Computational Economics
https://doi.org/10.1007/s10614-024-10629-x

PCA-ICA-LSTM: A Hybrid Deep Learning Model Based on Dimension Reduction Methods to Predict S&P 500 Index Price

Mehmet Sarıkoç[1] · **Mete Celik**[2]

Accepted: 9 May 2024

Abstract

In this paper, we propose a new hybrid model based on a deep learning network to predict the prices of financial assets. The study addresses two key limitations in existing research: (1) the lack of standardized datasets, time scales, and evaluation metrics, and (2) the focus on prediction return. The proposed model employs a two-stage preprocessing approach utilizing Principal Component Analysis (PCA) for dimensionality reduction and de-noising, followed by Independent Component Analysis (ICA) for feature extraction. A Long Short-Term Memory (LSTM) network with five layers is fed with this preprocessed data to predict the price of the next day using a 5 day time horizon. To ensure comparability with existing literature, experiments employ an 18 year dataset of the Standard & Poor's 500 (S&P500) index and include over 40 technical indicators. Performance evaluation encompasses six metrics, highlighting the model's superiority in accuracy and return rates. Comparative analyses demonstrate the superiority of the proposed PCA-ICA-LSTM model over single-stage statistical methods and other deep learning architectures, achieving notable improvements in evaluation metrics. Evaluation against previous studies using similar datasets corroborates the model's superior performance. Moreover, extensions to the study include adjustments to dataset parameters to account for the COVID-19 pandemic, resulting in improved return rates surpassing traditional trading strategies. PCA-ICA-LSTM achieves a 220% higher return compared to the "hold and wait" strategy in the extended S&P500 dataset, along with a 260% higher return than its closest competitor in the comparison. Furthermore, it outperformed other models in additional case studies.

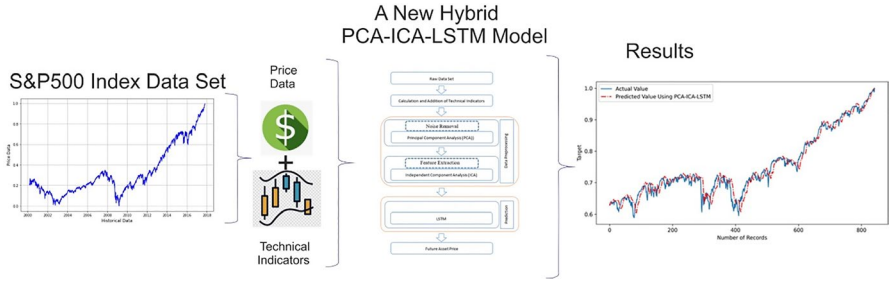
✉ Mehmet Sarıkoç
msarikoc@erciyes.edu.tr

[1] Distance Education Application and Research Center, Erciyes University, 38039 Kayseri, Turkey

[2] Department of Computer Engineering, Erciyes University, 38039 Kayseri, Turkey

Published online: 28 May 2024

Graphical Abstract



Keywords Financial time series · Price prediction · Principal component analysis (PCA) · Independent component analysis (ICA) · Deep learning · Long–short-term memory (LSTM)

Abbreviations

ANN	Artificial neural network
AR	Autoregressive
ARCH	Autoregressive conditional heteroskedasticity
ARIMA	Autoregressive integrated moving average
ARMA	Autoregressive moving average
CCC	Constant conditional correlation
CD	Correct down trend
CEEMD	Complementary ensemble empirical mode decomposition
CNN	Convolutional neural networks
CP	Correct up trend
DBN	Deep belief network
DQN	Deep Q-network
DS	Directional symmetry
EMA	Exponential moving average
EMD	Empirical mode decomposition
ESN	Echo state network
GRU	Gated recurrent units
ICA	Independent component analysis
ICs	Independent components
LASSO	Least absolute shrinkage and selection operator
LDA	Latent dirichlet allocation
LLY	Eli lilly and company
LSTM	Long–short-term memory
MA	Moving average
MAD	Mean absolute deviation
MAE	Mean absolute error
MAPE	Mean absolute percent error
MLP	Multi-layer perceptron

MSE	Mean square error
NB	Naive Bayes
NMSE	Normalized mean square error
NVDA	NVIDIA corporation
PCA	Principal component analysis
PCs	Principal components
PSO	Particle swarm optimization
R	Correlation coefficient
R^2	Coefficient of determination
RBFNN	Radial basis functions neural network
RBM	Restricted Boltzmann machine
RMSE	Root mean square error
RNN	Recurrent neural network
SE	Sample entropy
sICA	Scaled independent component analysis
sPCA	Scaled principal component analysis
SVM	Support vector machine
SVR	Support vector regression
S&P 500	Standard & poor's 500
TAR	Threshold autoregressive

1 Introduction

Data that changes over time is called time-series data. Time series data analysis is critical in the time-dependent application domains, such as health (Chialvo, 1987; Goldberger et al., 1988), finance (Hsieh, 1991; Peters, 1991), meteorology (Celik et al., 2014; Fraedrich, 1986; Nicolis & Nicolis, 1984; Özdoğan-Sarıkoç et al., 2023; Shekhar et al., 2008), industry (Huang et al., 2019; Mehdiyev et al., 2017), etc. Financial markets are one of the most challenging application areas to model and predict (Teixeira & De Oliveira, 2010). The complex nature of this field has led people to develop mathematical and statistical models that make the underlying structure clear, thereby making trade more efficient (Fayyad et al., 1996; Gandhmal & Kumar, 2019). Therefore, time series analysis and forecasting methods are frequently used in finance and economics (Abu-Mostafa & Atiya, 1996; Atsalakis & Valavanis, 2009; Kauffman et al., 2015; Kim, 2003; Pai & Lin, 2005; Rounaghi & Zadeh, 2016; Tay & Cao, 2001; Zhang, 2003). Recently, there has been remarkable progress in the development of deep learning models, which is a sub-branch of machine learning and has been shown to achieve successful results in various studies (Abu-Mostafa & Atiya, 1996; Nosratabadi et al., 2020; Ozbayoglu et al., 2020; Sezer et al., 2020).

The primary purpose of analyzing financial data is to predict stock market characteristics' effects and future directions for decision mechanisms based on market behavior (Cavalcante et al., 2016). In particular, financial time series forecasting constitutes the core of future decisions and transactions in the financial asset market. These forecasts help investors' decisions and reduce potential risks.

Investors always try to determine and predict the probable value of the stock or asset before deciding on their transactions (Gandhmal & Kumar, 2019; Teixeira & De Oliveira, 2010). Therefore, when the studies on financial data are analyzed, the stock market stands out as the most researched area (Nosratabadi et al., 2020). The stock market's primary target in the stock market research area is stock price or trend prediction (Cavalcante et al., 2016).

However, the large, complex, and variable structure of financial data makes it challenging to analyze and predict such data. In the reviews on financial markets, studies are generally classified according to the model used and the types of inputs (Bustos & Pomares-Quimbaya, 2020). The prediction models that can be used are divided into traditional prediction models and machine learning prediction models (Cavalcante et al., 2016; Liu et al., 2021; Nosratabadi et al., 2020). Some of the known traditional prediction models are autoregressive moving average (ARMA) and the autoregressive conditional heteroskedasticity (ARCH), while we can list some machine learning prediction models as support vector machine (SVM), naive Bayes (NB), artificial neural network (ANN). However, some authors state that traditional forecasting models are not as efficient as models based on artificial intelligence because they treat financial time series as linear systems (Atsalakis & Valavanis, 2009; Cavalcante et al., 2016; Li & Bastos, 2020). Studies in the literature are classified not only based on the mentioned model types but also according to input types. Figure 1 presents the input types to be used in the prediction models. The input type chosen for the forecasting model affects the forecasting model's performance. For this reason, a dataset previously used in the literature was preferred in our study (Sethia & Raut, 2019; Thakkar & Chaudhari, 2021). At the same

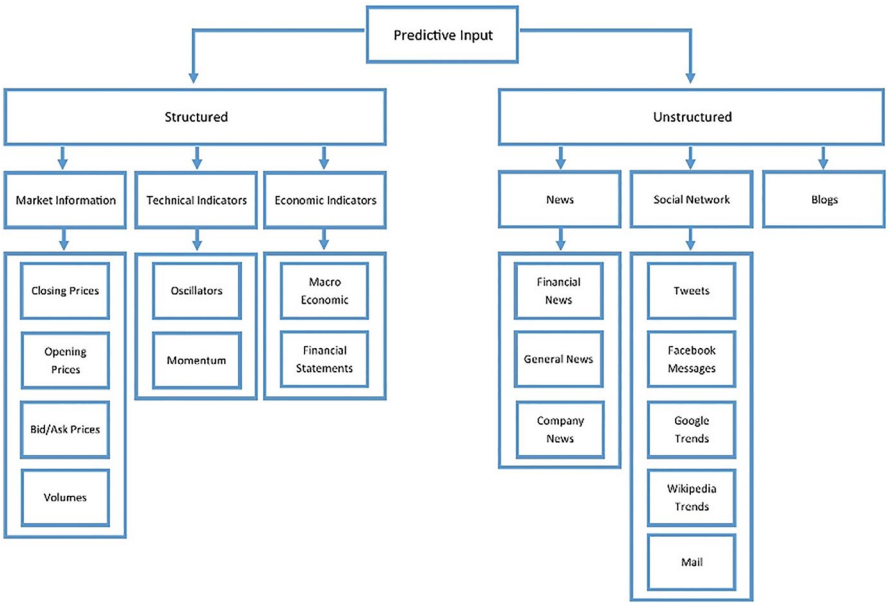


Fig. 1 Classification of existing studies according to input types

time, the authors emphasize that the dataset used is balanced (with a structure that includes both bullish (bull market) and bearish (bear market) movements) (Thakkar & Chaudhari, 2021). The data set used is of a structured input-type, consisting of market informations and technical indicators (Bustos & Pomares-Quimbaya, 2020). Our purpose in choosing a dataset with this input type is that the technical analysis approach is widely preferred in the literature (Atsalakis & Valavanis, 2009; Berradi & Lazaar, 2019; Gao & Chai, 2018; Gao et al., 2021; Kakade et al., 2023; Kwon & Moon, 2007; Li & Bastos, 2020; Sethia & Raut, 2019; Teixeira & De Oliveira, 2010; Thakkar & Chaudhari, 2021; Wei & Ouyang, 2024; Wen et al., 2020; Zheng & He, 2021). For this reason, machine learning techniques and deep learning models are very much recommended in the literature to analyze such data. In addition, in recent years, hybrid models have been proposed by combining both machine learning algorithms and deep learning models with different approaches instead of single models that cannot produce good results for every situation to improve the prediction performance of machine learning models (Nosratabadi et al., 2020; Ozbayoglu et al., 2020; Thakkar & Chaudhari, 2021).

Some of our motivations for preparing our research were as follows: (1) During our literature review, we observed that most deep learning-based asset price prediction models use different datasets and performance evaluation metrics. Although this situation limits the effectiveness or measurability of new technology prediction models, it can be considered a gap in the literature (Atsalakis & Valavanis, 2009; Bustos & Pomares-Quimbaya, 2020; Thakkar & Chaudhari, 2021). (2) Additionally, we noticed that most researchers in the existing literature focus on accuracy and low error rates in stock price prediction. However, the sole purpose of financial market participants is to reduce potential risks and achieve high returns in market volatility (Deng et al., 2023, 2024; Sethia & Raut, 2019; Zhang et al., 2020). Our research not only focuses on prediction accuracy but also calculates the return rate by establishing a simple trading strategy. Thus, the performance of the proposed method is presented to researchers as a simple decision support system. (3) Lastly, it is well-known that prediction models encounter challenges of dimensionality and overfitting due to large datasets and noisy data. In the literature, research aiming to address these structural issues and enhance the performance of prediction models is increasingly prevalent (Berradi & Lazaar, 2019; Chen et al., 2022; Guo et al., 2022; He & Dai, 2022; Huang et al., 2022; Jianwei et al., 2019; Kakade et al., 2023; Li et al., 2022; Ma et al., 2019; Sethia & Raut, 2019; Srijiranon et al., 2022; Wang et al., 2023; Wei & Ouyang, 2024; Zheng & He, 2021). To tackle these problems and improve the performance of prediction models, we aim to leverage the advantages of hybrid models to combine different methods and models. For all these reasons, we chose a dataset and model used in the literature in our study in addition to making efforts to keep the evaluation criteria of our study as broad as possible. Thus, we aimed to make our study comparable with other studies (Sethia & Raut, 2019; Thakkar & Chaudhari, 2021).

The novelty and contributions of the research to the literature are as follows: (1) We propose a hybrid PCA-ICA-LSTM model for predicting asset prices. While studies exist in the literature that combine PCA or ICA with different methods, to our knowledge, ours is the first study to combine these two statistical methods for a

two-stage preprocessing and integrate them with a recurrent deep learning network to create a hybrid model. The proposed model introduces a new framework that combines PCA and ICA statistical methods to provide input to an LSTM deep learning network used for prediction. We combine PCA for dimensionality reduction and noise removal, and ICA for feature extraction from processed data, leveraging the advantages of both methods to significantly enhance prediction performance. We support this claim with various experiments. (2) Many studies in the literature use different datasets, time scales, and evaluation metrics, making it challenging to compare studies fairly. Therefore, we divide our experiments into two stages initially. In the first stage, we use a well-established dataset and time scale from the literature (Sethia & Raut, 2019; Thakkar & Chaudhari, 2021). Additionally, we prefer commonly used evaluation metrics for a fair comparison. Thus, our goal is to achieve a directly comparable study in the literature concerning dataset, time scale, and evaluation metrics. (3) The second phase of experiments expands the utilized dataset to include the turbulent period experienced in financial markets during the COVID-19 pandemic, subsequently re-evaluating the model's effectiveness. Furthermore, two additional case studies are incorporated into our work to establish a new benchmark for validating the model's efficacy. While conducting these experiments, we emphasize that the primary objective of predicting a financial asset is to achieve high returns and mitigate risks. Therefore, we go beyond focusing solely on high prediction accuracy and low error rates by incorporating the return rate metric into our study, aiming to highlight this gap in the existing literature. (4) Our experiments yielded promising results when compared to existing models that do not utilize dimensionality reduction for predicting asset prices. Our findings suggest that our model has the potential to offer researchers higher accuracy and lower error rates when working with high-dimensional datasets, while also achieving competitive return rates when compared to state-of-the-art approaches.

The outline of this article is organized as follows: Sect. 2 reviews studies on various methods that use technical indicators and dimensionality reduction techniques to analyze financial data. Section 3 presents the Study Methodology. This section presents information about the operation of the study, baseline settings, the dataset used, the cross-validation method, machine learning models used in the study, statistical methods, and the latest proposed hybrid PCA-ICA-LSTM model. The experimental results of the proposed PCA-ICA-LSTM approach in Sect. 4 are analyzed in four parts. First, the proposed model is compared with models from the same family using single-stage statistical methods. The second part presents comparisons with state-of-the-art models in the literature. The third part compares the results of the proposed model with similar studies in the literature. Finally, in the last part, we repeat our experiments by expanding our dataset from 2000–2017 to 2000–2024 to include the COVID-19 pandemic. We also include two additional case studies in our research as a benchmark. The conclusions are summarized in Sect. 5.

2 Research Review

Studies on financial markets have been analyzed according to various classification methods. We believe that the two most important of these classifications are the classification according to the type of dataset input and the classification according to the type of forecasting model. We form our research analysis on this basis and try to justify our choices based on these two classifications.

2.1 Dataset Input Types

In their study, Bustos et al. grouped the input types under two main headings: structured and unstructured data (Bustos & Pomares-Quimbaya, 2020). The authors classified structured inputs as (1) market information, (2) technical indicators, and (3) economic indicators. Unstructured inputs are categorized as (1) news, (2) social networks, and (3) blogs. The classification of studies according to input types is shown in Fig. 1 (Bustos & Pomares-Quimbaya, 2020).

In the literature, studies on structured inputs are in the majority, and two approaches to structured inputs come to the fore (Bustos & Pomares-Quimbaya, 2020; Cavalcante et al., 2016). These approaches are called technical analysis and fundamental analysis. Technical analysis information is the approach that uses stock prices and indicators derived from this price information (Atsalakis & Valavanis, 2009). Researchers who adopt this approach argue that the effect of fundamental analysis indicators and news already exists in the price of financial assets (Bustos & Pomares-Quimbaya, 2020). According to this approach, it is sufficient to analyze price movements when forecasting asset prices. On the other hand, the fundamental analysis approach uses macroeconomic and financial situation information and consists of time series information that tries to understand the reasons for price movements (Bustos & Pomares-Quimbaya, 2020; Cavalcante et al., 2016). Fundamental analysis information is challenging to obtain and may require expertise to interpret. For this reason, it is not as widely used as the technical analysis approach**.** In addition to these approaches, studies utilizing data from social media as input and performing price prediction using sentiment analysis have also emerged in recent years(Deng et al., 2023, 2024; Srijiranon et al., 2022).

2.2 Prediction Model Types

In the literature, other than the estimation input, another approach to classifying time series forecasting models is based on the forecasting model itself (Atsalakis & Valavanis, 2009; Bustos & Pomares-Quimbaya, 2020; Cavalcante et al., 2016; Liu et al., 2021; Nosratabadi et al., 2020). According to the forecasting model used, two forecasting models come to the fore: traditional forecasting models and machine learning forecasting models.

Traditional forecasting methods are based on mathematical and statistical foundations. It is possible to examine traditional forecasting models in linear and

nonlinear classes. Famous traditional linear forecasting models are named as autoregressive (AR) model, moving average (MA) model, autoregressive moving average (ARMA) model, and autoregressive integrated moving average (ARIMA) models (Liu et al., 2021). Besides, well-known traditional nonlinear forecasting models are the Threshold Autoregressive (TAR) model, the Autoregressive Conditional Heteroskedasticity (ARCH) model, and the Constant Conditional Correlation (CCC) model, which can be listed (Cavalcante et al., 2016; Liu et al., 2021). Traditional forecasting methods assume the studied time series is produced after a linear process. They usually try to model the underlying process according to this assumption. However, financial time series are complex, noisy, and uncertain. Therefore, it exhibits non-linear behavior and makes it difficult to predict such data. This private nature of financial time series causes traditional statistical methods not to be applied effectively in the financial context (Cavalcante et al., 2016). For all these reasons, traditional forecasting models are not as reliable as necessary to predict the price of a financial asset (Nosratabadi et al., 2020).

Another predictive model classified according to the model used is the machine learning prediction model. Machine learning models provide the ability to learn from data and provide in-depth insight into problems (Nosratabadi et al., 2020). Some authors state that traditional forecasting models are not efficient because they treat financial time series as linear systems, and they get lower results than models based on artificial intelligence (Atsalakis & Valavanis, 2009; Cavalcante et al., 2016; Li & Bastos, 2020). The unpredictable dynamic nature of financial markets and the advantages as mentioned above of machine learning models have encouraged many researchers to work in this direction (Berradi & Lazaar, 2019; Chowdhury et al., 2018; Gao et al., 2021; Gudelek et al., 2017; Huang et al., 2019; Jianwei et al., 2019; Kao et al., 2013; Kauffman et al., 2015; Kim, 2003; Kwon & Moon, 2007; Long et al., 2019; Pai & Lin, 2005; Sarıkoç & Çelik, 2022; Sethia & Raut, 2019; Tay & Cao, 2001; Teixeira & De Oliveira, 2010; Thakkar & Chaudhari, 2021; Wen et al., 2020; Zhang, 2003). The deep learning models, a sub-branch of machine learning methods in recent years, have attracted attention with their successful results (LeCun et al., 2015; Schmidhuber, 2015). The advantage of deep learning models compared to other machine learning models is that deep learning models can effectively identify highly qualified features and outputs from a wide range of inputs (Nosratabadi et al., 2020). For this reason, studies focusing on deep learning models by keeping them separate from machine learning models are frequently seen in the literature (Bustos & Pomares-Quimbaya, 2020; Cavalcante et al., 2016; Li & Bastos, 2020; Nosratabadi et al., 2020; Ozbayoglu et al., 2020; Sezer et al., 2020; Thakkar & Chaudhari, 2021).

2.3 Related Work

Researchers on financial markets generally adopted a technical analysis approach and used market or technical analysis datasets(Berradi & Lazaar, 2019; Gao & Chai, 2018; Gao et al., 2021; Kakade et al., 2023; Kwon & Moon, 2007; Sethia & Raut, 2019; Thakkar & Chaudhari, 2021; Wei & Ouyang, 2024; Wen et al., 2020;

Zheng & He, 2021). Due to its widespread use, ease of calculation, and ability to show changes in price movements, we adopted the technical analysis approach in our study. In addition, as a prediction model, we focus on deep learning models, which have attracted attention with their successful results in recent years and have become popular compared to other machine learning methods. Accordingly, some of the studies using technical analysis information and machine learning models on the dataset are summarized in Table 1.

Kwon and Moon attempted to optimize an iterative neural network-based prediction model on a dataset consisting of a set of technical indicators by using a genetic algorithm. This study is one of the first to use technical indicators to forecast financial time series (Kwon & Moon, 2007). Lu et al. used the SVR model to forecast financial time series (Lu et al., 2009). The authors developed a forecasting model that uses the ICA method to remove noise from the data, which they call ICA-SVR. The Nikkei 225 Index and TAIEX Index data are used to evaluate the performance of the proposed model. The ICA-SVR model outperformed the SVR-only forecasting model and the random walk model. In another study, Liu and Wang built a prediction model by integrating dimension reduction methods into a back-propagation neural network (Liu & Wang, 2011). PCA and ICA methods are preferred as dimensionality reduction methods. The prediction models are trained and assessed using Shanghai Composite (SHCI) data for two datasets. It has been reported that the ICA-BPNN model proposed by the authors outperforms the PCA-BPNN and BPNN models. Kao et al. suggested connecting nonlinear independent component analysis to support vector regression (SVR) to examine the effect of feature extraction methods in predicting stock prices and obtained successful results in their experimental studies (Kao et al., 2013). Gao et al. used a dataset with a very similar scale to the dataset in our study (Gao et al., 2017). Accordingly, they tried to predict the next day's movement of the S&P 500 index using the LSTM deep learning network. Their model showed that compared to other systems (i.e., moving average (MA), exponential moving average (EMA), and support vector machine (SVM)), the proposed model yields a higher prediction accuracy for the next day's closing price of the stock. In another study, Chowdhury et al. proposed a new model that combines PCA and ICA, which are dimension reduction and feature extraction mechanisms, for stock price prediction with SVR and successfully applied (Chowdhury et al., 2018). Gao et al. conducted case studies on Standard & Poor's 500, NASDAQ, and Apple (AAPL), utilizing dimensionality reduction techniques and the LSTM model for stock price prediction. The authors recommend principal component analysis (PCA) as the dimensionality reduction method for removing unnecessary information in the technical indicators used in the dataset and extracting highly correlated features (Gao & Chai, 2018). Long et al. stated in their studies that deep learning methods may be more suitable for asset price prediction models since statistical methods depend on initial assumptions and machine learning techniques have performance and overfitting problems due to manual feature selection (Long et al., 2019). Berradi and Lazaar used a recurrent neural network (RNN) deep learning model for stock price prediction. In this study, they proposed the PCA-RNN model, which applies the PCA technique to reduce the dimension of the dataset consisting of 90-day historical data and technical indicators of a stock, and

Table 1 Summary of related works that include technical analysis information on the dataset and use machine learning models

Paper	Model	Dataset	Dataset time scale	Metric	Result
(Kwon & Moon, 2007)	GA-NN, GP, SVM	36 companies' stocks on NYSE and NASDAQ	1992 to 2004	Accuracy, Average Accuracy Improvement, Performance Consistency, P-Value	The study converts stock prices and volumes into numerous technical indicators. GA optimizes the weights of NN. The trading strategy created with the model significantly outperformed the "buy and hold" strategy
(Lu et al., 2009)	SVR, ICA-SVR, Random Walk	Nikkei 225 opening index and TAIEX closing index	October 4, 1999, to September 30, 2004, and January 2, 2003 to February 27, 2006	RMSE, Normalized Mean Square Error (NMSE), Mean Absolute Deviation (MAD), Directional Symmetry (DS), Correct up Trend (CP) and Correct down Trend (CD)	The ICA-SVR model achieves lower prediction error and higher accuracy than other models
(Liu & Wang, 2011)	BPNN, PCA-BPNN, ICA-BPNN	Shanghai Composite Index (SHCI)	Set 1: April 11, 2008, to November 30, 2009 Set 2: January 4, 2000, to November 30, 2009	MAE, RMSE, Correlation Coefficient (R)	The ICA-BPNN model outperforms the other two models for relatively small (Set 1) and large (Set 2) datasets
(Kao et al., 2013)	SVR, PCA-SCR, NLICA-SVR, LICA-SVR	Shanghai Stock Exchange Composite (SSEC) and Nikkei 225 Stock Indexes	November 6, 2007, to November 30, 2011, and October 26, 2007, to November 30, 2011	RMSE, MAD, MAPE, Root Mean Square Percentage Error (RMSPE), Directional Symmetry (DS)	NLICA-SVR outperforms PCA-SVR, LICA-SVR, and single-SVR methods

Table 1 (continued)

Paper	Model	Dataset	Dataset time scale	Metric	Result
(Gudelek et al., 2017)	CNN	17 Different Exchange-Traded Funds (ETF)(on the New York Stock Exchange)	April 6, 2000 to November 17, 2016	Accuracy, MSE, Profit	The model predicts the next day's prices with 72% accuracy
(Gao et al., 2017)	Moving Average (MA), Exponential Moving Average (EMA), SVM, LSTM	S&P500	January 3, 2000 to November 10, 2016	MAPE, MAE, RMSE, Average Mean Absolute Percentage Error (AMAPE)	LSTM model improved MAE, RMSE, MAPE, AMAPE according to MA, EMA, SVM
(Chowdhury et al., 2018)	SVR, PCA-SVR, ICA-SVR, PCA-ICA-SVR	Stock Prices of Of Square Pharmaceuticals Limited, AB Bank Limited, Bangladesh Lamps Limited (on the Dhaka Stock Exchange, Bangladesh)	January 2000 to December 2015	MAPE, MAE, rRMSE, MSE	The PCA-ICA-SVR model performs better with less prediction error than the other three
(Hossain et al., 2018)	LSTM, GRU, GRU-LSTM, LSTM-GRU	S&P500	1950 to 2016	MAE, MSE, MAPE	The LSTM-GRU model achieved 0,00098 MSE and outperformed all other neural network approaches by a very high margin
(Gao & Chai, 2018)	Moving Average (MA), Estimated Moving Average (EMA), ARMA, GARCH, SVM, FFNN, and LSTM	Standard & Poor's 500, NASDAQ, and Apple (AAPL)	2000-01-03 ~ 2016-11-10, 200-01-02 ~ 2016-08-15, 2000-01-03 ~ 2016-10-10	Percentage of Correct Trend (PCT), MAPE, AMAPE, MAE, RMSE	The authors stated that the utilization of fundamental stock trading data and technical indicator data with PCA-LSTM for stock price prediction represents a viable alternative to other techniques

Table 1 (continued)

Paper	Model	Dataset	Dataset time scale	Metric	Result
(Ma et al., 2019)	LSTM, PCA + LSTM and RF+LSTM	The Shanghai Composite Index (SCI)	January 4, 2013 to November 30, 2017	Precision, Recall, F1-Score and Accuracy	The authors report that the random forest feature extraction method is more effective than the classical PCA algorithm and that the RF+LSTM model achieves better results in stock prediction
(Long et al., 2019)	CNN, RNN, LTSM, MFNN, SVM, Random Forest(RF), Linear Regression, Logistic Regression	Chinese Stock Market İndex (CSI 300)	from December 9th, 2013 to December 7th, 2016	Accuracy, Profitability, Stability	MFNN outperforms other models in terms of accuracy, profitability, and stability
(Jianwei et al., 2019)	ARIMA, RBFNN, LSTM, GRUNN, ICA-LSTM, ICA-GRU	Gold Price	from January 1979 to December 2017	MAD, RMSE, MAPE, Adjusted R^2	ICA-GRU delivers high accuracy and outperforms all other models
(Berradi & Lazaar, 2019)	RNN, PCA-RNN	Stock Price Of Total Maroc (From Casablanca Stock Exchange)	08 February 2018 to 17 May 2018	MSE	PCA reduces the error obtained by the model from 0,011835 to 0,00596
(Sethia & Raut, 2019)	ICA-LSTM, ICA-GRU, ICA-SVM, ICA-MLP	S&P500	3rd January 2000 to 30th October 2017	MSE, R^2-Score, Return Ratio, Optimism Ratio, Pessimism Ratio	The ICA-LSTM model outperformed the other models with a return of 400% greater and an R^2 score of 0,9486

Table 1 (continued)

Paper	Model	Dataset	Dataset time scale	Metric	Result
(Wen et al., 2020)	PCA-LSTM, CNN, MLP, Moving Average (MA)	Stock Price Of Pingan Bank (on Shenzhen Stock Exchange)	January 4, 2016 to December 28, 2018	RMSE, MAPE	The PCA-LSTM model performs better than other models. It obtains 0.221 and 1.667% values for RMSE and MAPE, respectively
(Zhang et al., 2020)	CEEMD-PCA-LSTM, EEMD-PCA-LSTM, EMD-PCA-LSTM, LSTM, RNN	Shanghai Composite Index (000001.SH), SZSE Component Index (399,001.SZ), GEM (399,006.SZ), Hang Seng Index (HSI.HI), Dow Jones Industrial Average Index (DJI.GI) and S&P 500 Index (SPX.GI)	January 6, 2010 to April 27, 2018	RMSE, MAE, NMSE and DS	A novel financial time series prediction model based on deep learning has been proposed. The proposed model achieved the highest prediction accuracy and directional symmetry in comparative evaluations
(Zheng & He, 2021)	RNN, LSTM, FFN + Delay, FNN	Two kinds of companies in in the aerospace sector: the aerospace manufacturers (MFG Company) and the aerospace operators (OPR Company)	July 1, 2013 to June 29, 2018	MAE, MSE	PCA not only indicates a potential enhancement in prediction performance in terms of computational efficiency but also in prediction accuracy. Additionally, the results suggest that model parameters are associated with the stability of the stock

Table 1 (continued)

Paper	Model	Dataset	Dataset time scale	Metric	Result
(Thakkar & Chaudhari, 2021)	CNN, DQN, RNN, LSTM, GRU, ESN, DNN, RBM, DBN	S&P500	3rd January 2000 till 30th October 2017	Directional Accuracy (DA)	Regardless of the number of features, the highest average DA is achieved by the DQN model
(Gao et al., 2021)	LASSO-LSTM, PCA-LSTM, LASSO-GRU, PCA-GRU	Shanghai Composite Index (SSE:000001)	April 11, 2007, to August 3, 2021	MSE, RMSE, MAE	LSTM and GRU models are not superior to each other. LASSO models showed better prediction performance for the dimensionality reduction method than PCA models
(Srijiranon et al., 2022)	ARIMA, LSTM, EMD-LSTM, PCA-EMD-LSTM, PCA-EEMD-LSTM and PCA-CEEMDAN-LSTM	Part-1: Thai Financial News Data, Part-2: SET50 (The Stock Exchange of Thailand)	24 February 2018 to 24 February 2022	Part-1: Precision, Recall, F1-Score and Accuracy Part-2: RMSE, MAPE, MAE and coefficient of determination (R^2)	News sentiment analysis may enhance the performance of the original LSTM model and the proposed PCA-EMD-LSTM model demonstrates superior performance compared to traditional methods in predicting stock prices
(Chen & Hu, 2022)	AR, EGARCH, ANN, ANN(PCA), ANN(AE), LSTM, LSTM(PCA), and LSTM(AE)	CSI300 Index Futures, S&P500 Index Futures	January 1, 2011 to December 31, 2018	MSE, MAE, NMSE	The effectiveness of PCA is better than AE in predicting stock index futures fluctuations for both 5-day and 10-day forecast horizons, with the LSTM (PCA) model yielding the most successful results

Table 1 (continued)

Paper	Model	Dataset	Dataset time scale	Metric	Result
(He & Dai, 2022)	CEEMD-Multi-LSTM, EEMD-Multi-LSTM, EMD-Multi-LSTM, WT-Multi-LSTM, Multi-LSTM, Single-LSTM, Prophet, 1D-CNN and ICA-Multi-LSTM	5 Stocks Selected from the CSI300 Exchange (Stock Code: 000166, 002736, 601,198, 600,958, 601,211)	July 26, 2015 to December 31, 2020	RMSE, MSE, MAE	The experiments confirm that the proposed ICA-Multi-LSTM model outperforms ICA-less LSTM models in most cases, particularly from the perspective of individual stocks
(Sarıkoç & Çelik, 2022)	FA-LSTM, PCA-LSTM, ICA-LSTM	Borsa Istanbul 100 (BIST100)	September 20, 2002, to July 24, 2020	R^2, RMSE	The PCA-LSTM model outperformed other models and improved the results of the non-hybrid LSTM model
(Chen et al., 2022)	ARIMA, BP, SVM, LSTM, CNN, PSO-LSTM, CS-PSO-LSTM, and CS-ICA-PSO-LSTM	4 Stocks Selected from the Shanghai Stock Exchange (SSE) (Stock Code: SH600518, SH600519, SH600999, SH601988)	March 19, 2001 to March 16, 2021	MAE, MAPE, RMSE, DA, R^2	A new model, involving the integration of various methods including CEEMD, SE, ICA, PSO, and LSTM to predict stock prices, is proposed. The proposed model achieves successful results compared to seven different models that do not include ICA

Table 1 (continued)

Paper	Model	Dataset	Dataset time scale	Metric	Result
(Li et al., 2022)	SVM, XGBoost, LSTM, TreNet, PCA-LSTM	Standard Chartered Bank Dataset (in Stock Exchange of Hong Kong)	Jul 2018 to Jan 2021	Accuracy, Precision, Recall, Macro-F1, Kappa, Ham_distance	The experimental results indicate that the PCA-LSTM model, which utilizes PCA for noise reduction in the dataset and LSTM for feature extraction, outperforms the original LSTM model in the prediction task
(Huang et al., 2022)	PCA, sPCA, Target PCA (tPCA), PLS, LASSO, and Ridge regression	123 Macro Variables from the FRED-MD Database	January 1960 to December 2019	The Asymptotic Mean Square Forecast Error (MSFE), R^2 and R^2_{oos}	The scaled-PCA (sPCA), which assigns more weight to components with stronger predictive power considering the target information, is recommended and outperforms PCA
(Guo et al., 2022)	AR, AR-KS, AR-sPCA, AR-PCA, AR-PLS, Mean Combination (MC), Median Combination (MDC), Trimmed Mean Combination (TMC), and Discount Mean Square Prediction Error combinations (DMSPE)	The West Texas Intermediate (WTI) Price	January 1985 to April 2021	The out-of-sample R^2 (R^2_{oos}), MSE, MAE	The scaled-PCA method is introduced for predicting oil volatility, additionally integrated with AR for various variants and compared with PCA and PLS methods. It exhibits robust performance in resilience tests such as different window and lag selections

Table 1 (continued)

Paper	Model	Dataset	Dataset time scale	Metric	Result
(Li et al., 2023)	The variants of MLSTM, MBiLSTM, MGRU, MBiGRU models integrated with PCA or Sparse PCA(SPCA)	Shanghai Stock Exchange (SSE) Energy Sector Index and Shenzhen Stock Exchange (SZSE) Energy Sector Index	January 2014 to November 2022	MSE, MAPE, MAE and R^2	The performance of 12 models was tested to predict volatility in energy sector indices, with the SPCA-MLSTM model yielding the best predictions. Additionally, LSTM models generally outperform GRU models in prediction performance
(Mendoza et al., 2023)	GARCH, SVR, XGBR, MLP, RNN, LSTM	S&P 500, DAX (German stock index), AEX(Amsterdam Stock Exchange) and the SMI (Swiss Market Index) indexes	January 2000 to December 2020 (Daily)	MSE, Model Confidence Set (MCS)	The authors stated that the proposed Self-Similarity approach improved the performance in the predictive capacity of Deep Neural Network models, providing significant improvements of up to 23% for the S&P 500, 11.26% for the DAX, 21% for the AEX index, and 12% for the SMI index

Table 1 (continued)

Paper	Model	Dataset	Dataset time scale	Metric	Result
(Wang et al., 2023)	SVR, LSTM, GRU, IGRU, PCA-LSTM, PCA-GRU, CNN-LSTM	The Shanghai Composite Index (SCI)	January 1, 1992 to March 31, 2022	MAE, Directional Symmetry (DS), R^2	The proposed PCA-IGRU model, which utilizes PCA to reduce the dimensionality of input data, demonstrates better prediction accuracy and shorter training time compared to the 7-various models in the comparison
(Kakade et al., 2023)	LSTM, LSTM + PCA, and LSTM + PCA + ARIMA	Crude Oil Price	July 28, 2000 to May 13, 2019	RMSE, MAE, and MAPE	Including explanatory fundamental and technical variables as inputs helps improve prediction ability. The proposed LSTM + PCA + ARIMA model outperforms the LSTM model across all dimensions with an average improvement of 41% in prediction accuracy

Table 1 (continued)

Paper	Model	Dataset	Dataset time scale	Metric	Result
(Wei & Ouyang, 2024)	PCA, PLS, sPCA, Linear Regression, LSTM	The Hubei Emissions Exchange	April 28, 2014 to March 23, 2022	R^2, RMSE, MAE, and DM	The scaled PCA (s-PCA) method has been utilized on a multidimensional dataset to enhance the accuracy of carbon price predictions. The proposed model can yield higher average returns compared to other comparative strategies in terms of market timing

emphasized that the prediction accuracy of the proposed model achieves better results than does the RNN model (Berradi & Lazaar, 2019). Jianwei et al. proposed a new model (ICA-GRU) combining the ICA approach and gated recurrent unit (GRU) deep learning network to predict gold prices. It has been reported that the proposed model outperforms the integrated autoregressive moving average (ARIMA), radial basis function neural network (RBFNN), LSTM, and ICA-LSTM models (Jianwei et al., 2019). Sethia and Raut (2019) proposed a model that predicts prices after five days by establishing a simple trading strategy on a dataset consisting of 18 years of historical data and technical indicators of the S&P 500 index. Within the scope of the study, deep learning models such as LSTM and GRU were compared with models such as SVM and artificial neural networks (ANNs) using the ICA technique in data preprocessing steps, and it was emphasized that the performance of the LSTM deep learning price prediction model was superior to that of other models (Sethia & Raut, 2019). Ma et al. proposed a model based on an LSTM deep learning network and preprocessing with PCA to forecast the closing price of the Shanghai Composite Index. The experimental results highlight the success of the PCA method in removing noise from the data and improving the prediction accuracy (Ma et al., 2019). In their experimental study to develop a price prediction model, Wen et al. used the PCA approach with the LSTM deep learning network to reduce dependencies and reduce the data dimension in a dataset consisting of two years of financial data and several technical indicators of a stock (Wen et al., 2020). The PCA-LSTM model can predict asset prices more successfully than traditional price-prediction models. By adopting the concept of decomposition-reconstruction-synthesis to forecast complex financial time series, Zhang and colleagues proposed a new forecasting model based on deep learning called CEEMD-PCA-LSTM. Initially, the model decomposes the time series into intrinsic mode functions (IMFs) using the complementary ensemble empirical mode decomposition (CEEMD) method to identify trends. Then, principal component analysis (PCA) is applied for dimensionality reduction to extract high-level features from the data. The new features feed the long short-term memory (LSTM) network to predict the closing price of the next trading day. The authors emphasize the high prediction accuracy and directional symmetry of the proposed model while also performing trading simulations to evaluate its profitability performance (Zhang et al., 2020). In their study focusing on the stock prices of two aviation companies, Zheng et al. proposed the PCA + RNN model. This study aims to demonstrate the impact of technical indicators and the PCA method on the prediction performance of RNNs in forecasting stock prices (Zheng & He, 2021). In another study, Gao et al. studied a dataset consisting of financial data, technical indicators, and investor sentiment indicators to improve stock forecasting. Accordingly, researchers have used approaches such as least absolute shrinkage and selection operator (LASSO) and principal component analysis (PCA) in dimension reduction processes and examined the effects of these approaches on the performance of long short-term memory (LSTM) and gated recurrent units (GRUs) deep learning prediction models (Gao et al., 2021). Building on the study by Sethia and Raut (2019), Thakkar and Chaudhari (2021) conducted a comparative analysis of deep neural networks for stock price trend prediction. They compared deep learning models by applying the

same parameters and dataset across the models examined in their study. (Thakkar & Chaudhari, 2021). Huang et al. proposed a novel scaled PCA (sPCA) method that assigns higher weights to components with strong predictive power. The authors aim to address the weak aspect of the PCA method that does not consider target information. Experiments conducted on 123 macroeconomic variables from the FRED-MD database indicate that the sPCA method outperforms PCA (Huang et al., 2022). Guo et al. introduced the scaled PCA method for forecasting oil volatility in their study. This study compares the introduced s-PCA method with two other dimensionality reduction methods, PCA and PLS. Additionally, a series of experiments were conducted for hybrid models by integrating these methods with AR models for various variants. The proposed AR-sPCA model demonstrated robust performance in robustness tests, such as different window and lag selections (Guo et al., 2022). In another study demonstrating the effectiveness of hybrid models, Srijiranon et al. proposed the PCA-EMD-LSTM model to forecast the closing price of the Thai stock market. The model performs feature engineering using PCA and empirical mode decomposition (EMD) methods while utilizing LSTM for prediction tasks. The authors emphasized that the application of PCA to the EMD-LSTM model reduces prediction errors. Additionally, the model incorporates sentiment analysis of economic news to enhance performance based on news sensitivity (Srijiranon et al., 2022). In their study, Chen and Hu conducted a series of experiments on various models based on ANN and LSTM for predicting volatility in stock index futures trading in China and the United States utilizing feature extraction methods such as AE and PCA. The findings suggest that PCA outperforms AE in terms of prediction efficacy. In comparison, the LSTM (PCA) model achieved the most successful results (Chen & Hu, 2022). He and Dai conducted experiments on models combining ICA and LSTM to predict the prices of 5 stocks selected from the CSI 300 stock exchange. While the ICA method was used to eliminate noise, the predictive effect of the LSTM model was examined. The experiments confirm that the proposed ICA-Multi-LSTM model outperforms non-ICA-LSTM models in most cases, particularly from the perspective of individual stocks (He & Dai, 2022). Chen et al. proposed a new model that integrates various methods, such as CEEMD, sample entropy (SE), ICA, particle swarm optimization (PSO), and LSTM, to predict stock prices. The experiments utilize data from the Shanghai Stock Exchange (SSE) involving 4 selected stocks. The ICA technique is responsible for extracting the primary features of stock price data by separating the IMFs created with CEEMD. The proposed model achieves successful results compared to seven other models that do not include ICA (Chen et al., 2022). Li et al. proposed an optimized PCA-LSTM hybrid model for price prediction tasks. The experimental results emphasize the use of PCA to reduce noise in the dataset, improve sample quality, and eliminate input set correlations. The authors concluded that the PCA-LSTM model outperforms the original LSTM model in the prediction task (Li et al., 2022). Mendoza et al. aimed to enhance the prediction performance in time series of S&P 500, DAX, AEX, and SMI indices by leveraging fractal and self-similarity behaviors using simple recurrent neural networks, multilayer perceptron, and long short-term memory architectures. The authors noted that their proposed self-similarity approach improved the predictive capacity of deep neural network models, resulting in

significant improvements of 23%, 11.26%, 21%, and 12% for the S&P 500, DAX, AEX, and SMI indices, respectively (Mendoza et al., 2023). Li and colleagues tested the performance of 12 models based on LSTM and GRU to predict volatility in energy sector indices. The authors employed PCA and SPCA methods for feature extraction, with the SPCA-MLSTM model providing the best predictions in comparison. Additionally, while LSTM models generally outperform GRU models in terms of prediction performance, the SPCA method is superior to PCA in terms of prediction efficacy (Li et al., 2023). Wang et al. proposed a PCA-IGRU-based model for predicting the Shanghai Composite Index (SCI) closing price. The PCA method is utilized to reduce the high dimensionality of the data while minimizing information loss. To prevent overfitting, an anti-overfitting conversion module (ACM) is incorporated into the GRU, resulting in an enhanced gated recurrent unit (IGRU). The experimental results demonstrate that the PCA-IGRU model outperforms seven other models in terms of prediction accuracy and shorter training time (Wang et al., 2023). Kakade et al. suggested that including explanatory fundamental and technical variables as inputs to prediction models helps enhance their predictive ability. They proposed a hybrid LSTM + PCA + ARIMA model for forecasting crude oil prices. The PCA method is utilized to reduce the input dimensionality of the LSTM network, mitigating the impact of multicollinearity. The proposed LSTM + PCA + ARIMA model outperforms the LSTM model across all dimensions, with an average improvement of 41% in prediction accuracy (Kakade et al., 2023). Wei and Ouyang employed a scaled principal component analysis (s-PCA) method on a multidimensional dataset to improve carbon price prediction accuracy. The authors utilize factors such as technical indicators, financial indicators, and commodity indicators to characterize carbon prices. This study employed the s-PCA method to reduce the dimensionality of these factors and integrated it with the linear regression method and the LSTM model. The proposed model can achieve higher average returns than other benchmark strategies in terms of market timing (Wei & Ouyang, 2024).

As mentioned, ICA, PCA, and similar techniques are used for dimension reduction and feature extraction in data preprocessing stages with machine learning algorithms. Dimension reduction techniques simplify the dataset and can eliminate the dimensionality problem by selecting the most relevant attributes (Zhong & Enke, 2017). At the same time, it provides an additional contribution by increasing the prediction accuracy of machine learning prediction models with which it is used (Bustos & Pomares-Quimbaya, 2020; Singh & Srivastava, 2017). In this way, hybrid forecasting models with successful results have been developed (Cavalcante et al., 2016; Nosratabadi et al., 2020; Ozbayoglu et al., 2020; Thakkar & Chaudhari, 2021).

However, when the studies conducted to date are examined, many different approaches and different datasets are used for these models in financial time series forecasting. Model performance depends on the data characteristics used. Accordingly, the selection of inappropriate model parameters, feature sets, and training-test intervals reduces the comparability of the studies. For these reasons, our proposed PCA-ICA-LSTM model, which we use for financial time series forecasting, is prepared by considering the work of Thakkar and Chaudhari (2021),

who wanted to perform a comparative experimental study between deep neural networks (Thakkar & Chaudhari, 2021).

3 Methodology

The first goal of our work is to predict the price of a financial asset using a deep learning network. Accordingly, this section consists of six subsections: the basic settings of the study, the dataset, the method/models, the functioning of the prediction model, our proposed model, and the evaluation criteria. The first subsection provides information about basic settings and operations. The second subsection provides detailed information about data collection and datasets. The following subsection explains the statistical methods and deep learning models used. The fourth subsection provides information about the operation of the prediction model. The fifth subsection introduces the proposed model. The last subsection includes the evaluation metrics used to compare the performance of the proposed model with other models.

3.1 Baseline Settings

In the previous sections, we emphasized that we selected the LSTM network used in our study from the literature. Optimizing the selected LSTM model to improve it may be desirable in future studies, but we have not made any changes. Thus, we maintain our aim to directly compare our proposed hybrid deep learning model with previous similar studies. Thakkar and Chaudhari (2021), who has a similar goal to ours, used the same deep learning model and hyperparameters as Sethia and Raut in his work to make the comparison fair (Sethia & Raut, 2019; Thakkar & Chaudhari, 2021). We have applied the same basic settings as Thakkar and Chaudhari (2021) & Sethia and Raut (2019) in their work to our study. Table 2 provides a summary of the baseline settings of our study.

Accordingly, the prediction model architecture we have created in this study has a structure consisting of five consecutive layers. Dropout layers are added between the layers to prevent over-fitting. The model consists of two LSTM layers of 64 and 128 nodes in the first stage and provides long-term dependency handling. The subsequent two layers consist of dense layers consisting of 256 and 512 nodes. All the nodes in this layer are connected to the nodes of the previous layer and show a fully connected structure. In the last layer of the model, there is the output layer that will give the prediction value and consists of a single node. For the estimation model, the epoch of the training process was determined as 125, the heap size was 50, and the dropout value was 0.3. While the target attribute is determined as the adjusted closing price, the linear activation function, and Adam optimization algorithm are used. Work is carried out with the mean square error (MSE) loss function to update the parameters.

Table 2 The values of baseline setting used in this study (Thakkar & Chaudhari, 2021)

Parameters	Values
Prediction frequency	5 days
Dataset	S&P500
Data description	03.01.2000 ~ 30.10.2017
Cross-validation techniques	Holdout
Train: validate: test (number of records-date)	3053-(07.03.2000 ~ 01.05.2012): 525-(02.05.2012 ~ 11.06.2014): 847-(12.06.2014 ~ 23.10.2017)
Number of input attributes	48
Statistical methods used for feature extraction	PCA and ICA
Feature extraction number of components	12
Target attribute	Adjusted Closing Price
Normalization	[0, 1]
Deep neural network model	LSTM, RNN, GRU, CNN
Model architecture	64-128-256-512-1
Model weight initial values	Random
Number of training rounds (Epoch)	125
Batch size	50
Dilution (Dropout)	0.3
Activation function	Linear
Optimization algorithm	ADAM
Loss function	Mean squared error (MSE)

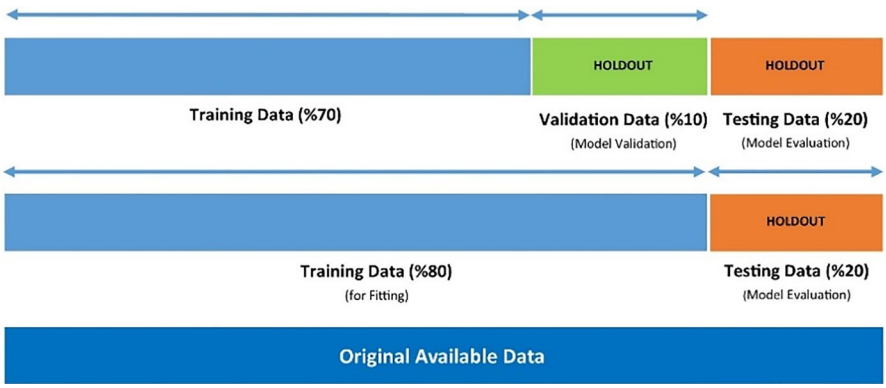


Fig. 2 Data splitting for training, validation, and test periods using the holdout method

In addition, Thakkar and Chaudhari & Sethia and Raut used the holdout method to parse the dataset in their study (Sethia & Raut, 2019; Thakkar & Chaudhari, 2021). The holdout method is the simplest type of cross-validation. In

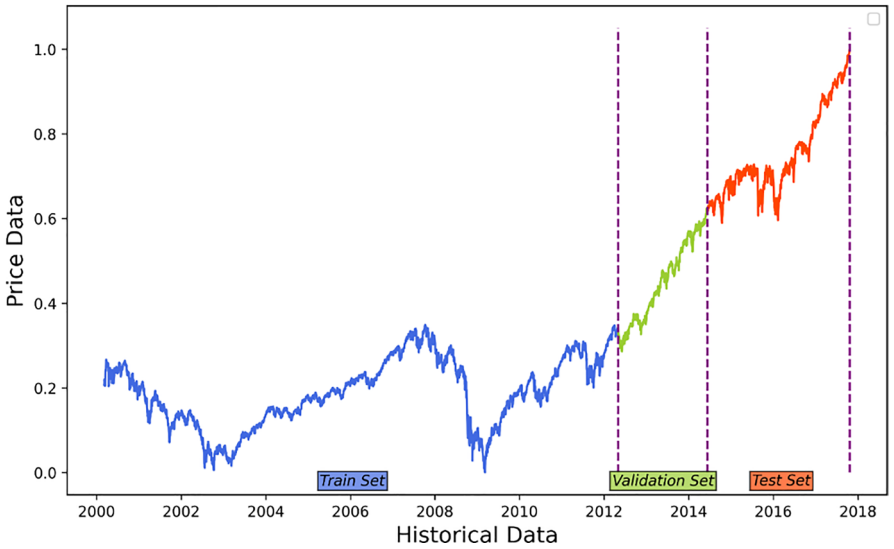


Fig. 3 Visualization of the S&P500 index data used in this study

its simplest form, the dataset is divided into two sets, called the training set (the validation set optional) and the test set. The main advantage of this method is that it takes a brief time to compute. However, the evaluation can have high variance. We apply the method precisely to our study. Figure 2 illustrates the partitioning of the dataset into periods using the holdout method within the scope of the study. Accordingly, the whole dataset consists of 4425 records: 3053 records for the training set (~69%), 525 records for the validation set (~11%), and 847 records for the test set (~20%). The visual for this is presented in Fig. 3.

In addition, there are statistical methods used in the pre-processing phase of the study. We include these methods in the pre-processing part because a new dataset is created after applying each statistical method. Therefore, after these statistical methods, the final dataset given as input to the prediction model will be quite different from the initial dataset at the beginning of the study. The most critical parameter we use in the basic settings of the statistical methods is the number of components. We apply this adjustment through PCA. The number of components for PCA determines the principal components found in a dataset. By using this parameter, we both reduce the size of our dataset and control the number of components of the ICA to be used afterward. Sethia and Raut (2019), who first used the dataset we used in our study, used the dimensionality reduction method for different numbers of attributes, such as 7, 12, 18, 25, 32, and 45 for the dataset. The authors reported reducing the number of features to 12 for the final dataset after the experiments (Sethia & Raut, 2019). Thakkar and Chaudhari followed the same path as Sethia and Raut (2019) in his study and performed similar tests (Thakkar & Chaudhari, 2021). Unlike Sethia and Raut (2019), the authors also conduct experiments for cases where the number of features is 5 and 48 without dimensionality reduction, but they compare the models based on a single metric.

The evaluation metric used by the authors is different from Sethia and Raut's (2019) study. However, Sethia and Raut (2019) conducted his experiments with increasingly common metrics. Therefore, we follow Sethia and Raut's (2019) study because one of the goals of our work was to have a comparable study. For this purpose, before the experiments, we checked whether the number of 12 components was sufficient to represent our dataset. Accordingly, when we reduced the number of components by 12 by applying the PCA method to our dataset with 48 features, we found that the total variance explained was 95.539%. In other words, we can represent 95% of the dataset with 12 principal components. Since this ratio is a sufficient total variance value, we use the value of the number of components parameter exactly. The effect of each component on the total variance is shown in Fig. 4.

3.2 Dataset

This study aims to develop a price forecasting model for financial time series forecasting. Therefore, we are conducting a study aiming to predict the price after 5 days in the Standard & Poor's 500 (S&P500) dataset, relying on the work of Thakkar and Chaudhari (Thakkar & Chaudhari, 2021). This study used the dataset that Sethia and Raut prepared (Sethia & Raut, 2019). Thakkar and Chaudhari state that the dataset used in their study is balanced for up and downtrends (Thakkar & Chaudhari, 2021). The dataset covers 18 years, including daily data between 01.01.2000 and 23.10.2017, and is obtained from the Yahoo Finance website (SPY, 2024) (Fig. 3).

The dataset obtained from Yahoo Finance consists of the opening, closing, highest, lowest, and adjusted closing prices for each trading day in the S&P500 index, as well as volume information (SPY, 2024). An example set of the Yahoo

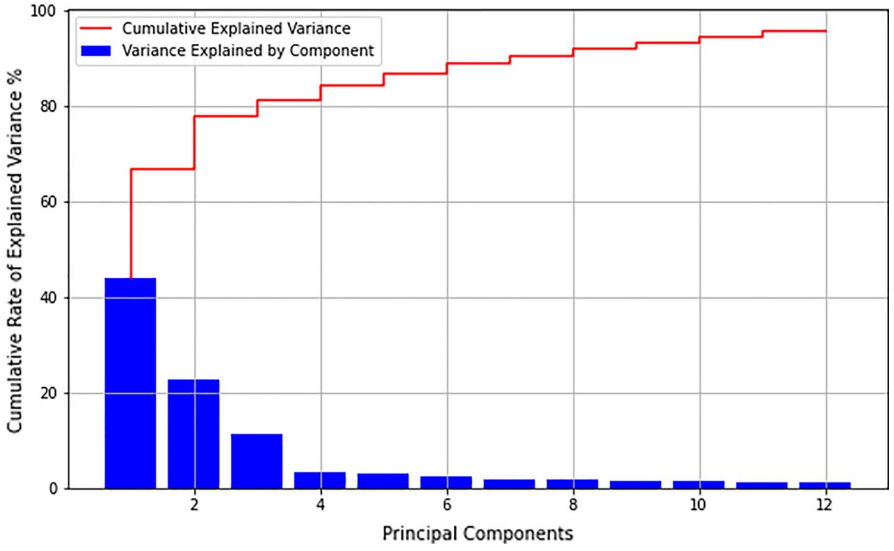


Fig. 4 Effect of components on total variance after dimension-reduction

Table 3 An example of data obtained from Yahoo Finance

Date	Open	High	Low	Close	Adj Close	Volume
2000-01-03	148.250000	148.250000	143.875000	145.437500	97.506668	8,164,300
2000-01-04	143.531250	144.062500	139.640625	139.750000	93.693573	8,089,800
2000-01-05	139.937500	141.531250	137.250000	140.000000	93.861176	12,177,900
2000-01-06	139.625000	141.500000	137.750000	137.750000	92.352676	6,227,200
2000-01-07	140.312500	145.750000	140.062500	145.750000	97.716209	8,066,500
2000-01-10	146.250000	146.906250	145.031250	146.250000	98.051422	5,741,700
2000-01-11	145.812500	146.093750	143.500000	144.500000	96.878143	7,503,700
2000-01-12	144.593750	144.593750	142.875000	143.062500	95.914406	6,907,700
2000-01-13	144.468750	145.750000	143.281250	145.000000	97.213379	5,158,300
2000-01-14	146.531250	147.468750	145.968750	146.968750	98.533318	7,437,300

Table 4 List of attributes of the dataset

Attributes	Number of attributes
Open, high, low, close, volume	5
Daily change	1
5 days momentum	1
14 day simple and exponential (k = 2/5) moving average	2
14 day Bollinger bands	2
14 day fast, slow, and smoothed slow stochastic indicators	3
Last 8 weeks and last 2 months returns	10
14 and 21 day moving average convergence divergence	1
Pivot point	1
14 day average true difference (ATR)	1
14 day relative strength index (RSI)	1
Balance volume (OBV)	1
3 day exchange rate (ROC)	1
7, 14, and 21 days uptrend and downtrend and their Fibonacci retracement levels 38.2%, 50%, and 61.8% (18 features)	18

Finance dataset is shown in Table 3. All of this information obtained on the Yahoo Finance web page regarding a financial asset is called market information. However, market information alone is often insufficient to determine the financial asset's future price trend. For this reason, researchers aim to increase forecast performance in many studies by adding technical indicators calculated using market information to datasets. Sethia and Raut conducted their studies on a dataset of 4425 records and 48 attributes, created using market information and technical indicators (Sethia & Raut, 2019). The list of attributes in the S&P500 dataset used in this study is presented in Table 4.

Each attribute in the dataset can be expressed with different ranges. This can result in a feature set of extremely high or low values. Sethia and Raut, in their study, first standardized the data attributes with Z-score standardization (Sethia & Raut, 2019). For this, Eq. (1) uses the formula (Furey, 2023). In the equation, $\bar{x}$ represents the population mean, E[X] is the population mean of a known sample, σ(X) is the population standard deviation of a known sample, and n is the sample size. Then, it is normalized with the Min–Max scaling method to scale each data feature in the range of [0,1]. In our study, we apply the same method for our dataset. The sample representation of our dataset, which is formed as the result of these processes, is shown in Table 5.

$$Z = \frac{\overline{X} - EX}{\sigma(\mathrm{X})/\sqrt{n}} \tag{1}$$

Because of the multidimensional nature of the dataset, the noise in it is minimized by using the dimension reduction method, and then efficient features are extracted. In the last step, the price is estimated using a deep learning model for forecasting.

3.3 Methods and Models

3.3.1 Principal Component Analysis (PCA) for Dimension Reduction and Noise Removal

Principal Component Analysis (PCA) is a statistical technique introduced by Karl Pearson and is frequently used in areas such as image compression, face recognition, and classification (Pearson, 1901). The purpose of the use is to eliminate the low-efficiency features that will occur due to high input sizes when working with large datasets in experimental studies, to increase interpretability by reducing the data dimension, and to minimize information loss (Gao & Chai, 2018; Huang et al., 2022; Kakade et al., 2023; Li et al., 2022; Ma et al., 2019; Srijiranon et al., 2022; Wang et al., 2023; Wei & Ouyang, 2024; Zheng & He, 2021). The basic idea of this technique is to reduce the dimension of the dataset while preserving the diversity in the dataset. For this reason, PCA forms a new variable set that provides the greatest possible diversity in the dataset and is equal to or less than the original number of variables called the principal components.

A financial time series is a multi-dimensional particular time series formed due to complex interactions of many factors. The dataset we will use is a multidimensional dataset that is formed by adding over 40 technical analysis information to the S&P500 Index information, which is the financial time series. In our study, the PCA method allows us to eliminate the existing noise and low-efficiency features by reducing the dimension of this dataset (Bustos & Pomares-Quimbaya, 2020; Li et al., 2022; Ma et al., 2019; Singh & Srivastava, 2017; Zheng & He, 2021; Zhong & Enke, 2017). However, it allows us to create a new dataset by minimizing the loss of information in the dataset used while doing this.

Table 5 An example representation of the dataset before dimension reduction in data preprocessing steps

Date	Open	High	Low	Adj Close	Volume	Daily_returns	Momentum	Sma	Ema	…	Adx
7.03.2000	0.380151	0.373390	0.359826	0.206338	0.021419	0.324673	0.632962	0.195936	0.212343	…	0.862784
8.03.2000	0.361519	0.361082	0.358836	0.205724	0.011927	0.398914	0.608093	0.196569	0.204165	…	0.862925
9.03.2000	0.365641	0.377215	0.364614	0.220013	0.004674	0.524006	0.690991	0.198002	0.206041	…	0.862024
10.03.2000	0.381140	0.383203	0.382607	0.217334	0.007461	0.382211	0.620030	0.200099	0.217544	…	0.858016
13.03.2000	0.362673	0.375053	0.362303	0.211864	0.010469	0.359211	0.616716	0.201849	0.214388	…	0.861415
14.03.2000	0.376359	0.373057	0.364779	0.204831	0.007851	0.345758	0.632300	0.202015	0.208637	…	0.857399
15.03.2000	0.363663	0.374887	0.364283	0.216218	0.010194	0.499817	0.703592	0.202804	0.204660	…	0.861675
16.03.2000	0.388725	0.408983	0.389705	0.239549	0.027789	0.595794	0.757309	0.205431	0.218085	…	0.857655

3.3.2 Independent Component Analysis (ICA) for Feature Extraction

Independent Component Analysis (ICA) is a linear feature extraction technique that produces new statistically independent features by aiming to reduce first- and second-order dependencies in a dataset (Anowar et al., 2021). Applicable to many different datasets, ICA can generally analyze digital images, audio streams, radio signals, and biofeedback (brain wave, etc.) information and time series (Tharwat, 2021). ICA is a technique that can separate and recover unknown source signals from a complex signal without having sufficient prior knowledge of source signals and mixing mechanisms, especially in signal processing such as blind source separation (Tharwat, 2021) and cocktail party problems.

ICA's investigation of non-normally distributed and statistically independent features are the two most prominent features distinguishing it from other feature extraction mechanisms. For example, PCA searches for aspects representative of the data, while ICA searches for aspects independent of each other (Anowar et al., 2021). Based on these features of the ICA method, Sethia and Raut state that the financial dataset they use in their studies consists of different individual components and exhibits a non-normal distribution (Sethia & Raut, 2019). ICA is a valuable method as a dimension-preserving transformation because it produces statistically independent components in pattern recognition (Chen et al., 2022; Draper et al., 2003; Jianwei et al., 2019; Kao et al., 2013; Liu & Wang, 2011; Lu et al., 2009; Sethia & Raut, 2019; Thakkar & Chaudhari, 2021). Studies in the literature emphasize that ICA is a better feature extraction method than other statistical methods (Chen et al., 2022; Draper et al., 2003; Kao et al., 2013; Kwak, 2008; Liu & Wang, 2011; Reza & Ma, 2016). In our study, the ICA method was used to extract features because the dataset used had appropriate features, as stated in Sethia and Raut's study (Sethia & Raut, 2019), and it was a successful feature extraction technique.

3.3.3 Deep Learning Using Long Short-Term Memory (LSTM) Network

Long Short-Term Memory (LSTM), which is essentially an improvement of Recurrent Neural Networks (RNN), was introduced by Hochreiter and Schmidhuber (Hochreiter & Schmidhuber, 1997). What makes LSTM networks different is the ability to use historical information on time series efficiently. It can also solve the disappearing gradient problem when processing long-term dependencies in RNNs. LSTM network consists of neural network cells that repeat each other just like RNN networks, but unlike RNNs, it can remember and forget past information with mechanisms called gates in the memory cell (Hochreiter & Schmidhuber, 1997; Nosratabadi et al., 2020). This way, desired historical information can be discarded or stored in the LSTM network. Figure 5 presents the internal structure of a memory cell belonging to the LSTM network (Ozkok & Celik, 2022).

In an LSTM memory cell, three gate mechanisms, input, output, and forgetting, control the storage state. The functions and calculations of the gates are as follows:

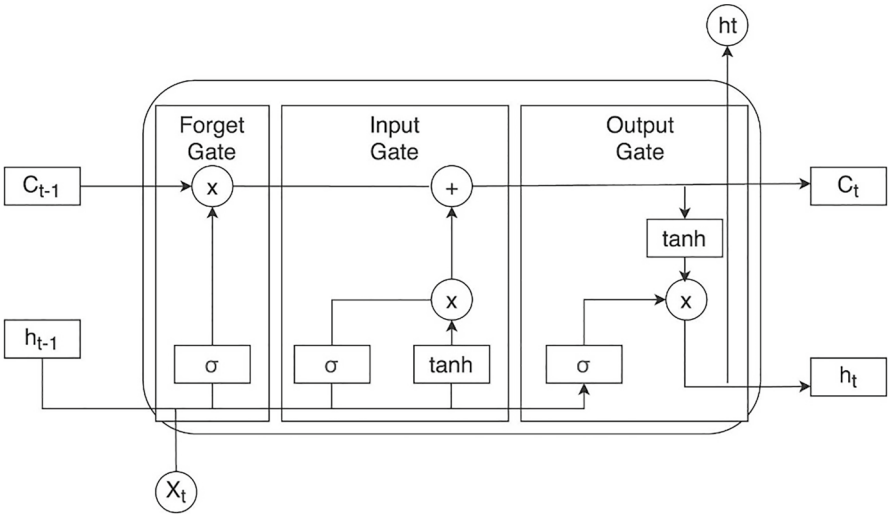


Fig. 5 The internal structure of an LSTM memory cell

- The first structure of an LSTM cell is the forget gate. Forget gate performs the function of determining the information to be discarded. σ represents the activation function, w weight, and b offset in Eq.(2). The sigmoid function produces values in the range of 0 ~ 1. Accordingly, 0 information allows the information of the previous cell to be forgotten, and 1 allows it to be transferred to the next cell completely. Thus, f_t determines how much of the previous cell's information should be remembered.

$$f_t = \sigma(w_f \times \left[h_{t-1}, x_t\right] + b_f) \tag{2}$$

- The following structure in the LSTM cell is the entry gate responsible for determining the information to be updated. First, in Eq. (3), with the help of a sigmoid function, as in the forget gate, i_t decides which values to update. Then, the vector $\overline{c_t}$ of the new candidate values that can be added to the situation with the tanh function in Eq. (4) is obtained.

$$i_t = \sigma(w_i \times \left[h_{t-1}, x_t\right] + b_i) \tag{3}$$

$$\overline{c_t} = tanh(w_c \times \left[h_{t-1}, x_t\right] + b_c \tag{4}$$

- Output Gate keeps the output information of the cell module. Entry and exit gates often use tanh or logistic sigmoid functions to perform their tasks. The information from the forget and entrance gates is combined in the first stage. Thus, the old cell state c_{t-1} is updated, and the new cell state c_t is obtained with Eq. (5). o_t, which decides which parts of the previous cell state to output, is calculated with the help of the sigmoid function in Eq. (6). The new cell

state (c_t) is subjected to the *tanh* function and multiplied by o_t to output the parts decided in Eq. (7). This resulting output (h_t) is based on the cell state but in a filtered state.

$$c_t = f_t * c_{t-1} + i_t * \overline{c_t} \tag{5}$$

$$o_t = \sigma(w_o \times [h_{t-1}, x_t] + b_o) \tag{6}$$

$$h_t = o_t * tanh(c_t) \tag{7}$$

3.4 Prediction Model Steps

The models used in our study aim to predict the price of a financial asset. To this end, we conduct a series of experiments to evaluate the effects of dimensionality reduction and feature extraction mechanisms on prediction performance in the data preprocessing phase. In the first part of our experiments, we compare our proposed model with the plain LSTM, PCA-LSTM, and ICA-LSTM prediction models derived from the same deep learning network family. In this way, we aim to show the effectiveness of our proposed model, which uses a two-stage statistical method, against prediction models that do not use statistical methods or use a single statistical method. In the second part of our experiments, we evaluate the performance of our proposed model against different deep learning networks in the literature, namely RNN, LSTM, GRU, and CNN. In the last part of our experiments, we expand the time scale of our dataset and aim to show the effectiveness of our proposed model for two new additional cases. We conduct all these experiments by following the same procedure steps below. Figure 6 presents the pseudocode we used for our experiments.

3.5 Proposed Hybrid PCA-ICA-LSTM Model

In this section, we introduce one of the main goals of our work, a hybrid deep learning model, which we call PCA-ICA-LSTM. The proposed model is built upon the LSTM neural network due to its capabilities among RNNs. RNNs, a type of deep learning model, have drawn attention for their success in analyzing and predicting datasets based on sequential data such as time series**.** Recurrent neural networks (RNNs), which can remember past time series and make decisions accordingly, have some disadvantages when dealing with large datasets. Next-generation recurrent neural networks like LSTM and GRU have become famous for mitigating these drawbacks. One of the most well-known disadvantages is the vanishing gradient problem when dealing with long-term dependencies in RNNs. This feature of LSTM networks has enabled them to be used in many fields, including finance. Sethia and Raut (2019) compared LSTM and GRU models and demonstrated that the LSTM model, an improved version of RNNs, achieved more successful results

Identify Financial Asset: S&P 500 index
Input Data: F={F_1:*open*, F_2:*close*, F_3:*high*, F_4:*low*, F_5:*volume*}
Output Data: O={F_{adj_close}}
Generalized Proposed Algorithm:

Step 1. Data Preparation:
Data Collection:
Collect historical price data of the financial asset from various sources (Yahoo Finance)
Add New Features to the Dataset:
Calculate and add technical indicators (*RSI*, *EMA*, *MACD*…) shown in Table 4 to increase the number of features ($F=\{F_1,F_2,...,F_{48}\}$)
Step 2. Data Preprocessing:
Make the Data Consistent and Complete
Standardize and Normalize the Data [0,1]
Split Data Sets for Cross Validation (*4425 samples*):
Split the data into training *(%70)*, validation *(%10)*, and test *(%20)* sets using the hold-out method.
for i ϵ [1, 10] do:
    Use PCA, which is a dimension reduction method (*component_number=12*):
    Reduce the number of components from 48 to 12 to complete the dimensionality reduction process
    Create a new dataset by removing redundant information and noise from the dataset (*Table 6*)
    Use ICA, which is a dimension reduction method (*component_number=12*):
    Preserve the number of components (dimensions) and use ICA for feature extraction
    Select practical features using ICA, creating a new dataset different from the previous step (*Table 7*)
    Step 3. Build Prediction Model:
    Select Model (*LSTM*):
    Create the LSTM deep learning prediction model according to the parameters specified in Table 2
    Train the Model (*epoch=125*)
    Step 4. Prediction:
    Validation and Test:
    Evaluate the model's performance with validation and test sets.
    Evaluate the model's performance according to Equations (8)-(14).
    Output:
    Predict the price (O) for the next day based on the data from five days before the specified date.
end for

Fig. 6 Pseudocode of the proposed PCA-ICA-LSTM prediction model

than GRU. Following this study, in our work, we opted for the classic LSTM deep learning network and used the same model architecture settings.

However, there are still challenges that the LSTM model has not yet overcome, such as the curse of dimensionality and issues like overfitting. To address these challenges, various methods and mechanisms are being developed. One of these methods is the hybrid model approach.

Recently, to enhance the prediction performance of machine learning models, hybrid models have been proposed by integrating both machine learning algorithms and deep learning models with different approaches, instead of relying solely on single models that may not yield satisfactory results in every scenario (Chen et al., 2022; He & Dai, 2022; Kakade et al., 2023; Li et al., 2022, 2023; Nosratabadi et al., 2020; Ozbayoglu et al., 2020; Srijiranon et al., 2022; Thakkar & Chaudhari, 2021; Wei & Ouyang, 2024). While certain solutions have been partially provided for

specific issues such as exploding gradients with advanced recurrent neural network variants like LSTM and GRU in the analysis of time series data, these models still face unresolved challenges. While LSTMs, due to their advanced design, can regularly eliminate invalid information and preserve crucial information when ingesting time series data, experimental findings have indicated that direct use of raw data is not beneficial for training, and hence, subjecting them to a series of statistical methods beforehand is advantageous (Berradi & Lazaar, 2019). The literature suggests that combining dimensionality reduction techniques with various machine learning methods positively impacts the performance of machine learning methods (Chen et al., 2022; Gao & Chai, 2018; He & Dai, 2022; Jianwei et al., 2019; Kakade et al., 2023; Kao et al., 2013; Li et al., 2022, 2023; Liu & Wang, 2011; Lu et al., 2009; Ma et al., 2019; Srijiranon et al., 2022; Wei & Ouyang, 2024; Zhang et al., 2020; Zheng & He, 2021; Zhong & Enke, 2017).

Deep learning models capable of working with big data often utilize dimensionality reduction techniques to mitigate the dimensionality problem. Researchers continue to investigate a multitude of novel approaches to address the aforementioned structural limitations of advanced recurrent neural network variants such as LSTM and GRU. The novelty of our work is to address the structural problems faced by the LSTM deep learning network, such as overfitting and the curse of dimensionality when working with multidimensional data, by combining them with dimensionality reduction methods through our PCA-ICA preprocessing mechanism. Leveraging the advantages of both statistical methods, our proposed hybrid PCA-ICA-LSTM approach aims to overcome these issues and improve prediction performance.

PCA and ICA methods have been widely used by the research community due to their ability to achieve effective results in many past and current studies. Their capacity to identify crucial components across diverse data types, including time series, speech and image data, and medical signals, among others, is particularly notable (Jianwei et al., 2019). While both methodologies are grounded in elementary statistical techniques, they employ distinct strategies for problem resolution. PCA generates a new feature set, equal to or fewer than the original number of variables, aimed at maximizing the variance within the dataset, known as principal components. Conversely, ICA is a linear feature extraction technique that endeavors to minimize first- and second-order dependencies within a dataset, thereby generating statistically independent new features. While PCA endeavors to identify representative data directions, ICA seeks out independent directions (Anowar et al., 2021). Upon scrutinizing studies focused on the financial domain, PCA typically finds favor in dimensionality reduction and feature selection tasks (Bustos & Pomares-Quimbaya, 2020; Gao & Chai, 2018; Huang et al., 2022; Kakade et al., 2023; Li et al., 2022; Ma et al., 2019; Singh & Srivastava, 2017; Srijiranon et al., 2022; Wang et al., 2023; Wei & Ouyang, 2024; Zheng & He, 2021; Zhong & Enke, 2017); whereas ICA is employed for feature selection and noise reduction mechanisms (Chen et al., 2022; Draper et al., 2003; Jianwei et al., 2019; Kao et al., 2013; Kwak, 2008; Liu & Wang, 2011; Lu et al., 2009; Reza & Ma, 2016; Sethia & Raut, 2019; Thakkar & Chaudhari, 2021).

The S&P 500 dataset we are working on has transformed into a multidimensional dataset as a result of adding more than 40 technical indicators, as presented in Table 4. The use of technical indicators in the dataset contributes significantly to the predictive performance of the models (Gao & Chai, 2018; Kakade et al., 2023). However, the use of technical indicators also leads to certain disadvantages. One of these disadvantages is the expansion of the feature set due to the use of numerous technical indicators, resulting in the dimensionality problem. The dimensionality problem is also known as the curse of dimensionality and implies that as the number of features increases, the risk of overfitting also increases (Srijiranon et al., 2022; Zheng & He, 2021). The second disadvantage is the proliferation of redundant information due to the similar calculation techniques used by technical indicators, ultimately leading to noise in the data. Principal Component Analysis (PCA) is a dimensionality reduction technique that utilizes statistical methods to represent the entirety of a dataset using a minimal number of fundamental components while minimizing data loss. PCA is the oldest and most well-known statistical method employed for dimensionality reduction (Kakade et al., 2023; Wang et al., 2023; Wei & Ouyang, 2024). PCA not only reduces the dimensionality of the dataset but also decreases the dimensionality of noise in the data, making it commonly used for reducing noise in the data (Li et al., 2022; Ma et al., 2019; Zheng & He, 2021). For these reasons, PCA was employed in the initial stage of the proposed model to address the high dimensionality of the dataset and eliminate noise.

While the PCA method has been successfully applied for feature extraction in numerous studies, it has some drawbacks. The most notable disadvantage is that PCA assigns equal weights to all components, potentially disregarding the prediction target (Guo et al., 2022; Huang et al., 2022). For instance, in extreme cases, assigning equal weights to all components by PCA may overlook strong components, or conversely, it may assign excessive weight to irrelevant or weak components for the prediction target, leading to the generation of noisy information. This aspect could hinder PCA from achieving stable prediction results (Wei & Ouyang, 2024). Furthermore, PCA has limitations in extracting higher-order statistical information due to its utilization of second-order statistical properties (Zare et al., 2018). Therefore, exploring complex and multifaceted data like financial time series directly with the PCA method is challenging. In our study, we aimed to address this limitation of PCA by utilizing another statistical method. To this end, we preferred the ICA method for feature extraction due to its ability to utilize higher-level statistical properties and positive effects on the performance of using ICA after PCA (Draper et al., 2003).

As a result, in the second preprocessing stage of our proposed prediction model, the new feature set reduced in size and denoised by PCA is directly transferred to the ICA method for feature extraction. Several benefits of applying PCA before ICA have been emphasized in the literature (Draper et al., 2003). Firstly, the PCA method allows us to control the number of independent components obtained from ICA by reducing the data dimensionality. Secondly, using PCA for dimension reduction before whitening helps eliminate low-variance features. Another advantage of using the PCA method is that it enhances

Table 6 An example representation of the principal components (PC) obtained after denoising

PC_1	PC_2	PC_3	PC_4	PC_5	PC_6	PC_7	PC_8	PC_9	PC_10	PC_11	PC_12
−0.131759	0.157004	0.143374	−0.125015	−0.144010	−0.121097	0.084826	−0.064723	0.023769	−0.043126	−0.110306	0.048014
−0.125271	0.159870	0.139613	−0.085572	−0.106780	−0.134241	0.121804	−0.099813	0.026525	−0.018552	−0.066560	0.015207
−0.146400	0.087084	−0.047768	−0.187706	−0.065575	−0.076797	0.084923	−0.063739	0.044847	−0.101160	−0.110777	−0.011790
−0.164176	0.091152	−0.055753	−0.114202	−0.036984	−0.121347	0.153870	−0.138945	0.075040	−0.144566	−0.040068	−0.013843
−0.158987	0.034435	−0.061456	−0.060665	−0.155881	−0.068647	0.158155	−0.040594	−0.039295	−0.184862	0.008404	0.093014
−0.149122	0.057226	0.050689	−0.029755	−0.207158	−0.068038	0.181567	−0.053354	−0.042394	−0.030394	0.141699	0.058448
−0.166912	−0.014247	−0.122627	−0.120229	−0.157980	0.025844	0.154778	−0.026713	−0.055265	−0.052304	0.097478	0.021996
−0.233175	0.028306	−0.299010	−0.225892	−0.109615	0.061815	0.018859	−0.131631	−0.086128	−0.006955	0.088175	−0.038197
−0.257711	0.046991	−0.295458	−0.270803	−0.123967	0.055398	0.022397	−0.132845	−0.150014	0.028848	0.148784	−0.110596
−0.262618	0.030055	−0.281024	−0.236996	−0.173445	0.001407	−0.084360	−0.052005	−0.036076	0.104333	0.192957	−0.069466

Table 7 An example representation of the independent components (ICs) obtained after feature extraction

IC_1	IC_2	IC_3	IC_4	IC_5	IC_6	IC_7	IC_8	IC_9	IC_10	IC_11	IC_12
0.005975	0.007480	−0.002308	0.020501	−0.032336	−0.020911	−0.015116	−0.020364	−0.016642	−0.004474	−0.011668	−0.004135
0.006729	0.008196	0.004539	0.013276	−0.024052	−0.019643	−0.015250	−0.024370	−0.008562	−0.006596	−0.010724	−0.016629
0.001643	0.007404	0.008470	0.005045	−0.036724	−0.022803	−0.000791	−0.016554	−0.025307	0.009812	−0.000572	−0.006356
0.003614	0.008966	0.031045	−0.007292	−0.031944	−0.030203	−0.008019	−0.026623	−0.016197	−0.000212	0.000455	−0.010994
0.005348	−0.009310	0.028629	−0.000952	−0.036827	−0.003848	−0.023106	−0.032307	−0.006830	−0.003095	−0.005963	0.023368
0.010458	−0.013101	0.026149	0.004392	−0.003199	0.013285	−0.042738	−0.036181	0.006103	−0.004289	−0.002448	−0.001917
0.005573	−0.015165	0.020308	−0.005893	−0.011788	0.015763	−0.030190	−0.027113	−0.003047	0.016563	0.003375	0.000408
0.001203	0.017766	0.007793	−0.021470	−0.008400	0.023847	−0.028331	−0.016664	−0.011045	0.033493	0.011541	−0.009199
0.004277	0.022687	0.011885	−0.020386	−0.003500	0.045444	−0.027624	−0.021151	−0.006000	0.042243	0.018686	−0.024561
−0.000834	0.018494	0.002960	−0.005866	0.026743	0.041234	−0.035362	−0.010783	−0.022876	0.024200	0.028159	−0.019003

computational efficiency by reducing computational complexity by minimizing pairwise dependencies. All these benefits encourage the use of ICA after PCA and improve ICA performance (Draper et al., 2003). There are many studies in the literature demonstrating the effectiveness of the ICA method in feature extraction (Chen et al., 2022; Jianwei et al., 2019; Kao et al., 2013; Liu & Wang, 2011; Lu et al., 2009; Sethia & Raut, 2019; Thakkar & Chaudhari, 2021). The ICA method generates a new input set with a different number of features and a high-importance explanatory feature set after the initial preprocessing (refer to Table 6 and Table 7). The new input set, which cannot be processed by PCA due to its second-order feature processing capability, undergoes statistically higher-order feature processing with ICA preprocessing and is applied to the LSTM deep learning network forming the prediction system. Thus, the ability to combine the advantages of two different dimensionality reduction methods makes the hybrid PCA-ICA-LSTM prediction model remarkably effective in the conducted experimental studies.

While studies exist in the literature that combine PCA or ICA with different methods, to our knowledge, ours is the first study to combine these two statistical methods for a two-stage preprocessing and integrate them with a recurrent deep learning network to create a hybrid model. This highlights the novelty and theoretical contribution of our study. The proposed model is straightforward and presents a new framework that combines PCA and ICA statistical methods to provide input to an LSTM deep learning network used for prediction. The PCA-ICA-LSTM prediction model is constructed per the operation of the prediction model and basic adjustments described in the previous sections. Figure 7 shows the prediction framework of the proposed PCA-ICA-LSTM model.

If we briefly summarize the operation of the proposed model, it is first made into the original dataset by adding the calculated technical indicator information to the raw dataset consisting of market information. After the dataset is normalized, it is input to statistical methods for two-stage pre-treatment. In the first stage, the dataset, whose dimension is reduced by the PCA method, is purified from noise. The resulting essential components (PCs) are current information compressed by removing unnecessary information and noise. In the second stage, before the PCs are used as input to the LSTM network, they are subjected to a decomposition process to find the practical features. Therefore, they are decomposed into their independent components (ICs) by the ICA method, and ICs are obtained. This results in a feature set that contains noise-free, efficient information for the prediction system. This new feature set consists of essential and independent features that can represent 95% of the total variance of our original dataset. Example representations of the PCs and ICs obtained during the two-stage preprocessing of our dataset are given in Tables 6 and 7, respectively.

In the last stage, the obtained dataset is used for training and evaluating the LSTM deep learning network to predict prices five days after the determined date.

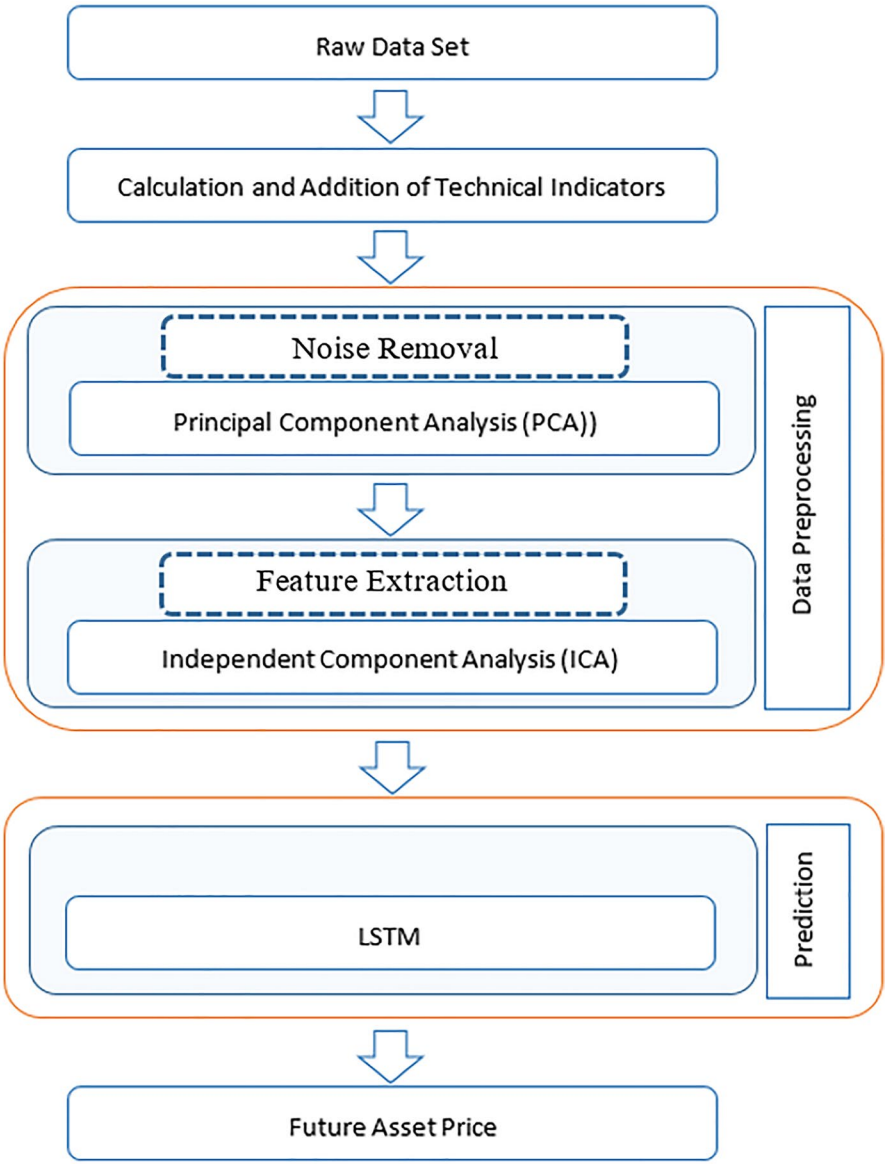


Fig. 7 Proposed PCA-ICA-LSTM model

3.6 Evaluation Criteria

Evaluation criteria are needed to measure the predictability of the forecast model and the accuracy of the forecasts found. Error estimation methods such as root mean square error (RMSE), mean square error (MSE), mean absolute error (MAE), and Mean Absolute Percent Error (MAPE) are frequently used in the literature to evaluate the performance of deep learning models. As evaluation metrics within the scope of the study, coefficient of determination (R^2), MSE, MAE, MAPE, Max Error, and Return Ratio are aimed to be compared with other studies (Chowdhury et al., 2018; Gao et al., 2021; Jianwei et al., 2019; Sethia & Raut, 2019; Wen et al., 2020). We present Eqs. (8–13) for the calculation of evaluation criteria below. Accordingly, $\hat{y}_i$, i. is the predicted value of the sample, and y_i represents the corresponding actual value.

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2} \tag{8}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{9}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{10}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{11}$$

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{|y_i|} \tag{12}$$

$$Max\ Error = \max(|y_i - \hat{y}_i|) \tag{13}$$

Accordingly, the study uses two well-known evaluation metrics, such as R^2 score and MSE, to see the model performance. The coefficient of determination measures how successful our prediction model is, and the higher its value, the higher the model's performance will be. The MSE metric, on the other hand, is a positive measure that shows that the prediction model's performance increases as its value approaches zero. In addition to R^2 and MSE metrics, evaluation metrics used to measure regression performance, such as MAE, MAPE, and Max Error, were examined to compare with other studies in the literature. MAE performs the same task as MSE and is interpreted similarly. However, MAE looks at the absolute difference between the data and the model's predictions, and outlier residuals do not contribute as much to the total error as the MSE. MAPE is the percent equivalent of MAE. MAPE can eliminate the disadvantages when comparing models with different unit values. In addition, MAPE, since it expresses the estimation errors as

a percentage, makes sense and can be interpreted on its own, making it different from other metrics. The Max Error metric calculates the maximum residual error, capturing the worst-case error between the predicted and actual values. So, the Max Error shows the extent of the model's error when fitted.

We add a different evaluation metric to our study for use in the second part of the experimental work outside of these evaluation metrics. In the second part of the experiments, we will compare our proposed model with state-of-the-art models and examine the return rates of the models through a simple trading strategy. This offers researchers the opportunity to evaluate models from a unique perspective. Equation (14) below demonstrates the success rate of models in trading strategy compared to the "hold and wait" strategy through the return ratio metric.

$$ReturnRatio = (return_{tradestrategy} x1.0)/return_{holdandwaitstrategy} \tag{14}$$

4 Experimental Results and Discussion

In the experiments, we test the success of the proposed PCA-ICA-LSTM model in predicting the price of a financial asset against models that use single-stage statistical methods and state-of-the-art models in the literature. At this point, we divide our experiments into four parts in line with the goals stated. In the first part, we prepare plain (LSTM) and hybrid (PCA-LSTM, ICA-LSTM, and PCA-ICA-LSTM) price prediction models and compare the proposed model against models derived from the same family. In this way, we aim to demonstrate the superiority of our PCA-ICA-LSTM model with two-stage preprocessing capability over prediction models that perform single-stage preprocessing. In the second part of our experiments, we compare our proposed model with commonly used deep learning models in the literature, including RNN, GRU, LSTM, and CNN. We also create a simple trading strategy for this comparison and evaluate the performance of the models in terms of return rates. In the third part of our experiments, we compare our proposed model with previous studies that utilized the same dataset in the literature. In the fourth part of our experiments, we make several changes. We extend the time scale of our dataset, selected from the literature for comparability, to update it. We set the time scale of our dataset to 2000–2024 to include the COVID-19 Pandemic. We investigate the most suitable dimensionality reduction component number for the changing time scale in our dataset and repeat our experiments for the found component number value. We also add two additional case studies to our research as a benchmark. All models used in the experiments follow the basic settings presented in Sect. 3.2. To ensure a fair comparison, we ran the experiments ten times and calculated the averages of the metrics. The experiments were performed on a system with Intel Core i7—2.5 GHz CPU and 32 GB RAM, using Python programming language and Keras library in a Google-Collaboratory environment.

Table 8 Deep learning price prediction models and results

Models	Criterion	Average	Best	Worst	Standard deviation
LSTM	R^2	0.859670	0.942713	0.727342	0.086950
	MSE	0.001457	0.000525	0.002831	0.000903
	RMSE	0.036300	0.022920	0.053208	0.011808
	MAE	0.029175	0.017617	0.043833	0.010106
	MAPE (%)	3.833056	2.433490	5.627846	1.228259
	Max Error	0.130058	0.118767	0.146149	0.009610
PCA-LSTM	R^2	0.956055	0.960098	0.950946	0.002828
	MSE	0.000456	0.000414	0.000509	0.000029
	RMSE	0.021350	0.020355	0.022569	0.000682
	MAE	0.016652	0.015636	0.018036	0.000704
	MAPE (%)	2.301951	2.168703	2.491193	0.092918
	Max Error	0.107193	0.103521	0.114547	0.003153
ICA-LSTM	R^2	0.961800	0.964421	0.958064	0.001909
	MSE	0.000397	0.000369	0.000435	0.000020
	RMSE	0.019910	0.019221	0.020867	0.000493
	MAE	0.015100	0.014464	0.016010	0.000442
	MAPE (%)	2.103943	2.023489	2.204409	0.054250
	Max Error	0.106107	**0.099213**	**0.110055**	0.003658
PCA-ICA-LSTM	R^2	**0.963297**	**0.964900**	**0.961943**	**0.000906**
	MSE	**0.000381**	**0.000364**	**0.000395**	**0.000009**
	RMSE	**0.019521**	**0.019091**	**0.019879**	**0.000241**
	MAE	**0.014667**	**0.014306**	**0.014992**	**0.000212**
	MAPE (%)	**2.049538**	**2.000409**	**2.083329**	**0.026623**
	Max Error	**0.104625**	0.099255	0.110564	**0.002820**

4.1 Comparison of Our Proposed Model and Prediction Models Derived from the Same Family

The performance of the PCA-ICA-LSTM model, which we propose within the scope of experimental studies, was compared with the performances of LSTM, PCA-LSTM, and ICA-LSTM deep learning price-prediction models used in the literature to predict the price of a financial asset. The experimental results obtained for three different deep learning price prediction models and the proposed hybrid deep learning prediction model are shown in Table 8. PCA-ICA-LSTM price prediction model, the two procedures for dimension reduction and feature extraction, which we introduced and proposed in Sect. 3, sequentially link each other and the deep learning network. Models were run 10 times during the experiments, and the results of the evaluation criteria were analyzed under four headings: average, best, worst, and standard deviation. The best results for the evaluation metrics are shown in bold in Tables 8, 9, 10, 11.

When Table 8 is examined, it is possible to evaluate the forecasting performances of the models. Accordingly, we can see that the proposed hybrid PCA-ICA-LSTM price prediction model for all categories realizes the highest R^2 and lowest MSE values, so much so that the hybrid PCA-ICA-LSTM model managed to reduce the average MSE value obtained by the LSTM price prediction model from 0.001457 to 0.000381 by improving it by 73.85%. An analogous situation is observed in the R^2 score. The proposed hybrid model increased the average R^2 value obtained by the LSTM price prediction model from 0.859670 by 12.05% to 0.963296.

When we compare the results obtained by the PCA-ICA-LSTM model with those of the PCA-LSTM model, the effect of using the ICA method for feature extraction will be more pronounced. According to the results, the hybrid PCA-ICA-LSTM model increased the average R^2 value obtained with the PCA-LSTM price prediction model from 0.956055 to 0.963297 by improving it by 0.75%. At the same time, the proposed hybrid PCA-ICA-LSTM model reduced the average MSE value obtained with the PCA-LSTM price prediction model by 16.45% from 0.000456 to 0.000381.

The graphical representations of the experimental results are presented in Fig. 8. The blue solid line represents the actual value of the financial asset used in the test set, and the red dotted line represents the value estimated by the relevant deep-learning model. Accordingly, when the graphs in Fig. 8 are examined, it is seen that the most successful model is the proposed PCA-ICA-LSTM model shown in Fig. 8d, and the most unsuccessful model is the LSTM model shown in Fig. 8a. In the graphical representation presented in Fig. 8a, it is noteworthy that there are vast differences between the estimated values obtained by the LSTM model and the actual values. In addition, in the graphical representation of the proposed PCA-ICA-LSTM model presented in Fig. 8d, it is seen that the values estimated by the model are remarkably close to the actual values.

We can say that the proposed PCA-ICA-LSTM model is more stable than other models in sudden price changes. We understand this situation because when error metrics such as MSE, RMSE, MAE, and MAPE are examined in Table 8, the best and worst results of the proposed model are remarkably close to each other compared to other models. When Fig.8a–d are scrutinized, it will be seen that the success of the proposed PCA-ICA-LSTM hybrid deep learning model is more noticeable graphically around the 800th record.

The estimated price values obtained by the actual and deep learning prediction models of the S&P500 index were compared during the experimental study. In addition, the estimated price values and residuals obtained by the models were also compared. The results show that the proposed hybrid PCA-ICA-LSTM model outperforms other models. The proposed hybrid PCA-ICA-LSTM model's actual and estimated prices often match well and center around the diagonal. The results of this situation are presented in Fig. 9d. Accordingly, the graphics on the left side of the page show the actual and estimated prices of the relevant models, and the graphs on the right show the remnants of the respective models.

Each deep learning price prediction model compared in the experimental study achieves acceptable, successful results. Apart from R^2 and MSE, values less than 5% obtained by our evaluation metric, MAPE, also support this situation (Montaño Moreno et al., 2013). Since the results obtained are remarkably close to each other

Fig. 8 S&P 500 Index actual and predicted price values of Models: **a** the result of the LSTM model, **b** the result of the PCA-LSTM model, **c** the result of the ICA-LSTM model, and **d** the result of the proposed PCA-ICA-LSTM model

and it is not easy to distinguish the results on the graphs (Fig. 9), we present new charts showing the density of the residues in Fig. 10.

When the distribution of residues in Figs. 9a and 10a is examined, it is seen that the plain LSTM model mostly has a scattered residue density in the range of $-0.10 \sim 0$. However, in a successful model, the residuals are expected to be in the 0-line as much as possible, for example, in Figs. 9d and 10d. So, when Fig. 10b–d, are examined, deep learning models that include dimension reduction and feature extraction methods in the distribution of residuals show a normal distribution compared to the plain LSTM deep learning model. When Fig. 10d is carefully examined, it can be seen that the proposed hybrid PCA-ICA-LSTM deep learning model shows a more balanced and dense distribution around the 0-line in terms of the distribution of residuals compared to the PCA-LSTM model in Fig. 10a and the ICA-LSTM model in Fig. 10c. This result demonstrates the success of our proposed PCA-ICA-LSTM hybrid deep learning model against other models in experimental studies.

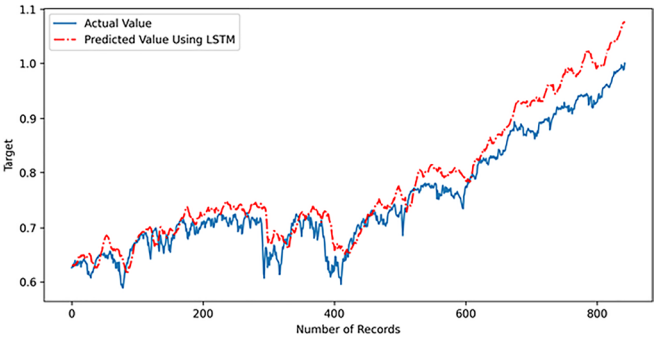
These results reveal that deep learning price prediction models created with the addition of dimension reduction and feature extraction techniques are more effective than plain deep learning price prediction models (LSTM) in making the prediction values close to the actual price of the financial asset. In comparing the four models, the plain LSTM model obtained the worst rates in the evaluation metrics. In addition, it is observed that the proposed hybrid PCA-ICA-LSTM model reaches the highest R^2 and lowest MSE, MAE, MAPE, and Max Error values even if the models obtain remarkably close results during the comparison.

4.2 Comparison of Our Proposed PCA-ICA-LSTM Model with Widely Used State-of-the-Art Models

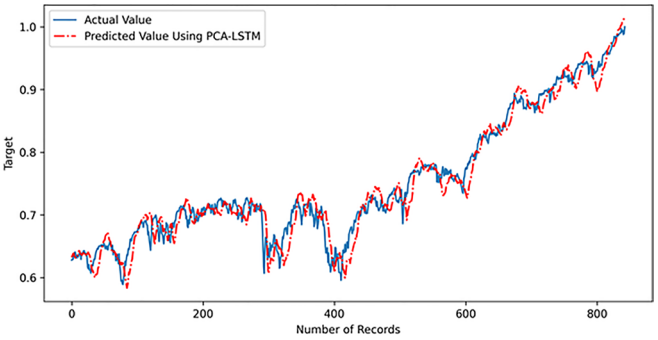
The importance of prediction models in financial markets lies in the fact that they guide investors and assist in assessing new opportunities while mitigating risks in a dynamic market. The stock market can present a lucrative investment environment for investors, but there is a clear correlation between profitability and risk. As a result, investors seek to maximize returns while minimizing risk by predicting the probable value of financial assets. In this segment of our study, we will attempt to calculate the return ratio of models.

First, we develop a straightforward trading strategy to compute the returns of our models. Based on this strategy, the model issues a buy signal if it predicts a higher price for the financial asset on day $t+5$ than on day t. Therefore, the model purchases the financial asset on day t and sells it on day $t+5$. Conversely, if the model anticipates a lower price, it generates a sell signal and short-sells the financial asset. The difference between the two prices represents the model's profit or loss for that particular trade. Ultimately, the model's rate of return is calculated as the

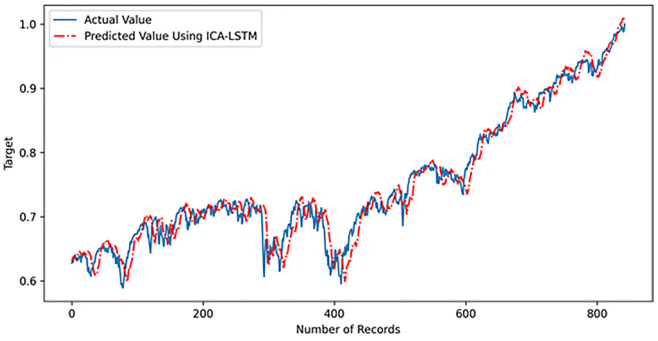
(a)


Actual Value
Predicted Value Using LSTM
Target
Number of Records

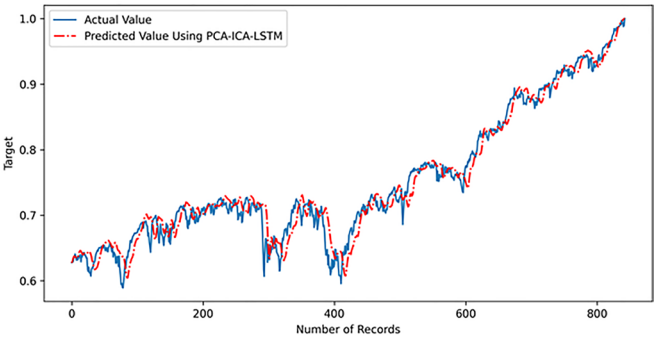
(b)


Actual Value
Predicted Value Using PCA-LSTM
Target
Number of Records

(c)


Actual Value
Predicted Value Using ICA-LSTM
Target
Number of Records

(d)


Actual Value
Predicted Value Using PCA-ICA-LSTM
Target
Number of Records

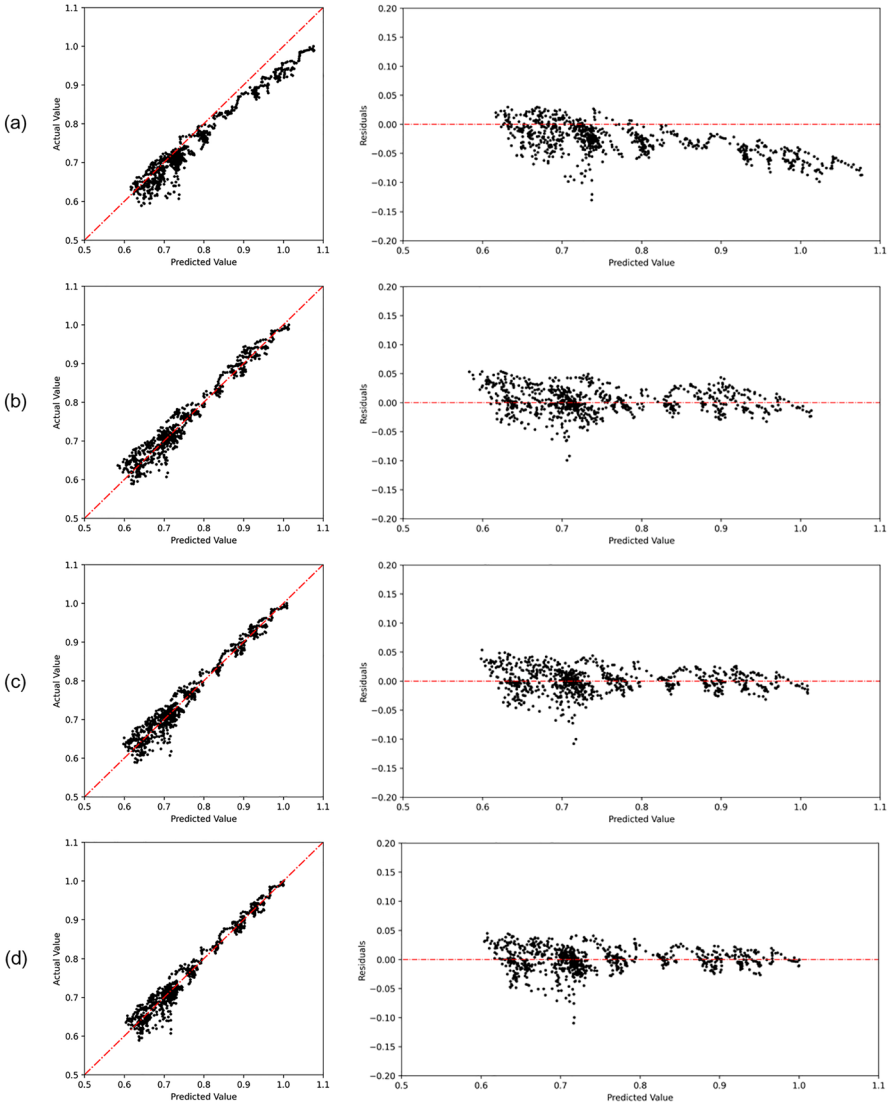


Fig. 9 S&P 500 Index actual and predicted price values and residuals with Deep Learning Price Models: **a** LSTM Model, **b** PCA-LSTM Model, **c** ICA-LSTM Model, **d** PCA-ICA-LSTM Model

ratio between the total value earned at the end of each trade and the value obtained through the "hold and wait" approach. The equation for this calculation is provided in Sect. 3.6, Eq. (14).

This section will compare our proposed PCA-ICA-LSTM model with state-of-the-art models commonly used in financial data studies. Specifically, we created RNN, LSTM, GRU, and CNN models based on the "baseline settings" subsection

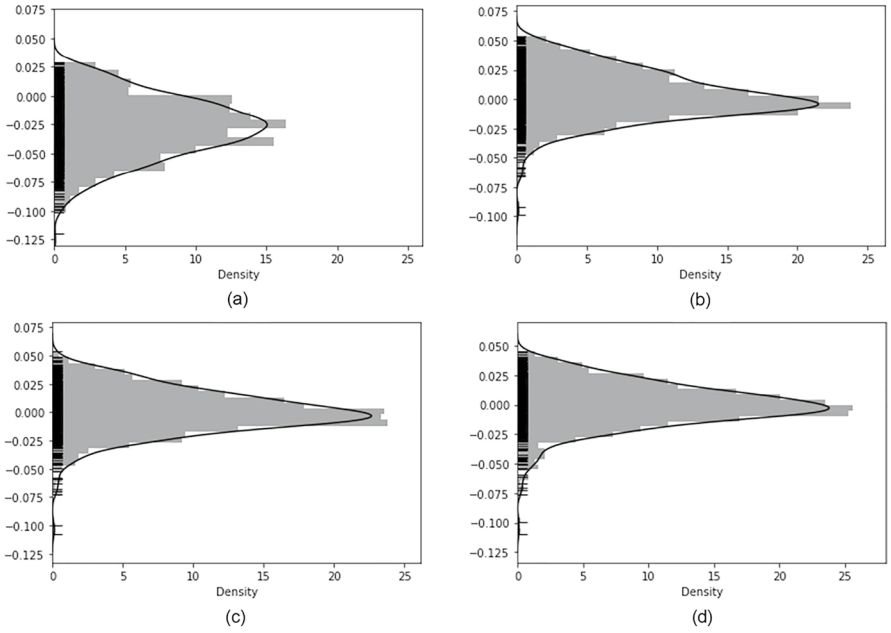


Fig. 10 Residual Density of Deep Learning Price Prediction Models: **a** LSTM Model Residual Density, **b** PCA-LSTM Model Residual Density, **c** ICA-LSTM Model Residual Density, **d** PCA-ICA-LSTM Model Residual Density

from Sect. 2. We used 1D convolutional networks for the CNN model while adhering to these baseline settings. Since there are drawbacks to using dropout operations in CNN networks, we applied max-pooling operations instead of dropout operations. As a result, we created RNN, LSTM, GRU, and CNN models with a 5-layer architecture without any statistical preprocessing.

When we examine Fig. 11, which shows the S&P 500 Index's actual value and the prediction values of all models for comparison, the region between the 250th and 650th records becomes crucial as it displays sudden changes in the S&P 500 Index and allows us to observe the models' performance more clearly. According to this, we can observe that our proposed PCA-ICA-LSTM model closely mimics the actual price movements. Additionally, we can conclude that the CNN model also performs quite well. This is evident from the R^2, MSE, and MAPE values obtained by the models and our observations in Fig. 11. Figure 12a–c display the R^2, MSE, and MAPE values obtained by the models, respectively. When assessing model performance, a high R^2 value indicates better performance. On the other hand, low MSE and MAPE values indicate successful model predictions. Out of the models we tested, the RNN model performed the worst, followed by the GRU model. Specifically, the RNN model tended to make pessimistic predictions and deviate from actual values.

Based on the rate of return metric results shown in Fig. 12d, it can be observed that the RNN model, which had pessimistic forecasts compared to the actual values of the S&P 500 Index, faced significant losses against the "hold and wait" strategy

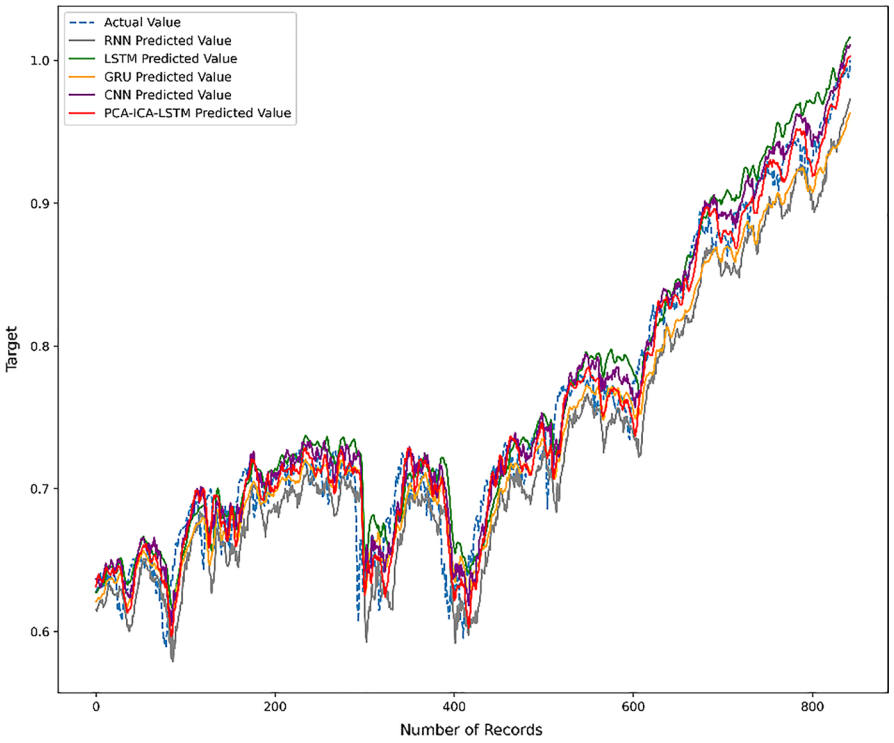


Fig. 11 S&P 500 Index's actual value and the prediction values of all models

with the simple trading strategy that was designed. Similarly, the GRU model did not provide better returns than the hold-and-wait strategy. On the other hand, the other three models achieved much more profitable returns than the "hold and wait" strategy. Among these models, the LSTM and CNN models were particularly successful, with returns exceeding 200%. We believe this outcome is because the LSTM and CNN models produced overly optimistic results, especially after the 700th record in Fig. 11, but were still able to make profitable trades due to the significant upward trend. The predictions made by these models were far from the actual values.

Our analysis shows that while our proposed PCA-ICA-LSTM model may not be as effective as the LSTM and CNN models in this aspect, it has generated returns that are 165% higher compared to the "hold and wait" strategy. This signifies that our model is highly competitive regarding both rate of return and error rate metrics. However, it is essential to note that successful predictions with low error rates alone are not enough to calculate the rate of return. In our experiment, all models had R^2 values above 0.90 and very low MSE values, but one model had negative returns compared to the "hold and wait" strategy, as illustrated in Fig. 12d. Developing a well-thought-out trading strategy is paramount in creating a model with high returns. We believe that with a well-developed trading strategy, our proposed PCA-ICA-LSTM model can achieve even more competitive results.

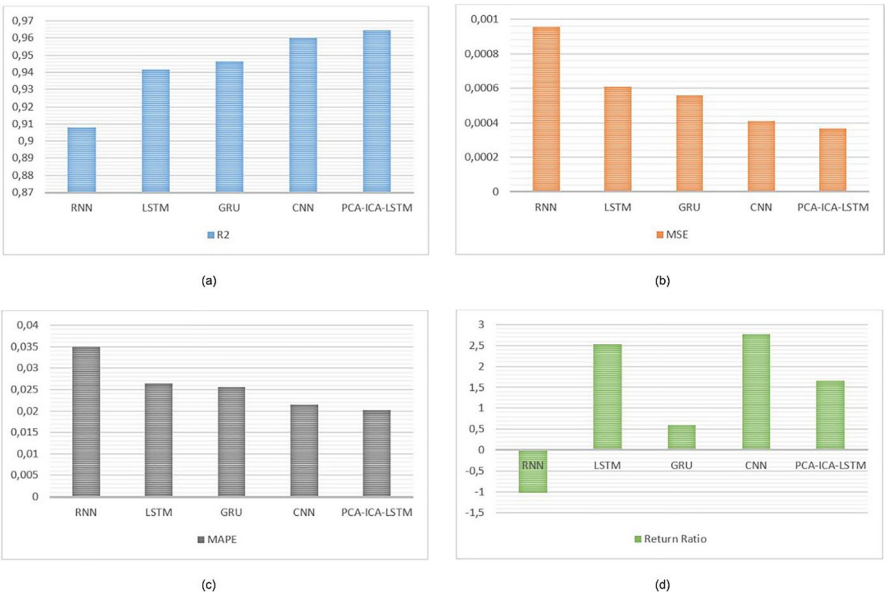


Fig. 12 Comparison of state-of-the-art models for post-experiment evaluation metric results: **a** R^2 metric results, **b** MSE metric results, **c** MAPE metric results, **d** Return ratio metric results

4.3 Comparison of the Recommended PCA-ICA-LSTM Model Similar Studies from the Literature

Under the study's first objective, we compared the PCA and ICA techniques used in the data preprocessing stage and the combined deep learning price prediction models with each other. In the first section, our PCA-ICA-LSTM hybrid model, distinguished as the most successful model, will be compared with similar studies in the literature in this section.

Although there have been many studies for the S&P500 dataset in the past, it is impossible to compare most of them. We can summarize the reasons for this under a few headings. The first reason is that although they examine the same financial time series, they are studies covering different periods. The second is the differences seen in the dataset properties. While some of the datasets contain only market information, some add diverse information to the market information to form the dataset. Examples are datasets created by adding technical indicators, fundamental indicators, or text information (news, social media messages, etc.). Another reason is the diversity in the evaluation metrics used in the models. In our study, we give many metrics that can evaluate price-prediction models. In this way, we aimed to increase the comparability of the proposed hybrid model with the models in different studies that have been made or will be done. However, even if the same evaluation metrics are used in some studies, the differences in the hyper-parameters used in the models can be considered a limitation of our comparison, for example, specifying a different window size or forecasting different time scales.

Table 9 The model we recommend and the results of the models in the literature (Sethia & Raut, 2019)

Criterions	MODELS				
	PCA-ICA-LSTM (Average)	Models in the study by Sethia and Raut			
		ICA-LSTM	ICA-GRU	ICA-SVM	ICA-MLP
R^2	**0.963296**	0.948616	0.938698	0.934952	0.874004
MSE	**0.000381**	0.000428	0.000511	0.000543	0.001052

For the above reasons, we aimed to prepare a comparative study using a dataset and deep learning architecture with similar characteristics. We mentioned earlier that we reviewed the comparable experimental work by Thakkar and Chaudhari (Thakkar & Chaudhari, 2021). The dataset and deep learning architecture used in the related study were the same as in the study by Sethia and Raut (Sethia & Raut, 2019). Therefore, we first compare our study with that of Sethia and Raut (Sethia & Raut, 2019). The related study uses LSTM and GRU deep learning models and SVM and multi-layer perceptron (MLP) models. All models were subjected to dimension reduction using the ICA technique, and the number of components was set to 12. Two of the five metrics determined as evaluation criteria are the R^2 and MSE evaluation metrics used in our study. The results of the related study revealed that the LSTM model is the most successful. The results obtained from this study and the results of the hybrid PCA-ICA-LSTM deep learning price prediction model we recommend are shown in Table 9.

If we recall our evaluation criteria in general terms, we can say that the model with the highest R^2 score follows the actual price movements of the financial asset more closely. MSE, one of our evaluation criteria, confirms this with the low values it will receive. Accordingly, when Table 9 is examined, Sethia and Raut (Sethia & Raut, 2019). It is observed that the ICA-LSTM model, which was declared the most successful model in their studies, obtained 0.948616 and 0.000428 for R^2 and MSE values, respectively. It is known that the hybrid PCA-ICA-LSTM model that we propose gets 0.963297 and 0.000381 for R^2 and MSE values, respectively. According to these results, it is seen that the hybrid PCA-ICA-LSTM model provides an improvement of 1.55% for the R^2 score and 10.91% for the MSE value compared to the model proposed by Sethia and Raut (Sethia & Raut, 2019).

Table 10 The model we recommend and the results of the models in the literature (Gao et al., 2017)

Criterions	Models				
	PCA-ICA-LSTM (Average)	Models (performance of 400 days) in the study by Gao et al			
		MA	EMA	SVM	LSTM
MAE	**0.0147**	33.9279	19.1532	16.1231	14.7709
RMSE	**0.0195**	40.9691	24.6726	21.8863	20.4668
MAPE	**0.0205**	1.6310	0.9329	0.7907	0.7240

Another study using a similar scale and time interval in the literature is that of Gao et al. (Gao et al., 2017). The authors tried to predict the next day's stock movement using the LSTM deep learning network in this study. The experimental study dataset consists of market information of 4243 trading days between the S&P500 between January 3, 2000, and November 10, 2016. In our study, the dataset size is 4425 trading days and covers the dates of January 3, 2000, and October 30, 2017. The datasets are of remarkably similar scales to each other. Gao et al. (2017) used the MAE, RMSE, MAPE, and AMAPE scales as the evaluation metric. Accordingly, the experimental results showed that the prediction system proposed by the authors gave higher prediction accuracy for the closing price of the next day's stock compared to other systems (moving average (MA), exponential moving average (EMA), support vector machine (SVM)). The results of our proposed price prediction model and the results of the study of Gao et al. (2017) are given in Table 10.

When Table 10 is examined, we see significant differences between the results. The main factor here comes from the dataset used. Although the dataset size is similar in scale and is divided into training and test sets at similar rates, there are differences in the dataset features. While the study of Gao et al., (2017) consists of only market information features, the dataset in our study consists of both market information and over forty technical indicator information. Thus, more suitable features were obtained for estimating price information, and our study's deep learning prediction models achieved better results. Another difference between studies is the size of the sliding window. While predicting the next day's price from the data of the past 20 days in Gao et al., (2017), this parameter was determined as 5 days in our study. We can show this situation as another factor affecting the results obtained in evaluation metrics.

In another study on the S&P 500 dataset, Hossain et al. performed stock price prediction using two deep learning networks: LSTM and GRU (Hossain et al., 2018). In the related study, a large dataset consisting of market information covering the years 1950 ~ 2016 used by Di Persio and Honchar in their studies was compared with the same study (Di Persio & Honchar, 2016). Researchers used MAE, MSE, and MAPE metrics to demonstrate the success of the proposed model. Accordingly, the researchers' results were much better than the Di Persio and Honchar (2016) studies. The study of Hossain et al. (2018) covers a much

Table 11 The model we recommend and the results of the models in the literature (Hossain et al., 2018)

Criterions	Models				
	PCA-ICA-LSTM (Average)	Models in the study by Hossain et al			
		Proposed LSTM + GRU	GRU + LSTM	GRU(2L)	LSTM(2L)
MAE	**0.0147**	0.023	0.063	0.028	0.086
MSE	**0.00038**	0.00098	0.008	0.001	0.018
MAPE	**0.0205**	0.0413	0.0936	0.0464	0.1158

larger time series than ours. Due to the scale difference, comparing studies with metrics such as MAE and MSE would not be correct. However, since the MAPE metric is scale-independent and can be used to compare different series or forecast scenarios, it will be possible to evaluate these two studies because MAPE calculates the percentage of error between actual values and predicted values. Table 11 compares the proposed PCA-ICA-LSTM model with some of the models in the Hossain study. Hossain et al. reported their MAPE result for the proposed model as 4.13%. Our proposed PCA-ICA-LSTM model shows a much more successful performance than the model suggested by Hossain et al., with an average MAPE value of 2.05%. We can explain this difference between the two studies in two ways. In the study of Hossain et al., the dataset consists of market information, while the dataset of our study consists of both market information and technical indicators. In this way, it contributed to forming more compelling features in the dataset. It is possible to see the effect of technical indicators with the 3.83% average MAPE value obtained by the lean LSTM model in pageour study. Another difference between the two studies is the effect of our dimension reduction methods in deep learning models. Because the average MAPE criterion obtained in all of our price prediction models using dimension reduction methods is below 2.31%. These results reveal the effectiveness of both the technical indicators used in the dataset and the dimension reduction methods used in prediction models.

4.4 The Expanded New S&P 500 Dataset and Additional Case Studies

As mentioned in previous sections, one of the objectives of our study was to achieve comparability with other studies in the literature. Therefore, we took care to select a dataset with the same time scale as the datasets used in other comparable studies in the literature. This is crucial because the performance of forecasting systems is significantly affected by the characteristics of the dataset used, as well as the parameter values defined for the models. This is particularly sensitive for studies that employ the holdout method for cross-validation, as is the case in our study. Altering the time scale of the dataset in our study would also change the training and test set partitioning, which would subsequently weaken the comparability of our study with other studies in the literature. For these reasons, we maintained the scale of the dataset in the experimental studies conducted in the previous section consistent with the studies of Sethia and Raut (2019) and Thakkar and Chaudhari (2021), thereby achieving one of our objectives.

In this section of our study, we continue our experiments by expanding the datasets. Firstly, we update our dataset to the most recent version and extend the time scale of the dataset until 2024 to encompass the fluctuations in financial markets caused by the COVID-19 pandemic. As a reminder, in the previous section, experiments were conducted with a parameter value of 12 for the number of components for dimensionality reduction and feature extraction, following the recommendation in the study by Sethia and Raut (2019) for direct comparability (See Table 2). Due to the change in the time scale of our dataset and consequently

Table 12 The descriptions of the expanded new S&P 500 dataset

Index	Open	High	Low	Close	Volume
Count	6060.0	6060.0	6060.0	6060.0	6060.0
Mean	198.363	199.549	197.077	198.381	108,031,146.963
Std	106.902	107.444	106.335	106.943	92,049,014.022
Min	67.949	70.0	67.099	68.110	1,436,600.0
Max	490.559	496.049	490.109	494.350	871,026,300.0

its acquisition of new dynamics, we are conducting some trials to determine the optimal number of components to be used in dimensionality reduction and feature extraction operations. To maintain the scope of the study during these trials, we are using the parameter list exactly as utilized in the studies by Sethia and Raut (2019) and Thakkar and Chaudhari (2021). Subsequently, to demonstrate the effectiveness of our method, we repeat our previous experiments by adjusting the parameter for the optimal new number of components calculated for our updated S&P 500 dataset, which now includes the pandemic period. Finally, to further validate the effectiveness of our method, we include additional case studies of selected stocks from the S&P 500 index as a comparative point in our research.

4.4.1 The Descriptions of the Expanded New S&P 500 Dataset

In the previous section, the dataset used in our experiments spanned 18 years, covering daily data from January 1, 2000, to October 23, 2017. By updating our study, we aim to demonstrate the impact of the fluctuating trajectory experienced in financial markets due to the worldwide COVID-19 pandemic on the performance of our proposed model. To achieve this objective, we extend our S&P 500 dataset to encompass the period from January 1, 2000, to January 25, 2024, and once again utilize the Yahoo Finance website to obtain this new dataset (SPY, 2024). The obtained new dataset comprises 6060 data points, and the dataset descriptions are presented in Table 12.

We apply the same processing steps to our new dataset as in the initial experiments. First, we calculate the technical indicators listed in Table 4 and

Table 13 Splitting the new S&P 500 dataset into training, validation, and test sets using the holdout method

Dataset split name	Date start	Date finish	Dataset split rate	Dataset sample size
Training set	2000.03.07	2016.11.22	0.70	4207
Validation set	2016.11.23	2019.04.16	0.10	601
Test set	2019.04.17	2024.01.25	0.20	1202

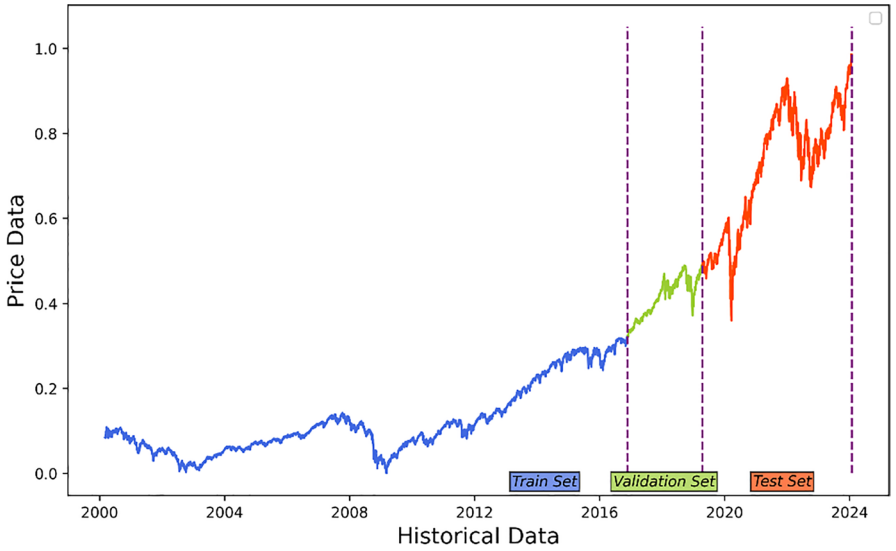


Fig. 13 The graphical representation of the expanded new S&P 500 dataset

incorporate them into the dataset, resulting in a dataset with 48 features in total. Subsequently, we perform standardization and normalization using Eq. (1).

We partition our dataset into training, validation, and test sets using the holdout method for cross-validation. To accomplish this, we employ the training (70%), validation (10%), and test (20%) ratios previously established by Sethia and Raut (2019). Initially, we utilize a two-month time frame covering January and February at the beginning of the dataset for calculating technical indicators. Accordingly, the new S&P 500 dataset is segmented into 4207 data points for training, 601 data points for validation, and 1202 data points for testing. Information regarding the splitting of the dataset into training, validation, and test sets using the holdout method is presented in Table 13, while the overall overview of the dataset after splitting is depicted in Fig. 13.

4.4.2 Experimental Setup

We conduct our experiments on the new dataset in two steps: in the first step, we search for the optimal number of components for dimensionality reduction and feature extraction for this dataset. In the subsequent step, we repeat the experiments from the previous section for the optimal number of components on the new dataset. To accomplish this, we utilize the model and parameter adjustments outlined in Table 2 of the 3.1 Baseline Settings section. Each experiment is executed 10 times, and we compute the average values of the obtained results for a fair comparison.

4.4.3 Investigating the Optimal Number of Components for Dimension Reduction and Feature Extraction on the Extended S&P 500 Dataset

In the preprocessing stage for the model proposed in our study, PCA and ICA statistical methods are sequentially employed. We define these statistical methods in the preprocessing stage because the final dataset that undergoes these two methods and is input into the prediction model differs significantly from the initial dataset at the beginning of the study. This difference arises from both methods producing entirely different new components as output from the datasets used as input. Previously, we noted that the critical parameter for both PCA and ICA methods is the number of components. We utilize the number of components as a parameter for PCA. The number of components for PCA determines the principal components present in a dataset. By using this parameter, we both reduce the dimensionality of our dataset and control the number of components to be used in the subsequent ICA method. In their study, Sethia and Raut (2019) expressed using different numbers of features (components) such as 7, 12, 18, 25, 32, and 45 for dimensionality reduction in the dataset covering the years 2000–2017, as stated in Sect. 3.1: Baseline Settings. Now, we apply the same experiments for the model proposed in our study on the S&P 500 index dataset covering the years 2000–2024, aiming to investigate the most suitable number of dimension reduction and feature extraction components for the updated dataset. We conduct this intending to achieve the lowest error rate and highest return rate for predicting the closing price of the next day with a 5 day forecasting horizon. To maintain the integrity of our study, we conduct experiments for the specified values in Table 2 without making any changes to the model structure and deep learning hyper-parameters.

For dimensionality reduction and feature extraction, 10 trials were conducted for the values of 7, 12, 18, 25, 32, and 45, and the average error and return ratios obtained with these values are present in Table 14. According to this, the top three numbers of features (components) with the lowest error rates are 45, 18, and 32, respectively.

As mentioned when outlining the objectives of our study, many studies in the literature focus on the accuracy and error rates of models. However, the primary goal of developing forecasting systems in financial markets, which is to provide

Table 14 Investigation of the optimum number of components for dimensionality reduction and feature extraction

Attributes number	Performance metrics			
	Return ratio	MSE	MAE	MAPE
7	2.807169	0.004039	0.051699	0.071063
12	−0.672271	0.005827	0.057276	0.080112
18	**3.069398**	0.003739	0.046944	0.064930
25	−2.617254	0.004998	0.052744	0.072964
32	−2.224981	0.003781	0.050644	0.070564
45	−1.904098	**0.002343**	**0.035966**	**0.050986**

Table 15 The performance of prediction models for the new S&P 500 dataset

Models	Performance metrics				
	Return ratio	R^2	MSE	MAE	MAPE
RNN	−0.387600	**0.957758**	**0.000887**	0.022616	0.03304501
LSTM	−1.010822	0.941867	0.001221	0.026137	0.03790223
GRU	−1.837704	0.853020	0.003087	0.044390	0.05953948
CNN	−1.627166	0.956025	0.000923	0.023232	0.03398207
PCA-ICA-LSTM	**2.216798**	0.951818	0.001012	**0.022374**	**0.03221122**

Table 16 Complexity analysis of the proposed method versus other deep learning models

Models	Complexity analysis		
	Params	Training(s)	Testing(s)
RNN	328,193	190.570	0.617
LSTM	424,193	301.941	0.647
GRU	392,769	249.021	0.631
CNN	353,025	139.803	0.342
PCA-ICA-LSTM	416,257	289.035	0.652

high returns and reduce potential risks, is not given much attention. In our study, we particularly focus on this issue and calculate the return rate of our model using a simple trading system, as will be recalled from previous experiments. As can be observed in Table 13, profitable transactions were only achievable for the models with 7 and 18 features, compared to the "hold and wait" strategy, across the experiments conducted for 6 different feature counts. However, we have stated above that the model with 45 features achieved the best error rates. This situation also reveals the weakness of studies that focus solely on accuracy or low error rates in real-world market conditions. We identify the optimal number of features (components) that enable our efforts to create a model that can provide high returns or reduce potential risks as 18, 7, and 12, respectively.

Upon examination of the lowest error rate list (in order: 45, 18, 32) and the highest return rate list (in order: 18, 7, 12) obtained from our experiments, we observe that there is only one feature (component) count value present in both lists, and we report this value to be 18. In light of these findings, we determine the optimum component (feature) count for use in dimensionality reduction and feature extraction to be 18, and we utilize this parameter setting in our subsequent experiments.

4.4.4 Predictive Performance of Models for the Expanded New S&P 500 Dataset

Having determined the optimal dimensionality reduction and feature extraction feature (component) count to be 18 for our S&P 500 dataset extending to 2024, we proceed to our comparative experiments for the new dataset in this section. Similar to the previous section, we compare the performance of our proposed PCA-ICA-LSTM

model with RNN, LSTM, GRU, and CNN models. For this purpose, we establish the model and parameter settings provided in Table 2 of Sect. 3.1 Baseline Settings, except for setting the feature extraction feature (component) count parameter value to 18 and adjusting the dataset training-test split to that of Table 13.

The average prediction performances of the models, based on the obtained experimental results, are presented in Table 15, with the best results highlighted in bold**.** Additionally, the number of trainable parameters and the average training and test times of the models for complexity analysis of the models in the comparison are provided in Table 16.

In the previous experiment, we showed that the optimal value for the dimensionality reduction and feature extraction component parameter for the expanded new dataset was 18. When conducting our new experiments according to this value, we observe that, with very slight differences, the highest R^2 values are obtained by the RNN, CNN, and PCA-ICA-LSTM models, respectively. It is also noted that the models achieving the lowest MSE values are again in the same order and with the same model names. We remark that the GRU model exhibits the worst performance for both R^2 and MSE evaluation metrics. Observing slight variations, we find that the ranking changes for the lowest MAE and MAPE values obtained from the experiments. Accordingly, for these two evaluation metrics, the best results are achieved by the PCA-ICA-LSTM, RNN, and CNN models, respectively.

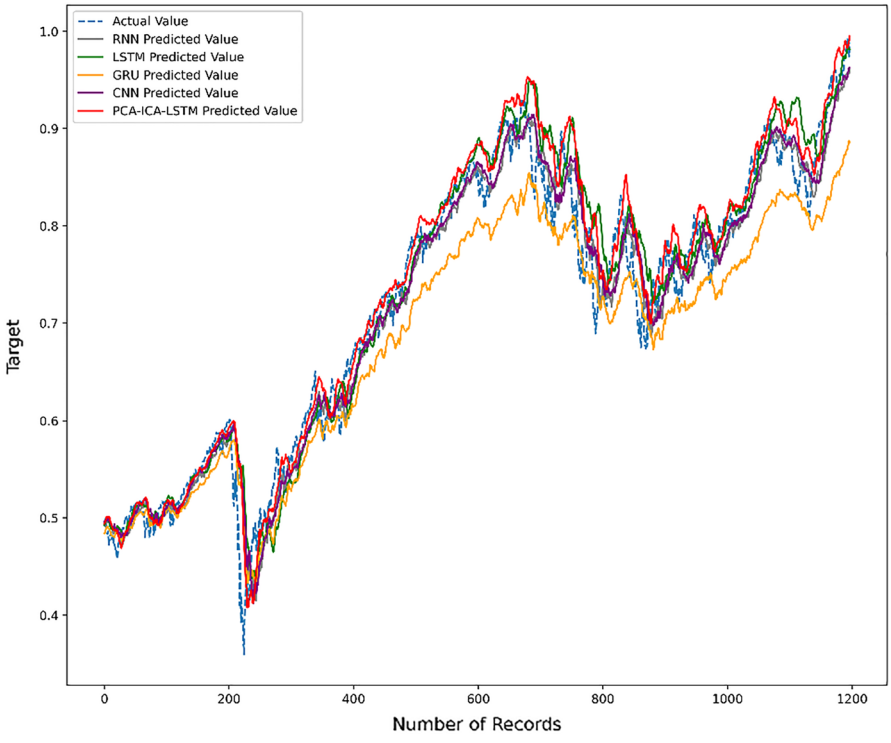


Fig. 14 Actual values of the extended new S&P 500 dataset and predicted values of all models

As emphasized in the previous section, we highlight that the primary goal of developing forecasting systems in financial markets should be to provide high returns and mitigate potential risks. When we examine our return ratio evaluation metric that helps us achieve this goal, we observe that 4 out of 5 models could not provide positive returns for this dataset according to the "hold and wait" strategy. Our results demonstrate that only our proposed model was able to provide a positive return according to the "hold and wait" strategy. We report that our proposed PCA-ICA-LSTM model achieved a 220% better return rate than the "hold and wait" trading strategy and 260% better than the RNN model, which achieved the closest result to our model in the comparison.

The comparative analysis of the actual value of the expanded new S&P 500 Index and the prediction values of all models is presented in Fig. 14. It is evident from Fig. 14 that the GRU model exhibits the poorest performance in the comparison, while the performance of the RNN model in the new S&P 500 dataset, which includes the COVID-19 pandemic period, is noteworthy. Additionally, we can state that our proposed PCA-ICA-LSTM model demonstrates more successful results in the expanded S&P 500 new dataset, which encompasses the period of significant market fluctuations caused by the COVID-19 pandemic, compared to the initial S&P 500 dataset covering the years 2000–2017.

4.4.5 Additional Case Studies Conducted to Demonstrate the Effectiveness of the Proposed Model

In this section of our study, we aim to further validate the effectiveness of our proposed method by incorporating additional case studies into our research. Since the focus of our study, as the title suggests, is the S&P 500 index, we did not consider focusing on different world indices. Instead, we decided that selecting stocks from the S&P 500 index would be more suitable for the additional case study. For this purpose, we identified two stocks that existed between 2000 and 2024 in the S&P 500 index.

Table 17 Dataset descriptions of the additional case studies included in the study

Dataset	Index	Open	High	Low	Close	Volume
LLY	Count	6060.0	6060.0	6060.0	6060.0	6060.0
	Mean	102.737	103.862	101.643	102.798	4,799,281.369
	Std	105.077	106.312	103.841	105.159	3,722,030.965
	Min	27.65	28.23	27.209	27.469	494,400.0
	Max	662.380	672.619	659.739	667.650	74,822,500.0
NVDA	Count	6060.0	6060.0	6060.0	6060.0	6060.0
	Mean	45.841	46.674	44.992	45.879	62,287,660.0
	Std	95.418	97.080	93.732	95.529	43,337,112.469
	Min	0.608	0.656	0.6	0.614	4,564,400.0
	Max	639.739	666.0	636.900	661.599	923,085,600.0

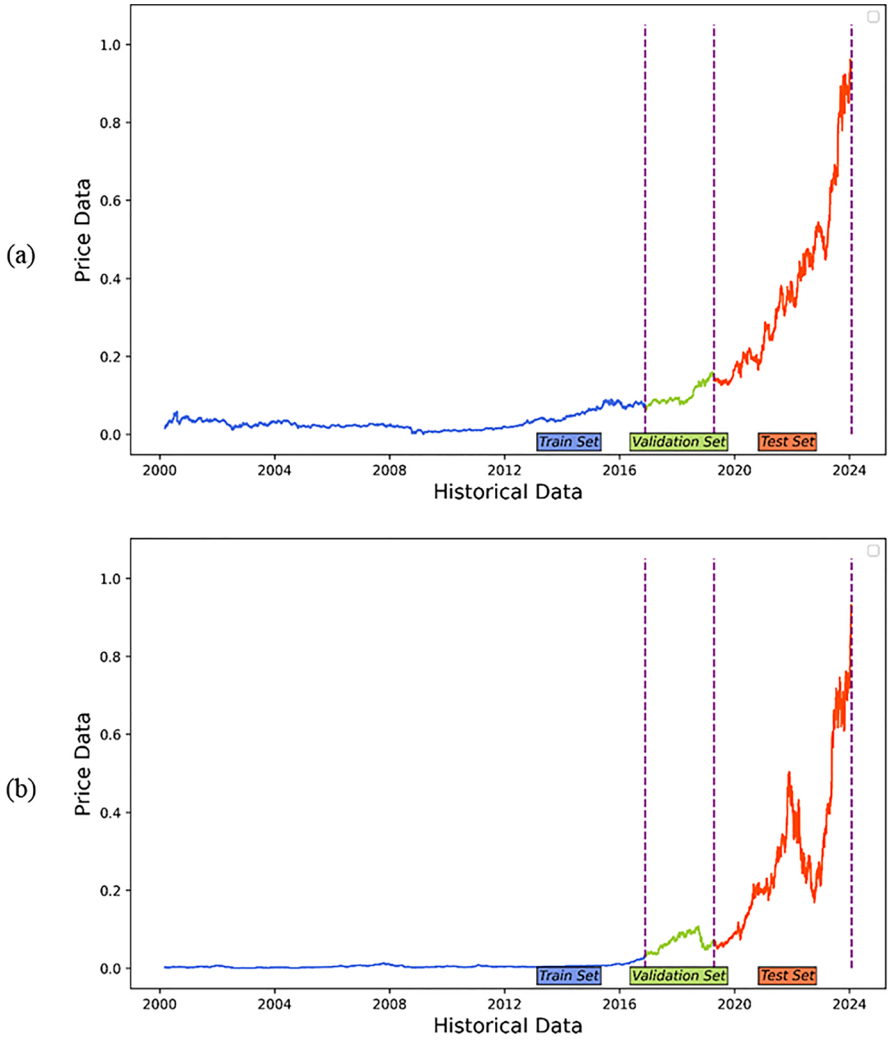


Fig. 15 Graphical representation of the datasets identified in the additional case study: **a** LLY stock dataset, **b** NVDA stock dataset

Table 18 Comparison of the performance of prediction models for the LLY stock identified in the additional case study

Models	Return Ratio	R^2	MSE	MAE	MAPE
RNN	−0.581181	0.981255	0.000920	0.020436	0.052011
LSTM	−0.376686	0.949153	0.002496	0.031663	0.072148
GRU	−4.366575	0.718864	0.013800	0.073957	0.142639
CNN	−0.567346	0.981674	0.000899	0.020313	0.051495
PCA-ICA-LSTM	**2.363911**	**0.986615**	**0.000657**	**0.017844**	**0.047265**

In the selection of these stocks, the top 10 companies in the S&P 500 index by market capitalization were examined. It was observed that these companies belonged to 4 different sectors (technology, retail, finance, and healthcare) (SPX, 2024). Companies that were not active between 2000 and 2024 were excluded. Furthermore, the world was affected by the COVID-19 pandemic from the end of 2019 to approximately the beginning of 2023. Considering the pandemic period, companies from the healthcare and technology sectors among the four sectors mentioned above were included in our study as a benchmark for the additional case study. The companies we included as an additional case study were identified as Eli Lilly and Company (LLY) and NVIDIA Corporation (NVDA).

The datasets for the respective companies were prepared by applying the processing steps outlined in Sect. 4.4.1. Accordingly, the dataset descriptions for the relevant companies are presented in Table 17. The datasets for both stocks were divided into training, validation, and test sets using the hold-out method, utilizing the values in Table 13 for cross-validation. The data for the datasets was obtained from the Yahoo Finance website, and the graphical representations of both datasets are presented in Fig. 15 (LLY, 2024; NVDA, 2024).

When examining the graphs of both stocks, it is observed that the overall trend is upward, particularly during the COVID-19 pandemic period. While the LLY dataset exhibits volatile short-term fluctuations, the general trend remains upward. On the other hand, the NVDA dataset shows volatile long-term fluctuations, eventually continuing its overall upward trend. The two datasets demonstrate different dynamics in terms of the fluctuations they experience.

Table 18 presents the prediction results of the models for the LLY stock identified in the additional case study. The best results for the evaluation metrics are shown in bold in the table. Accordingly, we observe that the proposed PCA-ICA-LSTM model provides the highest R^2 of 0.986615, while the GRU model provides the lowest R^2 of 0.718864 among the compared models for the LLY stock dataset. In addition, the PCA-ICA-LSTM model stands out as the most successful model by obtaining the lowest MSE, MAE, and MAPE values, while the GRU model stands out as the most unsuccessful model among the compared models with the highest MSE, MAE, and MAPE values. Furthermore, the PCA-ICA-LSTM model, while predicting the next day's price with a 5-day prediction horizon for the LLY stock, provides a return of over 230% compared to the "hold and wait" strategy, while the other models in the comparison achieve returns below the "hold and wait" strategy. Our proposed model has achieved a return rate of approximately 275% better than the LSTM model, which provided the closest return to it in the comparisons. In this way, the proposed model stands out significantly from the other compared models. The actual values of the LLY stock and the predicted values of all models are shown comparatively in Fig. 16a.

Table 19 presents the prediction results of the models for the NVDA stock, which is the other additional case study within the scope of the study. The best results for the evaluation metrics are shown in bold in the table. Similar comparison results were obtained for the NVDA stock as in the previous additional case study for the LLY stock. Accordingly, Table 19 shows that the proposed PCA-ICA-LSTM model achieved the highest R^2 value of 0.958137, while the GRU model achieved the lowest

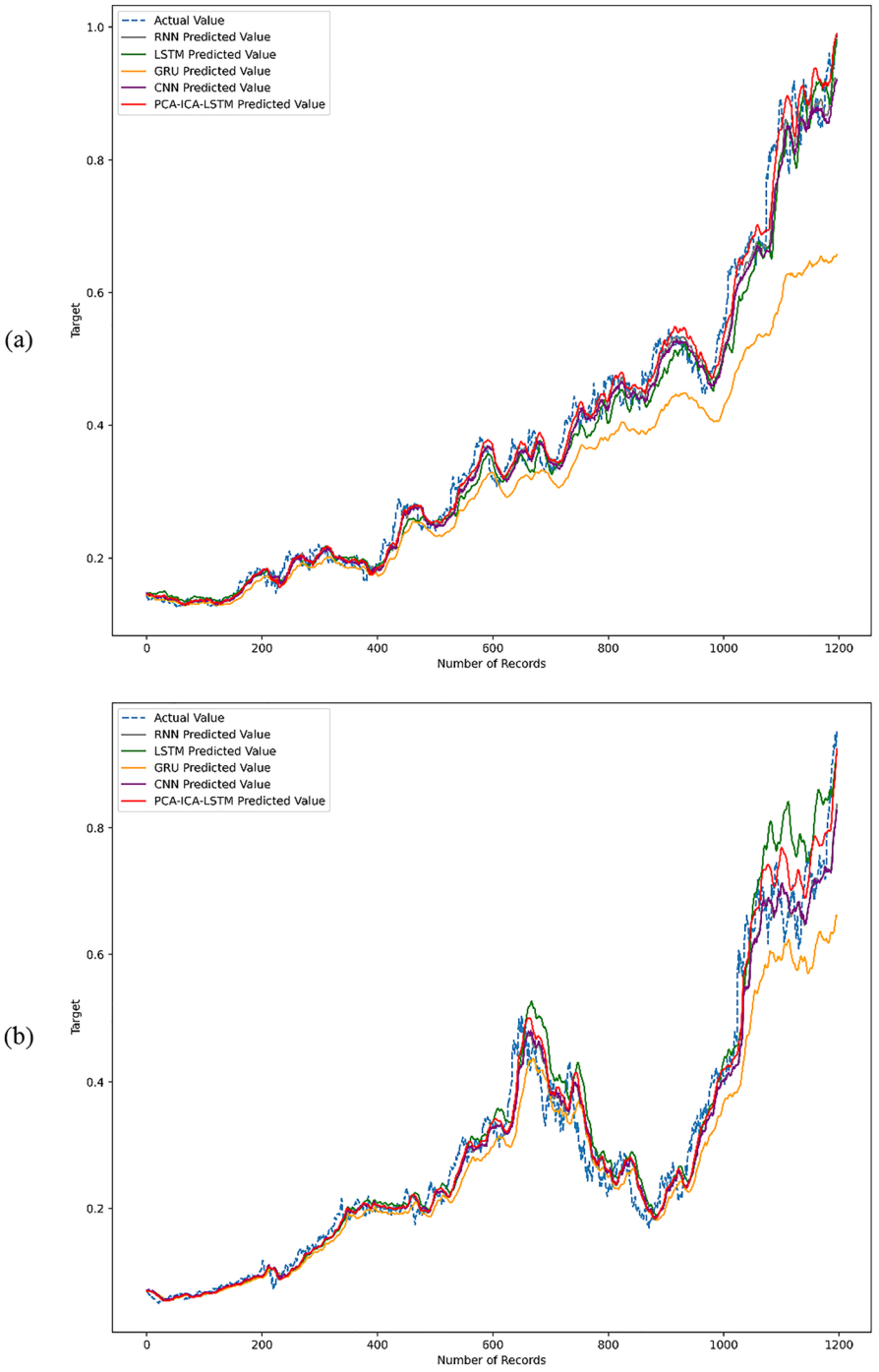


Fig. 16 Actual values of the stocks identified in the additional case study and predicted values of all models: **a** LLY stock dataset, **b** NVDA stock dataset

Table 19 Comparison of the performance of prediction models for the NVDA stock identified in the additional case study

Models	Return Ratio	R^2	MSE	MAE	MAPE
RNN	−3.991306	0.945346	0.002199	0.031746	0.105060
LSTM	−1.090105	0.940593	0.002390	0.031889	0.103503
GRU	−4.145328	0.871512	0.005170	0.043983	0.120123
CNN	−3.723013	0.955421	0.001793	0.028173	0.092280
PCA-ICA-LSTM	**−0.598597**	**0.958137**	**0.001684**	**0.026500**	**0.086708**

R^2 value of 0.871512 among the compared models for the NVDA stock dataset. In the same comparison, the PCA-ICA-LSTM achieved the lowest MSE, MAE, and MAPE values, becoming the most successful model, while the GRU model was the least successful model with the highest error values. Unlike the previous additional case study, the proposed PCA-ICA-LSTM model, while predicting the price of the NVDA stock 5 days later, provided a return rate 60% lower than the "hold and wait" strategy in the return ratio metric. Although our proposed model lags behind the "hold and wait" strategy in the return rate metric, it achieved a return rate 50% better than the LSTM model, which provided the highest return among the compared models. The actual values of the NVDA stock and the predicted values of all models are shown comparatively in Fig. 16b.

5 Conclusions

This study proposes a novel deep learning model, PCA-ICA-LSTM, for predicting financial asset prices. The model incorporates a two-stage preprocessing procedure utilizing PCA and ICA statistical methods to enhance prediction accuracy. A five-layer LSTM network utilizes preprocessed data to predict the price of the next day using a 5 day time horizon The research addresses two primary objectives: firstly, to establish comparability with existing literature by employing a standardized dataset and evaluation metrics; and secondly, to prioritize risk mitigation and return generation in financial forecasting. For this purpose, the study utilized an 18-year dataset of the S&P 500 spanning from 2000 to 2017, incorporating over 40 technical indicators. Evaluation of the model was conducted using six criteria, including a simple trading strategy as well as R^2-Score, MSE, MAE, MAPE, Maximum Error, and Return Ratio.

Comparative analyses demonstrate PCA-ICA-LSTM's outperformance against single-stage statistical methods and prevalent deep-learning models like RNN, GRU, LSTM, and CNN. While PCA-ICA-LSTM achieved the highest accuracy metrics, the CNN model obtained the highest return rate. The CNN model provided the highest return rate, surpassing the "hold and wait" strategy by 270%, whereas the PCA-ICA-LSTM model achieved a competitive result by yielding a 165% return rate compared to the "hold and wait" strategy. Remarkably, the fact that our model with the highest accuracy and lowest error rates lags behind other models in terms

of return rate underlines the second objective of our study. Moreover, comparisons with prior studies reveal significant improvements in MSE and MAPE metrics. The performance of the PCA-ICA-LSTM model was investigated over an extended time scale of 2000–2024, including the COVID-19 pandemic, to test its robustness under varying market conditions. Experiments determined the optimal number of dimension reduction components for the new dataset to be 18. The PCA-ICA-LSTM model demonstrated strong performance, achieving a return rate 220% higher than the "hold and wait" strategy and 260% higher than the closest competitor, the RNN model, on the extended S&P 500 data. In supplementary case studies involving LLY and NVDA stocks, the PCA-ICA-LSTM model outperformed other models, demonstrating its continued success. The proposed model achieved a return rate of approximately 275% and 50% higher than the LSTM model, which provided the closest return in the LLY and NVDA dataset comparisons, respectively.

This study offers a PCA-ICA-LSTM model that achieves high accuracy, low error rates, and competitive return rates. The study contributes to advancing predictive accuracy and return rates in financial forecasting, offering valuable insights for researchers and practitioners alike. Future studies could explore optimizing the LSTM network and applying the model to various datasets and deep learning architectures. Additionally, developing trade mechanisms for improved return rates and investigating hybrid forecasting models are promising areas for further research.

Acknowledgements The authors thank Proof Reading & Editing Office of Erciyes University for their proofreading support.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK).

Availability of Data and Materials The datasets generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Code Availability Custom code will be available upon request from the corresponding author.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

Abu-Mostafa, Y. S., & Atiya, A. F. (1996). Introduction to financial forecasting. *Applied Intelligence, 6*(3), 205–213. https://doi.org/10.1007/BF00126626

Anowar, F., Sadaoui, S., & Selim, B. (2021). Conceptual and empirical comparison of dimensionality reduction algorithms (PCA, KPCA, LDA, MDS, SVD, LLE, ISOMAP, LE, ICA, t-SNE). *Computer Science Review, 40*, 100378. https://doi.org/10.1016/j.cosrev.2021.100378

Atsalakis, G. S., & Valavanis, K. P. (2009). Surveying stock market forecasting techniques–Part II: Soft computing methods. *Expert Systems with Applications, 36*(3), 5932–5941. https://doi.org/10.1016/j.eswa.2008.07.006

Berradi, Z., & Lazaar, M. (2019). Integration of principal component analysis and recurrent neural network to forecast the stock price of casablanca stock exchange. *Procedia Computer Science, 148*, 55–61. https://doi.org/10.1016/j.procs.2019.01.008

Bustos, O., & Pomares-Quimbaya, A. (2020). Stock market movement forecast: A systematic review. *Expert Systems with Applications, 156*, 113464. https://doi.org/10.1016/j.eswa.2020.113464

Cavalcante, R. C., Brasileiro, R. C., Souza, V. L. F., Nobrega, J. P., & Oliveira, A. L. I. (2016). Computational intelligence and financial markets: A survey and future directions. *Expert Systems with Applications, 55*, 194–211. https://doi.org/10.1016/j.eswa.2016.02.006

Celik, M., Çelik, F. D., & Dokuz, A. S. (2014). Discovery of hydrometeorological patterns. *Turkish Journal of Electrical Engineering and Computer Sciences, 22*(4), 840–857. https://doi.org/10.3906/elk-1210-20

Chen, X., & Hu, Y. (2022). Volatility forecasts of stock index futures in China and the US–A hybrid LSTM approach. *PLoS ONE, 17*(7), e0271595.

Chen, Y., Zhao, P., Zhang, Z., Bai, J., & Guo, Y. (2022). A stock price forecasting model integrating complementary ensemble empirical mode decomposition and independent component analysis. *International Journal of Computational Intelligence Systems, 15*(1), 75. https://doi.org/10.1007/s44196-022-00140-2

Chialvo, D. R., & Jalife, J. (1987). Non-linear dynamics of cardiac excitation and impulse propagation. *Nature, 330*(6150), 749–752. https://doi.org/10.1038/330749a0

Chowdhury, U. N., Chakravarty, S. K., & Hossain, M. T. (2018). Short-term financial time series forecasting integrating principal component analysis and independent component analysis with support vector regression. *Journal of Computer and Communications, 6*(03), 51. https://doi.org/10.4236/jcc.2018.63004

Deng, S., Huang, X., Zhu, Y., Su, Z., Fu, Z., & Shimada, T. (2023). Stock index direction forecasting using an explainable extreme gradient boosting and investor sentiments. *The North American Journal of Economics and Finance, 64*, 101848. https://doi.org/10.1016/j.najef.2022.101848

Deng, S., Zhu, Y., Yu, Y., & Huang, X. (2024). An integrated approach of ensemble learning methods for stock index prediction using investor sentiments. *Expert Systems with Applications, 238*, 121710. https://doi.org/10.1016/j.eswa.2023.121710

Di Persio, L., & Honchar, O. (2016). Artificial neural networks architectures for stock price prediction: Comparisons and applications. *International Journal of Circuits, Systems and Signal Processing, 10*(2016), 403–413.

Draper, B. A., Baek, K., Bartlett, M. S., & Beveridge, J. R. (2003). Recognizing faces with PCA and ICA. *Computer Vision and Image Understanding, 91*(1), 115–137. https://doi.org/10.1016/S1077-3142(03)00077-8

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). The KDD process for extracting useful knowledge from volumes of data. *Communications of the ACM, 39*(11), 27–34.

Fraedrich, K. (1986). Estimating the dimensions of weather and climate attractors. *Journal of Atmospheric Sciences, 43*(5), 419–432. https://doi.org/10.1175/1520-0469(1986)043%3c0419:ETDOWA%3e2.0.CO;2

Furey, E. (2023, September 19). *Z Score Calculator*. CalculatorSoup. Retrieved March 18, 2024, from https://www.calculatorsoup.com/calculators/statistics/zscore-calculator.php.

Gandhmal, D. P., & Kumar, K. (2019). Systematic analysis and review of stock market prediction techniques. *Computer Science Review, 34*, 100190. https://doi.org/10.1016/j.cosrev.2019.08.001

Gao, T., Chai, Y., & Liu, Y. (2017). Applying long short term momory neural networks for predicting stock closing price. In: 2017 8th IEEE international conference on software engineering and service science (ICSESS).

Gao, T., & Chai, Y. (2018). Improving stock closing price prediction using recurrent neural network and technical indicators. *Neural Computation, 30*(10), 2833–2854.

Gao, Y., Wang, R., & Zhou, E. (2021). Stock prediction based on optimized LSTM and GRU models. *Scientific Programming*. https://doi.org/10.1155/2021/4055281

Goldberger, A., Rigney, D., Mietus, J., Antman, E., & Greenwald, S. (1988). Nonlinear dynamics in sudden cardiac death syndrome: Heartrate oscillations and bifurcations. *Experientia, 44*(11), 983–987. https://doi.org/10.1007/BF01939894

Gudelek, M. U., Boluk, S. A., & Ozbayoglu, A. M. (2017). A deep learning based stock trading model with 2-D CNN trend detection. In: 2017 IEEE symposium series on computational intelligence (SSCI).

Guo, Y., He, F., Liang, C., & Ma, F. (2022). Oil price volatility predictability: New evidence from a scaled PCA approach. *Energy Economics, 105*, 105714. https://doi.org/10.1016/j.eneco.2021.105714

He, H., & Dai, S. (2022). A prediction model for stock market based on the integration of independent component analysis and multi-LSTM. *Electronic Research Archive, 30*(10), 3855–3871.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hossain, M. A., Karim, R., Thulasiram, R., Bruce, N. D. B., & Wang, Y. (2018). Hybrid deep learning model for stock price prediction. In: 2018 IEEE symposium series on computational intelligence (SSCI).

Hsieh, D. A. (1991). Chaos and nonlinear dynamics: Application to financial markets. *The Journal of Finance, 46*(5), 1839–1877. https://doi.org/10.1111/j.1540-6261.1991.tb04646.x

Huang, D., Jiang, F., Li, K., Tong, G., & Zhou, G. (2022). Scaled PCA: A new approach to dimension reduction. *Management Science, 68*(3), 1678–1695.

Huang, X., Zanni-Merk, C., & Crémilleux, B. (2019). Enhancing deep learning with semantics: An application to manufacturing time series analysis. *Procedia Computer Science, 159*, 437–446. https://doi.org/10.1016/j.procs.2019.09.198

Jianwei, E., Ye, J., & Jin, H. (2019). A novel hybrid model on the prediction of time series and its application for the gold price analysis and forecasting. *Physica a: Statistical Mechanics and Its Applications, 527*, 121454. https://doi.org/10.1016/j.physa.2019.121454

Kakade, K. A., Ghate, K. S., Jaiswal, R. K., & Jaiswal, R. (2023). A novel approach to forecast crude oil prices using machine learning and technical indicators. *Journal of Advances in Information Technology, 14*(2), 302.

Kao, L.-J., Chiu, C.-C., Lu, C.-J., & Yang, J.-L. (2013). Integration of nonlinear independent component analysis and support vector regression for stock price forecasting. *Neurocomputing, 99*, 534–542. https://doi.org/10.1016/j.neucom.2012.06.037

Kauffman, R. J., Liu, J., & Ma, D. (2015). Technology investment decision-making under uncertainty. *Information Technology and Management, 16*(2), 153–172. https://doi.org/10.1007/s10799-014-0212-2

Kim, K.-J. (2003). Financial time series forecasting using support vector machines. *Neurocomputing, 55*(1–2), 307–319. https://doi.org/10.1016/S0925-2312(03)00372-2

Kwak, N. (2008). Feature extraction for classification problems and its application to face recognition. *Pattern Recognition, 41*(5), 1701–1717. https://doi.org/10.1016/j.patcog.2007.10.012

Kwon, Y.-K., & Moon, B.-R. (2007). A hybrid neurogenetic approach for stock forecasting. *IEEE Transactions on Neural Networks, 18*(3), 851–864. https://doi.org/10.1109/TNN.2007.891629

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444. https://doi.org/10.1038/nature14539

Li, A. W., & Bastos, G. S. (2020). Stock market forecasting using deep learning and technical analysis: A systematic review. *IEEE Access, 8*, 185232–185242. https://doi.org/10.1109/ACCESS.2020.3030226

Li, H., Zhou, D., Hu, J., Li, J., Su, M., & Guo, L. (2023). Forecasting the realized volatility of energy stock market: A multimodel comparison. *The North American Journal of Economics and Finance, 66*, 101895. https://doi.org/10.1016/j.najef.2023.101895

Li, J., Zhou, T., & Hu, X. (2022). Prediction algorithm of stock holdings of hong kong-funded institutions based on optimized PCA-LSTM model. *International Journal of Innovative Computing Information, 18*, 999–1008.

Liu, H., & Wang, J. (2011). Integrating independent component analysis and principal component analysis with neural network to predict chinese stock market. *Mathematical Problems in Engineering, 2011*, 382659. https://doi.org/10.1155/2011/382659

Liu, Z., Zhu, Z., Gao, J., & Xu, C. (2021). Forecast methods for time series data: A survey. *IEEE Access, 9*, 91896–91912. https://doi.org/10.1109/ACCESS.2021.3091162

LLY. (2024). *Eli lilly and company, historical data*. YahooFinance. Retrieved [27.02.2024] from https://finance.yahoo.com/quote/LLY/history.

Long, W., Lu, Z., & Cui, L. (2019). Deep learning-based feature engineering for stock price movement prediction. *Knowledge-Based Systems, 164*, 163–173. https://doi.org/10.1016/j.knosys.2018.10.034

Lu, C.-J., Lee, T.-S., & Chiu, C.-C. (2009). Financial time series forecasting using independent component analysis and support vector regression. *Decision Support Systems, 47*(2), 115–125. https://doi.org/10.1016/j.dss.2009.02.001

Ma, Y., Han, R., & Fu, X. (2019). Stock prediction based on random forest and LSTM neural network. In: 2019 19th international conference on control, automation and systems (ICCAS).

Mehdiyev, N., Lahann, J., Emrich, A., Enke, D., Fettke, P., & Loos, P. (2017). Time series classification using deep learning for process planning: A case from the process industry. *Procedia Computer Science, 114*, 242–249. https://doi.org/10.1016/j.procs.2017.09.066

Mendoza, C., Kristjanpoller, W., & Minutolo, M. C. (2023). Market index price prediction using deep neural networks with a self-similarity approach. *Applied Soft Computing, 146*, 110700. https://doi.org/10.1016/j.asoc.2023.110700

Montaño Moreno, J. J., Palmer Pol, A. L., Sesé Abad, A. J., & Cajal Blasco, B. (2013). Using the R-MAPE index as a resistant measure of forecast accuracy. *Psicothema*. https://doi.org/10.7334/psicothema2013.23

Nicolis, C., & Nicolis, G. (1984). Is there a climatic attractor? *Nature, 311*(5986), 529–532. https://doi.org/10.1038/311529a0

Nosratabadi, S., Mosavi, A., Duan, P., Ghamisi, P., Filip, F., Band, S. S., & Gandomi, A. H. (2020). Data science in economics: comprehensive review of advanced machine learning and deep learning methods. *Mathematics, 8*(10), 1799. https://doi.org/10.3390/math8101799

NVDA. (2024). *NVIDIA corporation, historical data*. YahooFinance. Retrieved 27 Feb 2024 from https://finance.yahoo.com/quote/NVDA/history.

Ozbayoglu, A. M., Gudelek, M. U., & Sezer, O. B. (2020). Deep learning for financial applications: A survey. *Applied Soft Computing, 93*, 106384. https://doi.org/10.1016/j.asoc.2020.106384

Özdoğan-Sarıkoç, G., Sarıkoç, M., Celik, M., & Dadaser-Celik, F. (2023). Reservoir volume forecasting using artificial intelligence-based models: Artificial neural networks, support vector regression, and long short-term memory. *Journal of Hydrology, 616*, 128766. https://doi.org/10.1016/j.jhydrol.2022.128766

Ozkok, F. O., & Celik, M. (2022). A hybrid CNN-LSTM model for high resolution melting curve classification. *Biomedical Signal Processing and Control, 71*, 103168. https://doi.org/10.1016/j.bspc.2021.103168

Pai, P.-F., & Lin, C.-S. (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. *Omega, 33*(6), 497–505. https://doi.org/10.1016/j.omega.2004.07.024

Pearson, K. (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 2*(11), 559–572. https://doi.org/10.1080/14786440109462720

Peters, E. E. (1991). A chaotic attractor for the S&P 500. *Financial Analysts Journal, 47*(2), 55–62. https://doi.org/10.2469/faj.v47.n2.55

Reza, M. S., & Ma, J. (2016). ICA and PCA integrated feature extraction for classification. In: 2016 IEEE 13th international conference on signal processing (ICSP).

Rounaghi, M. M., & Zadeh, F. N. (2016). Investigation of market efficiency and financial stability between S&P 500 and London stock exchange: Monthly and yearly forecasting of time series stock returns using ARMA model. *Physica a: Statistical Mechanics and Its Applications, 456*, 10–21. https://doi.org/10.1016/j.physa.2016.03.006

Sarıkoç, M., & Çelik, M. (2022). BIST100 index price prediction with dimension reduction techniques and LSTM deep learning network. *European Journal of Science and Technology, 34*, 519–524. https://doi.org/10.31590/ejosat.1083255

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks, 61*, 85–117. https://doi.org/10.1016/j.neunet.2014.09.003

Sethia, A., & Raut, P. (2019). Application of LSTM GRU and ICA for stock price prediction. *Information and Communication Technology for Intelligent Systems*. https://doi.org/10.1007/978-981-13-1747-7_46

Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied Soft Computing, 90*, 106181. https://doi.org/10.1016/j.asoc.2020.106181

Shekhar, S., Vatsavai, R. R., & Celik, M. (2008). Spatial and spatiotemporal data mining: Recent advances. *Next Generation of Data Mining*. https://doi.org/10.1201/9781420085877.ch26

Singh, R., & Srivastava, S. (2017). Stock prediction using deep learning. *Multimedia Tools and Applications, 76*(18), 18569–18584. https://doi.org/10.1007/s11042-016-4159-7

SPX. (2024). *S&P 500 Components*. Tradingview. Retrieved 04 Feb 2024 from https://tr.tradingview.com/symbols/SPX/components/.

SPY. (2024). *SPDR S&P 500 ETF trust, historical data*. YahooFinance. Retrieved 27 Feb 2024 from https://finance.yahoo.com/quote/SPY/history.

Srijiranon, K., Lertratanakham, Y., & Tanantong, T. (2022). A hybrid framework using PCA, EMD and LSTM methods for stock market price prediction with sentiment analysis. *Applied Sciences, 12*(21), 10823.

Tay, F. E., & Cao, L. (2001). Application of support vector machines in financial time series forecasting. *Omega, 29*(4), 309–317. https://doi.org/10.1016/S0305-0483(01)00026-3

Teixeira, L. A., & De Oliveira, A. L. I. (2010). A method for automatic stock trading combining technical analysis and nearest neighbor classification. *Expert Systems with Applications, 37*(10), 6885–6890. https://doi.org/10.1016/j.eswa.2010.03.033

Thakkar, A., & Chaudhari, K. (2021). A comprehensive survey on deep neural networks for stock market: The need, challenges, and future directions. *Expert Systems with Applications, 177*, 114800. https://doi.org/10.1016/j.eswa.2021.114800

Tharwat, A. (2021). Independent component analysis: An introduction. *Applied Computing and Informatics, 17*(2), 222–249. https://doi.org/10.1016/j.aci.2018.08.006

Wang, J., Liu, D., Jin, L., Sun, Q., & Xue, Z. (2023). A PCA-IGRU model for stock price prediction. *Journal of Internet Technology, 24*(3), 621–629.

Wei, X., & Ouyang, H. (2024). Carbon price prediction based on a scaled PCA approach. *PLoS ONE, 19*(1), e0296105.

Wen, Y., Lin, P., & Nie, X. (2020). Research of stock price prediction based on PCA-LSTM model. In: IOP conference series: materials science and engineering.

Zare, A., Ozdemir, A., Iwen, M. A., & Aviyente, S. (2018). Extension of PCA to higher order data structures: An introduction to tensors, tensor decompositions, and tensor PCA. *Proceedings of the IEEE, 106*(8), 1341–1358. https://doi.org/10.1109/JPROC.2018.2848209

Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing, 50*, 159–175. https://doi.org/10.1016/S0925-2312(01)00702-0

Zhang, Y. A., Yan, B., & Aasma, M. (2020). A novel deep learning framework: Prediction and analysis of financial time series using CEEMD and LSTM. *Expert Systems with Applications, 159*, 113609. https://doi.org/10.1016/j.eswa.2020.113609

Zheng, L., & He, H. (2021). Share price prediction of aerospace relevant companies with recurrent neural networks based on pca. *Expert Systems with Applications, 183*, 115384.

Zhong, X., & Enke, D. (2017). A comprehensive cluster and classification mining procedure for daily stock market return forecasting. *Neurocomputing, 267*, 152–168. https://doi.org/10.1016/j.neucom.2017.06.010