Journal of Open Innovation: Technology, Market, and Complexity 10 (2024) 100398

Contents lists available at ScienceDirect

Journal of Open Innovation: Technology, Market, and Complexity

journal homepage: www.sciencedirect.com/journal/journal-of-open-innovation-technology-market-and-complexity

Technical indicator empowered intelligent strategies to predict stock trading signals

Arjun Singh Saud [a,*], Subarna Shakya [b]

[a] Central Department of Computer Science and Information Technology, IOST, Tribhuvan University, Nepal
[b] Department of Electronics and Computer Engineering, IOE, Tribhuvan University, Nepal

ARTICLE INFO

Keywords:
MACD
DMI
KST
LSTM
GRU
Intelligent stock trading strategies
Trading signal prediction

ABSTRACT

Technical analysis is widely employed in stock trading, relying on popular indicators such as MACD, DMI, KST etc. to predict stock trends. Despite their common use, these lagging indicators can occasionally generate misleading signals. In the literature, machine learning researchers developed many intelligent strategies for predicting stock trading signals using these indicators as inputs. However, significant differences exist in how these indicators are applied by technical analysts and machine learning experts. Building on this knowledge, this study developed intelligent stock trading signal prediction strategies using MACD, DMI, and KST indicators, and implemented these strategies with LSTM and GRU networks due to their ability to manage long-term dependencies and maintain context. The proposed intelligent trading strategies were assessed using ARR, SR, and win rate metrics, based on historical trading data from 18 stocks—six each from NEPSE, BSE, and NYSE—leading to four key insights. (1) For predicting stock trading signals, a 5-day lookback period is optimal for intelligent strategies based on MACD and DMI, while a 10-day period is best for the KST-based strategy. (2) Intelligent trading strategies implemented with GRU networks demonstrated superior performance compared to those implemented with LSTM. (3) The intelligent trading strategies based on MACD, DMI, and KST indicators outperform their peer classical stock trading methods. (4) Among the three proposed intelligent strategies, the MACD-based approach is found to be the safest and most effective.

1. Introduction

Stock market prediction involves forecasting future stock prices or trading signals. Researchers from finance, statistics, and machine learning work on developing predictive models. Financial experts use fundamental and technical analysis, statistical experts use time series analysis, and computer science researchers apply machine learning models for this purpose.

Fundamental analysts determine a stock's intrinsic value by analyzing the broader economy, specific sectors, industries, and individual companies. They review financial metrics like revenue, earnings per share, and price-to-earnings ratio to assess whether a stock is under or overvalued. This analysis focuses on long-term trends rather than pinpointing precise entry or exit points, making it less suitable for traders who need to make quick decisions (Baresa et al., 2013; Jakpar et al., 2018; Lam, 2004; Nti et al., 2019). On the contrary, technical analysts seek to forecast future stock prices by examining historical trading data with tools such as candlestick charts, trend and momentum indicators, volume indicators, and volatility indicators. They analyze data over different time frames, ranging from 5-minute intervals to monthly periods, to pinpoint optimal entry and exit points for trades. This method is particularly favored by short-term traders (Jakpar et al., 2018; Lam, 2004; Nti et al., 2019; Farias Nazário et al., 2017; Li and Bastos, 2020). While no single tool offers complete accuracy, traders often rely on a combination of indicators to bolster their confidence in making trading decisions.

Time series analysis predicts future values using models based on past data. In stock market forecasting, common tools include moving averages, autoregression, ARIMA family models, and exponential smoothing (Shah et al., 2019; Siami-Namini et al., 2018, 2019; Yamak et al., 2019). Analyzing large volumes of trading data, whether manually or through computer-assisted tools, is time-consuming, leading to the widespread use of machine learning models for automated and rapid stock market forecasting. Popular techniques include regression, decision trees, SVM, MLP, etc (Bustos and Pomares-Quimbaya, 2020; Kumar et al., 2021). Recently, deep learning models like LSTM, GRU, CNN, and

* Corresponding author.
E-mail address: arjunsaud@cdcsit.edu.np (A.S. Saud).

https://doi.org/10.1016/j.joitmc.2024.100398
Received 22 August 2024; Received in revised form 26 September 2024; Accepted 5 October 2024
Available online 9 October 2024

Echo State Networks (ESN) have gained more popularity over traditional methods (Nabipour et al., 2020; Nikou et al., 2019).

The research in stock market prediction can be broadly divided into three main areas: forecasting stock prices, predicting stock market indices, and anticipating stock trading signals. The third area, anticipating stock trading signals, is particularly important for traders as it directly influences their decisions to buy, hold, or sell, making it more relevant than the other two areas. Although trend and momentum indicators are widely recognized in stock market forecasting, their potential is not fully explored by machine learning researchers. Machine learning researchers typically create feature vectors that combine various types of data from the following list: historical trading data, technical indicators, fundamental indicators, and market sentiment, which are then fed into the predictive models. In contrast, technical analysts focus on analyzing the relationships between components of trend and momentum indicators to make buy/sell decisions. This difference highlights a clear divergence in the methodologies used by technical analysts and machine learning experts in stock market prediction. Moreover, trend indicators such as MACD, DMI, and KST are lagging indicators that often generate false trading signals.

To address this research gap, the study devised three intelligent trading strategies utilizing the relationships between various components of the MACD, DMI, and KST indicators, enabling the devised strategies to filter out false trading signals and learn the true patterns from the indicators. The performance of the devised intelligent strategies was assessed against traditional trading strategies based on the same indicators. Specifically, this study hybridized technical analysis with machine learning so that trading signals predicted from trend indicators like MACD, DMI, and KST can be enhanced by machine learning models by identifying true patterns. The intelligent strategies developed in this study were implemented using LSTM and GRU networks due to their ability to remember context and identify long-term dependencies (Hochreiter and Schmidhuber, 1997; Sezer et al., 2020; Yu et al., 2019; Cho et al., 2014).

The key contributions of this research are threefold. First, this study devised and evaluated intelligent stock trading strategies based on the MACD, DMI, and KST indicators. These strategies are referred to as MACD-GRU, MACD-LSTM, DMI-GRU, DMI-LSTM, KST-GRU, and KST-LSTM throughout the article. Second, the study identified the most effective trend indicator among MACD, DMI, and KST for devising intelligent stock trading strategy. Finally, it determined which recurrent neural network variant, LSTM or GRU, is more suitable for implementing these intelligent trading strategies.

The effects of technological advancements on various domains, including stock market prediction, are often explored in research on innovation dynamics. In this research, deep learning techniques like LSTM and GRU are applied to features and outcomes derived from technical indicators to forecast stock trading signals. This demonstrates how technological innovation can enhance technical analysis and improve stock market prediction.

2. Technical indicators

This section describes the various trend indicators used as the basis for formulating intelligent stock trading strategies in the study.

2.1. Moving average convergence divergence

The Mean Average Convergence Divergence (MACD) is a popular technical indicator in stock trading, used to analyze trend and momentum in stock prices. It consists of three data series: the MACD Line, Signal Line, and Histogram. The MACD Line is calculated by subtracting the 26-period EMA from the 12-period EMA, while the Signal Line is the 9-period EMA of the MACD. Traders often use the MACD-based strategy, buying when the MACD Line crosses above the Signal Line and selling when it falls below. The MACD Histogram represents the stock's bullish or bearish momentum. This strategy is referred to as the classical MACD-based trading strategy in this article. The calculation for the MACD indicator is presented in Equation (1) (Chakrabarty et al., 2014).

$$\begin{aligned} &MACD = EMA_{12}(CP) - EMA_{26}(CP) \\ &Signal = EMA_{9}(MACD) \\ &Histogram = MACD - MACD\ Signal \end{aligned} \tag{1}$$

Where, CP is close price and EMA_p is p-period EMA.

2.2. Directional movement index

$$\begin{aligned} &Up = CH - PH, Down = PL - CL \\ &if\ Up > 0 and\ Up > Down\ then + DM = Up\ else + DM = 0 \\ &if\ Down > 0 and\ Down > Up\ then - DM = Down\ else - DM = 0 \\ &+/-SDM = \sum_{t-1}^{n} DM - \frac{\sum_{t-1}^{n} DM}{n} + Current\ DM \\ &+/-DI = \frac{+/-SDM}{ATR} \times 100 \\ &DX = \left(\frac{|(+DI) - (-DI)|}{|(+DI) + (-DI)|} \right) \times 100 \\ &ADX = \frac{Previous\ ADX \times (n-1) + Current\ DX}{n} \end{aligned} \tag{2}$$

Where, CH is current high, PH is previous high, CL is current low, PL is previous low, n is time period, +DM is positive directional movement, -DM is negative directional movement, SDM is smoothed directional movement, ATR is average true range, DX is directional index and ADX is average directional index.

The Directional Movement Index (DMI) is a technical indicator used to identify the direction of a security's price movement. It consists of three lines: the positive directional indicator (+DI), negative directional indicator (-DI), and the average directional movement index (ADX). A 14-period timeframe is commonly used for calculations. The +DI line rises as prices increase, while the -DI line rises as prices decrease. An ADX value below 25 indicates a weak trend, while a value above 25 signifies a strong trend. Traders typically buy when the +DI line crosses above the -DI line and sell when the -DI line crosses above the +DI line, a strategy referred to as the classical DMI-based trading strategy in this article. Equation (2) presents the mathematical formulation of DMI indicator (Tharavanij et al., 2015).

2.3. Know sure thing indicator

The Know Sure Thing (KST) is a momentum and trend indicator derived from Rate of Change (ROC) indicators. It combines the Simple Moving Averages (SMAs) of ROCs from four different periods into a single momentum oscillator. The KST consists of the KST oscillator, the KST signal (a 9-period SMA of the KST oscillator), and the KST histogram (the difference between the KST oscillator and the KST signal). The calssical KST-based trading strategy involves buying when the KST oscillator crosses above the KST signal and selling when it crosses below. If the KST oscillator is above zero, it indicates bullish momentum; if below, it signals bearish momentum. Equation (3) is utilized for computing KST indicators (Oyewola et al., 2019).

$$KST = SMA_{10}(R_1) \times 1 + SMA_{10}(R_2) \times 2 + SMA_{10}(R_3) \times 3 + SMA_{15}(R_4) \times 4$$

$$Signal = SMA_9(KST) \tag{3}$$

$$Hitogram = KST - Signal$$

Where, $SMA_p(x)$ is p-period SMA of x and R_1, R_2, R_3, and R_4 are ROC's of 10, 15, 20, and 30-period respectively.

3. Related works

Chen and Hao developed PCA-WSVM, a weighted support vector machine that uses principal component analysis (PCA) to predict stock trading signals. They framed the problem as a four-class classification based on closing price changes and used PCA to reduce dimensionality of data that contains daily trading data and nine technical indicators. The performance of PCA-WSVM was evaluated using accuracy, profit percentage, and the Sharpe ratio, and was compared to WSVM, PCA-ANN, and the Buy-and-Hold strategy. Their experiments conducted on stocks from Shanghai and Shenzhen stock exchange from May 2, 2012, to May 31, 2014, showed that PCA-WSVM outperformed the other methods (Chen and Hao, 2018).

Wu et al. developed the LSTM-GA Stock Trading Suggestion System, which uses the LSTMLI framework and genetic algorithms (GA) to predict stock trading signals. The LSTMLI framework is an LSTM neural network with a leading index, processing historical price data, futures data, and option data to generate output labels (+1, 0, and −1) indicating stock price trends. The system's performance, assessed by profit margin, was compared with the Stock Sequence Array Convolutional Neural Network (SSACNN). Experiments on ten stocks from the US and Taiwan stock exchanges demonstrated that the LSTM-GA system achieved higher returns than SSACNN (Wu et al., 2021).

D. Lv et al. developed a feature vector combining historical trading data and 44 technical indicators. They applied four dimensionality reduction techniques—PCA, CART, AE, and LASSO—on stock datasets from the American and Chinese markets. The rise and fall of the next day's closing price were then predicted using six deep neural network (DNN) algorithms: MLP, SAE, DBN, RNN, GRU, and LSTM. A stock trading strategy was then formulated based on the predicted signals. The results showed that LASSO significantly improved the performance of RNN, GRU, and LSTM (Lv et al., 2020).

Ayala et al. improved stock trading strategies by integrating machine learning with traditional technical indicators. The authors combined the trading signals generated by the MACD or TEMA indicator with the trend predicted by machine learning algorithms to formulate a more reliable trading strategy. The study compared the effectiveness of different machine learning techniques—LM, ANN, SVR, and RF—using daily trading data from IBEX, DAX, and DJI indices. The hybrid approach was found to increase profits, reduce the number of trades, and lower the risk of losses (Ayala et al., 2021).

Y. Chen and Hao introduced a method called PLR–FW-SVM to detect trading points by combining piecewise linear representation (PLR) with a feature-weighted support vector machine (FW-SVM). They framed the prediction of stock trading points as a weighted four-class classification problem, using historical data and 15 technical indicators. The PLR-FW-SVM model was then employed to predict future turning points in the stock market. Experiments comparing PLR-FW-SVM with PLR-WSVM and PLR-ANN on over 30 stocks across different investment strategies demonstrated that the proposed approach achieved the highest average accuracy and profits (Chen and Hao, 2020).

Z. Yang et al. proposed a multi-indicator channel convolutional neural network (MICNN) that uses 9 technical indicators to forecast stock trading signals. They utilized piecewise linear representation (PLR) to classify data into Buy, Hold, or Sell categories. The study compared MICNN's performance with a three-layer MLP and an RSI-based trading method, using metrics like accuracy, recall, and profit percentage. Experiments on 10 equities from the Shanghai Stock Exchange showed that the MICNN-based strategy outperformed the RSI-based approach (Yang et al., 2019).

Silva et al. developed intelligent stock index trading strategies using LSTM combined with various risk management techniques. They introduced five strategies: LSTM-N, LSTM-PH, LSTM-RMO, LSTM-RMOD, and LSTM-RMODV. These strategies were tested on 5-minute interval data containing historical trading information and 108 technical indicators. The experimental results demonstrated that the LSTM-RMODV strategy outperformed the others in stock index trading performance (Silva et al., 2020).

W. Chen et al. introduced the GC-CNN model, a stock trend prediction method that integrates an enhanced graph convolutional network (EGCN) with a Dual-CNN. In this model, EGCN processes broader market information, while Dual-CNN handles individual stock data. The model's effectiveness was tested on six randomly selected Chinese stocks, using data that included OHCL values and 10 technical indicators. The experimental results demonstrated that GC-CNN outperformed other stock trend prediction techniques examined in the study (Chen et al., 2021).

Chandar introduced TI-CNN, a stock trading system that combines technical indicators with a convolutional neural network (CNN). Ten technical indicators were extracted from historical data to create feature vectors, which were then converted into images and used as input for the CNN. The stock's closing prices were manually classified into sell, buy, or hold points by identifying top and bottom points in a sliding window. The experimental results showed that TI-CNN outperforms other intelligent trading systems in prediction accuracy and profitability (Chandar, 2022).

Long et al. introduced the multi-filter neural network (MFNN), a model designed for extracting features from financial time series and predicting price movements. The MFNN integrates convolutional and recurrent neurons: CNNs perform feature engineering on 6-dimensional input data (OHCLVA), while RNNs extract information on past stock behavior. These features were then processed by a fully connected layer to predict stock trend labels, which inform buy or sell decisions. Experimental results showed that MFNN outperformed conventional machine learning models, statistical models, and single-structure networks (convolutional, recurrent, and LSTM) in accuracy, profitability, and stability (Long et al., 2019).

Lien Minh et al. introduced the Two-Stream Gated Recurrent Unit (TGRU) neural network for predicting short-term stock market trends. TGRU was used to enhance text processing and capture information more effectively. The authors then used the proposed Stock2Vec sentiment embedding method to improve the quality of sentiment prediction. Finally, the model was evaluated using a dataset that included historical trading data, sentiment data, and technical indicators. The results showed that the TGRU outperformed LSTM and GRU models in terms of accuracy (Lien Minh et al., 2018).

Zhang and Tan introduced the DeepStockRanker, a model for building portfolios by predicting future stock returns. The model's performance was compared with other models using daily trading data (OHCLV) and 11 technical indicators. Simulation results showed that DeepStockRanker outperformed other stock trading strategies, delivering superior returns (Zhang and Tan, 2018).

Lee et al. used an attention-based Bidirectional LSTM (AttBiLSTM) to develop a trading strategy by creating feature vectors from historical trading data and technical indicators. The study also introduced two trading approaches that combine technical indicators with deep neural networks (DNNs). Empirical results demonstrated that using time series deep neural networks in conjunction with technical analysis significantly improves stock price prediction accuracy and return on investment (Lee et al., 2022).

Touzani and Douzi developed a trading strategy for the Moroccan stock market using short-term price predictions from LSTM models and medium-term predictions from GRU models. They applied this strategy to daily trading data from the SP500 and CAC40. The experimental results showed that their method significantly outperformed local benchmark indices. During the test period, the strategy achieved an annualized return of 27.13 %, surpassing the Moroccan All Share Indices' 0.43 %, the pharmaceutical industry indices' 19.94 %, and the distributor sector indices' 15.24 % (Touzani and Douzi, 2021).

Wen et al. proposed a new method for predicting financial time series trends by reconstructing time series with high-order structures like motifs. They used convolutional neural networks to identify patterns in these reconstructed sequences, offering insights into predicting market movements. The method demonstrated lower computational complexity compared to traditional sequential models like recurrent neural networks. Experimental results confirmed the method's effectiveness, showing superior performance on real financial time series datasets in capturing stock trends (Wen et al., 2019).

Shynkevich et al. studied the relationship between a predictive system's performance and the combination of forecast horizon and input window length. They used various technical indicators as input features in machine learning algorithms to predict future price movements. The results showed that the most accurate trading signals were generated when the input window length closely matched the forecast horizon (Shynkevich et al., 2017).

Stoean et al. used historical trading data and 11 technical indicators to predict close prices for 25 companies on the Bucharest Stock Exchange. They developed two deep learning models: a temporal convolutional neural network (CNN) and a long short-term memory network (LSTM). A trading strategy was created based on these predictions, with buy/sell decisions guided by two thresholds, optimized using a hill climbing strategy. The CNN outperformed the LSTM in terms of the proportion of profitable transactions, while the LSTM achieved better profit margins. The HC-CNN had a higher Sharpe ratio, and the HC-LSTM secured a higher annualized return (Stoean et al., 2019).

Arevalo et al. introduced a high-frequency stock trading strategy using Deep Neural Networks (DNNs). The DNN was trained to predict the next one-minute average price based on current prices, n-lagged one-minute pseudo-returns, standard deviations, and trend indicators. The strategy involved buying a stock when the predicted price exceeded the latest closing price and selling when it fell below. The model was trained and tested on tick-by-tick AAPL transactions from September to November 2008. The DNN achieved a directional accuracy of 66 %, while the trading strategy had an 81 % success rate during testing (Arévalo et al., 2016).

Arevalo et al. introduced an automatic and adaptable trading rule based on flag pattern identification, eliminating the need for traders to predict the optimal arrangement of the rule. They also integrated the Exponential Moving Average (EMA) indicator to filter trades, considering both 15-minute and 1-day timeframes for short and medium term trends. The trading rule was tested on a large intraday database for the DJIA index. Experimental results showed that this strategy was more profitable and risk-averse compared to previous flag pattern strategies and the Buy-Hold strategy (Arévalo et al., 2017).

Troiano et al. investigated the use of a Long Short-Term Memory (LSTM) neural network to learn a trading rule by analyzing the correlation between market indicators and decisions to enter or exit a position. They developed a deep learning-based, model-free robot to understand how market sentiment, reflected in technical indicators, influences investment actions. Their experimental results demonstrated the feasibility and potential of this approach (Troiano et al., 2018).

L. Chen et al. developed a stock trading system with three key components. First, an LSTM neural network classified stock price fluctuations into three categories (+1, 0, −1) using leading indicators. Next, a genetic algorithm was used to find the optimal threshold for trading signals. Finally, the Kelly criterion was applied to determine the best investment score, aiming to reduce transaction risk. Experimental results showed that using the Kelly criterion effectively minimized trading risk while boosting returns (Chen et al., 2021).

Lee's model integrated a Gate Recurrent Unit (GRU) with an Attention Mechanism to predict stock trends. It uses technical indicator for transition prediction and stock prices and trading volume to estimate the probability of future price movements. The study found that this model could effectively forecast significant stock price fluctuations, making it useful for trading strategies (Lee, 2022).

Sang and Pierro enhanced a traditional technical analysis trading strategy by integrating a Long Short-Term Memory (LSTM) neural network. They combined technical indicators like MACD, RSI, and SMA with LSTM and compared the system's performance against standard technical analysis strategies. The results showed that the LSTM-augmented strategy outperformed traditional methods in terms of annual cumulative profit. The study also explored LSTM hyper-parameters, finding that the Adam optimizer with a learning rate of 0.0001 was the most effective (Sang and Di Pierro, 2019).

Shi and Zhao analyzed weekly market price fluctuations, focusing on identifying true and false golden crosses and death crosses, key indicators in technical analysis. They also explored the impact of breaking news on stock prices using deep neural networks and investigated how profit expectations and holding thresholds affect profitability. Experiments with twenty S&P 500 stocks showed that deep neural networks help investors achieve higher profits, with optimal results occurring when balancing the holding threshold with profit expectations (Shi and Zhao, 2020).

Kuo et al. developed a two-stage model to improve pair-wise trading systems (PTS) by optimizing trigger thresholds and eliminating underperforming stock pairs. In the first stage, they used a multi-scale ResNet to identify the best threshold from a set generated by a representative labeling mechanism (RLM). In the second stage, another multi-scale ResNet was trained to filter out unprofitable stock pairs based on the profitability determined in the first stage. The experimental results showed that this strategy could achieve a higher win rate and average profit compared to other trading methods (Kuo et al., 2022).

From the above discussion, it is evident that while most research has included technical analysis indicators as features in predictive models and few studies have overlooked them entirely. However, these studies have not fully leveraged the relationships between various components of technical indicators to generate stock trading signals, a common practice in the field of technical analysis. For instance, crossovers between the MACD line and signal line, the +DI and -DI lines, and the KST line and its signal line are frequently used as triggers for buying and selling in technical analysis. This approach, however, is often ignored by machine learning researchers when developing intelligent trading strategies. Addressing this research gap is the primary focus of this study.

4. Proposed strategy

Technical analysts often use trend and momentum indicators like MACD, KST, and DMI to predict buy/sell signals. Although many machine learning researchers have incorporated these indicators into intelligent stock trading strategies, they typically use them only as inputs, overlooking the relationships between different signal lines within the indicators. This reflects a difference in how machine learning experts and technical analysts use trend indicators. Additionally, a notable drawback of the trend indicator-based trading systems is their propensity to generate false signals. To address these issues, this research developed three intelligent stock trading methods using the MACD, KST, and DMI indicators. These strategies were implemented using LSTM and GRU networks, referred to as MACD-LSTM, MACD-GRU, KST-LSTM, KST-GRU, DMI-LSTM, and DMI-GRU strategies. Fig. 1 presents schematic representations of these intelligent trading strategies.

Historical trading data for 18 selected stocks was downloaded from various sources and preprocessed as detailed in Sections 5.1 and 5.2. Next, MACD, KST, and DMI indicators were calculated as described in

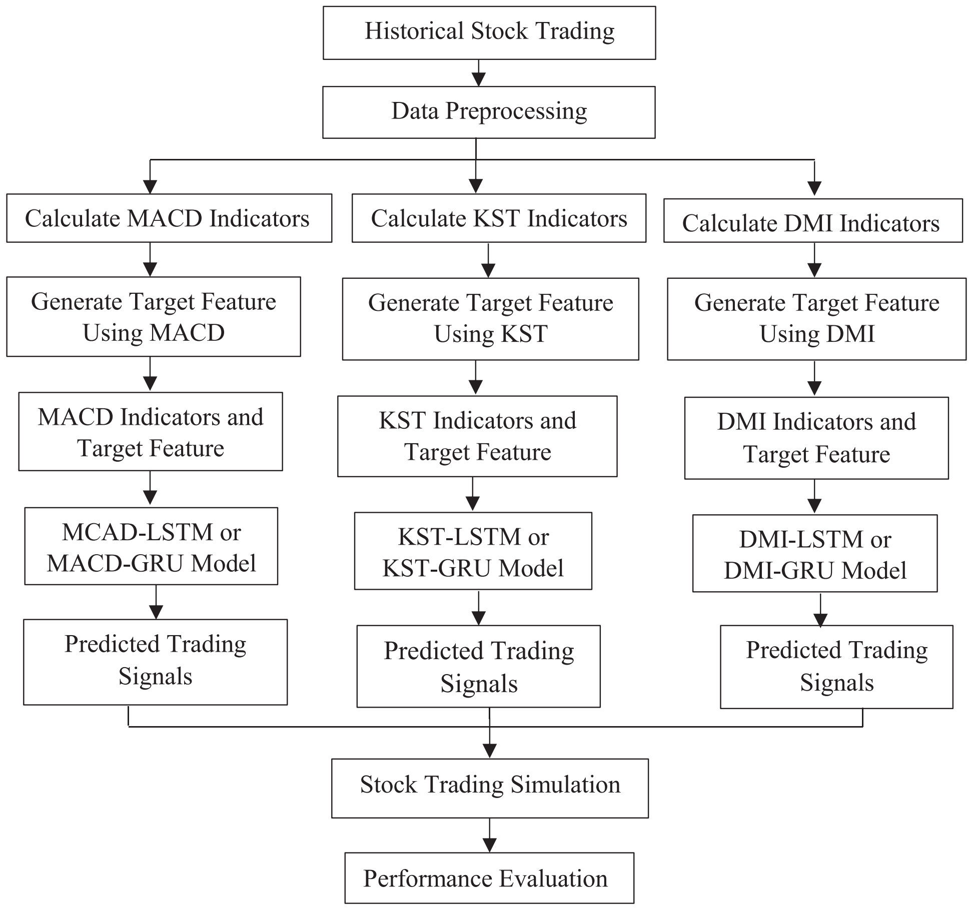

Fig. 1. Schematic representation of proposed intelligent trading strategies.

Sections 2.1 to 2.3, and target features for MACD, KST, and DMI-based intelligent strategies were generated using practices employed by technical analysts. These are trading signals predicted from technical indicators, used by the formulated intelligent strategies to learn true buy/sell signals and filter out false trading signals. The exact formulation for generating trading signals is provided in Section 4.1. Then, all features except the MACD Line, MACD Signal Line, MACD Histogram, and 'Trading Signal' for MACD-based intelligent strategies; the KST Line, KST Signal Line, KST Histogram, and 'Trading Signal' for KST-based intelligent strategies; and the +DI Line, -DI Line, ADX Line, and 'Trading Signal' for DMI-based intelligent strategies were removed from the dataset. This is explained in detail in Sections 5.3 and 5.4. Finally, trading signals predicted by the trained intelligent strategies were used by the stock trading simulator to perform automated trading on the test data, as discussed in Section 4.2. The simulator generates profit/loss, number of trades, and number of profitable trades, which were then used to calculate the annual rate of return (ARR), Sharpe Ratio (SR), and Win Rate (WR), forming the basis for evaluating the proposed intelligent strategies.

4.1. Target feature generation

This study developed models to generate the target feature "Trading Signal" for training the intelligent trading strategies. The relationship between the MACD line and the MACD signal line served as the foundation for generating the "Trading Signal" for the MACD-LSTM and MACD-GRU strategies, as outlined in Equation (4). Similarly, the relationship between the KST line and the KST signal line was utilized to generate the "Trading Signal" for the KST-LSTM and KST-GRU methods, as detailed in Equation (5). Furthermore, the relationship between the +DI line and the –DI line was employed to generate the "Trading Signal" for the DMI-LSTM and DMI-GRU strategies, as shown in Equation (6). Each of these models produces a sequence of buy, hold, and sell actions. Whenever there is a crossover between the lines, the models incorporate three buy and sell signals in the target feature. This approach aims to simplify the identification of trading signals by the intelligent models developed for stock trading.

$$\begin{aligned} &If\ m > ms\ and\ turn = 0 then\ ts[i : i+3] = 'Buy' \\ &If\ ms > m and\ turn = 1 then\ ts[i : i+3] = 'Sell' \\ &Otherwise\ ts[i] = 'Hold' \end{aligned} \tag{4}$$

Where, m is MACD line, ms is MACD signal line, ts is trading signal vector, and turn is variable representing Buy or Sell turn.

$$\begin{aligned} &If k > ks\ and\ turn = 0 then\ ts[i : i+3] = 'Buy' \\ &If\ ks > k and\ turn = 1 then\ ts[i : i+3] = 'Sell' \\ &Otherwise\ ts[i] = 'Hold' \end{aligned} \tag{5}$$

Where, k is KST line and ks is KST signal line.

$$\begin{aligned} &If\ pdi > ndi\ and\ turn = 0 then\ ts[i : i+3] = 'Buy' \\ &If\ ndi > pdi\ and\ turn = 1 then\ ts[i : i+3] = 'Sell' \\ &Otherwise\ ts[i] = 'Hold' \end{aligned} \tag{6}$$

Where, pdi is +DI line and ndi is –DI line.

4.2. Stock trading simulation

This study also designed and implemented a stock trading simulator to assess the effectiveness of trading strategies based on predicted signals. The simulator assumes that a stock trader begins with no equity and some seed capital, requiring the first trading action to be a "Buy." It further assumes that traders invest their entire capital in a single purchase and sell all their equities at once, resulting in a series of "Buy" and "Sell" transactions. After completing a "Sell," the simulator reinvests the entire capital in the next "Buy" transaction. If the last action is a "Sell," the sequence must contain an equal number of "Buy" and "Sell" transactions. Otherwise, there should be one more "Buy" than "Sell". In this case, the simulator discards the last "Buy" transaction and then calculates the profit/loss percentage from trading. For this purpose, the simulator uses the closing price as the "Buy/Sell" price. The mathematical formula for calculating the gross profit/loss from trading, excluding transaction costs and income gain taxes, is provided in Equation (7), while Equation (8) gives the profit/loss percentage calculation.

$$gpl = y - x \tag{7}$$

Where, y is capital obtained from last "Sell" operation and x is seed capital that the stock trader initially invested.

$$plp = \frac{gpl}{x} \times 100 \tag{8}$$

This research work applied the aforementioned formulation for trading simulation across all the intelligent and classical trading strategies that were studied in the study.

5. Methodology

5.1. Dataset description

This study utilized historical stock trading data sourced from the New York Stock Exchange (NYSE), Bombay Stock Exchange (BSE), and Nepal Stock Exchange (NEPSE). A total of eighteen stocks were examined, with six selected from each stock market. Three stock markets were selected to represent a developed country (NYSE), a developing country (BSE), and an underdeveloped country (NEPSE). Six stocks were randomly selected from each stock market, with careful consideration given to represent various sectors and price ranges. Specific stock statistics are detailed in Table 1. Historical trading data for equities listed on the NEPSE, BSE, and NYSE was obtained from NepseAlpha (Nepse price export, 2024) BSE India (Stock prices, 2024), and Yahoo Finance (Yahoo Finance - Stock Market Live, 2024), respectively. The data for all stocks spans the period from January 1, 2000, to October 4, 2022, except for the NEPSE stocks, due to unavailability of data for them. The selected 18 stocks exhibited bullish, bearish, and mixed trends during the test period, as depicted in Fig. 2.

Table 1
Historical trading data of stocks.

Stock Exchange	Stock Name	Date	
		From	To
NEPSE	Sanima Bank Ltd. (SANIMA)	10/4/2010	10/4/2022
	Nepal Bank Ltd. (NBL)	10/4/2010	10/4/2022
	Muktinath Bikas Bank Ltd. (MNBBL)	10/4/2010	10/4/2022
	National Life Insurance Company (NLICL)	10/4/2010	10/4/2022
	Nirdhan Utthan Laghubitta Ltd. (NUBL)	10/4/2010	10/4/2022
	Butwal Power Company Ltd. (BPCL)	10/4/2010	10/4/2022
BSE	ICICI Bank Ltd. (ICICI)	1/1/2000	10/4/2022
	Housing Development Finance Ltd. (HDFC)	1/1/2000	10/4/2022
	Hindustan Unilever Ltd. (HUNL)	1/1/2000	10/4/2022
	JSW Steel Ltd. (JSW)	1/1/2000	10/4/2022
	Vedanta Limited (VDNTA)	1/1/2000	10/4/2022
	Pidilite Industries Ltd. (PIDI)	1/1/2000	10/4/2022
NYSE	Avista Corporation (AVA)	1/1/2000	10/4/2022
	Bank of America Corporation (BAC)	1/1/2000	10/4/2022
	Carnival Corporation and plc (CCL)	1/1/2000	10/4/2022
	Davon Energy Corporation (DVN)	1/1/2000	10/4/2022
	Ford Motor Company (FORD)	1/1/2000	10/4/2022
	The Kroger Co.(KR)	1/1/2000	10/4/2022

5.2. Data preprocessing

The historical trading data for NEPSE, BSE, and NYSE stocks was sourced from NepseAlpha, BSE India, and Yahoo Finance, respectively, with varying column structures. For consistency, the data was preprocessed to retain only the essential columns: Date, Open, High, Low, Close, and Volume (OCHLV). The data was organized chronologically, checked for missing values, and further preprocessing specific to the prediction module was incorporated into the algorithms discussed in Section 5.4.

5.3. Data preparation

In this study, data were divided into training, validation, and test sets in an 8:1:1 ratio. The primary goal of stock market prediction was to forecast stock price or stock trading signal for the day t+1 by using daily trading data from the $(t\text{-}N+1)^{th}$ day to the t^{th} day, where N is window size or look back period. The time series sequence of the input features was therefore created using the trading data from the previous N days. Thus, the dataset for the prediction systems was made up of N independent variables (d_{t-N+1}, $d_{t-N+2}\ldots$ d_{t-1}, d_t) and a dependent a_{t+1}, where d_i is a tuple of input features and a_i represents the trading signal for the i^{th} trading day. Each prediction system in the study featured slight variations in the input features, with the specific input tuple detailed in the algorithm provided in Section 5.4.

Intelligent Trading Signal Prediction Algorithm**Input:** CSV file of Stock trading data containing Date and OHLCV**Output:** Stock Trading Signal for Next trading day

1. Read CSV file of stock trading data.
2. For the case "MACD-Based Strategy"

a. Calculate MACD indicators using Eq. (1).
b. Generate Output Label "Trading Signal" using Eq. (4).
c. Drop tuples with missing values and normalize input features using standard scalar.
d. Set, m_i,ms_i,mh_i, where m_i, ms_i and mh_i are MACD line, MACD signal line, and MACD histogram for i^{th} trading day respectively, as input features.
e. Prepare time series data as $inp = (d_{t-n+1}, d_{t-n+2}, \ldots, d_{t-1}, d_t)$, where d_i is independent variable and is given as $d_i = (m_i, ms_i, mh_i)$ and n is look back period.

3. For the case "KST-Based Strategy"

a. Calculate KST indicators using Eq. (2).
b. Generate Output Label "Trading Signal" using Eq. (5).
c. Drop tuples with missing values and normalize input features using standard scalar.

(continued on next page)

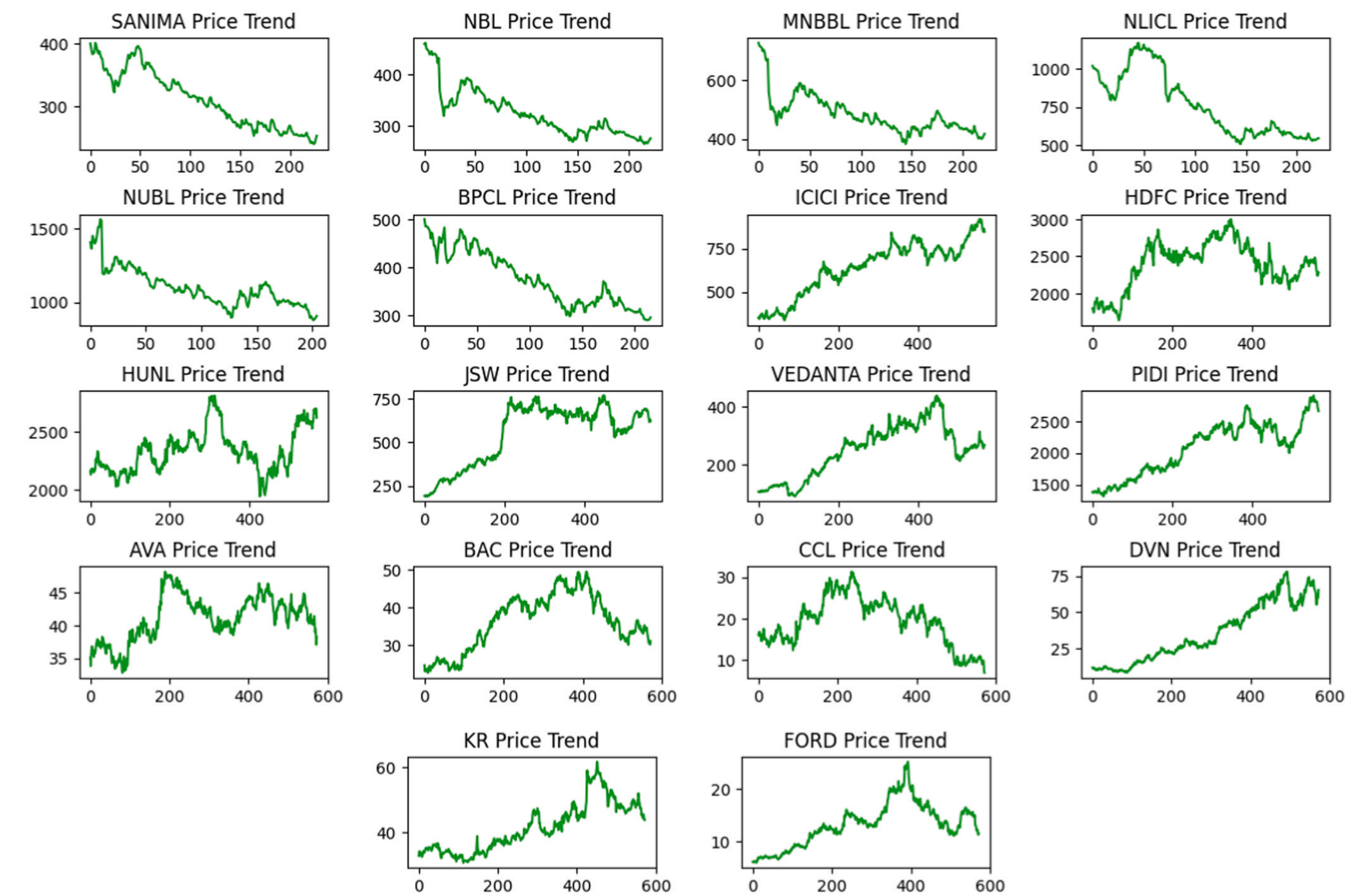

Fig. 2. Trends of stocks in test period.

(*continued*)

d. Set, k_i,ks_i,kh_i, where k_i, ks_i and kh_i are KST line, KST signal line, and KST histogram for i^{th} trading day respectively, as input features.
e. Prepare time series data as $inp = (d_{t-n+1}, d_{t-n+2}, \ldots, d_{t-1}, d_t)$, where d_i is independent variable and is given as $d_i = (k_i, ks_i, kh_i)$ and n is look back period.

4. For the case "DMI-Based Strategy"

a. Calculate DMI indicators using Eq. (3).
Generate Output Label "Trading Signal" using Eq. (6)

b. Drop tuples with missing values and normalize input features using standard scalar.
c. Set, pd_i,nd_i,ad_i, where pd_i, nd_i and ad_i are values of +DMI line, -DMI line, and ADX line for i^{th} trading day respectively, as input features.
d. Prepare time series data as $inp = (d_{t-n+1}, d_{t-n+2}, \ldots, d_{t-1}, d_t)$, where d_i is independent variable and is given as $d_i = (pd_i, nd_i, ad_i)$ and n is look back period.

5. Encode Output feature using one hot encoding.
6. Set a_{t+1}, where a_{t+1} is trading signal for (t+1)th trading day, as output feature.
7. Split dataset into training, validation, and test sets in 8:1:1 ratio.
8. Train GRU/LSTM Neural Network and predict stock trading signals for the test dataset.
9. Perform trading simulation using predicted trading signals.
10. Compute and display gross profit/loss its percentage using Eq. (7) and Eq. (8) respectively.
11. Compute and display total number of trades and number of profitable trades.

5.4. Configuration of intelligent stock trading systems

The LSTM and GRU networks used for implementing intelligent trading strategies were configured with a node structure of 4×128×128×128×128×128×3. Mini-batch Gradient Descent with a batch size of 32 and a look-back period of 20 was used as the gradient descent variant. The Adam optimizer was applied for optimizing gradient descent. The hidden layers utilized the ReLU activation function, while the output layer employed a Softmax activation function. Categorical Cross-entropy was chosen as the loss function.

5.5. Evaluation metrics

The stock trading signal prediction systems developed in this research were evaluated using three key metrics: the annual rate of return (ARR), the Sharpe ratio (SR), and the win rate (WR). A brief overview of these indicators is provided below.

5.5.1. Annual rate of return (ARR)

The annual rate of return (ARR), often expressed as a percentage, denotes the profit earned on an investment during a 12-month duration. Whether this percentage is positive or negative depends on the gained or lost amount relative to the principal. The formula for calculating ARR is outlined in Eq. (9).

$$ARR = \left\{ \left(\frac{P+G}{P} \right)^{1/n} - 1 \right\} \times 100 = \left\{ (1 + return)^{1/n} - 1 \right\} \times 100 \tag{9}$$

Where n is number of years, P is principle, and G is the profit or loss obtained over the period.

5.5.2. Sharpe ratio

The Sharpe ratio (SR) evaluates the performance of an investment, such as securities or a portfolio, relative to a risk-free asset, accounting for its risk. Also known as the Sharpe index, Sharpe measure, or reward-to-volatility ratio, it is calculated by dividing the difference between the investment's returns and the risk-free return by the standard deviation of the investment's returns, as shown in Eq. (10). In essence, it quantifies the risk-adjusted return of a financial portfolio. A portfolio with a higher Sharpe ratio is generally regarded as superior to its peers.

$$SR = \frac{R_t - R_f}{\sigma} \tag{10}$$

Where, R_t is the return from stock trading, R_f is return from risk-free investment, and σ is standard deviation of return from stock trading

5.5.3. Win rate

The win rate (WR) represents the percentage of profitable trades conducted within a specified timeframe out of all trades made during that same period. It is typically expressed as a percentage, as depicted in Eq. (11).

$$WR = \frac{P_t}{T} \times 100 \tag{11}$$

Where, P_t is total number of profitable trades, and T is total number of trades executed in the specified period.

6. Experimental results and discussion

This section presents the experimental results and interpretations derived from the proposed intelligent trading strategies. Additionally, the hyperparameter lookback period had a noticeable impact on the trading performance of the strategies, so an analysis of the look back period is also included.

6.1. Effect of look back period in intelligent trading strategies

The look back period for each intelligent trading strategy was varied from 3 to 15. Each experiment was repeated 10 times for each look back period value, and the average profit percentage was recorded. The annual rate of return (ARR) was then calculated and analyzed.

Figs. 3 and 4 illustrate that the MACD-GRU strategy produced the highest ARR for 15 out of 18 stocks with a look back period of 5, while the MACD-LSTM strategy achieved the highest ARR for 12 out of 18 stocks. Similarly, Figs. 5 and 6 show that with a look back period of 5, the DMI-GRU strategy resulted in the highest ARR for 17 out of 18 stocks, and the DMI-LSTM strategy did so for 16 out of 18 stocks. Finally, Figs. 7 and 8 indicate that the KST-GRU strategy obtained the highest ARR for 15 out of 18 stocks, and the KST-LSTM strategy achieved the highest ARR for 13 out of 18 stocks with a look back period between 9 and 11. Mean values of the intelligent trading strategies at various values of lookback period is presented in Table 2. Based on these findings, a look back period of 5 is recommended for MACD and DMI-based intelligent trading strategies, while a look back period between 9 and 11 is recommended for the KST-based intelligent strategies.

6.2. Performance evaluation of GRU/LSTM based intelligent trading strategies

This section evaluated the performance of proposed intelligent trading strategies to determine which architecture is better suited for predicting trading signals. Following the recommendations from the previous section, MACD and DMI-based strategies were implemented with a look back period of 5, while the KST-based strategy used a look back period of 10. Each experiment was conducted ten times, and the average profit percentage was recorded. Subsequently, the annual rate of return (ARR) was calculated and analyzed.

Figs. 9 to 11 show that the MACD-GRU strategy achieved a slightly higher ARR than the MACD-LSTM strategy for 12 of the 18 stocks. The mean ARR for MACD-GRU was 114.27, and it was 110.95 for MACD-LSTM. Similarly, the DMI-GRU strategy outperformed the DMI-LSTM strategy for 14 of the 18 stocks, and the KST-GRU strategy yielded higher ARR than the KST-LSTM strategy for 12 of the 18 stocks. The mean ARR values for the DMI-GRU and DMI-LSTM strategies were 87.08 and 84.57, respectively, while for the KST-GRU and KST-LSTM strategies, they were 84.98 and 78.92. These results indicate that intelligent trading strategies implemented with the GRU network are slightly more profitable. Additionally, GRU networks train faster than LSTM networks and have better generalization capability with a moderate volume of data. Therefore, the study suggests implementing intelligent trading strategies using the GRU network.

6.3. Evaluation of MACD based intelligent trading strategy

This section compares the effectiveness of the MACD-based intelligent trading strategy with the classical MACD-based trading strategy in terms of ARR, SR, and win rate.

Fig. 12 shows that the ARR from the intelligent MACD-based trading strategy is consistently higher than that from the classical MACD-based trading strategy. Unlike the classical strategy, which produced positive ARR for only 7 out of 18 stocks, the MACD-GRU strategy consistently yielded positive ARR. The figure also indicates that the ARR for NEPSE stocks is typically lower than ARR obtained from stocks listed in BSE and NYSE, likely due to NEPSE stocks being in an extreme bearish trend during the test period. Fig. 13 demonstrates that the Sharpe Ratio (SR) from the MACD-GRU strategy was consistently higher than that from the classical strategy. While the classical strategy achieved an SR greater than 3 just for one stock, the MACD-GRU strategy did so for 15 out of 18 stocks. In finance, SR values above 3 are considered excellent. Lastly, Fig. 14 shows that the win rate of the intelligent trading strategy is significantly higher than that of the classical strategy, indicating that the

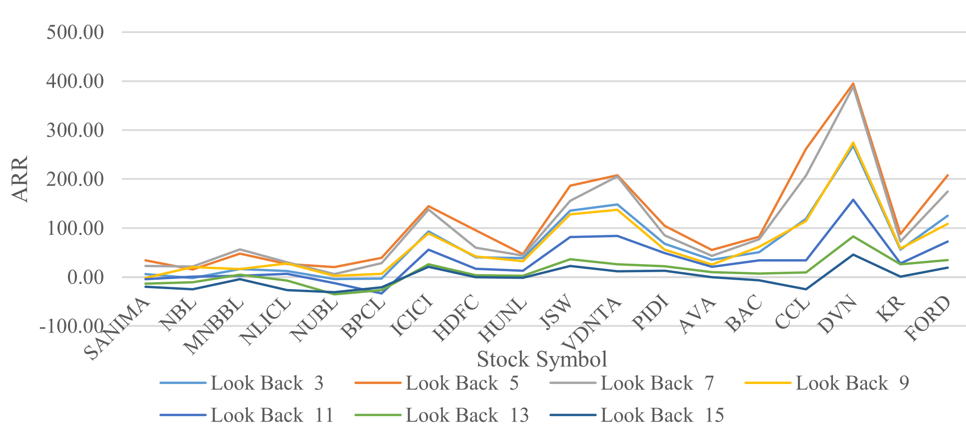

Fig. 3. ARR obtained from MACD-GRU for various values of look back period.

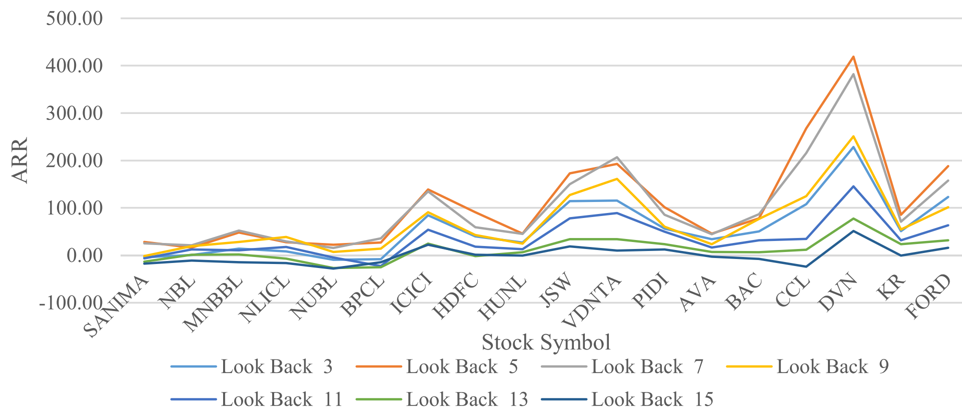

Fig. 4. ARR obtained from MACD-LSTM for various values of look back period.

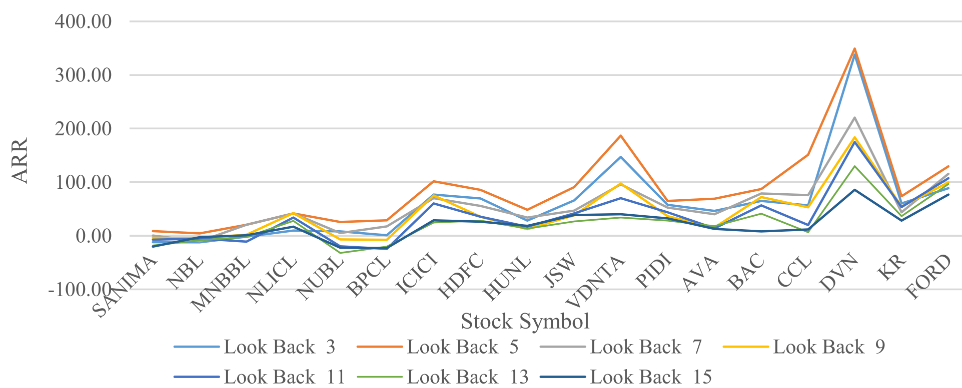

Fig. 5. ARR obtained from DMI-GRU for various values of look back period.

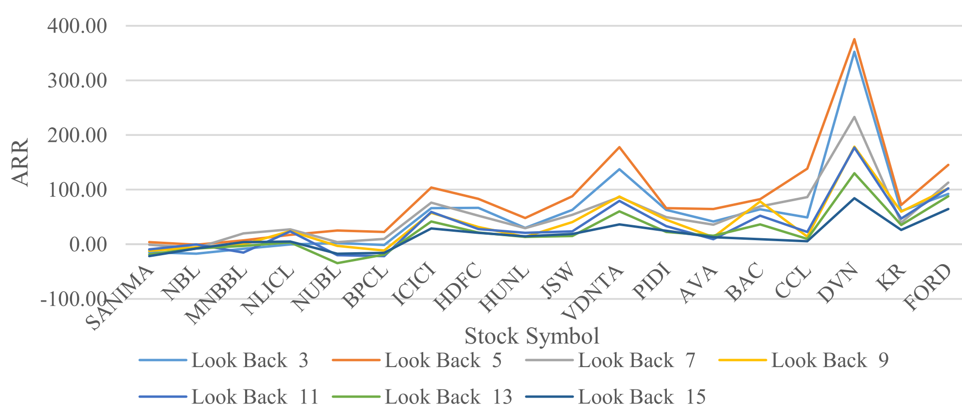

Fig. 6. ARR obtained from DMI-LSTM for various values of lookback period.

intelligent strategy effectively filtered out many false signals from the MACD indicator. Table 3 presents the mean values of ARR, SR, and *WR* for both the MACD-based intelligent and classical strategies. Based on these findings, the study concluded that the intelligent MACD-based trading strategy is more profitable and less risky than the classical strategy.

6.4. Evaluation of DMI based intelligent trading strategy

This study also developed an intelligent trading strategy based on DMI indicators. This section compares the performance of this intelligent trading strategy with the classical DMI-based strategy in terms of ARR, SR, and win rate.

The experimental outcomes of the DMI-based intelligent trading strategy showed patterns similar to those of the MACD-based intelligent trading strategy. As illustrated in Fig 15, the DMI-GRU strategy significantly outperformed the classical DMI strategy in terms of ARR for all stocks. Only 9 out of the 18 stocks in the classical strategy yielded positive ARR, while all 18 stocks in the DMI-GRU strategy produced positive ARR. Fig 16 demonstrated that the DMI-GRU strategy had positive SR values for 16 out of 18 stocks, compared to only 9 out of 18 for the classical strategy. The intelligent strategy achieved an SR greater than 3 for 13 of the 18 stocks, whereas the classical strategy did so for just one stock. In terms of win rate, the DMI-GRU strategy also outperformed the classical strategy, consistently achieving a higher win rate as depicted in Fig 17. This indicates that the DMI generates numerous false trading signals, which the intelligent strategy effectively filters out. Table 4 presents the mean values of ARR, SR, and *WR* for the DMI-based intelligent and classical strategies. Based on these findings, the study concludes that the DMI-based intelligent trading strategy is far

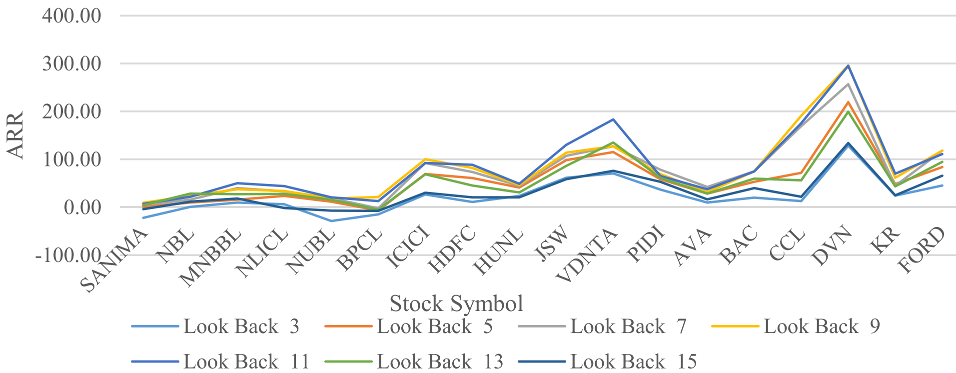

Fig. 7. ARR obtained from KST-GRU for various values of lookback period.

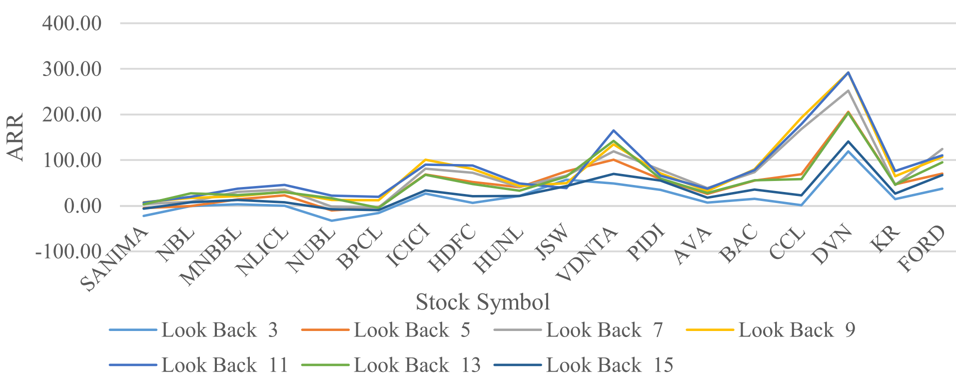

Fig. 8. ARR obtained from KST-LSTM for various values of look back period.

Table 2
Mean ARR at various lookback values.

Look-Back PeriodMean ARR	3	5	7	9	11	13	15
MACD-GRU	66.74	**114.27**	100.74	66.44	33.61	10.77	−1.65
MACD-LSTM	57.43	**110.95**	101.07	69.22	35.16	11.77	−0.26
DMI-GRU	60.68	**87.08**	55.84	44.15	36.73	23.94	19.85
DMI-LSTM	58.22	**84.57**	54.35	39.71	34.07	22.88	16.39
KST-GRU	23.52	55.75	74.46	80.92	**84.98**	56.02	31.70
KST-LSTM	17.80	49.29	68.52	75.03	**78.92**	55.42	31.11

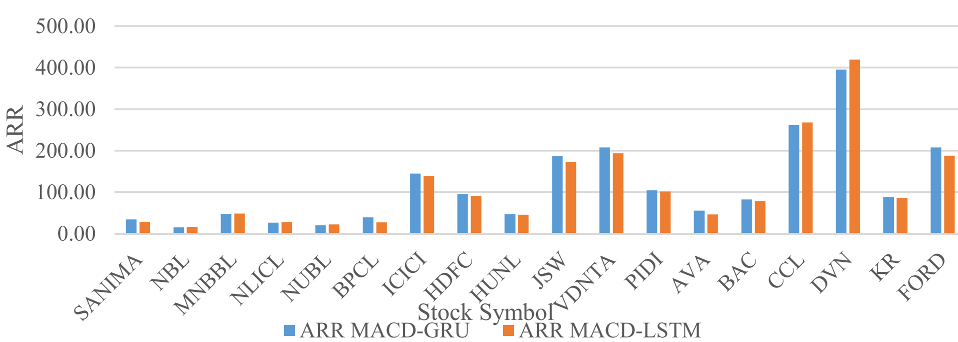

Fig. 9. ARR achieved from MACD-GRU and MACD-LSTM.

superior to the classical DMI-based strategy for stock trading.

6.5. Evaluation of KST based intelligent trading strategy

Another intelligent trading strategy based on the KST indicator was also developed in this study, and its performance was compared to that of the classical KST-based trading strategy. This section presents a comparative analysis of these trading strategies using ARR, SR, and win rate.

In terms of ARR, SR, and win rate, the intelligent trading strategy based on KST outperformed the classical KST-based trading strategy. According to Fig 18, the intelligent strategy consistently achieved much higher ARR than the classical strategy. While the intelligent strategy generated positive ARR for all 18 stocks, the classical strategy only did so for 10 out of 18 stocks. Fig 19 illustrates that the SR values generated by the classical KST-based trading strategy never exceeded those generated by the KST-GRU strategy. The classical strategy produced positive SR for only 8 out of 18 stocks, whereas the intelligent trading

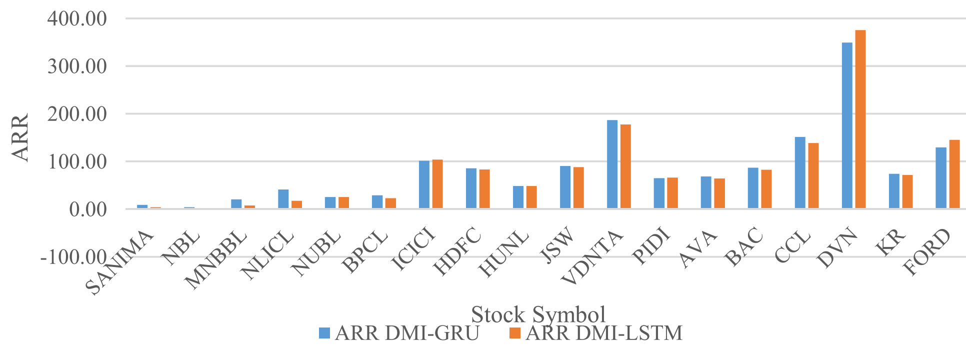

Fig. 10. ARR achieved from DMI-GRU and DMI-LSTM.

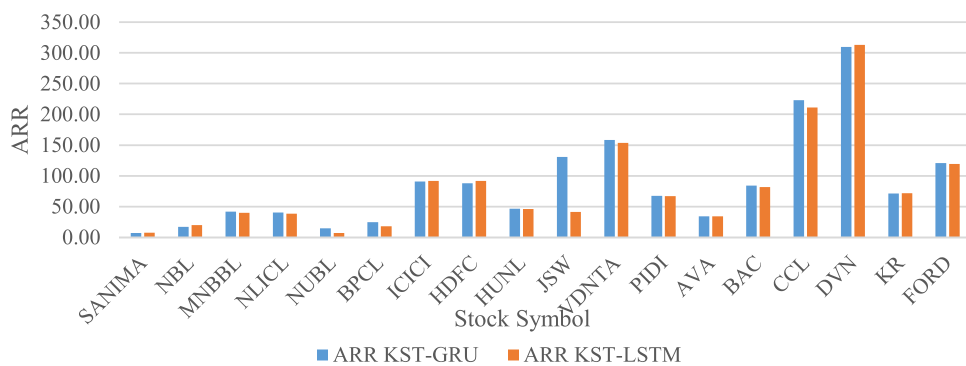

Fig. 11. ARR achieved from KST-GRU and KST-LSTM.

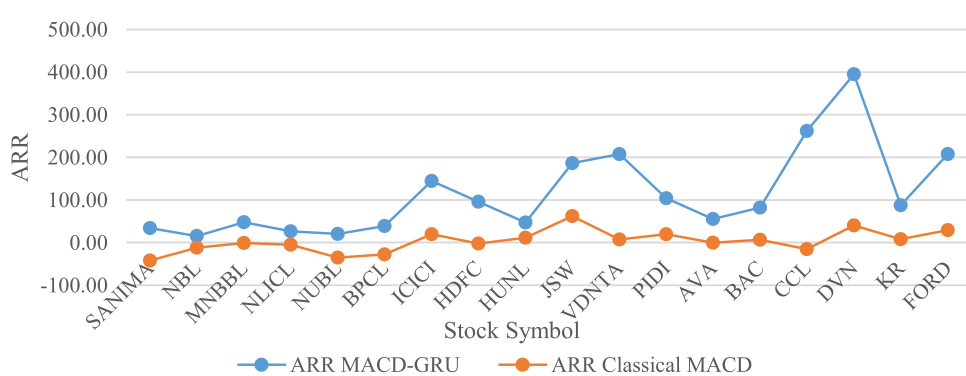

Fig. 12. ARR achieved from MACD-GRU and classical MACD trading strategies.

strategy achieved positive SR for 17 out of 18 stocks. Additionally, while only one out of 18 stocks achieved an SR greater than 3 using the classical strategy, 12 out of 18 stocks achieved this using the KST-GRU strategy. Fig 20 shows that the win rate from the intelligent trading strategy was consistently higher than that from the classical strategy. The mean values of ARR, SR, and WR for the KST-based intelligent and classical strategies are presented below in Table 5. Therefore, based on these observations, the KST-based intelligent trading strategy is clearly

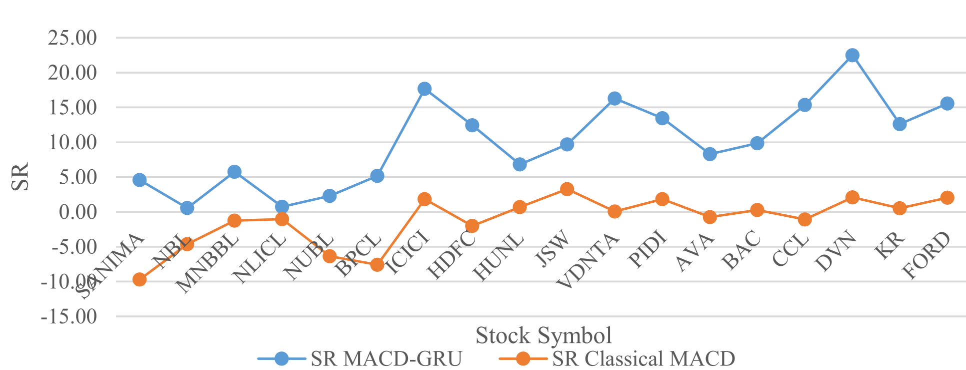

Fig. 13. SR achieved from MACD-GRU and Classical MACD Based Trading Strategies.

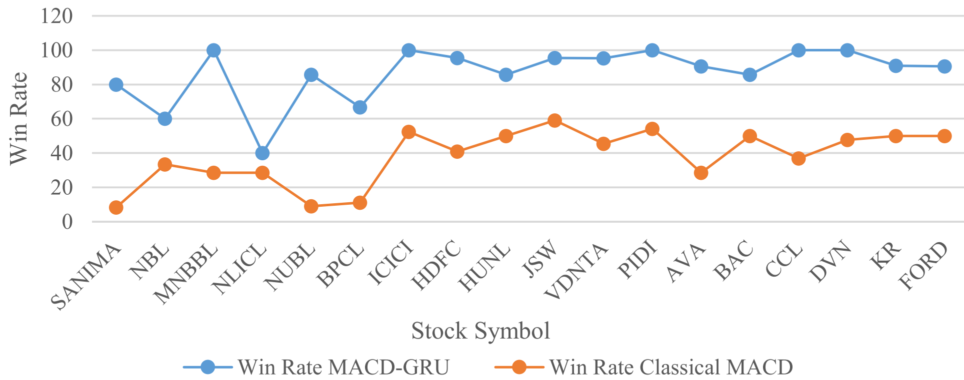

Fig. 14. Win rate of MACD-GRU and classical MACD based trading strategies.

Table 3
Mean values of performance measures for MACD based trading strategies.

Performance MeasuresTrading Strategy	Mean ARR	Mean SR	Mean WR
Intelligent Strategy	114.27	9.97	86.77
Classical Strategy	3.57	−1.21	38.00

superior to the classical KST-based strategy for stock trading.

6.6. Evaluation of the intelligent trading strategies

This section evaluates the three intelligent trading strategies developed in this research study. These strategies include the MACD-GRU, the DMI-GRU, and the KST-GRU.

As depicted in Fig 21 the MACD-GRU intelligent trading strategy outperformed the DMI-GRU and KST-GRU strategies in terms of ARR. The DMI-GRU and KST-GRU strategies achieved the highest ARR for only 5 and 1 stocks, respectively, whereas the MACD-GRU strategy achieved the highest ARR for 12 out of 18 stocks. The average ARR values achieved with the MACD-GRU, DMI-GRU, and KST-GRU strategies were 114.27, 87.08, and 84.98, respectively. Additionally, the MACD-based intelligent trading strategy demonstrated superior performance in terms of SR compared to the other two strategies, as shown in Fig 22. While the DMI and KST-based strategies achieved the highest SR values for only 6 and 1 stock, respectively, the MACD-GRU strategy attained the highest SR value for 11 out of 18 stocks. Unlike the DMI-GRU strategy, which resulted in negative SR for two stocks, and the KST-GRU strategy, which did so for only one stock, the MACD-based strategy never yielded negative SR for any stock. In summary, the MACD, DMI, and KST-based intelligent trading strategies achieved average SR values of 9.97, 7.35, and 6.8, respectively. For most stocks, the MACD-GRU strategy executed a higher percentage of profitable trades, as illustrated in Fig 23. It achieved a higher win rate for 8 stocks, whereas the DMI and KST-based strategies achieved this for only 2 and 4 stocks, respectively. For the remaining 4 stocks, an equal proportion of profitable trades was observed using both MACD and KST-based intelligent methods. On average, the MACD, DMI, and KST-based strategies

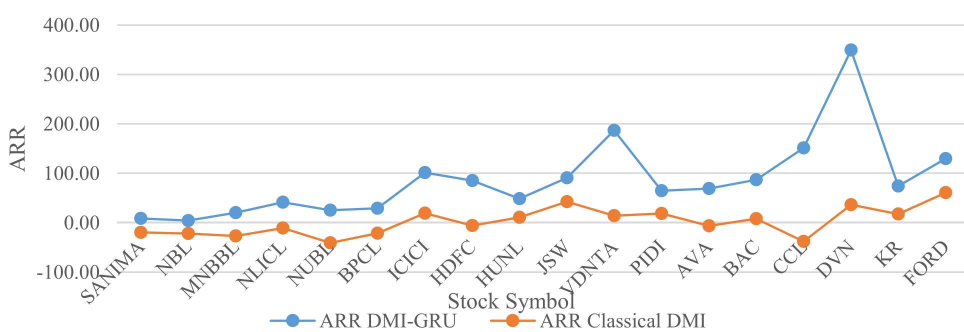

Fig. 15. ARR Obtained from DMI-GRU and Classical DMI Trading Strategies.

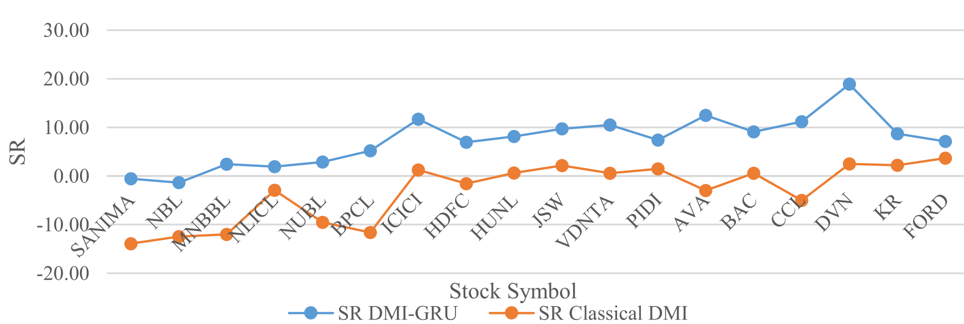

Fig. 16. SR obtained from DMI-GRU and classical DMI trading strategies.

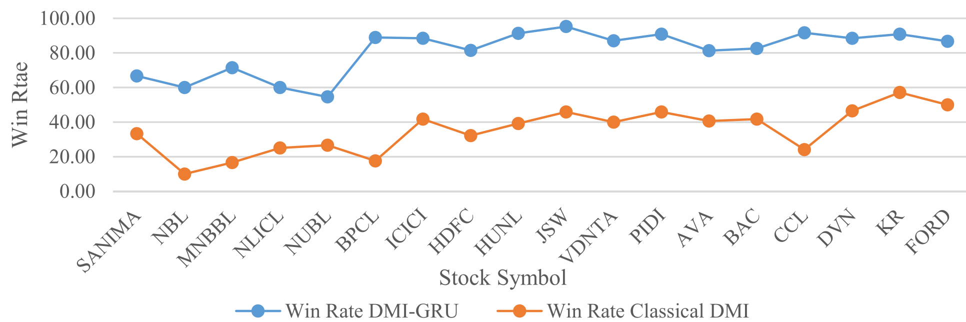

Fig. 17. Win rate of DMI-GRU and classical DMI trading strategies.

Table 4
Mean values of performance measures for DMI based trading strategies.

Performance MeasuresTrading Strategy	Mean ARR	Mean SR	Mean WR
Intelligent Strategy	87.08	7.35	80.97
Classical Strategy	1.98	−3.16	35.22

executed successful trades at rates of 86.76 %, 80.97 %, and 83.2 %, respectively. Based on these findings, the study recommends using a MACD-based intelligent trading strategy for stock trading.

7. Conclusion

This study focuses on improving stock market prediction by developing three intelligent trading strategies using the MACD, DMI, and KST indicators to predict stock trading signals. These strategies were compared to traditional stock trading methods using the same indicators. The research also highlighted the significant impact of the look back period on the performance of the strategies. The effectiveness of these strategies was measured by annual return rate (ARR), Sharpe ratio, and the win rate. Based on the experimental results presented in Section

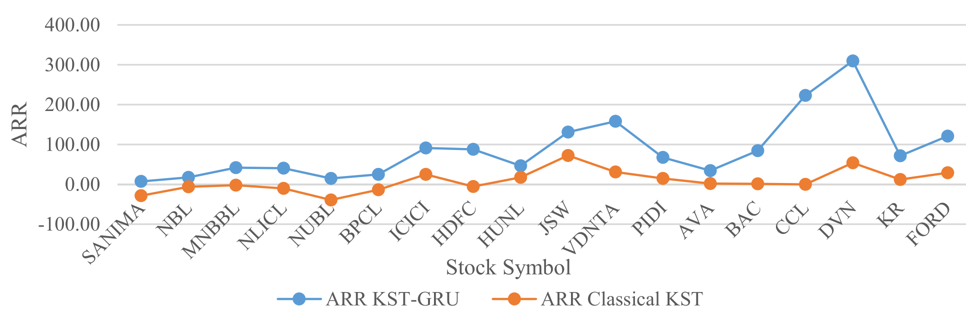

Fig. 18. ARR Obtained from KST-GRU and classical KST trading strategies.

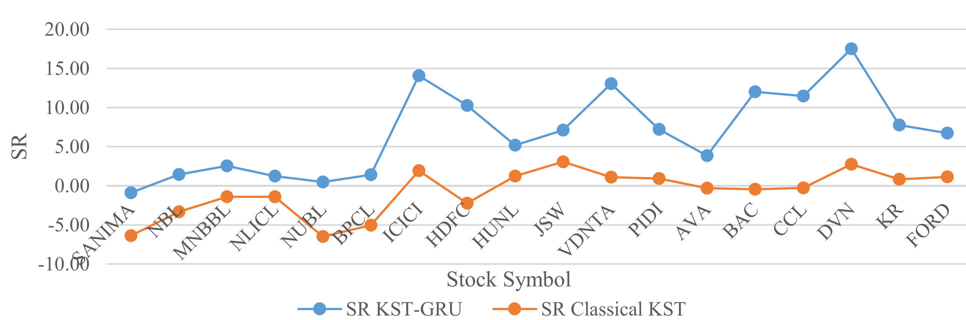

Fig. 19. SR Obtained from KST-GRU and classical KST trading strategies.

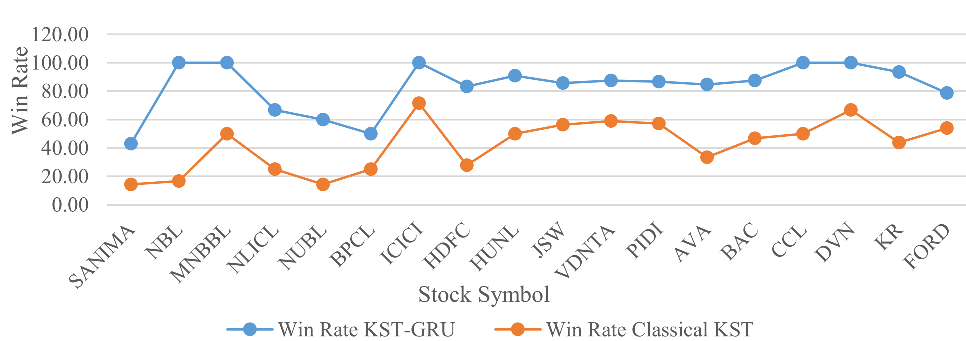

Fig. 20. Win Rate of KST-GRU and classical KST trading strategies.

Table 5
Mean values of performance measures for KST based trading strategies.

Performance MeasuresTrading Strategy	Mean ARR	Mean SR	Mean WR
Intelligent Strategy	84.98	6.80	83.20
Classical Strategy	8.46	−0.80	42.27

6, this study concludes the following insights:

- Look back period value 5 is optimal for MACD and DMI based intelligent trading strategies, and look back period value between 9 and 11 is optimal for KST based intelligent trading strategies, because these strategies produced higher ARR for the majority of stocks when the aforementioned look back period values were used.
- Intelligent trading strategies based on trend indicators implemented with GRU are better than the strategies implemented with LSTM. MACD-GRU strategy outperformed MACD-LSTM strategy for 12 of the 18 stocks, DMI-GRU strategy outperformed DMI-LSTM strategy for 14 of the 18 stocks, and KST-GRU strategy outperformed KST-LSTM strategy for 12 of the 18 stocks.
- Intelligent trading methods using MACD, DMI, and KST indicators consistently outperformed classical trading strategies in terms of ARR, SR, and win rate. The performance indicators from these intelligent techniques were significantly better than those from classical strategies.
- Among the three intelligent trading methods studied, the MACD-based approach stands out as the most profitable, least risky, and most successful. The MACD-GRU approach achieved the highest ARR for 12 out of 18 equities. Intelligent trading techniques using MACD, DMI, and KST showed average SR values of 9.97, 7.35, and 6.8 respectively. On average, these techniques resulted in profitable stock transactions of 86.76 %, 80.97 %, and 83.2 %.

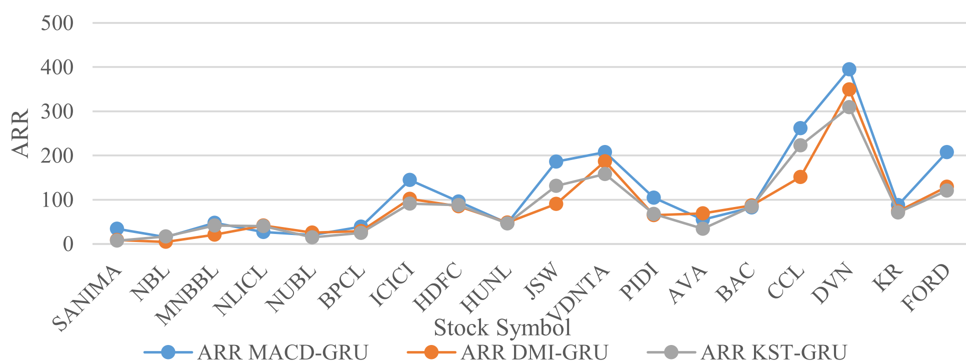

Fig. 21. ARR Obtained from intelligent trading strategies.

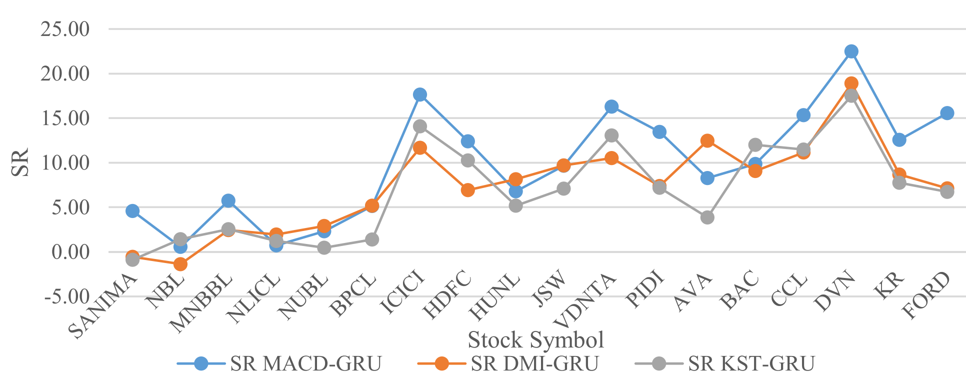

Fig. 22. SR Obtained from intelligent trading strategies.

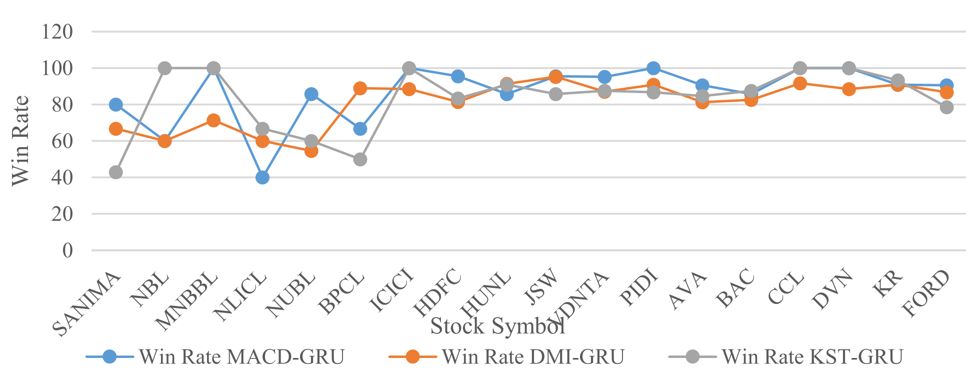

Fig. 23. Win rates achieved for intelligent trading strategies.

Funding sources

This research is not funded by any agency.

Permission note

No copyrighted work is used in this research article

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used ChatGPT and QuillBot in order to improve language quality. After using this tool/service, the authors reviewed and edited the content as needed and take (s) full responsibility for the content of the publication.

Author contributions

Arjun Singh Saud is the primary individual responsible for conceptualization, implementation, data analysis, and writing the research report. Subarna Shakya served as an advisor throughout all stages of the research and also reviewed the report.

Ethical statement

This study does not involve any research with human participants or animals.

CRediT authorship contribution statement

Arjun Singh Saud: Writing – original draft, Validation, Software, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Subarna Shakya:** Writing – review & editing, Validation, Supervision.

Declaration of Competing Interest

There is no conflict of interest

References

Arévalo, R., García, J., Guijarro, F., Peris, A., 2017. A dynamic trading rule based on filtered flag pattern recognition for stock market price forecasting. Expert Syst. Appl. vol. 81, 177–192. https://doi.org/10.1016/j.eswa.2017.03.028.

Arévalo, A., Niño, J., Hernández, G., Sandoval, J., 2016. High-frequency trading strategy based on deep neural networks. Intell. Comput. Methodol. 424–436. https://doi.org/10.1007/978-3-319-42297-8_40.

Ayala, J., García-Torres, M., Noguera, J.L.V., Gómez-Vela, F., Divina, F., 2021. Technical analysis strategy optimization using a machine learning approach in stock market indices. Knowl. Based Syst. vol. 225, 107119. https://doi.org/10.1016/j.knosys.2021.107119.

Baresa, S., Bogdan, S., Ivanovic, Z., 2013. Strategy of stock valuation by fundamental analysis. UTMS J. Econ. vol. 4 (1), 45–51. ⟨https://hdl.handle.net/10419/105304⟩ (Available).

Bustos, O., Pomares-Quimbaya, A., 2020. Stock market movement forecast: a systematic review. Expert Syst. Appl. vol. 156, 113464. https://doi.org/10.1016/j.eswa.2020.113464.

Chakrabarty, A., De, A., Dubey, R., 2014. A flexible approach towards multi-frequency Re-engineering of the moving average convergence divergence indicator. Glob. J. Flex. Syst. Manag. vol. 15 (3), 219–234. https://doi.org/10.1007/s40171-014-0068-7.

Chandar, S.K., 2022. Convolutional neural network for stock trading using technical indicators. Autom. Softw. Eng. vol. 29 (1). https://doi.org/10.1007/s10515-021-00303-z.

Chen, Y., Hao, Y., 2018. Integrating principle component analysis and weighted support vector machine for stock trading signals prediction. Neurocomputing vol. 321, 381–402. https://doi.org/10.1016/j.neucom.2018.08.077.

Chen, Y., Hao, Y., 2020. A novel framework for stock trading signals forecasting. Soft Comput. vol. 24 (16), 12111–12130. https://doi.org/10.1007/s00500-019-04650-8.

Chen, W., Jiang, M., Zhang, W.-G., Chen, Z., 2021. A novel graph convolutional feature based convolutional neural network for stock trend prediction. Inf. Sci. vol. 556, 67–94. https://doi.org/10.1016/j.ins.2020.12.068.

Chen, L., Sun, C.-M., Chen, M.-E., Wu, Jimmy Ming-Tai Wu, Dec. 2021. Stock trading system based on machine learning and kelly criterion in internet of things. Wirel. Commun. Mob. Comput. vol. 2021, 1–9. https://doi.org/10.1155/2021/7632052.

Cho, K., van Merrienboer, B., Bahdanau, D., Bengio, Y., 2014. On the properties of neural machine translation: encoder–decoder approaches. Proc. SSST-8 Eighth Workshop Syntax Semant. Struct. Stat. Transl. https://doi.org/10.3115/v1/w14-4012.

Farias Nazário, R.T., e Silva, J.L., Sobreiro, V.A., Kimura, H., 2017. A literature review of technical analysis on stock markets. Q. Rev. Econ. Financ. vol. 66, 115–126. https://doi.org/10.1016/j.qref.2017.01.014.

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

Jakpar, S., Tinggi, M., Tak, A.H., Chong, W.Y., 2018. Fundamental analysis VS technical analysis: the comparison of two analysis in malaysia stock market. UNIMAS Rev. Account. Financ. vol. 1 (1). https://doi.org/10.33736/uraf.1208.2018.

Kumar, D., Sarangi, P.K., Verma, R., 2021. A systematic review of stock market prediction using machine learning and statistical techniques. Mater. Today. Proc. https://doi.org/10.1016/j.matpr.2020.11.399.

Kuo, Wei-Lun, Chang, W.-L., Dai, T.-S., Chen, Y., Chang, H.-H., Jan. 2022. Improving pairs trading strategies using two-stage deep learning methods and analyses of time (In)variant inputs for trading performance. IEEE Access vol. 10, 97030–97046. https://doi.org/10.1109/access.2022.3204056.

Lam, M., 2004. Neural network techniques for financial performance prediction: integrating fundamental and technical analysis. Decis. Support Syst. vol. 37 (4), 567–581. https://doi.org/10.1016/s0167-9236(03)00088-5.

Lee, M.-C., 2022. Research on the feasibility of applying GRU and attention mechanism combined with technical indicators in stock trading strategies. Appl. Sci. vol. 12 (3), 1007. https://doi.org/10.3390/app12031007.

Lee, M.-C., Chang, J.-W., Yeh, S.-C., Chia, T.-L., Liao, J.-S., Chen, X., 2022. Applying attention-based BiLSTM and technical indicators in the design and performance analysis of stock trading strategies. Neural Comput. Appl. vol. 34 (16), 13267–13279. https://doi.org/10.1007/s00521-021-06828-4.

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

Lien Minh, D., Sadeghi-Niaraki, A., Huy, H.D., Min, K., Moon, H., 2018. Deep learning approach for short-term stock trends prediction based on two-stream gated recurrent unit network. IEEE Access vol. 6, 55392–55404. https://doi.org/10.1109/ACCESS.2018.2868970.

Long, W., Lu, Z., Cui, L., 2019. Deep learning-based feature engineering for stock price movement prediction. Knowl. Based Syst. vol. 164, 163–173. https://doi.org/10.1016/j.knosys.2018.10.034.

Lv, D., Wang, D., Li, M., Xiang, Y., 2020. DNN models based on dimensionality reduction for stock trading. Intell. Data Anal. vol. 24 (1), 19–45. https://doi.org/10.3233/ida-184403.

Nabipour, M., Nayyeri, P., Jabani, H., S. S, Mosavi, A., 2020. Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data; a comparative analysis. IEEE Access vol. 8, 150199–150212. https://doi.org/10.1109/access.2020.3015966.

Nepse price export, https://nepsealpha.com/nepse-data (accessed Aug. 22, 2024).

Nikou, M., Mansourfar, G., Bagherzadeh, J., 2019. Stock price prediction using DEEP learning algorithm and its comparison with machine learning algorithms. Intell. Syst. Account. Financ. Manag. vol. 26 (4). https://doi.org/10.1002/isaf.1459.

Nti, K., Adekoya, A.F., Weyori, B.A., 2019. A systematic review of fundamental and technical analysis of stock market predictions. Artif. Intell. Rev. vol. 53 (1), 3007–3057. https://doi.org/10.1007/s10462-019-09754-z.

Oyewola, D.O., Gbenga Dada, E.G., Olaoluwa, O.E., Al-Mustapha, K.A., 2019. Predicting Nigerian stock returns using technical analysis and machine learning. Eur. J. Electr. Eng. Comput. Sci. vol. 3 (2). https://doi.org/10.24018/ejece.2019.3.2.65.

Sang, C., Di Pierro, M., 2019. Improving trading technical analysis with TensorFlow long short-term memory (LSTM) neural network. J. Financ. Data Sci. vol. 5 (1), 1–11. https://doi.org/10.1016/j.jfds.2018.10.003.

Sezer, O.B., Gudelek, M.U., Ozbayoglu, A.M., 2020. Financial time series forecasting with deep learning: a systematic literature review: 2005–2019. Appl. Soft Comput. vol. 90, 106181. https://doi.org/10.1016/j.asoc.2020.106181.

Shah, D., Isah, H., Zulkernine, F., 2019. Stock market analysis: a review and taxonomy of prediction techniques. Int. J. Financ. Stud. vol. 7 (2), 26. https://doi.org/10.3390/ijfs7020026.

Shi, M., Zhao, Q., 2020. Stock market trend prediction and investment strategy by deep neural networks. 2020 11th Int. Conf. Aware. Sci. Technol. https://doi.org/10.1109/icast51195.2020.9319488.

Shynkevich, Y., McGinnity, T.M., Coleman, S.A., Belatreche, A., Li, Y., 2017. Forecasting price movements using technical indicators: investigating the impact of varying input window length. Neurocomputing vol. 264, 71–88. https://doi.org/10.1016/j.neucom.2016.11.095.

S. Siami-Namini, N. Tavakoli, and Namin, Akbar Siami, "A Comparative Analysis of Forecasting Financial Time Series Using ARIMA, LSTM, and BiLSTM," *arXiv.org*, 2019. http://arxiv.org/abs/1911.09512 (accessed Aug. 21, 2024).

Siami-Namini, S., Tavakoli, N., Siami Namin, A., 2018. A Comparison of ARIMA and LSTM in Forecasting Time Series. 2018 17th IEEE Int. Conf. Mach. Learn. Appl. https://doi.org/10.1109/icmla.2018.00227.

Silva, T.R., Li, A.W., Pamplona, E.O., 2020. Automated trading system for stock index using LSTM neural networks and risk management. 2020 Int. Conf. Neural Netw. https://doi.org/10.1109/ijcnn48605.2020.9207278.

Stock prices. Available at: https://www.bseindia.com/markets/equity/EQReports/StockPrcHistori.aspx (Accessed: 22 August 2024).

Stoean, C., Paja, W., Stoean, R., Sandita, A., 2019. Deep architectures for long-term stock price prediction with a heuristic-based strategy for trading simulations. PLOS ONE vol. 14 (10), e0223593. https://doi.org/10.1371/journal.pone.0223593.

Tharavanij, P., Siraprapasiri, V., Rajchamaha, K., 2015. Performance of technical trading rules: evidence from Southeast Asian stock markets. SpringerPlus vol. 4 (1). https://doi.org/10.1186/s40064-015-1334-7.

Touzani, Y., Douzi, K., 2021. An LSTM and GRU based trading strategy adapted to the Moroccan market. J. Big Data vol. 8 (1). https://doi.org/10.1186/s40537-021-00512-z.

Troiano, L., Villa, E.M., Loia, V., 2018. Replicating a trading strategy by means of LSTM for financial industry applications. IEEE Trans. Ind. Inform. vol. 14 (7), 3226–3234. https://doi.org/10.1109/tii.2018.2811377.

Wen, M., Li, P., Zhang, L., Chen, Y., 2019. Stock market trend prediction using high-order information of time series. IEEE Access vol. 7, 28299–28308. https://doi.org/10.1109/access.2019.2901842.

Wu, Jimmy Ming-Tai, Sun, L., Srivastava, G., Lin, Jerry Chun-Wei, 2021. A novel synergetic LSTM-GA stock trading suggestion system in internet of things. Mob. Inf. Syst. vol. 2021, 1–15. https://doi.org/10.1155/2021/6706345.

Yahoo Finance - Stock Market Live, quotes, Business & Finance News Yahoo! Finance. Available at: https://finance.yahoo.com/ (Accessed: 22 August 2024).

Yamak, P.T., Yujian, L., Gadosey, P.K., 2019. A Comparison between ARIMA, LSTM, and GRU for time series forecasting. Proc. 2019 2nd Int. Conf. Algorithms Comput. Artif. Intell. https://doi.org/10.1145/3377713.3377722.

Yang, Z., Hao, K., Cai, X., Chen, L., Ren, L., 2019. Prediction of stock trading signal based on multi-indicator channel convolutional neural networks. 2019 IEEE 8th Data Driven Control Learn. Syst. Conf. https://doi.org/10.1109/ddcls.2019.8908881.

Yu, Y., Si, X., Hu, C., Zhang, J., 2019. A review of recurrent neural networks: LSTM cells and network architectures. Neural Comput. vol. 31 (7), 1235–1270. https://doi.org/10.1162/neco_a_01199.

Zhang, X., Tan, Y., 2018. Deep stock ranker: a LSTM neural network model for stock selection. Third Int. Conf. *2018*, 614–623. https://doi.org/10.1007/978-3-319-93803-5_58.