Heliyon 10 (2024) e40819

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Hybrid preprocessing for neural network-based stock price prediction [☆]

Jian-Lei Li [*], Wei-Kang Shi

North China University of Water Resources and Electric Power, Zhengzhou, Henan, 450011, PR China

ARTICLE INFO

Dataset link: https://www.kaggle.com/datasets/harbhajansingh21/top-50-canadian-stocks-since-2010

MSC:
TP183

Keywords:
Empirical wavelet transform (EWT)
Dynamic time warping (DTW)
Principal component analysis (PCA)
Neural network
Stock price prediction

ABSTRACT

In the domain of stock price prediction, the intricate interdependencies within multivariate time series data present significant challenges for accurate forecasting. This paper introduces a groundbreaking hybrid preprocessing technique to tackle this issue. By leveraging the Empirical Wavelet Transform (EWT), we adeptly extract both low-frequency and high-frequency components from the time series. We then apply Dynamic Time Warping (DTW) and Differential Dynamic Time Warping (DDTW) to measure component similarities, identifying correlated patterns within the stock price series. High-frequency components are managed using sliding windows and Principal Component Analysis (PCA), while PCA is directly applied to low-frequency components. Integrating these techniques into neural network models, our approach yields a substantial 30% improvement in prediction accuracy compared to traditional methods. This significant advancement underscores the potential of our hybrid preprocessing method in enhancing stock price prediction accuracy, offering valuable insights for financial market analysis.

1. Introduction

Time series forecasting [1] is a crucial research area focusing on analyzing and processing data sequences that vary over time [2], to predict future trends [3], patterns [4], or values [5]. Research in this field is vital for various industries, including financial market analysis [6], weather forecasting, energy demand prediction [7], and inventory management. Accurate time series forecasting can help businesses and organizations make more informed decisions [8], optimize resource allocation [9], reduce uncertainty, and enhance efficiency.

In recent years, neural networks have become prevalent in stock price prediction due to their powerful nonlinear modeling capabilities. Despite their strengths, neural networks often require substantial computational resources and may sometimes fall short in prediction accuracy.

Stock price prediction is inherently challenging due to the noise and volatility of financial markets. Many models have been proposed to tackle this issue, but they often struggle to capture the complex patterns in stock price data, especially without adequate preprocessing. Recent studies have integrated preprocessing techniques with advanced models to enhance prediction accuracy.

For instance, Quilty et al. [10] utilized classification algorithms with data preprocessing to reduce prediction errors. Ma et al. [11] introduced the Multi-source Aggregated Classification (MAC) method, which uses graph convolutional networks and pre-trained em-

[☆] This research was supported by Basic Research Projects of Key Scientific Research Projects Plan in Henan Higher Education Institutions (No. 25ZX013) and Henan Province Postgraduate Quality Courses (YJS2022KC01).

[*] Corresponding author.
E-mail address: hnmaths@163.com (J.-L. Li).

https://doi.org/10.1016/j.heliyon.2024.e40819
Received 23 March 2024; Received in revised form 28 November 2024; Accepted 28 November 2024

Available online 2 December 2024

bedding feature generators to outperform state-of-the-art baselines in prediction accuracy and financial metrics. Similarly, Sonkamble et al. [12] employed Principal Component Analysis (PCA) to simplify data before applying linear regression, which improved predictive precision for Tesla stock data. Wang [13] compared machine learning models for Apple stock, finding that linear regression and Long Short-Term Memory (LSTM) networks were most effective, with LSTM showing robustness for time-series data.

Other notable studies include Rajpurohit et al. [14], who integrated LSTM with sentiment analysis for data preprocessing to enhance prediction accuracy. Liang et al. [15] combined LSTM with wavelet transform-based preprocessing, showing significant improvements in performance. Chen et al. [16] proposed the XCEEMDAN-Bidirectional LSTM-Spline model, which effectively utilized feature extraction for superior performance. Cai [17] introduced a CNN-GRU-attention model employing various decomposition methods like EMD, EEMD, and CEEMDAN, which outperformed conventional models.

Furthermore, Jiang et al. [18] proposed a model integrating affinity propagation clustering with convolutional neural networks (CNN) for feature extraction, and LSTM for multifactor analysis, achieving superior prediction performance. Rezaei et al. [19] combined empirical mode decomposition (EMD) techniques with CNN and LSTM models, significantly enhancing feature extraction and prediction accuracy. Lastly, Zarandi et al. [20] developed a hybrid fuzzy intelligent agent-based system for stock price prediction, demonstrating the effectiveness of combining fuzzy logic with intelligent agents for financial forecasting.

These studies highlight the continuous evolution of methodologies aimed at improving stock price prediction, emphasizing the importance of innovative preprocessing and modeling techniques. Our proposed hybrid preprocessing technique aligns with these advancements, offering a novel combination of Empirical Wavelet Transform (EWT), Dynamic Time Warping (DTW), Derivative Dynamic Time Warping (DDTW), and PCA. This approach addresses both low-frequency trends and high-frequency fluctuations, significantly enhancing prediction accuracy and model robustness.

In conclusion, our hybrid preprocessing technique demonstrates superior performance in stock price prediction, as evidenced by closer predicted-actual curves and lower error indicators. This method not only captures complex patterns in stock data but also improves the overall robustness of the prediction models. Future research should continue optimizing preprocessing techniques and model hyperparameters to further enhance prediction accuracy and reliability for investors and decision-makers.

Abbreviation	Definition
EWT	Empirical Wavelet Transform
DTW	Dynamic Time Warping
DDTW	Differential Dynamic Time Warping
PCA	Principal Component Analysis

This paper introduces a novel hybrid preprocessing technique for stock price prediction [21], utilizing Empirical Wavelet Transform (EWT) [22] for decomposing time series into low-frequency trends and high-frequency details. Dynamic Time Warping (DTW) [23] and Derivative Dynamic Time Warping (DDTW) are combined for advanced similarity measurement, capturing both global similarity and local trends, which improves the identification of stock price patterns.

Applying these techniques to neural network models enhances their ability to learn complex patterns, significantly boosting prediction accuracy and robustness. By integrating EWT, DTW, and DDTW with neural networks, the method achieves superior precision in stock price prediction.

Key contributions of this paper include:

- Enhanced Predictive Accuracy: Demonstrating significant improvement in predictive accuracy and trend-tracking capabilities of neural network models through hybrid preprocessing.
- Comprehensive Model Comparison: Providing a detailed comparison of model performances with and without preprocessing.
- Future Research Directions: Highlighting the importance of preprocessing in multivariate time series forecasting and suggesting future optimization of preprocessing techniques and model hyperparameters.

The paper is structured as follows: Section 2 covers basic concepts like EWT, DTW, and PCA [24]. Section 3 details the hybrid preprocessing model. Section 4 presents experimental results and discussions. Section 5 concludes the paper.

2. Basic concepts

This section primarily discusses time series preprocessing techniques, mainly including decomposition techniques, similarity calculation, and principal component extraction.

2.1. Time series decomposition

In choosing EWT for decomposition, the main consideration is its ability to effectively handle nonlinear and non-stationary signals. Traditional wavelet transforms are less adaptive to signals and require predefined basis functions, whereas EWT dynamically adjusts basis functions based on actual data characteristics, making it more suitable for complex financial market data analysis.

The original signal is analyzed using FFT (Fast Fourier Transform), and according to the Shannon criterion, the signal's Fourier spectrum is normalized within $[0, \pi]$ and divided into continuous N segments $[A_1, A_2, \ldots, A_N]$, resulting in $N+1$ boundary lines on

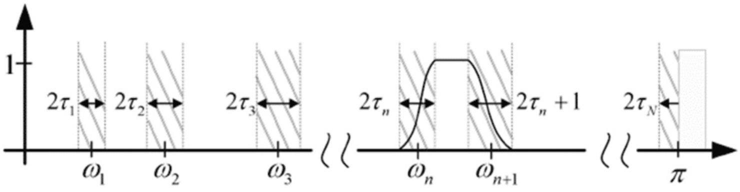


Fig. 1. Schematic diagram of the signal spectrum division.

the spectrum. Let ω_n be the interval boundary line, then $A_n = [\omega_{n-1}, \omega_n]$, where $\omega_0 = 0$ and $\omega_N = \pi$ are the interval boundary lines. The schematic diagram of the signal spectrum division is shown in Fig. 1.

The width of the transition phase is given by the equation:

$$\tau_n = \gamma \times \omega_n, \quad 0 < \gamma < \min\left(\frac{\omega_{n+1} - \omega_n}{\omega_{n+1} + \omega_n}\right) \tag{1}$$

After dividing the frequency band intervals, based on Meyer's theory, the scale function $\phi_n(\omega)$ and wavelet function $\psi_n(\omega)$ for each interval A_n are defined as follows:

$$\phi_n(\omega) = \begin{cases} 1, & |\omega| \le (1-\gamma)\omega_n \\ \cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\gamma\omega_n}\left(|\omega| - (1-\gamma)\omega_n\right)\right)\right], & (1-\gamma)\omega_n \le |\omega| \le (1+\gamma)\omega_n \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

$$\psi_n(\omega) = \begin{cases} 1, & (1+\gamma)\omega_n \le |\omega| \le (1-\gamma)\omega_{n+1} \\ \cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\gamma\omega_{n+1}}\left(|\omega| - (1-\gamma)\omega_{n+1}\right)\right)\right], & (1-\gamma)\omega_n \le |\omega| \le (1+\gamma)\omega_{n+1} \\ \sin\left[\frac{\pi}{2}\beta\left(\frac{1}{2\gamma\omega_n}\left(|\omega| - (1-\gamma)\omega_n\right)\right)\right], & (1-\gamma)\omega_n \le |\omega| \le (1+\gamma)\omega_n \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

In these expressions, we adopt $\beta(x) = x^4(35 - 84x + 70x^2 - 20x^3)$, $\tau_n = \gamma\omega_n$ with $0 < \gamma < 1$, and $\gamma < \min_n\left(\frac{\omega_{n+1}-\omega_n}{\omega_{n+1}+\omega_n}\right)$.

Constructing the detail coefficients $W_x(n,t)$ and approximation coefficients $W_x(0,t)$:

$$W_x(n,t) = \langle x(t), \psi_n(t)\rangle = \mathcal{F}^{-1}\left[\hat{x}(\omega)\overline{\psi_n(\bar{\omega})}\right] \tag{4}$$

$$W_x(0,t) = \langle x(t), \phi_0(t)\rangle = \mathcal{F}^{-1}\left[\hat{x}(\omega)\overline{\phi_0(\bar{\omega})}\right] \tag{5}$$

The original signal can be represented as the superposition of approximation and detail values, yielding the reconstructed signal of the Empirical Wavelet Transform:

$$X(t) = W_x(0,t)\phi_0(t) + \sum_{n=1}^{n} W_x(n,t)\psi_n(t) \tag{6}$$

The various modal components of the original signal after decomposition are:

$$c_0 = W_x(0,t)\phi_0(t) \tag{7}$$

$$c_n(t) = W_x(n,t)\psi_n(t), \quad n = 1, \ldots, n-1 \tag{8}$$

In conclusion, EWT as an innovative signal decomposition method not only enhances the performance and prediction accuracy of our model but also demonstrates its unique value in handling complex financial market data. It provides a powerful tool for researchers and engineers facing challenges in analyzing nonlinear and non-stationary signals.

In our model, EWT plays a crucial role in enhancing the preprocessing process: EWT simultaneously extracts the high-frequency and low-frequency components of the signal, crucial for capturing short-term fluctuations and long-term trends in stock price time series.

2.2. Similarity measurement

DTW is a widely used algorithm for comparing time series data. It's especially popular in fields like speech recognition and signal processing. DTW works by aligning two sequences, allowing for differences in timing and speed.

Given two time series $S = \{s_1, s_2, \ldots, s_n\}$ and $Q = \{q_1, q_2, \ldots, q_m\}$, an $n \times m$ distance matrix $D_{(n\times m)}$ is established, where $D(i,j)$ is determined by the distance between any two points, and $D(i,j) = ||s_i - q_i||_\omega$ represents the distance between points s_i and q_i, $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, m$, with $\omega = 2$ representing the Euclidean distance.

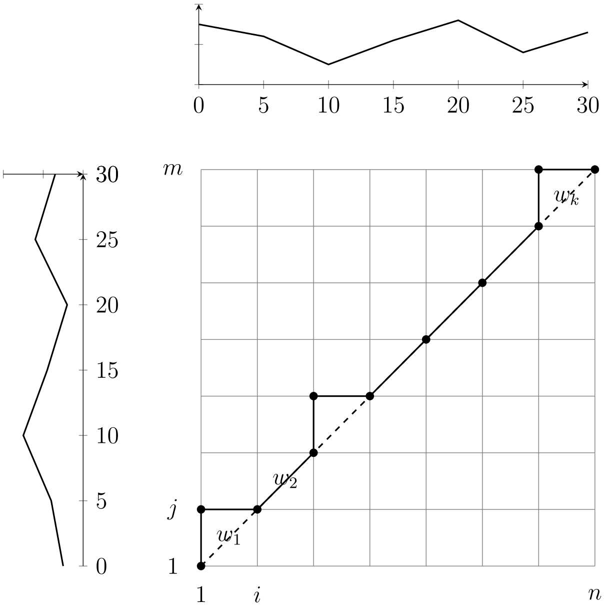


Fig. 2. Warping Path.

To calculate the Dynamic Time Warping distance $DTW(S,Q)$ between S and Q, it is necessary to find an optimal warping path:

$$P_{\text{best}} = \{p_1, p_2, \dots, p_k\} \quad (\max(n,m) \le K \le n+m+1) \tag{9}$$

such that the cumulative distance between S and Q is minimized. p_k represents the position of the warping path element in the distance matrix, $p_k = (i,j)_k$ indicates the matching relationship between s_i and q_i. Among the many valid paths, the unique optimal path is found that minimizes the cumulative distance, given by

$$D(S,Q) = \min\left\{\frac{1}{K}\sum_{k=1}^{K} D(p_k)\right\} \tag{10}$$

A cost matrix γ is constructed using dynamic programming, where each element is determined by:

$$\gamma(i,j) = D(i,j) + \min\{\gamma(i-1,j-1), \gamma(i-1,j), \gamma(i,j-1)\} \tag{11}$$

with $i = 1,2,\dots,n$, $j = 1,2,\dots,m$, and $\gamma(0,0) = 0$, $\gamma(i,0) = \gamma(0,j) = \infty$. $\gamma(n,m)$ is the minimum cumulative cost of measuring S and Q using Dynamic Time Warping, thus $DTW(S,Q) = \gamma(n,m)$.

As shown in Fig. 2, the Warping Path illustrates the alignment path between two time series using Dynamic Time Warping (DTW).

Derivative Dynamic Time Warping (DDTW) is an extension of DTW that improves alignment precision by computing the derivatives of time series. The specific steps are as follows:

- **Derivative Calculation:** Compute the derivatives of the time series to obtain the derivative sequences, which better reflect the signal's variation trends.
- **Distance Matrix Calculation:** Calculate the distance matrix based on the derivative sequences.
- **Cumulative Distance Matrix:** Compute the cumulative distance matrix to find the optimal matching path.
- **Path Planning:** Use a dynamic programming algorithm to find the optimal path that matches the derivative sequences.
- **Time Alignment:** Align the time series derivatives through the optimal path for precise comparison and analysis.
- **Advantages of DDTW:** DDTW captures subtle differences in time series changes, improving alignment accuracy and robustness.

Dynamic Time Warping (DTW) is effective for aligning time series data with low-frequency variations, finding the optimal alignment path that minimizes overall distance, and capturing broad trends and patterns.

DDTW extends DTW by computing derivatives of the time series, enhancing alignment precision for high-frequency data. This makes DDTW suitable for high-frequency data where subtle changes are crucial.

The selection of DTW for low-frequency components and DDTW for high-frequency components in our model is crucial for several reasons:

- **Capturing Long-term Trends:** DTW excels at aligning low-frequency variations, capturing overarching trends in stock price movements over extended periods.
- **Aligning Rapid Fluctuations:** DDTW aligns high-frequency components, such as short-term price fluctuations, for precise synchronization.
- **Enhancing Prediction Accuracy:** Leveraging both DTW and DDTW allows comprehensive analysis of stock price movements, improving prediction accuracy and robustness.

The main objective of this study is to leverage the strengths of both DTW and DDTW for improved stock price prediction. By using DTW for low-frequency components, we capture long-term trends, and by using DDTW for high-frequency components, we ensure accurate alignment of short-term fluctuations. This hybrid approach aims to enhance overall prediction accuracy by combining the benefits of both techniques.

2.3. MPCA

MPCA is a statistical method used to reduce the dimensionality of complex multivariate data while retaining as much variability as possible in the dataset. It achieves this by extracting a smaller set of orthogonal vectors called principal components that capture the main patterns and variations in the data.

Assume $X = (x_1, x_2, x_3, \dots, x_n)^T$ is an n dimensional random variable, and assuming the existence of the second moment, let $\mu = E(X), \Sigma = \text{var}(X)$. Then there exists the following linear function:

$$\begin{cases} y_1 = a_{11}x_1 + a_{21}x_2 + a_{31}x_3 + \dots + a_n x_n = A_1^T X \\ y_2 = a_{12}x_1 + a_{22}x_2 + a_{32}x_3 + \dots + a_{n2}x_n = A_2^T X \\ \vdots \\ y_n = a_{1n}x_1 n + a_{2n}x_2 + \dots + a_{nn}x_n = A_n^T X \end{cases} \tag{12}$$

Through linear transformation, we obtain:

$$\text{var}(y_i) = A_i^T \Sigma A_i, \quad i = 1, 2, \dots, n, \tag{13}$$

$$\text{cov}(y_i, y_j) = A_i^T \Sigma A_j, \quad i, j = 1, 2, \dots, n. \tag{14}$$

Generally, the greater the var(y), the more information is reflected by the variable y. Given the condition $||A_1|| = 1$, find A_1 to maximize var(y_1), under which y_1 is termed the first principal component variable. If the first principal component variable does not contain sufficient information, the second principal component variable is considered. To ensure the principal components are uncorrelated, it's required that $\text{cov}(y_1, y_2) = 0$, and so on for subsequent principal components.

The core idea of MPCA is to find a linear transformation that projects the original multidimensional data into a lower-dimensional space, which is composed of several principal components capturing the main variability in the data. Let the sample data matrix from the population be:

$$X = \begin{bmatrix} x_{11} & x_{21} & \cdots & x_{1p} \\ x_{12} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1n} & x_{2n} & \cdots & x_{np} \end{bmatrix} = [X_1, X_2^T, \dots, X_n^T] \tag{15}$$

The sample covariance matrix and the sample correlation matrix are respectively:

$$S = \frac{1}{(n-1)} \sum_{i=1}^{n} (x_i - \bar{x})(x_i - \bar{x})^T = (s_{ij})_{p \times p} \tag{16}$$

$$R = (r)_{p \times p} = \left(\frac{S_{ii}}{\sqrt{S_{ii}} \cdot \sqrt{S_{jj}}} \right)_{p \times p} \tag{17}$$

where $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} X_i$. The procedure for MPCA involves computing the covariance matrix S of the centered data, calculating its eigenvalues and eigenvectors, and selecting the principal components based on these eigenvalues.

The selection of MPCA is motivated by its ability to transform and condense complex data into a form that retains meaningful information for modeling tasks. This method is particularly useful when dealing with datasets containing correlated variables or when exploring relationships across multiple dimensions.

Fig. 3 illustrates the workflow for applying MPCA to data, highlighting the sequential steps involved in transforming and extracting principal components from the original dataset.

In the context of analyzing stock market data, MPCA serves as a robust technique for reducing the complexity of multidimensional data and extracting key features that enhance the interpretability and predictive performance of models. By transforming high-dimensional stock metrics into a smaller set of principal components, MPCA helps capture the underlying patterns and variations in

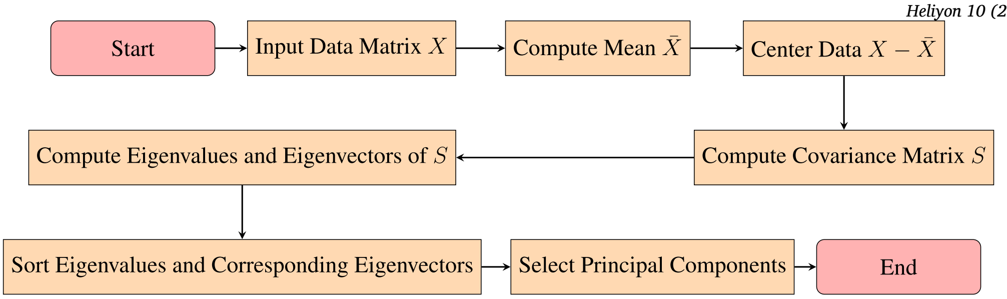

Start
Input Data Matrix X
Compute Mean $\bar{X}$
Center Data $X - \bar{X}$
Compute Covariance Matrix S
Compute Eigenvalues and Eigenvectors of S
Sort Eigenvalues and Corresponding Eigenvectors
Select Principal Components
End

Fig. 3. Workflow for Multivariate Principal Component Analysis (MPCA).

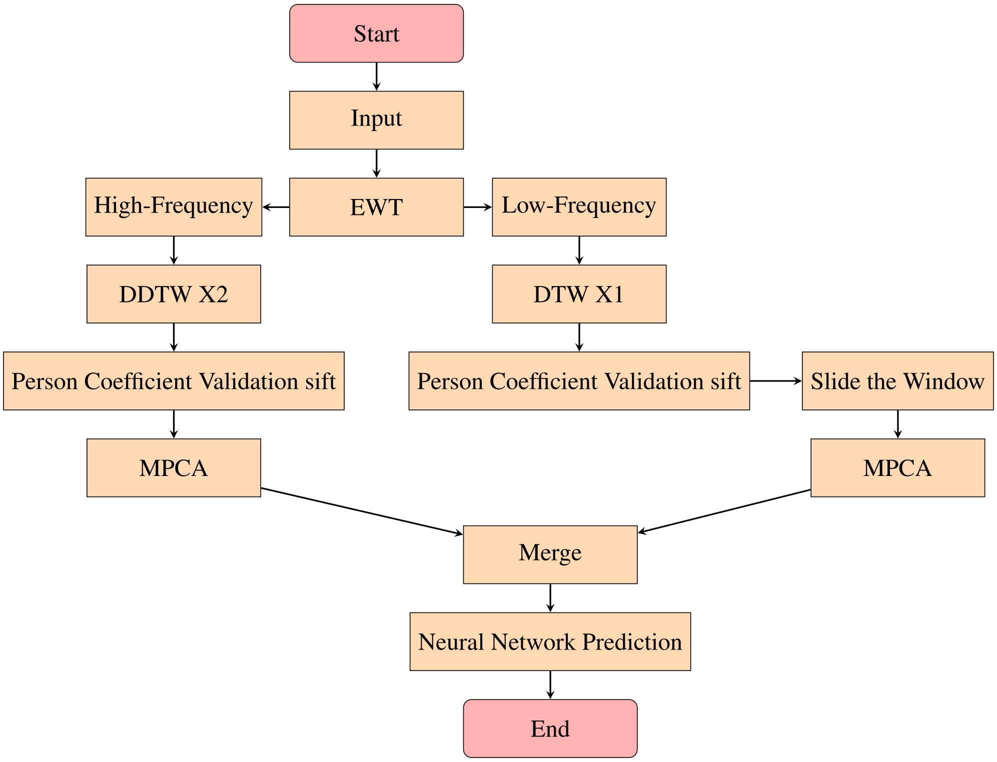

Start
Input
High-Frequency
EWT
Low-Frequency
DDTW X2
DTW X1
Person Coefficient Validation sift
Person Coefficient Validation sift
Slide the Window
MPCA
MPCA
Merge
Neural Network Prediction
End

Fig. 4. Process flow of the hybrid preprocessing model.

stock price movements. This approach not only simplifies the modeling process but also improves the accuracy of predicting market trends and fluctuations, making it a valuable tool in financial analytics.

3. Neural network model with hybrid preprocessing

This section introduces the hybrid preprocessing model framework, integrating concepts discussed in Section 2 to construct the model. Fig. 4 visually depicts the workflow, illustrating the entire process from input data through meticulously designed processing steps to preprocessed data output. The framework incorporates critical stages such as data cleaning, feature selection, normalization, ensuring high-quality data for subsequent analysis and modeling phases, thereby enhancing overall accuracy and reliability.

By adopting this hybrid preprocessing model, researchers effectively address challenges posed by complex datasets, particularly high-dimensional, non-stationary, or noisy signals. This approach significantly improves data usability and analytical outcomes, making it a potent tool for data scientists and analysts.

The hybrid preprocessing model framework integrates the following techniques:

- EWT: Decomposes the time series into high-frequency and low-frequency components, capturing both short-term fluctuations and long-term trends.

Table 1
Model parameters and configurations.

Model	Time Step	Batch Size	Epochs	Hidden Dim	Learning Rate	Dropout	Specifics
BP	20	32	500	64	0.01	-	Fully connected layers
LSTM	20	32	500	64	0.01	-	Tanh activation
RNN	20	32	500	64	0.01	-	Tanh activation
AM-LSTM	20	32	500	64	0.01	0.2	Tanh activation
CNN-BILSTM	20	32	500	64	0.01	0.1	ReLU activation

- Similarity Measurement: Measures similarity between high-frequency and low-frequency components. DTW is used for the low-frequency components, while DDTW is used for the high-frequency components. This alignment and comparison account for varying speeds and timing differences in the data.
- Pearson Coefficient Validation sift: Validates alignment and similarity from DTW and DDTW by calculating Pearson correlation coefficients, selecting the top three sequences with the highest coefficients, indicating strong alignment and robust preprocessing.
- MPCA: Reduces dimensionality by extracting principal components that retain significant variability. For low-frequency components, a sliding window of size 30 is applied before MPCA to enhance computational efficiency and interpretability.
- Neural Network Models: Uses preprocessed components from EWT, DTW, and PCA for training and predicting stock prices, leveraging comprehensive data insights.

The hybrid preprocessing model framework effectively addresses challenges in complex stock market datasets, such as high dimensionality, non-stationarity, and noise. It ensures thorough data preprocessing—cleaning, feature selection, and normalization—providing high-quality inputs for neural networks. This enhances prediction accuracy and model interpretability, making it a valuable tool for financial forecasting and decision-making.

This framework handles stock market complexity by decomposing time series data, processing frequency components, using DTW (Dynamic Time Warping) for comparison and alignment, and applying PCA (Principal Component Analysis) for dimensionality reduction. The preprocessed data are then used for neural network training and prediction. This method surpasses traditional models in managing high-dimensional, non-linear, and non-stationary data, leading to more accurate and understandable stock price forecasts. For financial professionals, this framework offers a systematic approach to market data analysis, aiding in more accurate predictions and better trading decisions.

4. Experiments and results

To validate the advantages of the hybrid preprocessing model, this study compares several neural network architectures: BP Neural Network, LSTM Neural Network, RNN, AM-LSTM Hybrid Neural Network, and CNN-BILSTM Hybrid Neural Network [25–27]. The models are evaluated using standardized parameters to assess the impact of hybrid preprocessing on their performance in time series forecasting tasks (Table 1).

- Activation Functions: Selected based on their suitability for the respective architectures; ReLU (Rectified Linear Unit) for introducing non-linearity and Tanh for bounded output range in LSTM.
- Learning Rate: Set to 0.01 to ensure stable convergence during training, balancing speed and accuracy.
- Dropout: Applied to prevent overfitting by randomly dropping units during training, with specific rates tailored to each model's complexity and data sensitivity.

This setup ensures each model configuration is optimized for the specific characteristics of the data and the architecture's requirements, aiming to maximize performance in time series forecasting tasks.

The following are the evaluation metrics used to assess the models:

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}_i| \tag{18}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} |(\hat{y}_i - y_i)^2|} \tag{19}$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N} \left(\frac{|y_i - \hat{y}_i|}{y_i} \times 100\right) \tag{20}$$

4.1. Experimental results

The provided explanation of the dataset is clear and comprehensive, but it can be slightly refined to ensure clarity and precision. Here's an improved version:

Table 2
Model evaluation results.

	BP	LSTM	RNN	AM-LSTM	CNN-BILSTM
Hybrid Preprocessing					
RMSE	10.68	3.07	2.85	4.12	0.44
MAE	8.48	1.87	1.75	2.81	0.32
MAPE	0.09	0.02	0.02	0.03	0.009
Without Hybrid Preprocessing					
RMSE	16.36	12.00	12.63	11.74	14.6
MAE	14.4	9.63	10.38	9.77	12.1
MAPE	0.16	0.10	0.11	0.11	0.13

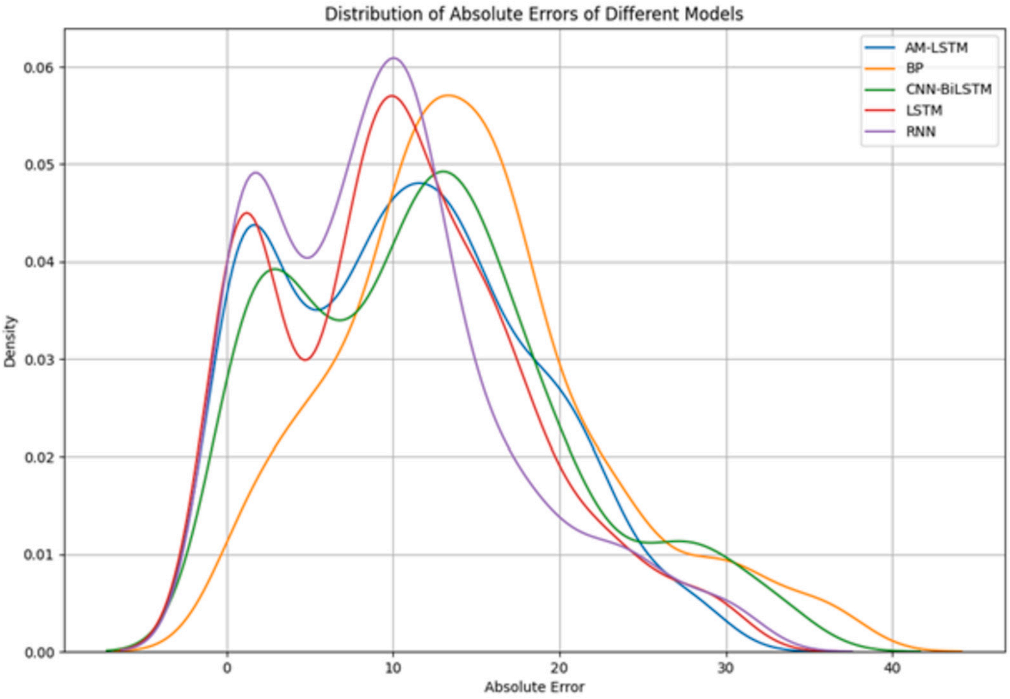


Fig. 5. No preprocessing.

We evaluated our model on a diverse dataset comprising high-cap, mid-cap, and small-cap stocks across multiple sectors, allowing us to test the robustness of our model under various market conditions. The dataset, sourced from Kaggle, includes daily stock data for Canada's top 30 stocks, spanning from January 2010 to October 2023. This dataset consists of 3460 data points, with 75% used for training and the remaining 25% for testing.

Two neural network models were employed in our evaluation: one utilizing hybrid preprocessing techniques and one without preprocessing. This setup allowed us to rigorously compare the performance and effectiveness of hybrid preprocessing in enhancing stock price prediction accuracy.

The evaluation metrics for both models on the test set are summarized in Table 2.

In the table, the hybrid preprocessing approach shows that the CNN-BILSTM model performs best with lower RMSE (0.44), MAE (0.32), and MAPE (0.009) values compared to other models like RNN, LSTM, and BP. Without preprocessing, all models perform worse, with AM-LSTM showing the best performance with RMSE (11.74), MAE (9.77), and MAPE (0.11) values, while BP performs poorly in both cases. Overall, CNN-BILSTM excels with hybrid preprocessing, while AM-LSTM performs relatively better without preprocessing, indicating the significant impact of preprocessing methods on model effectiveness.

Figs. 5 and 6 illustrate the absolute error distributions of five neural network models with and without hybrid preprocessing. With preprocessing, the error distributions flatten and shift closer to zero, notably for the CNN-BILSTM model, indicating improved alignment between predictions and actual values, especially for the BP model. Other models also show flatter and wider error curves post-preprocessing, suggesting enhanced generalization ability across different scenarios. Overall, hybrid preprocessing enhances these models' generalization characteristics, significantly boosting prediction accuracy and precision (Figs. 7 and 8).

We analyzed 100 initial predicted data points from different models, focusing on predictive accuracy and trend-tracking capabilities. For the AM-LSTM model without preprocessing, the prediction curve consistently underestimated actual value fluctuations.

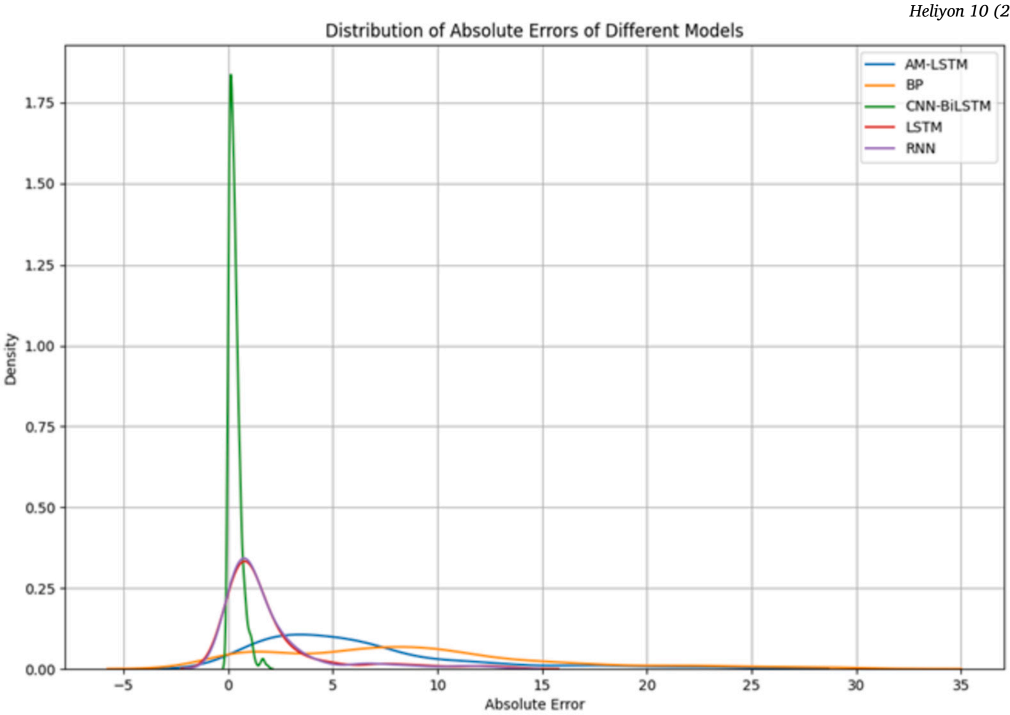


Fig. 6. Hybrid preprocessing.

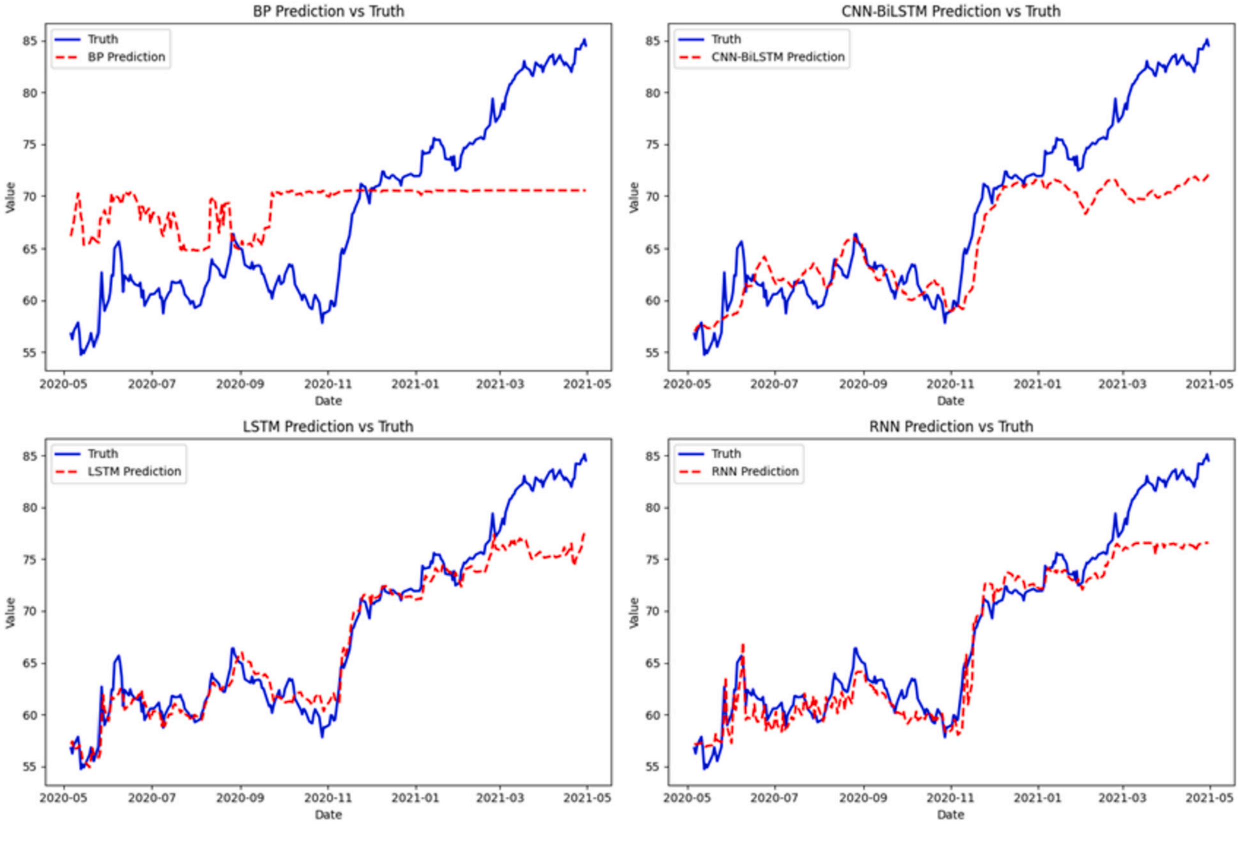
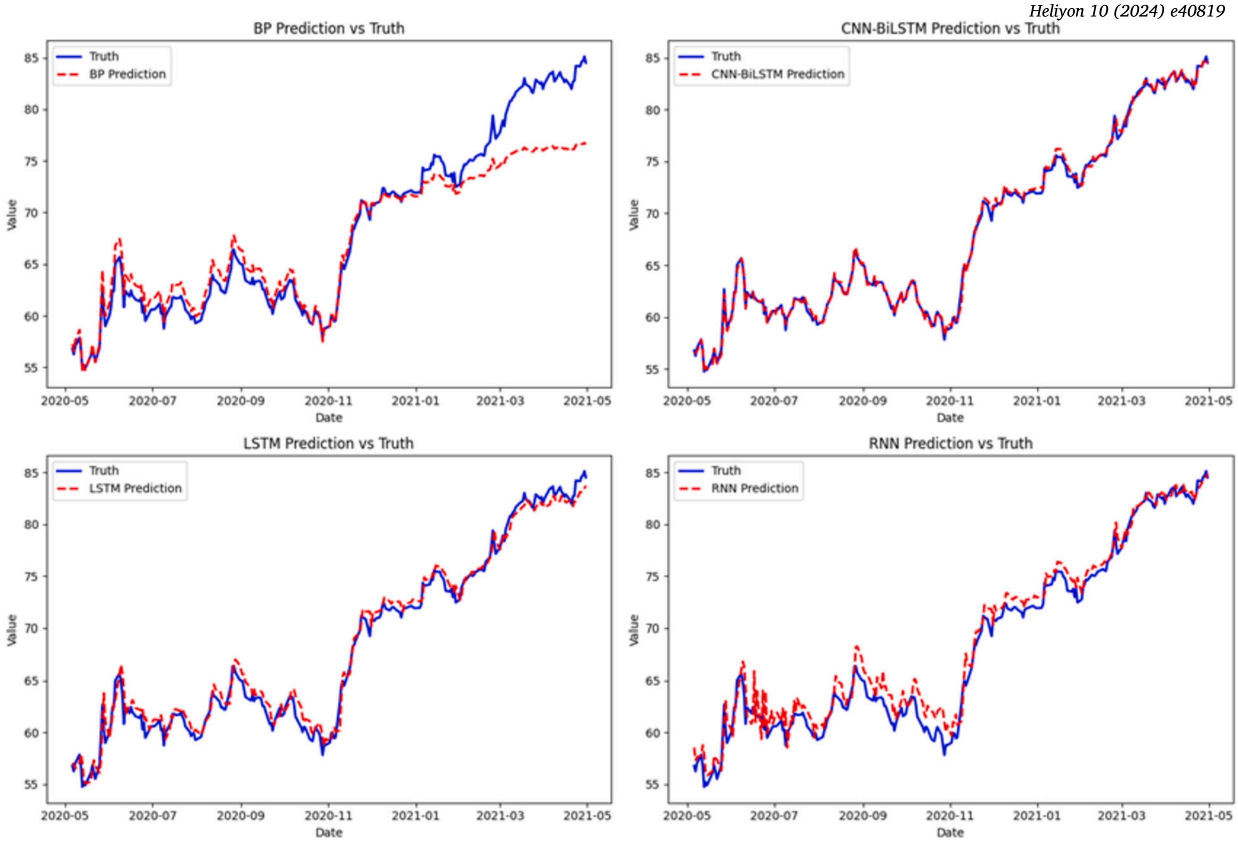


Fig. 7. No preprocessing prediction.


BP Prediction vs Truth

CNN-BiLSTM Prediction vs Truth

LSTM Prediction vs Truth

RNN Prediction vs Truth

Fig. 8. Hybrid preprocessing prediction.

With preprocessing, it aligned better with actual value trends, showcasing hybrid preprocessing's effectiveness. The BP model without preprocessing had prediction curve issues. Post-preprocessing, it closely matched actual values, improving trend tracking. The LSTM model without preprocessing diverged from actual values, especially at extreme points. After preprocessing, it performed better. The RNN model without preprocessing had lag and deviation issues. Preprocessing made its prediction curve smoother and more accurate. The CNN-BILSTM model with hybrid preprocessing outperformed others significantly. In summary, hybrid preprocessing consistently improved model performance in time series prediction, enhancing data understanding and trend tracking.

5. Conclusion

The analysis underscores the critical importance of preprocessing in stock price prediction research. Effective preprocessing enhances model understanding, sensitivity, and accuracy in capturing trends, which is particularly crucial in multivariate time series forecasting. Future research should focus on optimizing preprocessing techniques, as well as model architecture and hyperparameters, which require careful adjustments and testing.

In stock price prediction, comprehensive preprocessing is essential. Hybrid methods significantly boost prediction accuracy, as evidenced by closer predicted-actual curves and lower error indicators. To achieve optimal practical results, it is important to combine comprehensive preprocessing with powerful models. This approach can substantially enhance stock price prediction accuracy, providing reliable support for investors and decision-makers.

CRediT authorship contribution statement

Jian-Lei Li: Writing – review & editing, Methodology. **Wei-Kang Shi:** Writing – original draft.

Declaration of competing interest

The authors declare that they have no conflicts of interest.

Data availability

The dataset used in this study, which comprises the top 50 Canadian stocks since 2010, is publicly available on Kaggle at the following link: https://www.kaggle.com/datasets/harbhajansingh21/top-50-canadian-stocks-since-2010.

References

[1] V.S. Ediger, S. Akar, ARIMA forecasting of primary energy demand by fuel in Turkey, Energy Policy 35 (2007) 1701–1708.

[2] Kumarv Ujjwal, et al., ARIMA forecasting of ambient air pollutants (O_3, NO, NO_2 and CO), Stoch. Environ. Res. Risk Assess. (2010).

[3] C.W. Cheong, Modeling and forecasting crude oil markets using ARCH-type models, Energy Policy 37 (2009) 2346–2355.

[4] B. Krithikaivasan, Y. Zeng, K. Deka, et al., ARCH-based traffic forecasting and dynamic bandwidth provisioning for periodically measured nonstationary traffic, IEEE/ACM Trans. Netw. 15 (2007) 683–696.

[5] R.C. Garcia, J. Contreras, M.V. Akkeren, et al., A GARCH forecasting model to predict day-ahead electricity prices, IEEE Trans. Power Syst. 20 (2005) 867–874.

[6] X. Li, Application of neural networks in financial time series forecasting models, J. Funct. Spaces 2022 (2022) 1–9.

[7] M. Pegalajar, L.G.B. Ruiz, Time series forecasting for energy consumption, Energies 15 (2022) 773.

[8] C. Andreeski, D. Mechkaroska, Modelling, forecasting and testing decisions for seasonal time series in tourism, Acta Polytech. Hung. 17 (2020) 149–171.

[9] A. Ampountolas, Forecasting hotel demand uncertainty using time series Bayesian VAR models, Tour. Econ. 25 (2018) 734–756.

[10] J. Quilty, J. Adamowski, Addressing the incorrect usage of wavelet-based hydrological and water resources forecasting models for real-world applications with best practices and a new forecasting framework, J. Hydrol. (2018) 336–353, https://doi.org/10.1016/j.jhydrol.2018.05.003.

[11] Y. Ma, et al., Multi-source aggregated classification for stock price movement prediction, Inf. Fusion 91 (2023) 515–528, https://doi.org/10.1016/j.inffus.2022.10.025.

[12] S. Sonkamble, et al., Stock price prediction system, Int. J. Sci. Res. Comput. Sci. Eng. Inf. Tech. 9 (2) (2023) 273–277, https://doi.org/10.32628/cseit2390229.

[13] Y. Wang, Stock price prediction for technology company, Adv. Econ. Manag. Pol. Sci. 56 (1) (2023) 284–290, https://doi.org/10.54254/2754-1169/56/20231103.

[14] Aditya Singh Rajpurohit, Harshada Mhaske, Pradnya Sangitbabu Gaikwad, et al., Data preprocessing for stock price prediction using LSTM and sentiment analysis, in: 2023 6th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2023, pp. 1–6, https://doi.org/10.1109/ISCON57294.2023.10112026.

[15] Xiaodan Liang, Zhaodi Ge, Liling Sun, et al., LSTM with wavelet transform based data preprocessing for stock price prediction, Math. Probl. Eng. 2019 (2019) 1–8, https://doi.org/10.1155/2019/1340174.

[16] Kelvin Chen, Ronsen Purba, Arwin Halim, Stock price prediction using XCEEMDAN-bidirectional LSTM-spline, Indones. J. Artif. Intell. Data Min. 5 (1) (2022) 1, https://doi.org/10.24014/ijaidm.v5i1.14424.

[17] Chen Cai, Stock price prediction based on the fusion of CNN-GRU combined neural network and attention mechanism, in: 2023 6th International Conference on Electronics Technology (ICET), Chengdu, China, 2023, pp. 1166–1170, https://doi.org/10.1109/ICET58434.2023.10211379.

[18] N. Jiang, et al., Stock price prediction based on stock price synchronicity and deep learning, Int. J. Financ. Eng. 8 (2) (2021) 2141010, https://doi.org/10.1142/S2424786321410103.

[19] H. Rezaei, et al., Stock price prediction using deep learning and frequency decomposition, Expert Syst. Appl. 169 (2021) 114332, https://doi.org/10.1016/j.eswa.2020.114332.

[20] M.H. Fazel Zarandi, Esmaeil Hadavandi, I.B. Türkşen, A hybrid fuzzy intelligent agent-based system for stock price prediction, Int. J. Intell. Syst. 27 (11) (2012) 947–969, https://doi.org/10.1002/int.21554.

[21] H. Widiputra, A. Mailangkay, E. Gautama, Multivariate CNN-LSTM model for multiple parallel financial time-series prediction, Complexity 2021 (2021) 1–14.

[22] B. Xue, S. Zhou, C. Gu, et al., Morphological filtering enhanced empirical wavelet transform for mode decomposition, IEEE Access 7 (2019) 14283–14293.

[23] B.J. Jain, Making the dynamic time warping distance warping-invariant, Pattern Recognit. 94 (2019) 35–52.

[24] I.V. Nedaivoda, M.A. Primin, Y.V. Maslennikov, Algorithm for analysis of magnetocardiac signals: the principal component method, J. Commun. Technol. Electron. 64 (12) (2019) 1414–1421.

[25] E. Egriolgu, E. Bas, A new hybrid recurrent artificial neural network for time series forecasting, Neural Comput. Appl. 35 (2022) 2855–2865.

[26] J. Liu, X. Tang, X. Guan, Grain protein function prediction based on self-attention mechanism and bidirectional LSTM, Briefings in Bioinformatics 24 (1) (2023) bbac493, https://doi.org/10.1093/bib/bbac493.

[27] A.B. Farid, E. Fathy, A.S. Eldin, et al., Software defect prediction using hybrid model (CBIL) of convolutional neural network (CNN) and bidirectional long short-term memory (Bi-LSTM), PeerJ 7 (2021) e739.