performing model on the validation data.

2.4. LSTM Implementation with Random Search Optimization

Random Search is implemented for hyperparameter optimization of the LSTM model with the following search space:

Table 2. Hyperparameter Search Space of LSTM Model Random Search Optimization

Parameters	Search Space
Units	[50, 75, 100, 125, 150]
Dropout_rate	[0.1, 0.2, 0.3, 0.4, 0.5]
Learning rate	[0.0001, 0.001, 0.01, 0.1]
Batch_size	[16, 32, 64, 128]

A total of 15 Random Search iterations were performed, where each hyperparameter combination was evaluated using cross-validation with validation data. The best hyperparameter combination found can be seen in Table 2. The LSTM model with this optimal hyperparameter is then retrained using all training data and evaluated using test data.

2.5. Model Evaluation

The performance of both models was evaluated using several metrics to provide a comprehensive perspective:

1. Root Mean Square Error (RMSE): Measures the square root of the average square of the difference between the predicted value and the actual value .[24]

2. Mean Absolute Error (MAE): Measures the average absolute value of the difference between the predicted value and the true value .[25]
3. Mean Absolute Percentage Error (MAPE): Measures the average percentage absolute error relative to the true value.
4. Coefficient of Determination (R^2): Measures the proportion of variation in the dependent variable that can be explained by the independent variable.
5. Accuracy: Calculated as 100% - MAPE, shows the level of model accuracy in percentage form.

All evaluation metrics were calculated for both models on training and test data to provide a comprehensive performance comparison.

3. RESULTS AND DISCUSSION

3.1. Hyperparameter Search Results with Random Search

The results of 15 Random Search iterations for hyperparameter optimization of the LSTM model are shown in Table 3. Each iteration tries different hyperparameter combinations and evaluates the model performance using validation loss.

Table 3. Hyperparameter Search Process using Random Search

Iterations	Units	Dropout Rate	Learning Rate	Batch Size	Validation Loss
1	150	0.2	0.1	16	0.014381
2	100	0.5	0.0001	128	0.002677
3	100	0.5	0.0001	16	0.001234
4	75	0.2	0.001	32	0.000796
5	125	0.3	0.0001	64	0.001772
6	125	0.5	0.0001	32	0.001143
7	75	0.2	0.001	16	0.000652
8	150	0.1	0.001	64	0.000616
9	125	0.4	0.1	128	0.028873
10	75	0.4	0.0001	16	0.001205
11	50	0.4	0.1	64	0.087178
12	125	0.4	0.01	128	0.000775
13	150	0.2	0.0001	16	0.001108
14	150	0.1	0.001	128	0.000875
15	50	0.3	0.01	16	0.000680

Based on the Random Search results, the best hyperparameter combination is obtained at the 8th iteration with the lowest validation loss of 0.000616. The optimal hyperparameter combination is:

Table 4. Hyperparameter Search Results with Random Search

Parameters	Value
Units	150
Dropout_rate	0.1
Learning rate	0.001
Batch_size	64

From Table 3, it can be observed that too high a value of learning rate (0.1) tends to result in greater validation loss (iterations 1, 9, and 11), indicating that the model has difficulty in convergence. Meanwhile, the combination of larger units (150) with lower dropout rate (0.1) and moderate learning rate (0.001) provides the best performance.

3.2. Model Performance Comparison

The evaluation results for the standard LSTM model and the LSTM model with Random Search optimization on training data and test data are shown in Table 5.

.

Table 5. Comparison of Evaluation Metrics for Standard LSTM Model and LSTM with Random Search Optimization

Metrics	LSTM Without Optimization		LSTM with Grid Search	
	Training	Testing	Training	Testing
RMSE	0.05	0.04	0.06	0.04
MAE	0.04	0.03	0.04	0.03
MAPE	1.28%	0.73%	1.40%	0.78%
R^2	0.9885	0.8571	0.9861	0.8302
Accuracy	98.72%	99.27%	98.60%	99.22%

From the data in Table 2, several important points can be observed:

1. Standard LSTM Model: This model performed very well with a Train RMSE of 0.05 and a Test RMSE of 0.04. The low MAPE values (1.28% on training data and 0.73% on test data) indicate minimal prediction error. The high R^2 values (0.9885 in the training data and 0.8571 in the test data) indicate the ability of the model to explain variations in stock prices. The accuracy of the model reached 98.72% in the training data and 99.27% in the test data.
2. LSTM model with Random Search Optimization: This model showed comparable performance with Train RMSE of 0.06 and Test RMSE of 0.04. The MAPE value of 1.40% on the training data and 0.78% on the test data also indicates a low error rate. The R^2 values of 0.9861 in the training data and 0.8302 in the test data indicate good predictive ability. The accuracy of this model reached 98.60% in the training data and 99.22% in the test data.
3. Comparison of the two models: Both models performed very well in predicting Spark New Zealand's share price. The standard LSTM model performed slightly better on almost all evaluation metrics, with insignificant differences. On the test data, the standard LSTM model produced a MAPE of 0.73% compared to 0.78% for the optimized model, and an R^2 of 0.8571 compared to 0.8302. The difference in accuracy between the two models on the test data was also very small (99.27% vs 99.22%).

3.3. Model Performance Analysis

Figures 2 and 3 show the visual comparison between the actual price and the predicted price for the LSTM model without optimization and LSTM with Grid Search optimization on the test data.

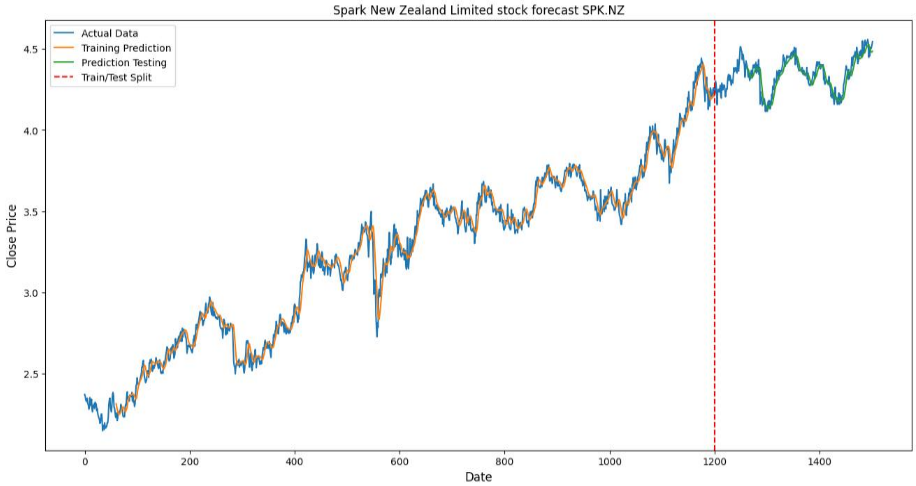

Figure 2. Comparison graph of train and test data of LSTM model with Random Search optimization.

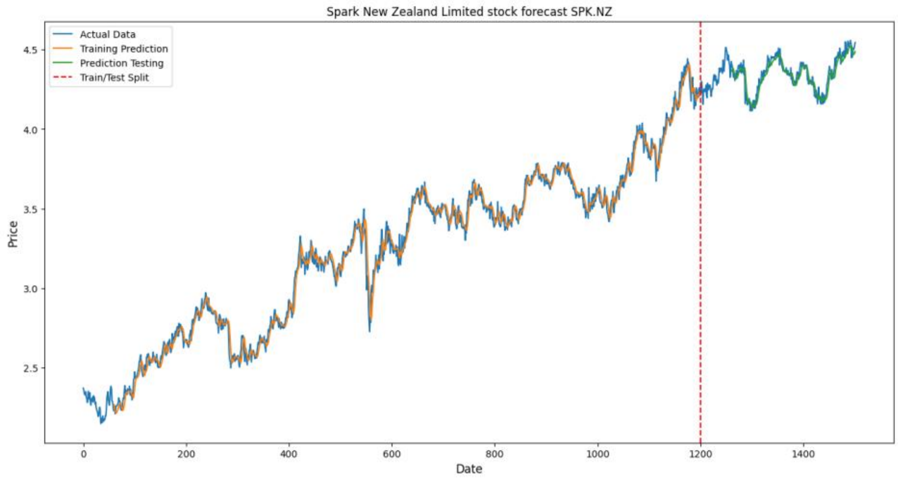

Figure 3. Comparison graph of train and test data of LSTM model without optimization

Although hyperparameter optimization using Random Search successfully found the optimal hyperparameter combination based on validation loss, the standard LSTM model still performed slightly better on the test data. This can be explained by several factors:

1. Overfitting vs Generalization: The LSTM model with Random Search optimization has a larger number of units (150 compared to 100 in the standard model) and a lower dropout rate (0.1 compared to 0.2). This combination may cause the model to be more complex and potentially overfitting the training data. This is reflected in the larger difference between the performance on the training and test data of the optimized model.
2. Prediction Stability: The standard LSTM model shows a smaller performance degradation when applied to the test data compared to the optimized model. This indicates that the standard model has better prediction stability and stronger generalization ability.
3. Complexity and Performance Trade-off: While optimized models have a more complex architecture, the increased complexity does not always result in significant performance improvements, especially if the data being analyzed has relatively simple or consistent patterns.
4. Effectiveness of Random Search: Random Search may not find a truly optimal combination of hyperparameters due to the limited number of iterations (15 iterations). Increasing the number of iterations or using more sophisticated hyperparameter optimization methods such as Bayesian Optimization may result in better performing models.

3.4. Implications for Investment and Trading

Both models showed very high accuracy (>99% on the test data), indicating significant potential for application in investment decision-making on Spark New Zealand shares. However, some practical implications need to be considered:

1. Short-term vs Long-term Prediction: The LSTM model developed in this study shows good performance for short-term prediction (based on the previous 60-day window). However, the performance for long-term prediction needs to be further tested.
2. Integration with External Factors: The model only uses historical stock price data without considering external factors such as news, company financial reports, or overall market trends. Integration of such factors can improve prediction accuracy especially during periods of high market volatility.
3. Application in Trading Strategies: High prediction accuracy can be integrated into algorithmic trading strategies, but it is necessary to consider factors such as transaction costs, bid-ask spreads, and stock liquidity that may affect the overall profitability of the strategy.

Based on research by Sanhatham (2024)[5] , deep learning models such as LSTM have been proven effective in stock price prediction in various markets and sectors. However, they also emphasized the importance of thorough model evaluation and consideration of practical factors in model implementation for real investment decision making.

4. CONCLUSION

This study compares the performance of the standard LSTM model with the LSTM model optimized using Random Search to predict the stock price of Spark New Zealand Limited based on Yahoo Finance historical data for the period 2018-2023. The results show that both models are able to provide highly accurate predictions, with accuracy rates exceeding 99% on the test data, confirming the effectiveness of the LSTM architecture in financial time series modeling. In particular, the standard LSTM model with manually set hyperparameters (units=100, dropout_rate=0.2, learning_rate=0.001, batch_size=32) showed slightly superior performance compared to the optimized model, with a Test RMSE value of 0.04, Test MAPE 0.73%, and Test R^2 0.8571. Meanwhile, Random Search successfully found the optimal hyperparameter combination (units=150, dropout_rate=0.1, learning_rate=0.001, batch_size=64), but the increased model complexity did not provide a significant improvement in accuracy, with a Test RMSE of 0.04, Test MAPE of 0.78%, and Test R^2 of 0.8302. This finding indicates that, in the context of Spark New Zealand stock price prediction, LSTM models with simple architectures can perform equally or even better than more complex and optimized models. In addition, both models show good generalization ability, with performance on test data remaining high despite a slight decrease compared to training data, demonstrating the reliability of the models in predicting new data. These results provide practical implications for researchers and practitioners in the field of computational finance, that a simple approach with appropriate hyperparameters can be an efficient solution for stock price prediction, without the need for complex optimization. Future research could explore alternative optimization methods or expand the dataset to validate these findings on other financial instruments.

REFERENCES

[1] K. Zhang, G. Zhong, J. Dong, S. Wang, and Y. Wang, "Stock market prediction based on generative adversarial network," *Procedia Comput. Sci.*, vol. 147, pp. 400-406, 2019.
[2] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, V. K. Menon, and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model," in *2017 international conference on advances in computing, communications and informatics (icacci)*, 2017, pp. 1643-1647.
[3] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735-

1780, 1997.

[4] R. Kepo and D. Okokpujie, "Performance Comparison of GRU and LSTM Methods for Predicting Bitcoin Exchange Rate against US Dollar," *Int. J. Artif. Intell. Informatics*, vol. 2, no. 1, pp. 17-24, 2024, doi: 10.33292/ijarlit.v2i1.32.

[5] S. Sanhatham, "Stock Price Prediction of Thai Oil Public Company Limited (TOP.BK) Using LSTM Model with Grid Search Hyperparameter optimization," *Int. J. Artif. Intell. Informatics*, vol. 2, no. 1, pp. 33-40, 2024, doi: 10.33292/ijarlit.v2i1.34.

[6] N. Reimers and I. Gurevych, "Optimal hyperparameters for deep lstm-networks for sequence labeling tasks," *arXiv Prepr. arXiv1707.06799*, 2017.

[7] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 281-305, 2012.

[8] E. Bochinski, T. Senst, and T. Sikora, "Hyper-parameter optimization for convolutional neural network committees based on evolutionary algorithms," in *2017 IEEE international conference on image processing (ICIP)*, 2017, pp. 3924-3928.

[9] T. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions," *Eur. J. Oper. Res.*, vol. 270, no. 2, pp. 654-669, 2018.

[10] Q. Zhang, C. Qin, Y. Zhang, F. Bao, C. Zhang, and P. Liu, "Transformer-based attention network for stock movement prediction," *Expert Syst. Appl.*, vol. 202, p. 117239, 2022.

[11] J. Xia *et al.*, "A Long Short-Term Memory Ensemble Approach for Improving the Outcome Prediction in Intensive Care Units," *Comput. Math. Methods Med.*, 2019, doi: 10.1155/2019/8152713.

[12] Y. Liu, W. Kimita, X. Shamaitijiang, L. Skudder-Hill, I. R. Sequeira, and M. S. Petrov, "Intra-pancreatic Fat Is Associated With Continuous Glucose Monitoring Metrics," *Diabetes Obes. Metab.*, 2024, doi: 10.1111/dom.15550.

[13] O. Noy *et al.*, "A Machine Learning Model for Predicting Deterioration of COVID-19 Inpatients," *Sci. Rep.*, 2022, doi: 10.1038/s41598-022-05822-7.

[14] S. A. Nagro, A. R. Raines, S. D. Hooks, D. W. Fraser, and S. Nagy, "The Connection Between Teacher Candidate Attitude and Video-Based Self-Evaluation Accuracy," *J. Spec. Educ. Technol.*, 2020, doi: 10.1177/0162643420947827.

[15] M.-T. Crafoord, K. Sundberg, M. Nilsson, M. Fjell, and A. Langius-Eklöf, "Patients' Individualized Care Perceptions and Health Literacy Using an Interactive App During Breast and Prostate Cancer Treatment," *Cin Comput. Informatics Nurs.*, 2023, doi: 10.1097/cin.0000000000001007.

[16] G. Güder *et al.*, "Establishing a Cardiac Training Group for Patients With Heart Failure: The 'HIP-in-Würzburg' Study," *Clin. Res. Cardiol.*, 2021, doi: 10.1007/s00392-021-01892-1.

[17] Y. S. Kwon and M. S. Baek, "Development and Validation of a Quick Sepsis-Related Organ Failure Assessment-Based Machine-Learning Model for Mortality Prediction in Patients With Suspected Infection in the Emergency Department," *J. Clin. Med.*, 2020, doi: 10.3390/jcm9030875.

[18] N. J. Tustison *et al.*, "The ANTsX Ecosystem for Quantitative Biological and Medical Imaging," *Sci. Rep.*, 2021, doi: 10.1038/s41598-021-87564-6.

[19] J. P. Czakert, D. Leiva, and R. Berger, "How Transformational Leadership Affects the Off-Work Recovery of Daily Personal Energy Resources via Work Engagement: Resource and Demand-Based Pathways," *Span. J. Psychol.*, 2024, doi: 10.1017/sjp.2024.12.

[20] A. R. Mohamed, P. S. Sterkenburg, J. G. Yeatman, E. van Rensburg, and C. Schuengel, "Using Attachment and Biobehavioral Catch-Up With Young Children With Developmental Delays: A Multiple-Baseline Trial of Attachment, Sensitivity, and Cortisol," *J. Intellect. Disabil.*, 2022, doi: 10.1177/17446295221104614.

[21] M. Young, G. MacGregor, and G. A. Tanguay, "Revisiting COVID-19's Impact on Cycling: An Examination of Bicycle Count Data in Montréal," *Findings*, 2024, doi: 10.32866/001c.118813.

[22] E. Passmore *et al.*, "Automated Identification of Abnormal Infant Movements From Smart Phone Videos," *Plos Digit. Heal*, 2024, doi: 10.1371/journal.pdig.0000432.

[23] M. P. Harris and J. R. Mahalik, "Dominant Masculinity and Marginalized Sexuality: Extending Minority Stress Theory for Gay and Bisexual+ Men," *Psychol. Sex. Orientat. Gend. Divers.*, 2023, doi: 10.1037/sgd0000656.

[24] L. Tian, "Forecast and Analysis for Stock Markets of the U.S., Canada, and Mexico Based on Time Series Forecasting Models," *Adv. Econ. Manag. Polit. Sci.*, vol. 13, no. 1, pp. 389-401, 2023, doi: 10.54254/2754-1169/13/20230759.

[25] G. Bekdaş, Y. Aydın, Ü. Işıkdağ, A. N. Sadeghifam, S. Kim, and Z. W. Geem, "Prediction of Cooling Load of Tropical Buildings With Machine Learning," *Sustainability*, vol. 15, no. 11, p. 9061, 2023, doi: 10.3390/su15119061.