JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 9, No. 1, JANUARI, 2024, Pp. 10 – 26
ISSN: 2527 – 5836 (print) | 2528 – 0074 (online)

Improving Stock Price Prediction Accuracy with StacBi LSTM

Mohammad Diqi [(1)*], Hamzah [(2)]
Informatika, Fakultas Sains dan Teknologi, Universitas Respati Yogyakarta, Yogyakarta
e-mail : {diqi,hamzah}@respati.ac.id.
* Penulis korespondensi.
Artikel ini diajukan 10 Agustus 2023, direvisi 31 Oktober 2023, diterima 1 November 2023, dan dipublikasikan 25 Januari 2024.

Abstract

This research aimed to enhance stock price prediction accuracy using the Stacked Bidirectional Long Short-Term Memory (StacBi LSTM) model. The study addressed the challenge of capturing long-term dependencies and temporal patterns inherent in stock price data. The research objectives were to evaluate the model's performance across different input sequence lengths and identify the optimal length for prediction. Leveraging a dataset from the Indonesian Stock Exchange, the model's predictions were evaluated using key metrics such as RMSE, MAE, MAPE, and R2. Results indicated that the StacBi LSTM model excelled in capturing stock price trends and demonstrated strengths over traditional methods. The optimal input sequence length was identified, balancing computational efficiency and prediction accuracy. This research contributes valuable insights into improving stock price prediction techniques and offers practical implications for traders and investors. Future research directions encompass hybrid models and integrating external factors to enhance predictive capabilities further.

Keywords: Stock Price Prediction, Stacked Bidirectional LSTM, Time Series Analysis, Indonesian Stock Exchange, Input Sequence Length

Abstrak

Penelitian ini bertujuan untuk meningkatkan akurasi prediksi harga saham menggunakan model *Stacked Bidirectional Long Short-Term Memory* (StacBi LSTM). Penelitian ini mengatasi tantangan dalam menangkap ketergantungan jangka panjang dan pola temporal yang inheren dalam data harga saham. Tujuan penelitian adalah mengevaluasi kinerja model dalam berbagai panjang urutan masukan dan mengidentifikasi panjang masukan yang optimal untuk prediksi. Dengan menggunakan *dataset* dari Bursa Efek Indonesia, prediksi model dievaluasi menggunakan metrik kunci seperti RMSE, MAE, MAPE, dan R2. Hasil penelitian menunjukkan bahwa model StacBi LSTM mampu dengan baik dalam menangkap tren harga saham dan memiliki keunggulan dibandingkan metode tradisional. Panjang masukan optimal diidentifikasi, menciptakan keseimbangan antara efisiensi komputasi dan akurasi prediksi. Penelitian ini memberikan wawasan berharga dalam meningkatkan teknik prediksi harga saham dan memberikan implikasi praktis bagi para trader dan investor. Arah penelitian di masa depan meliputi model hibrida dan integrasi faktor eksternal untuk meningkatkan kemampuan prediksi lebih lanjut.

Kata Kunci: Prediksi harga saham, Stacked Bidirectional LSTM, Analisis runtun waktu, Bursa Efek Indonesia, Panjang urutan masukan

1. INTRODUCTION

Stock price prediction is a crucial domain within financial research, holding substantial implications for global financial markets in an increasingly interconnected and dynamic economy (Miftahurrohmah et al., 2021). Accurate predictions of stock prices are imperative for enabling informed decision-making among investors, traders, and financial institutions, leading to enhanced portfolio management, effective risk mitigation, and optimal resource allocation (Aydin et al., 2022). Furthermore, such predictions are instrumental in uncovering potential profit avenues and formulating efficacious trading strategies. In this regard, advanced machine learning and deep learning techniques have been transformative, providing novel means to enhance

prediction precision and unravel complex patterns in historical stock data, thereby empowering market participants to make more lucrative investment decisions (Rajamoorthy et al., 2022).

In evaluating predictive methodologies, fundamental analysis emerges as a prominent approach, anchored in examining a company's financial and economic data to gauge its intrinsic value and determine the stock's market standing (Williams et al., 2020). Despite its merits, this method is not without its challenges; it is inherently time-intensive, subjective in nature, and often neglects short-term market sentiments, rendering it less effective for tracking swift market dynamics (Naumoski et al., 2022; Wang et al., 2021). Conversely, technical indicators offer real-time analyses rooted in historical price and volume data, shedding light on market trends and potential trading positions (Jamous et al., 2021). However, they have limitations, including a tendency to rely heavily on past data, produce delayed signals, and generate false signals in volatile markets. These challenges necessitate expertise in selecting and adapting indicators and parameters to individual stocks and resolving discrepancies when multiple indicators are employed simultaneously (Htun et al., 2023).

From a modeling perspective, AutoRegressive Integrated Moving Average (ARIMA) models and their seasonal variant, SARIMA, are widely recognized for their efficacy in time series forecasting, especially for data with linear dependencies and periodic fluctuations respectively (Brahma & Wadhvani, 2020; Jiang et al., 2019). Nevertheless, they exhibit limitations in addressing non-stationary data and capturing non-linear patterns, with performance challenges arising when dealing with long-term and noisy data (Musarat et al., 2021; Shuai et al., 2021). Support Vector Machine (SVM) and Decision Tree algorithms. However, capturing complex relationships and providing intuitive insights also present challenges regarding dataset balance, feature scaling, and model stability (Jamous et al., 2021; Sekiguchi et al., 2019).

Ensemble methods like Random Forest and XGBoost attempt to mitigate these issues by aggregating multiple models, enhancing predictive performance and robustness to overfitting (Campbell et al., 2020; Pamir et al., 2022). Nevertheless, they still grapple with challenges related to computational efficiency, hyperparameter tuning, and interpretability (Kim et al., 2021; Lind & Anderson, 2019). On the other hand, K-Nearest Neighbors (KNN) stands out for its simplicity and robustness, although it requires careful parameter selection and is computationally demanding for large datasets (Lokanan, 2022; Ma et al., 2020).

Delving into deep learning, Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) represent robust architectures for handling sequential data, each with its own set of challenges and areas of applicability (Gao et al., 2018; Succetti et al., 2022; Tang & Mahmoud, 2022). LSTM and GRU, as variants of RNNs, offer solutions to the vanishing gradient problem, facilitating the capture of long-term dependencies, albeit with considerations for computational cost and overfitting (Ahmad et al., 2022; Choi & Shin, 2019; Suleman & Shridevi, 2022). Autoencoders and Generative Adversarial Networks (GANs) further extend the deep learning repertoire, providing capabilities in feature extraction, dimensionality reduction, and data augmentation, but necessitate careful model design and training (Erizal & Diqi, 2023; Fathy et al., 2021; Mo et al., 2022; Pan et al., 2020; Zhang et al., 2022).

The research presented here revolves around the StacBi LSTM model, chosen for its proficiency in capturing long-term dependencies and temporal patterns in stock prices, integrating bidirectional processing and multiple LSTM layers to discern complex relationships in time series data (Suleman & Shridevi, 2022). The investigation is structured to predict stock prices using this model with varied input sequences, aiming to discern the impact of sequence length on predictive performance and employing a comprehensive suite of evaluation metrics to benchmark against conventional forecasting methods. This study contributes to the domain of stock price prediction by elucidating the potentials of the StacBi LSTM model, exploring the influence of input sequence variations, and advancing the evaluation methodology within this context.

2. METHODS

2.1 Long Short-Term Memory (LSTM)

LSTM is an RNN that addresses the vanishing gradient problem, allowing it to capture long-term dependencies in sequential data (Qaddoura et al., 2021). It achieves this by introducing a memory cell and three gating mechanisms: the input gate, forget gate, and output gate. Table 1 summarizes the mathematical notation used in the context of LSTM.

Table 1 Mathematical Notation for LSTM

Symbol	Description
x_t	The input at time step t.
h_t	The hidden state at time step t.
c_t	The cell state (memory) at time step t.
i_t	The input gate at time step t.
f_t	The forget gate at time step t.
o_t	The output gate at time step t.
σ	The sigmoid activation functions.
$tanh$	The hyperbolic tangent activation function.

The LSTM computation consists of four main steps for each time step t:

Input Gate. The input gate determines how much of the new input information is stored in the cell state, as calculated in Equation 1.

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \quad (1)$$

Forget Gate. The forget gate determines how much of the previous cell state to retain for the current time step, as calculated in Equation 2.

$$f_t = \sigma\left(W_{fx}x_t + W_{fh}h_{t-1} + b_f\right) \quad (2)$$

Cell State. The cell state is updated by combining the previous cell state with the new input information using the input and forget gates, as calculated in Equation 3.

$$\begin{gathered}\tilde{c}_t = \tanh(W_{cx}x_t + W_{ch}h_{t-1} + b_c) \\ c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t\end{gathered} \quad (3)$$

Output Gate. The output gate determines how much of the updated cell state to output as the hidden state for the current time step, as calculated in Equation 4.

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \quad (4)$$

Hidden State. The hidden state is obtained by applying the output gate to the updated cell state, as calculated in Equation 5.

$$h_t = o_t \cdot \tanh(c_t) \quad (5)$$

Here, W_{ix}, W_{ih}, W_{fx}, W_{fh}, W_{cx}, W_{ch}, W_{ox}, W_{oh} are weight matrices and b_i, b_f, b_c, b_o are bias vectors.

The LSTM memory cell's ability to control the flow of information through the input, forget, and output gates enables it to retain essential information over long sequences, making it effective in capturing long-term dependencies in time series data. The vanishing gradient problem is

mitigated by the constant error flow through the cell state, allowing for more stable and efficient training.

2.2 Stacked Bidirectional (StacBi) LSTM

The StacBi LSTM consists of N LSTM layers, $N/2$ in the forward direction and $N/2$ in the backward direction. Table 2 summarizes the mathematical notation used in the StacBi LSTM model, distinguishing between the forward and backward LSTM layers and their respective hidden states, cell states, and gates.

Table 2 Mathematical Notation for StacBi LSTM

Symbol	Description
x_t	The input at time step t.
$h_t^{(n,f)}$	The hidden state of the n-th forward LSTM layer at time step t.
$c_t^{(n,f)}$	The cell state (memory) of the n -th forward LSTM layer at time step t.
$h_t^{(n,b)}$	The hidden state of the n-th backward LSTM layer at time step t.
$c_t^{(n,b)}$	The cell state (memory) of the n-th backward LSTM layer at time step t.
$i_t^{(n,f)}$	The input gate of the n-th forward LSTM layer at time step t.
$f_t^{(n,f)}$	The forget gate of the n-th forward LSTM layer at time step t.
$o_t^{(n,f)}$	The output gate of the n-th forward LSTM layer at time step t.
$i_t^{(n,b)}$	The input gate of the n-th backward LSTM layer at time step t.
$f_t^{(n,b)}$	The forget gate of the n-th backward LSTM layer at time step t.
$o_t^{(n,b)}$	The output gate of the n-th backward LSTM layer at time step t.
$\tilde{c}_t^{(n,f)}$	The candidate cell state of the n-th forward LSTM layer at time step t.
$\tilde{c}_t^{(n,b)}$	The candidate cell state of the n-th backward LSTM layer at time step t.

Forward LSTM. The forward LSTM computes the hidden state $h_t^{(n,f)}$ and the cell state $c_t^{(n,f)}$ for the n-th forward layer at time step t, as shown in Equation 6.

$$\begin{aligned} i_t^{(n,f)} &= \sigma\left(W_{ix}^{(n,f)} x_t + W_{ih}^{(n,f)} h_{t-1}^{(n,f)} + b_i^{(n,f)}\right) \\ f_t^{(n,f)} &= \sigma\left(W_{fx}^{(n,f)} x_t + W_{fh}^{(n,f)} h_{t-1}^{(n,f)} + b_f^{(n,f)}\right) \\ \tilde{c}_t^{(n,f)} &= \tanh\left(W_{cx}^{(n,f)} x_t + W_{ch}^{(n,f)} h_{t-1}^{(n,f)} + b_c^{(n,f)}\right) \\ c_t^{(n,f)} &= f_t^{(n,f)} \cdot c_{t-1}^{(n,f)} + i_t^{(n,f)} \cdot \tilde{c}_t^{(n,f)} \\ o_t^{(n,f)} &= \sigma\left(W_{ox}^{(n,f)} x_t + W_{oh}^{(n,f)} h_{t-1}^{(n,f)} + b_o^{(n,f)}\right) \\ h_t^{(n,f)} &= o_t^{(n,f)} \cdot \tanh\left(c_t^{(n,f)}\right) \end{aligned} \tag{6}$$

Backward LSTM. The backward LSTM computes the hidden state $h_t^{(n,b)}$ and the cell state $c_t^{(n,b)}$ for the n-th backward layer at time step t, as shown in Equation 7.

$$\begin{aligned} i_t^{(n,b)} &= \sigma\left(W_{ix}^{(n,b)} x_t + W_{ih}^{(n,b)} h_{t+1}^{(n,b)} + b_i^{(n,b)}\right) \\ f_t^{(n,b)} &= \sigma\left(W_{fx}^{(n,b)} x_t + W_{fh}^{(n,b)} h_{t+1}^{(n,b)} + b_f^{(n,b)}\right) \\ \tilde{c}_t^{(n,b)} &= \tanh\left(W_{cx}^{(n,b)} x_t + W_{ch}^{(n,b)} h_{t+1}^{(n,b)} + b_c^{(n,b)}\right) \\ c_t^{(n,b)} &= f_t^{(n,b)} \cdot c_{t+1}^{(n,b)} + i_t^{(n,b)} \cdot \tilde{c}_t^{(n,b)} \\ o_t^{(n,b)} &= \sigma\left(W_{ox}^{(n,b)} x_t + W_{oh}^{(n,b)} h_{t+1}^{(n,b)} + b_o^{(n,b)}\right) \\ h_t^{(n,b)} &= o_t^{(n,b)} \cdot \tanh\left(c_t^{(n,b)}\right) \end{aligned} \tag{7}$$

StacBi LSTM. The StacBi LSTM is formed by stacking N LSTM layers, where each forward layer's output serves as the input to the next forward layer, and each backward layer's output is the input to the next backward layer, as shown in Equations 8-9.

Forward Pass. For the n-th forward layer:

$$x_t^{(n,f)} = h_t^{(n-1,f)}$$
$$h_t^{(n,f)} = \text{Forward LSTM computation using } x_t^{(n,f)} \text{ as input} \quad (8)$$

Backward Pass. For the n-th backward layer:

$$x_t^{(n,b)} = h_t^{(n+1,b)}$$
$$h_t^{(n,b)} = \text{Backward LSTM computation using } x_t^{(n,b)} \text{ as input} \quad (9)$$

The final hidden state h_t of the StacBi LSTM is obtained by concatenating the outputs from the last forward layer $h_t^{(N/2,f)}$ and the first backward layer $h_t^{(1,b)}$, as shown in Equation 10.

$$h_t = \left[h_t^{(N/2,f)}; h_t^{(1,b)}\right] \quad (10)$$

In this way, the StacBi LSTM captures past and future context, enabling it to model long-term dependencies and complex temporal patterns more effectively than a single LSTM layer.

2.3 Dataset

The dataset utilized in this research is sourced from Yahoo Finance (Erizal & Diqi, 2023). It encompasses the top 10 stocks listed in the Indonesia Stock Exchange within the period spanning from July 6, 2015, to October 14, 2021. The stocks and corresponding symbols and sectors are presented in Table 3.

Table 3 List of the Observed Stocks

Symbol	Company	Sector
ACES.JK	Ace Hardware Indonesia Tbk.	Consumer Non-Cyclicals
ADRO.JK	Adaro Energy Tbk.	Energy
EXCL.JK	XL Axiata Tbk.	Infrastructures
KLBF.JK	Kalbe Farma Tbk.	Healthcare
PGAS.JK	Perusahaan Gas Negara (Persero) Tbk.	Energy
PTBA.JK	Tambang Batubara Bukit Asam (Persero) Tbk.	Energy
PTPP.JK	PP (Persero) Tbk.	Infrastructures
PWON.JK	Pakuwon Jati Tbk.	Properties & Real Estate
SMRA.JK	Summarecon Agung Tbk.	Properties & Real Estate
TPIA.JK	Chandra Asri Petrochemical Tbk.	Basic Materials

The dataset includes essential features such as Date, Open, High, Low, Close, and Volume, with records collected daily. Notably, this study exclusively focuses on the Close Price variable, as depicted in Figures 1 to 10.

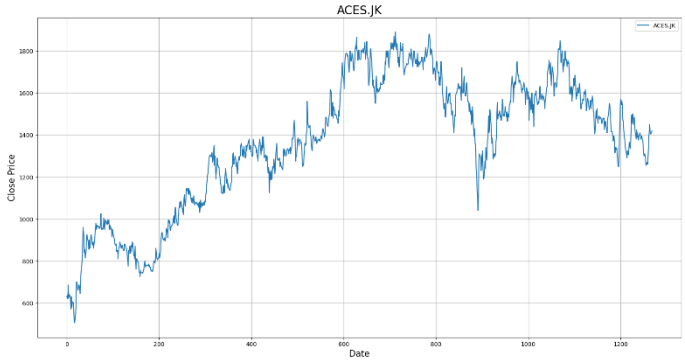

Figure 1 ACES.JK

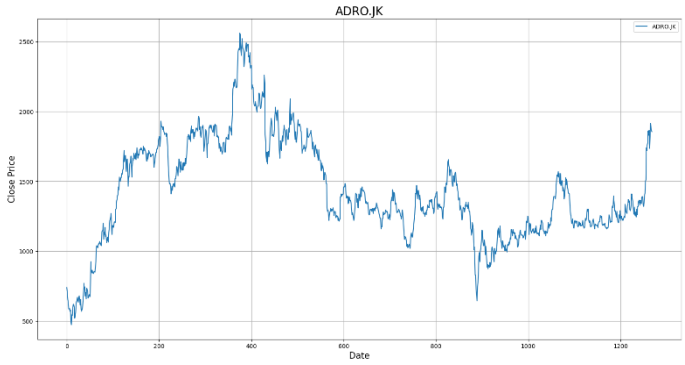

Figure 2 ADRO.JK

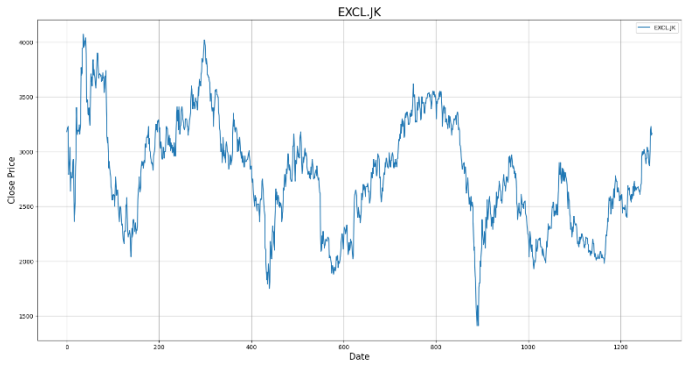

Figure 3 EXCL.JK

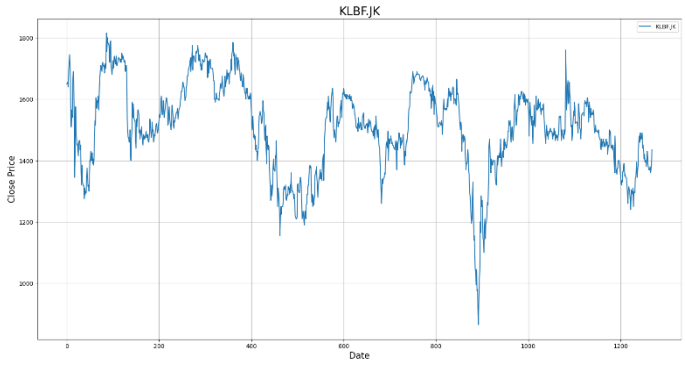

Figure 4 KLBF.JK

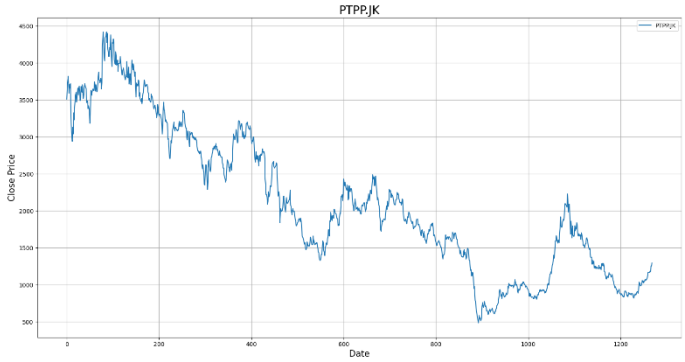

Figure 5 PGAS.JK

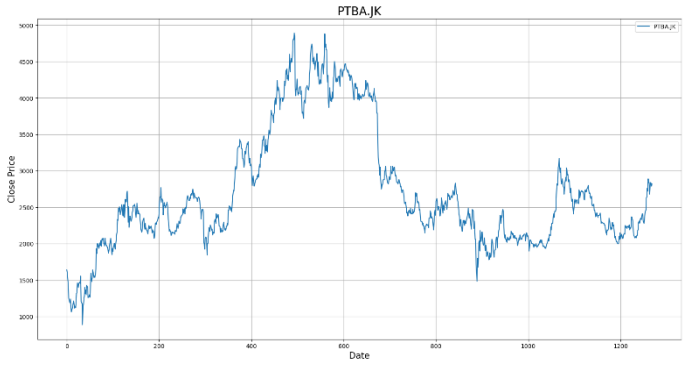

Figure 6 PTBA.JK

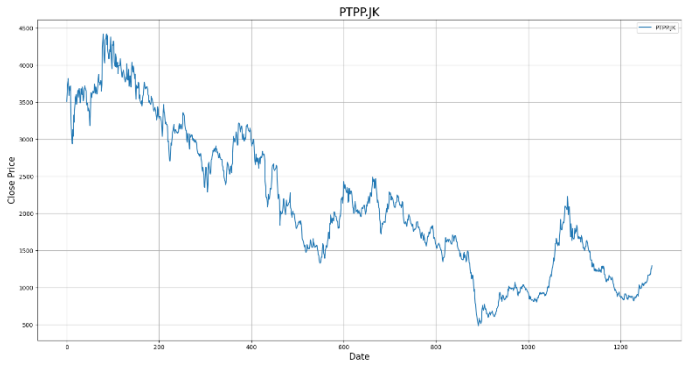

Figure 7 PTPP.JK

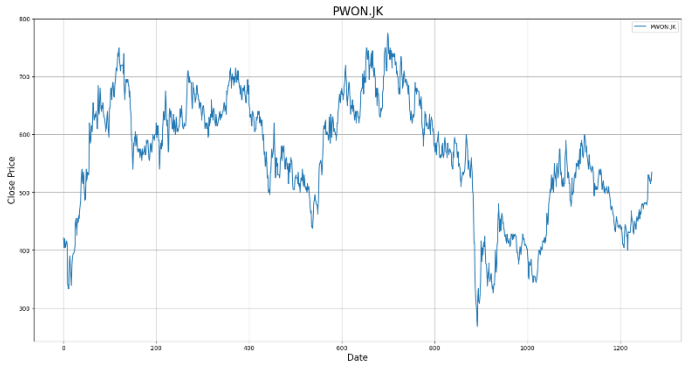

Figure 8 PWON.JK

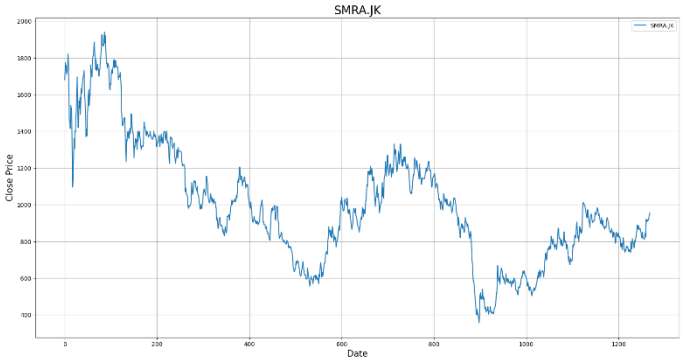

Figure 9 SMRA.JK

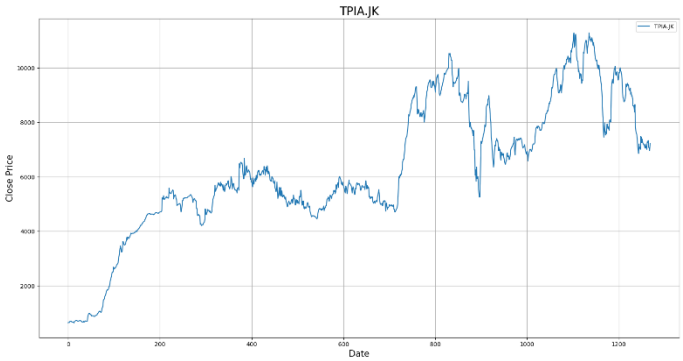

Figure 10 TPIA.JK

2.4 Data Processing

In the preparatory phase of data preprocessing, meticulous measures were implemented to refine the dataset for predicting stock prices, ensuring its quality and relevance. Instances with a trading volume of zero were deliberately removed to prevent potential distortions in the model's learning trajectory. The min-max scaler technique was applied in a normalization procedure to foster model convergence and maintain numerical stability. This method recalibrates the range of feature values to a standardized scale between 0 and 1, accommodating various data magnitudes and contributing to a more uniform and manageable dataset for the model. The mathematical formulation of the min-max scaler is provided in Equation 11, detailing the procedure of this normalization process.

$$X_{\text{normalized}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \tag{11}$$

Where X is the original feature value, X_{min} and X_{max} are the minimum and maximum values of the feature, respectively. This preprocessing strategy ensures that the dataset lacks missing values and zero volume instances and is normalized for effective utilization in the subsequent analysis.

2.5 Data Splitting

The dataset, consisting of 1269 data points, was split into distinct subsets to facilitate practical model training, validation, and testing. Specifically, 40 data points were reserved for testing the model's generalization performance on unseen data. Most of the data, totaling 983 instances, was allocated for training the model to learn patterns and relationships within the dataset. An additional subset comprising 246 data points was designated for validation, enabling the fine-tuning of model hyperparameters and preventing overfitting. This data-splitting strategy ensured

that the model's performance was rigorously assessed across different training and testing phases, enhancing its predictive capabilities for stock price forecasting.

2.6 Model Training Process

The model training process involves a StacBi LSTM architecture for stock price prediction. The model is structured as follows:

1) **Input Configuration**: The input features are defined by $n\_features = 1$, indicating that the model utilizes one feature (Close Price) for prediction. Three different input sequence lengths, $n\_steps = [30,50,70]$, are considered to capture varying historical information.
2) **Model Construction**: A Sequential model is established. The first layer is a Bidirectional LSTM with 256 units, employing the 'relu' activation function and 'return_sequences=True' to pass sequences to the subsequent layer. A dropout layer (dropout rate = 0.2) follows, aiding in regularization.
3) **Second Bidirectional LSTM**: Another Bidirectional LSTM with 128 units and "relu' activation is added, capturing complex temporal patterns. Another dropout layer (dropout rate = 0.2) helps prevent overfitting.
4) **Output Layer**: A Dense layer with 1 unit is employed for the final prediction.
5) **Compilation**: The model is compiled with the Adam optimizer (learning rate = 0.001) and mean squared error (MSE) loss function.
6) **Training**: The model is trained using the provided training data (X and y) for 100 epochs, with a batch size 32. Training progress is run in a quiet mode (verbose=0).

This training process enables the model to learn and capture intricate patterns in the input data, ultimately enhancing its ability to forecast stock prices accurately.

2.7 Evaluation Metrics

In this research, several evaluation metrics were employed to assess the performance of the StacBi LSTM model in predicting stock prices. The following metrics were utilized and formulated in Equations 12-15.

1) Root Mean Squared Error (RMSE) (Gutmann et al., 2021):

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{12}$$

2) Mean Absolute Error (MAE) (Gutmann et al., 2021):

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{13}$$

3) Mean Absolute Percentage Error (MAPE) (Patel et al., 2022):

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100 \tag{14}$$

4) Coefficient of Determination (R2) (Baek & Chung, 2023):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{15}$$

Here, y_i represents the actual stock price, $\hat{y}_i$ represents the predicted stock price, $\bar{y}$ is the mean of the actual stock prices, and n is the number of data points. These evaluation metrics collectively quantify the model's accuracy, precision, and goodness of fit in predicting stock prices.

3. RESULTS AND DISCUSSION

The StacBi LSTM model was employed to forecast stock prices for 40 days. The model's predictive accuracy was assessed using fundamental metrics: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Coefficient of Determination (R2), as illustrated in Table 4.

Table 4 Performance of the StacBi LSTM Model

Symbol	n_steps	RMSE	MAE	MAPE	R2
ACES.JK	30	0.01049	0.00769	0.01221	0.94227
	50	0.01014	0.00820	0.01319	0.94608
	70	0.01473	0.01131	0.01775	0.88621
ADRO.JK	30	0.01047	0.00758	0.01451	0.99064
	50	0.01123	0.00829	0.01594	0.98923
	70	0.00980	0.00763	0.01596	0.99180
EXCL.JK	30	0.01254	0.01131	0.02076	0.96542
	50	0.00901	0.00708	0.01328	0.98214
	70	0.01489	0.01397	0.02653	0.95126
KLBF.JK	30	0.01656	0.01449	0.02599	0.92209
	50	0.01141	0.00946	0.01763	0.96302
	70	0.01573	0.01366	0.02475	0.92972
PGAS.JK	30	0.01003	0.00641	0.03496	0.95224
	50	0.00888	0.00797	0.05998	0.96260
	70	0.00915	0.00858	0.06007	0.96025
PTBA.JK	30	0.01072	0.00897	0.02264	0.97277
	50	0.00690	0.00534	0.01334	0.98873
	70	0.01755	0.01644	0.04525	0.92700
PTPP.JK	30	0.00632	0.00513	0.03823	0.95554
	50	0.00507	0.00423	0.03271	0.97131
	70	0.00645	0.00557	0.04492	0.95361
PWON.JK	30	0.01153	0.01005	0.02550	0.95741
	50	0.01699	0.01434	0.03395	0.90751
	70	0.00710	0.00556	0.01407	0.98383
SMRA.JK	30	0.00631	0.00522	0.01633	0.95927
	50	0.00949	0.00911	0.02990	0.90789
	70	0.00397	0.00317	0.01030	0.98392
TPIA.JK	30	0.00875	0.00644	0.00982	0.97799
	50	0.01171	0.01045	0.01630	0.96063
	70	0.01474	0.01304	0.02052	0.93765

To visually depict the model's performance, Figures 11 to 20 illustrate the actual stock prices over the next 40 days (indicated by the red line) alongside the predicted prices for the same period using input sequence lengths of $n\_steps = 30$ (represented by the blue line), $n\_steps = 50$ (depicted by the yellow line), and $n\_steps = 70$ (illustrated by the green line). These figures provide a comprehensive insight into the model's predictive capabilities under varying input conditions, allowing for a comprehensive assessment of its effectiveness in capturing short-term stock price trends.

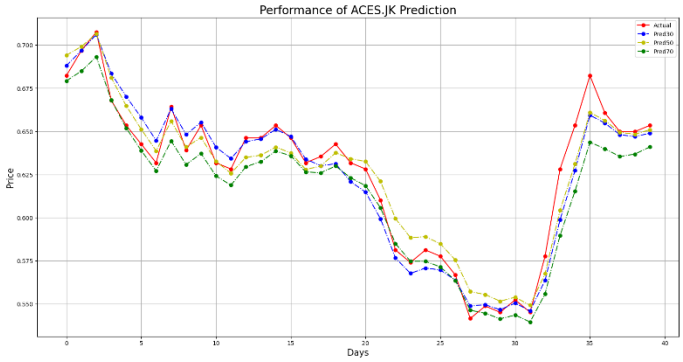

Figure 11 Performance of ACES.JK

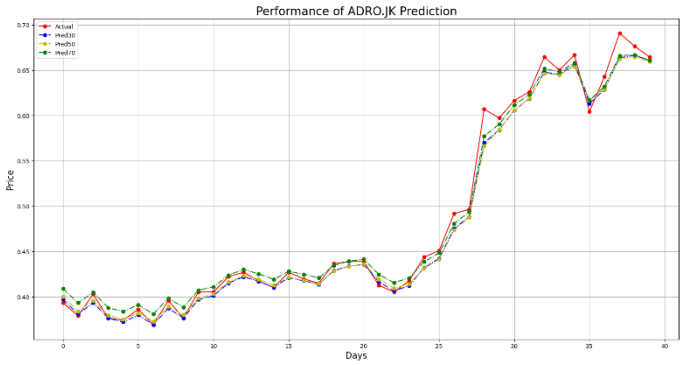

Figure 12 Performance of ADRO.JK

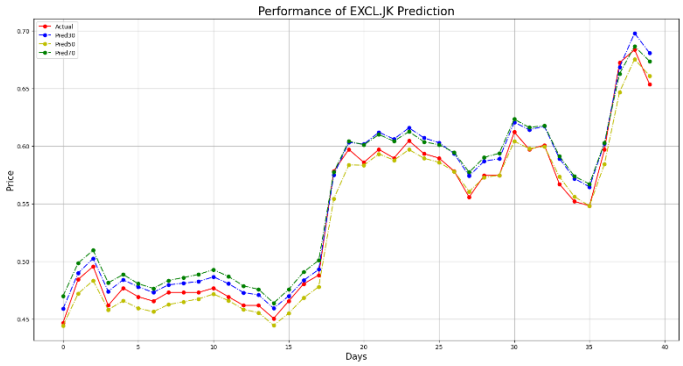

Figure 13 Performance of EXCL.JK

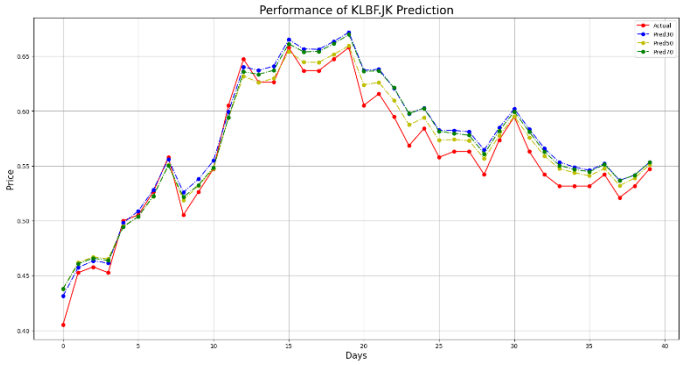

Figure 14 Performance of KLBF.JK

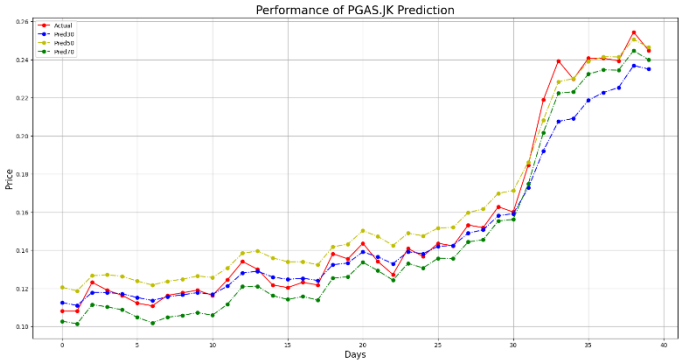

Figure 15 Performance of PGAS.JK

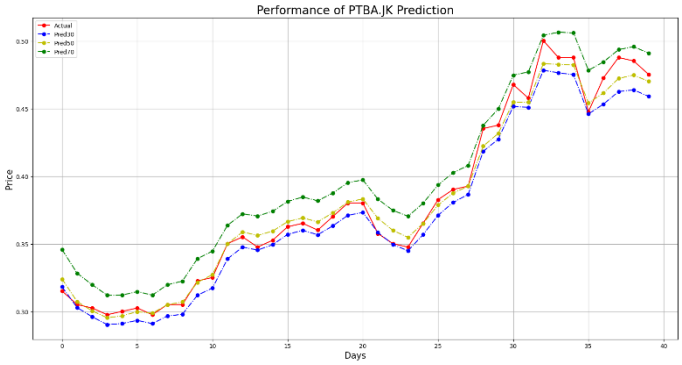

Figure 16 Performance of PTBA.JK

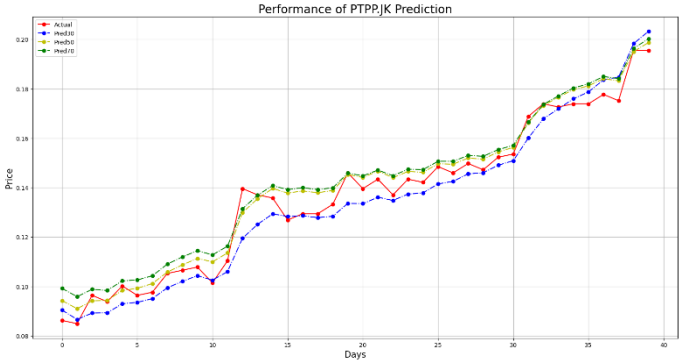

Figure 17 Performance of PTPP.JK

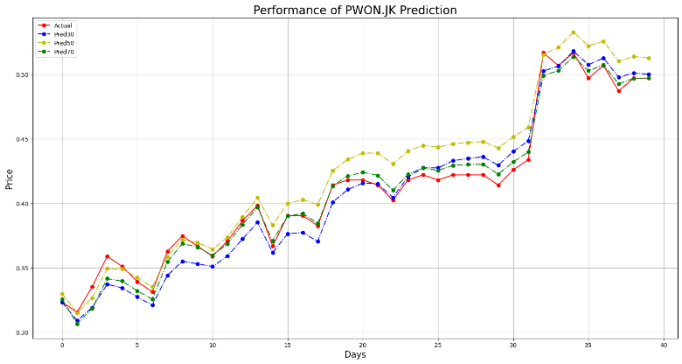

Figure 18 Performance of PWON.JK

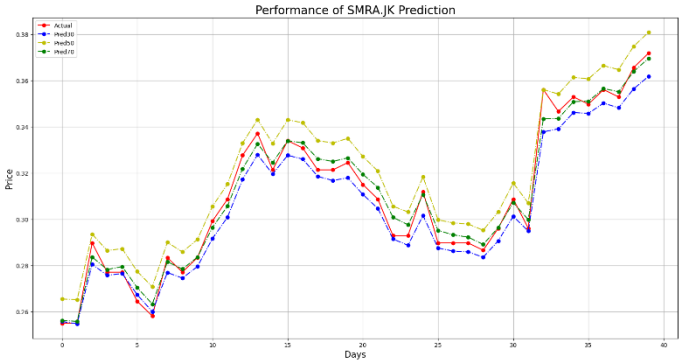

Figure 19 Performance of SMRA.JK

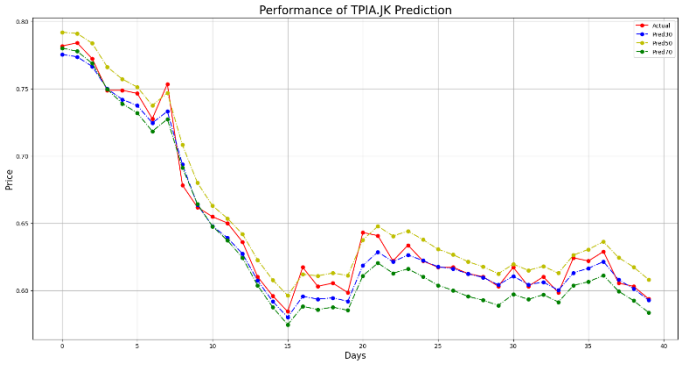

Figure 20 Performance of TPIA.JK

3.1 Impact of Input Sequence Lengths

The investigation into the StacBi LSTM model's performance across varied input sequence lengths (30, 50, and 70 days) elucidates the nuanced relationship between sequence length and predictive accuracy, assessed through key performance indicators including RMSE, MAE, MAPE, and R2. A general trend of enhanced predictive accuracy with increased input sequence length is discernible, as longer sequences give the model a more comprehensive historical context, reducing prediction errors. Nevertheless, this enhancement is not consistent across all stocks. For some, there may be a threshold beyond which extending the sequence length yields diminishing returns, as evidenced by fluctuations in RMSE and MAE values. This implies that overly extensive sequences may inadvertently introduce noise or foster overfitting, compromising prediction quality. The MAPE metric, denoting percentage errors, corroborates these findings, indicating that while longer sequences typically correlate with reduced percentage errors, the model may still grapple with predicting extreme market fluctuations, occasionally resulting in elevated MAPE scores.

Additionally, R2 values, reflecting the model's capacity to capture stock price variations, generally increase with longer sequences, signifying enhanced explanatory power. However, this trend may plateau or reverse beyond a certain sequence length. In summary, the relationship between input sequence length and model performance is complex and non-linear; optimizing sequence length necessitates a delicate balance between harnessing more historical information and mitigating the risks of noise introduction and overfitting.

3.2 Optimal Input Sequence Length

After analyzing the performance metrics of the StacBi LSTM model for predicting stock prices using input sequence lengths of 30, 50, and 70 days, an optimal range for prediction was observed. The 50-day input sequence length often results in lower RMSE, MAE, and MAPE

values than shorter or longer alternatives, making it a balanced choice. This implies that it effectively assimilates an adequate amount of historical data for precise predictions while concurrently circumventing the perils of noise and overfitting associated with unduly lengthy sequences. Shorter sequences like the 30-day option may enhance computational efficiency due to their reduced data and computational demands. Still, they risk neglecting longer-term trends vital for accurate forecasting, a drawback particularly pronounced in turbulent market conditions. Conversely, the 70-day sequence length holds the potential to discern more complex stock price patterns but at the expense of heightened computational requirements and elevated overfitting risk, especially in volatile and rapidly changing markets. Thus, the 50-day input sequence length balances efficiency with predictive accuracy, capturing a comprehensive range of short- to medium-term patterns while averting the extremes of sequence length. Nonetheless, it is crucial to acknowledge that the ideal sequence length may vary, contingent upon the individual characteristics of each stock and the broader market context.

3.3 Comparative Analysis

The comparative analysis of performance metrics across varying input sequence lengths (30, 50, and 70 days) for the StacBi LSTM model elucidates the inherent trade-offs between prediction precision and explanatory power and the challenges in optimizing both concurrently. Shorter sequences, such as the 30 days, exhibit higher precision in forecasting imminent stock price movements for certain stocks, as evidenced by lower RMSE, MAE, and MAPE values. However, these stocks tend to display slightly reduced R2 values, indicating a compromise in the model's ability to account for overall price variability due to the restricted historical context. Conversely, a 70-day sequence length often results in enhanced explanatory power, capturing a broader spectrum of price fluctuations as reflected in higher R2 values. Yet, it can also lead to increased prediction errors and potential overfitting, as denoted by elevated RMSE, MAE, and MAPE scores. Instances where the model excels in either precision or explanatory power, but not both, underscore the complexity of achieving an optimal balance and highlight the strategic nature of selecting an input sequence length tailored to specific stocks and market conditions. Shorter sequences may be preferable for day traders prioritizing immediate accuracy, while longer sequences could benefit investors seeking a comprehensive understanding of overall price trends. This analysis ultimately emphasizes the criticality of a nuanced understanding of each performance metric and the necessity of a strategic and informed approach to balance precision with explanatory power, aligning with market participants' specific objectives and strategies.

The StacBi LSTM model demonstrates notable strengths in stock price prediction, capitalizing on its unique architecture to capture long-term dependencies and temporal patterns within financial data. Its dynamic adaptability to changing market conditions, combined with a deep architecture skilled at handling non-linearities and noisy data, offers a comprehensive framework for precise predictions and insightful market analysis despite sudden market shifts and volatility challenges. In the Indonesian Stock Exchange context, the model showcases its versatility, effectively interpreting local market patterns while navigating periods of volatility characteristic of emerging markets. Nonetheless, the research process highlighted data-related challenges, necessitating meticulous preprocessing to maintain data integrity and prevent biases. Balancing model complexity with computational demands is crucial, as excessive complexity can lead to overfitting and inefficient resource use. The StacBi LSTM model is a valuable tool for traders and investors, with optimal input sequence length selection enhancing predictive accuracy and informing robust trading strategies. Integrating the model with other predictive techniques, external factors, and novel architectures presents promising avenues for advancing stock price prediction capabilities.

4. KESIMPULAN

This research paper investigated the application of the StacBi LSTM model for stock price prediction. The key findings highlight that the choice of input sequence length significantly impacts the model's performance. An optimal input sequence length was identified through rigorous evaluation, offering a balance between computational efficiency and predictive accuracy. Notably, the StacBi LSTM model demonstrated a remarkable ability to capture stock price trends by

effectively incorporating long-term dependencies and temporal patterns. The model's strengths surpassed traditional methods, enabling traders and investors to make more informed decisions.

This study opens avenues for practical applications and future enhancements in stock price prediction. The insights gained from the optimal input sequence length can guide decision-making for predictive models. At the same time, the StacBi LSTM's adeptness in capturing stock price trends underscores its potential in real-world financial forecasting scenarios. Future directions could involve hybrid approaches, integrating external factors, and addressing market shifts. In conclusion, this research underscores the significance of leveraging deep learning techniques like the StacBi LSTM for stock price prediction, presenting an impactful tool that bridges the gap between data-driven insights and effective financial strategies.

REFERENCES

Ahmad, I., Wang, X., Zhu, M., Wang, C., Pi, Y., Khan, J. A., Khan, S., Samuel, O. W., Chen, S., & Li, G. (2022). EEG-Based Epileptic Seizure Detection via Machine/Deep Learning Approaches: A Systematic Review. *Computational Intelligence and Neuroscience*, *2022*, 1–20. https://doi.org/10.1155/2022/6486570

Aydin, M., Pata, U. K., & Inal, V. (2022). Economic policy uncertainty and stock prices in BRIC countries: evidence from asymmetric frequency domain causality approach. *Applied Economic Analysis*, *30*(89), 114–129. https://doi.org/10.1108/AEA-12-2020-0172/FULL/PDF

Baek, J.-W., & Chung, K. (2023). Multi-Context Mining-Based Graph Neural Network for Predicting Emerging Health Risks. *IEEE Access*, *11*, 15153–15163. https://doi.org/10.1109/ACCESS.2023.3243722

Brahma, B., & Wadhvani, R. (2020). Solar Irradiance Forecasting Based on Deep Learning Methodologies and Multi-Site Data. *Symmetry*, *12*(11), 1830. https://doi.org/10.3390/sym12111830

Campbell, T., Dixon, K. W., Dods, K., Fearns, P., & Handcock, R. (2020). Machine Learning Regression Model for Predicting Honey Harvests. *Agriculture*, *10*(4), 118. https://doi.org/10.3390/agriculture10040118

Choi, J.-E., & Shin, D. W. (2019). The roles of differencing and dimension reduction in machine learning forecasting of employment level using the FRED big data. *Communications for Statistical Applications and Methods*, *26*(5), 497–506. https://doi.org/10.29220/CSAM.2019.26.5.497

Erizal, E., & Diqi, M. (2023). Performance Evaluation of Stock Prediction Models using EMAGRU. *Applied Computer Science*, *19*(3), 160–173. https://doi.org/10.35784/acs-2023-30

Fathy, Y., Jaber, M., & Brintrup, A. (2021). Learning With Imbalanced Data in Smart Manufacturing: A Comparative Analysis. *IEEE Access*, *9*, 2734–2757. https://doi.org/10.1109/ACCESS.2020.3047838

Gao, Y., Lian, J., & Gong, B. (2018). Automatic classification of refrigerator using doubly convolutional neural network with jointly optimized classification loss and similarity loss. *Eurasip Journal on Image and Video Processing*, *2018*(1), 1–11. https://doi.org/10.1186/S13640-018-0329-Z/FIGURES/9

Gutmann, S., Maget, C., Spangler, M., & Bogenberger, K. (2021). Truck Parking Occupancy Prediction: XGBoost-LSTM Model Fusion. *Frontiers in Future Transportation*, *2*, 693708. https://doi.org/10.3389/ffutr.2021.693708

Htun, H. H., Biehl, M., & Petkov, N. (2023). Survey of feature selection and extraction techniques for stock market prediction. *Financial Innovation*, *9*(1), 1–25. https://doi.org/10.1186/S40854-022-00441-7/FIGURES/3

Jamous, R., ALRahhal, H., & El-Darieby, M. (2021). A New ANN-Particle Swarm Optimization with Center of Gravity (ANN-PSOCoG) Prediction Model for the Stock Market under the Effect of COVID-19. *Scientific Programming*, *2021*, 1–17. https://doi.org/10.1155/2021/6656150

Jiang, H., Fang, D., Spicher, K., Cheng, F., & Li, B. (2019). A New Period-Sequential Index Forecasting Algorithm for Time Series Data. *Applied Sciences*, *9*(20), 4386. https://doi.org/10.3390/app9204386

Kim, B., Yuvaraj, N., Sri Preethaa, K. R., Hu, G., & Lee, D.-E. (2021). Wind-Induced Pressure Prediction on Tall Buildings Using Generative Adversarial Imputation Network. *Sensors*, *21*(7), 2515. https://doi.org/10.3390/s21072515

Lind, A. P., & Anderson, P. C. (2019). Predicting drug activity against cancer cells by random forest models based on minimal genomic information and chemical properties. *PLOS ONE*, *14*(7), e0219774. https://doi.org/10.1371/journal.pone.0219774

Lokanan, M. (2022). The determinants of investment fraud: A machine learning and artificial intelligence approach. *Frontiers in Big Data*, *5*, 961039. https://doi.org/10.3389/FDATA.2022.961039/BIBTEX

Ma, S.-C., Chou, W., Chien, T.-W., Chow, J. C., Yeh, Y.-T., Chou, P.-H., & Lee, H.-F. (2020). An App for Detecting Bullying of Nurses Using Convolutional Neural Networks and Web-Based Computerized Adaptive Testing: Development and Usability Study. *JMIR MHealth and UHealth*, *8*(5), e16747. https://doi.org/10.2196/16747

Miftahurrohmah, B., Wulandari, C., & Dharmawan, Y. S. (2021). Investment Modelling Using Value at Risk Bayesian Mixture Modelling Approach and Backtesting to Assess Stock Risk. *Journal of Information Systems Engineering and Business Intelligence*, *7*(1), 11. https://doi.org/10.20473/jisebi.7.1.11-21

Mo, S., Lu, P., & Liu, X. (2022). AI-Generated Face Image Identification with Different Color Space Channel Combinations. *Sensors*, *22*(21), 8228. https://doi.org/10.3390/s22218228

Musarat, M. A., Alaloul, W. S., Rabbani, M. B. A., Ali, M., Altaf, M., Fediuk, R., Vatin, N., Klyuev, S., Bukhari, H., Sadiq, A., Rafiq, W., & Farooq, W. (2021). Kabul River Flow Prediction Using Automated ARIMA Forecasting: A Machine Learning Approach. *Sustainability*, *13*(19), 10720. https://doi.org/10.3390/su131910720

Naumoski, A., Arsov, S., & Cvetkoska, V. (2022). Asymmetric Information and Agency Cost of Financial Leverage and Corporate Investments: Evidence from Emerging South-East European Countries. *Scientific Annals of Economics and Business*, *69*(2), 317–342. https://doi.org/10.47743/saeb-2022-0010

Pamir, Javaid, N., Akbar, M., Aldegheishem, A., Alrajeh, N., & Mohammed, E. A. (2022). Employing a Machine Learning Boosting Classifiers Based Stacking Ensemble Model for Detecting Non Technical Losses in Smart Grids. *IEEE Access*, *10*, 121886–121899. https://doi.org/10.1109/ACCESS.2022.3222883

Pan, D., Zeng, A., Jia, L., Huang, Y., Frizzell, T., & Song, X. (2020). Early Detection of Alzheimer's Disease Using Magnetic Resonance Imaging: A Novel Approach Combining Convolutional Neural Networks and Ensemble Learning. *Frontiers in Neuroscience*, *14*, 501050. https://doi.org/10.3389/FNINS.2020.00259/BIBTEX

Patel, R. K., Kumari, A., Tanwar, S., Hong, W.-C., & Sharma, R. (2022). AI-Empowered Recommender System for Renewable Energy Harvesting in Smart Grid System. *IEEE Access*, *10*, 24316–24326. https://doi.org/10.1109/ACCESS.2022.3152528

Qaddoura, R., M. Al-Zoubi, A., Faris, H., & Almomani, I. (2021). A Multi-Layer Classification Approach for Intrusion Detection in IoT Networks Based on Deep Learning. *Sensors*, *21*(9), 2987. https://doi.org/10.3390/s21092987

Rajamoorthy, R., Saraswathi, H. V., Devaraj, J., Kasinathan, P., Elavarasan, R. M., Arunachalam, G., Mostafa, T. M., & Mihet-Popa, L. (2022). A Hybrid Sailfish Whale Optimization and Deep Long Short-Term Memory (SWO-DLSTM) Model for Energy Efficient Autonomy in India by 2048. *Sustainability*, *14*(3), 1355. https://doi.org/10.3390/su14031355

Sekiguchi, Hayashi, Sugino, & Terada. (2019). The Effects of Differences in Individual Characteristics and Regional Living Environments on the Motivation to Immigrate to Hometowns: A Decision Tree Analysis. *Applied Sciences*, *9*(13), 2748. https://doi.org/10.3390/app9132748

Shuai, C., Pan, Z., Gao, L., & Zuo, H. (2021). Short-Term Traffic Flow Prediction of Expressway: A Hybrid Method Based on Singular Spectrum Analysis Decomposition. *Advances in Civil Engineering*, *2021*, 1–10. https://doi.org/10.1155/2021/4313970

Succetti, F., Rosato, A., Di Luzio, F., Ceschini, A., & Panella, M. (2022). A Fast Deep Learning Technique for Wi-Fi-Based Human Activity Recognition. *Progress In Electromagnetics Research*, *174*, 127–141. https://doi.org/10.2528/PIER22042605

Suleman, M. A. R., & Shridevi, S. (2022). Short-Term Weather Forecasting Using Spatial Feature Attention Based LSTM Model. *IEEE Access*, *10*, 82456–82468. https://doi.org/10.1109/ACCESS.2022.3196381

Tang, L., & Mahmoud, Q. H. (2022). A Deep Learning-Based Framework for Phishing Website Detection. *IEEE Access*, *10*, 1509–1521. https://doi.org/10.1109/ACCESS.2021.3137636

Wang, G., Cao, L., Zhao, H., Liu, Q., & Chen, E. (2021). Coupling Macro-Sector-Micro Financial Indicators for Learning Stock Representations with Less Uncertainty. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(5), 4418–4426. https://doi.org/10.1609/aaai.v35i5.16568

Williams, R. I., Smith, A., Aaron, J. R., Manley, S. C., & McDowell, W. C. (2020). Small business strategic management practices and performance: A configurational approach. *Economic Research-Ekonomska Istraživanja*, *33*(1), 2378–2396. https://doi.org/10.1080/1331677X.2019.1677488

Zhang, J., Olatosi, B., Yang, X., Weissman, S., Li, Z., Hu, J., & Li, X. (2022). Studying patterns and predictors of HIV viral suppression using A Big Data approach: a research protocol. *BMC Infectious Diseases*, *22*(1), 122. https://doi.org/10.1186/s12879-022-07047-5