International Journal of Electrical and Computer Engineering (IJECE)
Vol. 12, No. 4, August 2022, pp. 4185~4194
ISSN: 2088-8708, DOI: 10.11591/ijece.v12i4.pp4185-4194

Directional movement index based machine learning strategy for predicting stock trading signals

Arjun Singh Saud[1], Subarna Shakya[2]
[1]Central Department of Computer Science and Information Technology, Tribhuvan University, Kathmandu, Nepal
[2]Institute of Engineering, Tribhuvan University, Lalitpur, Nepal

Article Info

Article history:

Received Jun 11, 2021
Revised Dec 21, 2021
Accepted Jan 20, 2022

Keywords:

Direction movement index
Gated recurrent unit
Intelligent stock trading
Trading signal prediction

ABSTRACT

Intelligent stock trading systems are demand of the modern information age. This research paper proposed a directional movement index based machine learning (DMI-ML) strategy for predicting stock trading signals. Performance of the proposed strategy was evaluated in terms of annual rate of return (ARR), Sharpe ratio (SR), and percentage of profitable trades executed by the trading strategy. In addition, performance of the proposed model was evaluated against the strategies viz. traditional DMI, buy-hold. From the experimental results, we observed that the proposed strategy outperformed other strategies in terms of all three parameters. On average, the ARR obtained from the DMI-ML strategy was 52.58% higher than the ARR obtained from the buy-hold strategy. At the same time, the ARR of the proposed one was found 75.12% higher than the ARR obtained from the traditional DMI strategy. Furthermore, the Sharpe ratio for the DMI-ML strategy was positive for all stocks. On the other side, the percentage of profitable trades executed by the DMI-ML strategy soared in comparison to the percentage of such trades by the traditional DMI. This study also extended analysis of the proposed model with the various intelligent trading strategies proposed by authors in various literatures and concluded that the proposed DMI-ML strategy is the better strategy for stock trading.

Corresponding Author:

Arjun Singh Saud
Central Department of Computer Science and Information Technology, Tribhuvan University
Kirtipur, Kathmandu, Nepal
Email: arjunsaud@cdcsit.edu.np

1. INTRODUCTION

Many people around the world are involved in stock trading. Their main objective is to increase wealth. They use their knowledge and experience to predict the future direction of the stocks and trade accordingly. Researchers mainly from finance, statistics, and computer science are actively involved in devising suitable models for predicting the future direction of stocks. They use fundamental analysis, technical analysis, and time series forecasting approaches for stock forecasting. Fundamental analysts use company analysis, industry analysis, and economic analysis to forecast future stock directions [1]. Technical analysts use candlestick charts and various technical indicators to predict future stock movements [2], [3]. Time series forecasting is an approach that uses linear and non-linear models to predict the future direction of stocks [4], [5].

Researches related to stock market prediction can be basically divided into three categories: i) prediction of stock trading signals [6]–[13], ii) prediction of future stock price [14]–[31], and iii) prediction of the stock market index [16], [17], [30], [31]. In addition, the input features used by researchers to predict the stock market can be broadly divided into four categories: i) historical trading data that include open, high, low, close, and volume [6], [29]; ii) technical indicators computed from historical trading data [4], [16], [18]; iii) investor sentiment mined from financial news, blogs, and tweets, [19], [21], [28]; and iv) combination of input features from categories 1 to 3.

Technical indicators are pattern-based signals generated by the price and volume of historical trading. Some commonly used technical indicators include simple moving average (SMA), exponential moving average (EMA), moving average convergence and divergence (MACD), directional movement index (DMI), relative strength index (RSI), commodity channel index (CCI), and William %R [22], [23]. DMI is one of the preferred and widely used technical indicators among technical analysts to predict future stock price movements and trading signals [32]–[34]. They need to analyze a large volume of historical trading data to predict stock trading signals. Their approach is largely subjective also and hence very difficult to computerize [35]. Therefore, machine learning strategies became popular for predicting stock price trends and trading signals [36], [37]. These strategies are not able to remember the historical context, but recurrent neural networks (RNNs) have this capability. Therefore, researchers used RNNs to predict stock price movements [38], [39]. Since RNNs suffer from vanishing/exploding gradient descent problem, nowadays long short-term memory (LSTM) and gated recurrent unit (GRU) networks are among the widely used models for stock forecasting [6], [7], [14], [23], [39]. However, none of the aforementioned strategies have utilized DMI indicators for trading signal prediction.

One of the major problems associated with DMI based trading strategy used by technical analysts is that DMI indicators give false signals frequently, which are very difficult to recognize. On the other hand, looking at and analyzing the chart patterns of each stock is a time-consuming task. Therefore, this research paper proposed a novel directional movement index based machine learning (DMI-ML) strategy for predicting stock trading signals. Although many intelligent strategies were proposed to predict stock trading signals [21], [27], [31], none of them was based on DMI and similar technical indicators. The main motive of the proposed strategy is to devise a machine learning strategy on the basis of estimated trading signals from DMI indicators to reduce the false trading signals and increase trading profit. The detail of the proposed strategy is given in section 3. We used the GRU network as a machine learning tool to realize the proposed DMI-ML strategy and evaluated the strategy in terms of annual rate of return (ARR), Sharpe ratio (SR), and percentage of profitable trades. Moreover, the study compared the performance of the proposed model with the traditional DMI based trading strategy and the Buy-Hold strategy, a benchmark strategy.

This paper is organized as follows. Section 2 describes the DMI indicators briefly. The proposed strategy is explained in section 3. The research method adopted to carry out this research work is discussed in section 4. Section 5 sheds light on the results and discussion. Finally, section 6 provides the conclusion of the research work.

2. DIRECTIONAL MOVEMENT INDEX (DMI)

Directional movement index (DMI) is a trend and momentum indicator that signifies the strength of a stock price trend. The indicator does not determine the direction of the trend, rather it only determines the strength of the current trend. The average directional movement index (ADMI) is derived from the directional movement indicators: positive directional indicator (+DI) and negative directional indicator (–DI). +DI increases when movement is upward and –DI increases when movement is downward. Normally, the ADMI value below 25 suggests the absence of the trend and the ADMI value above 25 suggests the presence of the strong trend. It shows the strength of uptrend when +DI is above –DI and strength of downtrend when –DI is above +DI. The general strategy used by trend traders is to buy stocks when the +DI line crosses above the –DI line and sell stocks when the –DI line crosses above the +DI line [40]. The process of calculating DMI indicators is discussed below and is formulated in (1): i) compute positive directional movement (+DM) and negative directional movement (-DM) using current high (CH), previous high (PH), current low (CL) and previous low (PL); ii) compute smoothed +DM (+SDM) and smoothed -DM (-SDM) using 14 days period as suggested by Wilder [40]; iii) compute true range (TR), smoothed TR, and average TR. Here, TR is a market volatility indicator; iv) divide +SDM by ATR and multiply the value by 100 to get +DI; v) similarly, divide -SDM by ATR and multiply the value by 100 to get –DI; vi) compute DMI indicator, which is 100 times the absolute difference between DI indicators divided by the sum of DI indicators; and vii) finally, compute ADMI indicator, which is smoothed value of the DMI indicator.

$$\mathrm{Up} = \mathrm{CH} - \mathrm{PH} \ \mathrm{Down} = \mathrm{PL\text{-}CL}$$
$$\begin{cases} \text{if Up} > \text{Down and Up} > 0 \text{ then } +\mathrm{DM} = \text{Up else } +\mathrm{DM} = 0 \\ \text{if Down} > \text{Up and Down} > 0 \text{ then -DM} = \text{Down else -DM} = 0 \end{cases}$$
$$\pm\mathrm{SDM} = \sum_{i=1}^{n} \mathrm{DM} - \frac{1}{n}\sum_{i=1}^{n} \mathrm{DM} + \text{Current DM}$$

$$\mathrm{TR} = \max\{(\mathrm{CH} - \mathrm{CL}), \mathrm{abs}(\text{CH-PC}), \mathrm{abs}(\text{CL-PC})\}$$
$$\mathrm{STR} = \sum_{i=1}^{n} \mathrm{TR} - \frac{1}{n}\sum_{i=1}^{n} \mathrm{TR} + \text{Current TR} \quad \mathrm{ATR} = \frac{1}{n}\sum_{i=1}^{n} \mathrm{TR}_i$$
$$+\mathrm{DI} = \frac{+\mathrm{SDM}}{\mathrm{ATR}} \times 100 \quad -\mathrm{DI} = \frac{-\mathrm{SDM}}{\mathrm{ATR}} \times 100$$
$$\mathrm{DMI} = \left(\frac{|(+\mathrm{DI}) - (\text{-DI})|}{(+\mathrm{DI}) + (\text{-DI})}\right) \times 100 \quad \mathrm{ADMI} = \frac{\text{Prior ADMI} \times (\text{n-1}) + \text{Current ADMI}}{n} \tag{1}$$

3. PROPOSED STRATEGY

This study proposed a DMI-ML stock trading strategy as well as a target feature generation model for producing buy/hold/sell signals. In addition, we designed and constructed a stock trading simulation module. This section goes over each of these components in detail.

3.1. Proposed DMI-ML Strategy

A schematic diagram of the proposed directional movement index based machine learning (DMI-ML) strategy for predicting stock trading signals is presented in Figure 1. The target feature generation method is discussed in section 3.2. Details of dataset preparation using target feature and DMI indicators is provided in section 4.3. GRU network uses this dataset for training and predicting trading signals. Configuration of the GRU network is provided in section 4.4. Finally, based on the predicted signals, the trading simulation module performs stock trading as discussed in section 3.3.

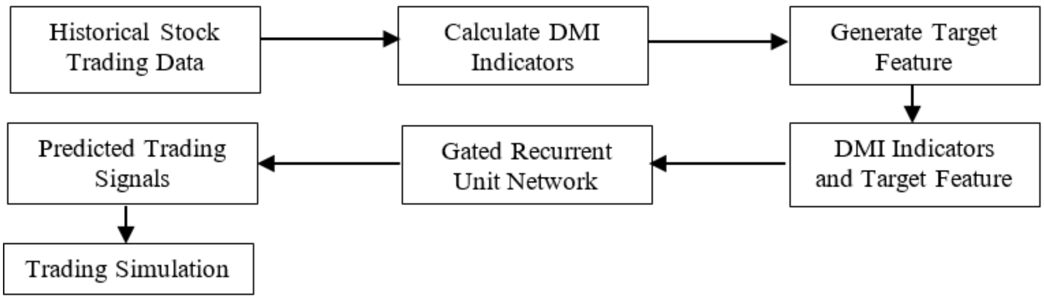

Figure 1. Proposed DMI-ML trading strategy

3.2. Target feature generation model

This paper also put forward a model for generating target feature "action" based on values of directional indices. The mathematical formulation of the model is given in (2). The model generates a sequence of Buy, Hold, and Sell actions. It generates Buy and Sell signals turn by turn. The model puts three Buy and Sell signals in the target feature when there is a crossover between the +DI line and the –DI line. The main reason behind this is to make the machine learning model easier to recognize trading signals.

$$\begin{cases} \text{if PDI} > \text{NDI} & \text{then} \quad \text{action[i:i} + 3] = \text{'Buy'} \\ \text{if NDI} > \text{PDI} & \text{then} \quad \text{action[i:i} + 3] = \text{'Sell'} \\ \text{otherwise} & \qquad\quad \text{action}[i] = \text{'Hold'} \end{cases} \tag{2}$$

Where PDI and NDI are +DI and –DI indicators respectively.

3.3. Trading simulation

During this study, a trading simulator has been designed and developed to perform automatic stock trading based on the trading signals predicted. The simulator performs buy and sell operations on the assumption that stock investors initially hold no stocks. Hence, the first trading operation of the simulator should be the buy operation. Thus, the simulator performs a sequence of buy and sell operations. If the last trading operation is the sell operation, then the sequence consists of the same number of buy and sell operations. However, if the last trading operation is the buy operation, then the sequence includes one more buy operation than the number of sell operations. In such situation, the simulator discards the last buy operation and performs profit/loss calculation derived from stock trading. The simulator treats the "close price" as the buy/sell price. Initially, the simulator assumes that investor holds some seed money for purchasing the stocks. Once the stocks are sold, the entire sold value is reinvested in the next buy operation. The formula used to calculate gross profit/loss from the trading is given in (3). The equation does not include transaction cost and income gain tax. After calculating the profit/loss, the simulator calculates profit/loss percentage using (4):

$$pl = y - x \tag{3}$$

where y is the amount of money obtained from the last sell operation and x is the amount of money that the investor decides to invest in a stock initially.

$$plp = \frac{pl}{x} x100 \tag{4}$$

The traditional DMI trading strategy also uses the above-mentioned technique for calculating profit/loss percentage. On the other hand, the buy-hold strategy operates buy and sell operations only once. The assumption for the strategy is that the buy operation is performed on the first day and the sell operation is performed on the last day of the test period. The amount of money obtained from the sell operation was calculated using (5). Then gross profit/loss and percentage of profit/loss for the buy-hold strategy were calculated using (3) and (4), respectively:

$$y = (C_L \times (ns + k) + \sum_{i=1}^{n} cd_i) \tag{5}$$

where, C_i is close price of i^{th} trading day, ns is number of stocks purchased initially, k is the number of bonus share issued and cd_i is the i^{th}cash dividend provided during the test period.

4. RESEARCH METHOD

This section describes the stock data that was used in the research work. Aside from that, the research work's data preprocessing and data preparation strategy are discussed. Finally, the configuration of the GRU network used in the study, as well as the performance measures used to evaluate the stock trading strategies, are described.

4.1. Stock data

In this research, historical trading data from Nepal stock exchange (NEPSE), Bombay stock exchange (BSE), New York stock exchange (NYSE), and Shanghai stock exchange (SSE) were used. This study experimented twenty stocks, five from each stock exchange. Detail of the stock data is given in Table 1. The historical trading data of NEPSE, BSE, NYSE and SSE stocks were collected from Nepal Stock Exchange [41], BSE India [42] and Yahoo Finance [43] respectively.

Table 1. Historical trading data of stocks

Stock Exchange	Stock Name	Date	
		From	To
NEPSE	Siddhartha Bank Ltd. (SBL)	4/15/2010	4/13/2021
	Everest Bank Ltd. (EBL)	4/15/2010	4/13/2021
	Garima Bikas Bank Ltd. (GBBL)	1/3/2011	4/13/2021
	Asian Life Insurance (ALICL)	5/12/2011	4/13/2021
	Butwal Power Company Ltd. (BPCL)	4/15/2010	4/13/2021
	Britannia Industries Ltd. (BRTN)	1/1/2000	4/13/2021
	HDFC Bank Ltd. (HDFC)	1/1/2000	4/13/2021
BSE	Hindustan Unilever Ltd. (HUNL)	1/1/2000	4/13/2021
	Shree Cement Ltd. (SHREE)	1/1/2000	4/13/2021
	Pidilite Industries Ltd. (PIDI)	1/1/2000	4/13/2021
	Avista Corporation (AVA)	1/1/2000	4/13/2021
	Boeing Company (BA)	1/1/2000	4/13/2021
NYSE	Bank of America Corporation (BAC)	1/1/2000	4/13/2021
	BlackBerry Ltd. (BB)	1/1/2000	4/13/2021
	Ford Motor Company (Ford)	1/1/2000	4/13/2021
	Fujian Qingshan Paper Industry Co. Ltd. (FQP)	1/1/2000	4/13/2021
	Huaxin Cement Co., Ltd. (HCC)v	1/1/2000	4/13/2021
SSE	Jiangsu Hengrui Medicine Co.Ltd. (JHM)	10/18/2000	4/13/2021
	China Minsheng Banking Corp. Ltd. (MSHEG)	12/19/2000	4/13/2021
	Sichuan Mingxing Electric (SME)	1/1/2000	4/13/2021

4.2. Data preprocessing

Initially, the data was arranged in the order of oldest to the newest date. Thereafter, the DMI indicators: +DI, -DI, and ADMI were calculated from the historical trading data and the unnecessary features were dropped from the datasets. Then, the target feature 'action' was generated using the target feature generation model discussed in section 3.2. Finally, the input features were normalized using standard scalar and the target attribute was encoded using the one-hot encoding strategy.

4.3. Data preparation

The datasets were divided into training, validation, and test sets in 8:1:1 ratio. This study aimed to predict the trading signal for the day t+1 using input features from $(t\text{-}N+1)^{th}$ day to t^{th} day, where t is the current trading day and N is the window size. As suggested by Saud and Shakya [44], window size 5 was used. Therefore, the input data was a combination of N independent variables $d_{t-N+1} \dots d_{t-1}$, d_t and a dependent variable a_{t+1}, where d_i is a tuple (pd_i, nd_i, ad_i) and a_i represents trading action for the i^{th} trading day. The symbols pd_i, nd_i, and ad_i, denote +DI, -DI, and ADMI indicators respectively for the i^{th} trading day. Thus, $t \in [1, L*0.8]$ for the training set, $t \in [L*0.8+1, L*0.9]$ for the validation set, and $t \in [L*0.9+1, L]$ for the test set, where L is the length of the dataset.

4.4. Configuration of GRU network

The configuration of the GRU network used in this research work was 3×100×100×3. The configuration is not guaranteed to be optimal, however, random experiments were performed with the various configurations before adopting this one. Each hidden layer of the network was followed by a dropout layer. The ReLU activation function was used in the hidden layers and the SoftMax activation function was used in the output layer of the network. In addition, Adam optimizer was used with the GRU network.

4.5. Performance measures

In this study, trading strategies were evaluated in terms of three measures: annual rate of return (ARR), Sharpe ratio (SR), and percentage of profit/loss trades executed by the trading strategies. SR is the risk-adjusted return of the investment [45]. It measures the volatility of investment returns. ARR and SR were computed from the profit earned from the trading strategies using the formulae given in (6) and (7), respectively. Since the generated target feature 'action' is simply an estimation based on the crossover between positive and negative directional indicators, classification accuracy is not a relevant performance measure for the proposed strategy.

$$ARR = \left\{(1 + Return)^{1/n} - 1\right\} \times 100 \tag{6}$$

Where, n is the number of years.

$$SR = \frac{R_t - R_f}{\sigma} \tag{7}$$

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

5. RESULTS AND DISCUSSION

This section presents the results derived from stock trading in terms of ARR and SR. Then, the percentage of profit/loss trades executed by the trading strategies is discussed. Finally, results obtained from the proposed DMI-ML strategy are compared with the results obtained from various intelligent trading strategies proposed in the literature.

5.1. Analysis of trading returns

A trading simulation was carried out on the test data based on the trading signals generated by the DMI-ML strategy and the traditional DMI trading strategy. Then, profit/loss obtained from the trading strategies as well as from the Buy-Hold strategy were computed. Furthermore, the standard deviation of the profit/loss obtained from the DMI-ML was calculated. Thereafter, ARR and SR were calculated for each strategy. The corresponding results are plotted in Figures 2 and 3, respectively. The annual rate of treasury bills issued by the central banks of respective countries was used as a risk-free return for calculating the Sharpe ratio.

Figures 2 and 3 clearly show that the DMI-ML Strategy outperformed the buy-hold strategy and the traditional DMI trading strategy in terms of ARR and SR. The DMI-ML strategy yielded the highest ARR for all companies. The strategy achieved ARR ranging from 18.17% to 158.28% from stock trading with SR values ranging from 4.14 to 58.24. The DMIL-ML strategy not only yielded a handsome return from the stocks that were in the bullish pattern but also from the stocks that were in the bearish and sideways pattern during the test period. On the other hand, the buy-hold strategy yielded ARR in the range of -21.24% to 183.06% and SR range from -6.06 to 57.8. From the experiments, we observed that the buy-hold strategy only yielded a handsome return from the stocks that were in the bullish pattern during the test period. However, the strategy could not achieve satisfactory returns from the stocks in the bearish and sideways pattern. The traditional DMI trading strategy performed worst among all three trading strategies. The strategy yielded ARR ranging from -1.89% to 19.38% from stock trading. The reason behind the unsatisfactory performance of the strategy is that the DMI is a lagging indicator and it also gives false trading signals frequently. The traditional strategy achieved the specified ARR with SR in the interval -1.31 to 7.83.

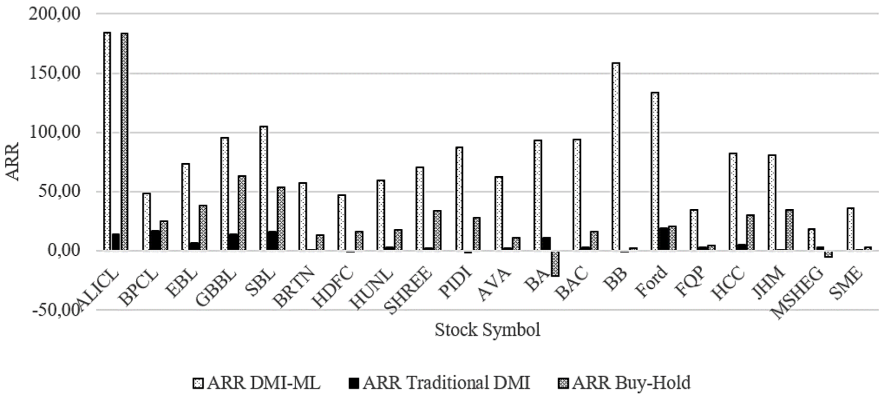

Figure 2. ARR obtained from trading

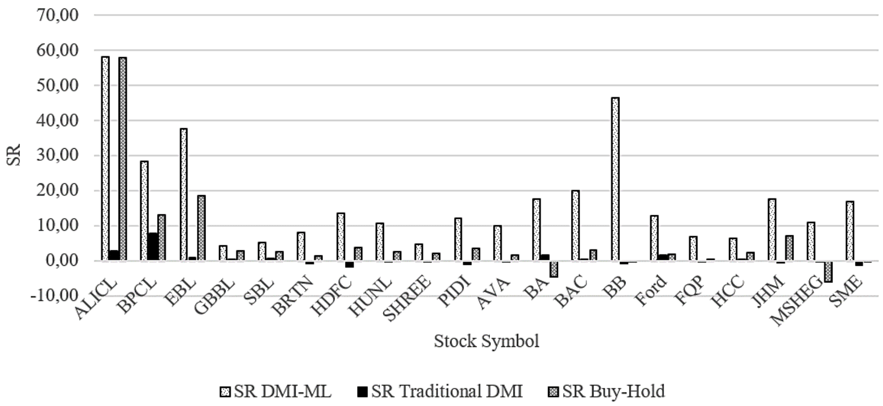

Figure 3. SR obtained from trading strategies

The DMI-ML strategy achieved 1.38% to 156.01% higher ARR than the Buy-Hold strategy and 15.54% to 170.86% higher ARR than the traditional DMI trading strategy. The strategy earned the specified ARR with positive SR for all stocks. The buy-hold strategy yielded the specified ARR with positive SR for 16 stocks and with negative SR for the remaining 4 stocks. At the same time, the traditional DMI based trading strategy yielded the above-mentioned ARR with positive SR only for 9 stocks and with negative SR for the remaining 11 stocks. Furthermore, the DMI-ML strategy obtained significantly higher SR than the Buy-Hold strategy and the traditional DMI based trading strategy.

In summary, the DMI-ML strategy achieved 80.97% average ARR with 17.36 average SR from trading, whereas the traditional DMI strategy and the buy-hold strategy respectively achieved 5.85% and 28.39% average ARR with an average SR value of 0.44 and 5.61 as shown in Figure 4. These observations point to the fact that the proposed DMI-ML strategy is a better approach for automated stock trading, as it achieved much higher ARR as well as risk-adjusted return compared to the other two stock trading strategies.

5.2. Analysis of percentage of profitable trades

The percentage of profit/loss trades is one of the key parameters for any stock trading strategy. This parameter was therefore measured through the trading simulation module. The graph illustrated in Figure 5 provides a comparison between the percentage of profitable trades executed by the DMI-ML strategy and the traditional DMI based trading strategy. The analysis of the buy-hold strategy is not included in this regard, as it always conducts only one trade for each stock during the test period.

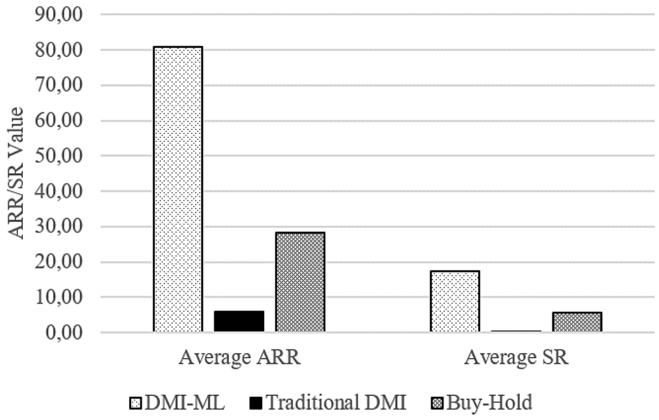

Figure 4. Average ARR and SR obtained from trading strategies

From Figure 5, we observed that the percentage of gain trades executed by the DMI-ML strategy is much higher than the percentage of gain trades executed by the traditional DMI strategy. 66.67% to 100% of trades executed by the DMI-ML strategy are gain trades, whereas only 19.05% to 100% of trades executed by the traditional DMI based trading strategy are gain trades. During analysis, it was observed that the traditional DMI strategy had an exceptional performance for the stock ALICL with 100% gain trades. However, for the rest of the stocks, the strategy executed a significantly lower percentage of gain trades than the DMI-ML strategy. The reason behind the outperformance of the traditional DMI strategy for the stock ALICL is that the stock was in an extremely bullish trend during the test period. Excluding ALICL, the DMI-ML strategy executed 8.7% to 64.71% more gain trades than the traditional DMI strategy. In summary, the DMI-ML trading strategy executed 89.15% gain trades on average, whereas the traditional DMI based trading strategy executed merely 41.33% gain trades. These observations indicated that numerous trade signals given by directional indicators were false signals and the proposed DMI-ML stock trading strategy was able to filter out those false trading signals significantly.

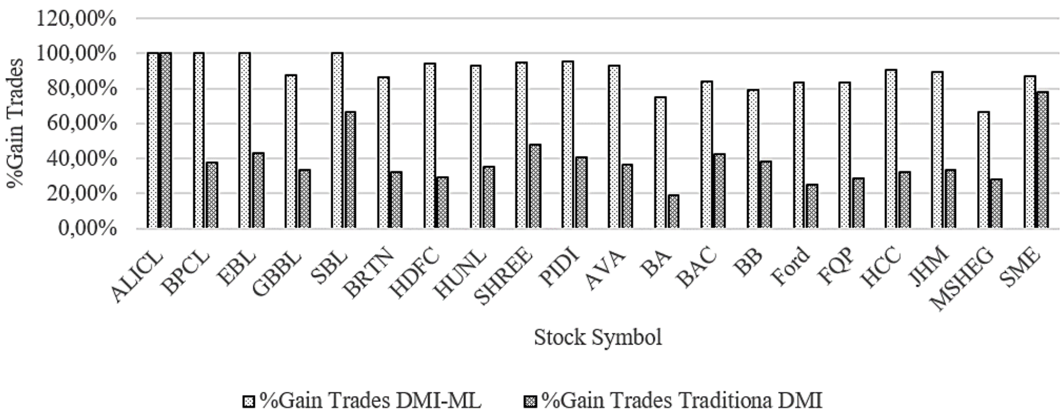

Figure 5. Percentage profitable trades

5.3. Comparison with other intelligent trading strategies

Although many researchers have proposed machine learning models for predicting stock price or index, only a few researchers have devised intelligent models for stock trading using machine learning strategies. Table 2 provides a comparison between the average value of results obtained from intelligent trading strategies devised in various literatures and DMI-ML strategy. From the table, it is clearly seen that the proposed DMI-ML strategy not only outperformed the Buy-Hold strategy and traditional DMI strategy, but also the returns derived from the strategy are superior to other intelligent trading strategies.

Table 2. Average value of results obtained from intelligent trading strategies

Proposed Strategy	Comparison with	%Return	SR of return	Gained for
HC-LSTM [6]	Bollinger Band based trading Strategy, Buy-Hold strategy	– 10.61% more return than the Buy-Hold strategy – 14.17% more return than the Bollinger Band based strategy	Not provided	48% stocks
HC-CNN [6]	Bollinger Band based trading strategy, Buy-Hold strategy	– 3.21% more return than the Buy-Hold strategy – 6.77% more return than the Bollinger Band based strategy	Not provided	68% stocks
MICNN [10]	RSI Based trading strategy	– 6.2% more return than the RSI based trading strategy	Not provided	70% stocks
GP Based Strategy [12]	Buy-Hold strategy	– 2.72% more return than the Buy-Hold strategy	Not provided	Not provided
Proposed DMI-ML Strategy	Traditional DMI trading strategy, Buy-Hold strategy	– 52.58% more return than Buy-Hold strategy – 75.12% more return than the traditional DMI strategy	– SR=17.36 for DMI-ML strategy – SR=0.44 for Traditional DMI strategy – SR=5.61 for Buy-Hold strategy	100% stocks

6. CONCLUSION

This research work proposed a directional movement index based machine learning (DMI-ML) strategy for predicting stock trading signals. The study evaluated the model against the Buy-Hold strategy and the traditional DMI based trading strategy. The evaluation was made in terms of annual rate of return (ARR), Sharpe ratio (SR), and percentage of profit/loss trades executed by trading strategies. From the experimental results, we observed that the DMI-ML strategy outperformed the other two models in terms of all three measures. The strategy yielded higher ARR than the Buy-Hold strategy, ranging from 1.38% to 156.01%. Meanwhile, the strategy resulted in higher ARR than the traditional DMI strategy, ranging from 15.54% to 170.86%. Further, The DMI-ML strategy obtained higher SR than the other two models. The strategy achieved at least 0.45 and at most 46.43 more SR than the Buy-Hold strategy. At the same time, the strategy achieved at least 3.74 and at most 55.57 more SR than the traditional DMI strategy. Except for the ALICL, the DMI-ML strategy executed more profitable trades, ranging from 8.7% to 64.71%, than the traditional DMI strategy.

From the summarized view of the experimental results, we observed that the proposed DMI-ML strategy achieved 80.97% ARR with 17.36 SR whereas the Buy-Hold strategy and the traditional DMI strategy respectively achieved 28.39% and 5.85% ARR with 5.61 and 0.44 SR. In addition, we observed the DMI-ML strategy delivered significantly better returns than the intelligent stock trading strategies proposed in the literatures. Hence, we concluded that the proposed DMI-ML strategy is the better strategy for automated trading in stock markets. The strategy is capable to make a decent profit from stock trading regardless of the trend of the stocks.

REFERENCES

[1] A. S. Wafi, H. Hassan, and A. Mabrouk, "Fundamental analysis models in financial markets – review study," *Procedia Economics and Finance*, vol. 30, pp. 939–947, 2015, doi: 10.1016/S2212-5671(15)01344-1.
[2] J. Stanković, I. Marković, and M. Stojanović, "Investment strategy optimization using technical analysis and predictive modeling in emerging markets," *Procedia Economics and Finance*, vol. 19, pp. 51–62, 2015, doi: 10.1016/S2212-5671(15)00007-6.
[3] F. B. Oriani and G. P. Coelho, "Evaluating the impact of technical indicators on stock forecasting," in *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, Dec. 2016, pp. 1–8, doi: 10.1109/SSCI.2016.7850017.
[4] G. Zhang, X. Zhang, and H. Feng, "Forecasting financial time series using a methodology based on autoregressive integrated moving average and Taylor expansion," *Expert Systems*, vol. 33, no. 5, pp. 501–516, Oct. 2016, doi: 10.1111/exsy.12164.
[5] M. Khashei and M. Bijari, "A new hybrid methodology for nonlinear time series forecasting," *Modelling and Simulation in Engineering*, vol. 2011, pp. 1–5, 2011, doi: 10.1155/2011/379121.

[6] C. Stoean, W. Paja, R. Stoean, and A. Sandita, "Deep architectures for long-term stock price prediction with a heuristic-based strategy for trading simulations," *PLOS ONE*, vol. 14, no. 10, Oct. 2019, doi: 10.1371/journal.pone.0223593.
[7] A. S. Saud, S. Shakya, and A. Ghosh, "Hill-climbing powered gated recurrent unit network to predict stock trading signals using the mean average convergence divergence indicator," *Journal of Theoretical and Applied Informtion Technology*, vol. 99, no. 7, pp. 1493–1504, 2021.
[8] C. D. Tilakaratne, M. A. Mammadov, and S. A. Morris, "Modified neural network algorithms for predicting trading signals of stock market indices," *Journal of Applied Mathematics and Decision Sciences*, vol. 2009, pp. 1–22, Jun. 2009, doi: 10.1155/2009/125308.
[9] H. Yang, X.-Y. Liu, S. Zhong, and A. Walid, "Deep reinforcement learning for automated stock trading: an ensemble strategy," *SSRN Electronic Journal*, 2020, doi: 10.2139/ssrn.3690996.
[10] Z. Yang, K. Hao, X. Cai, L. Chen, and L. Ren, "Prediction of stock trading signal based on multi-indicator channel convolutional neural networks," in *2019 IEEE 8th Data Driven Control and Learning Systems Conference (DDCLS)*, May 2019, pp. 912–917, doi: 10.1109/DDCLS.2019.8908881.
[11] A. Arevalo, J. Nino, G. Hernandez, and J. Sandoval, "High-frequency trading strategy based on deep neural networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, vol. 9773, pp. 424–436, doi: 10.1007/978-3-319-42297-8_40.
[12] P. Kroha and M. Friedrich, "Comparison of genetic algorithms for trading strategies," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8327, Novy Smokovec, 2014, pp. 383–394.
[13] D. Lv, S. Yuan, M. Li, and Y. Xiang, "An empirical study of machine learning algorithms for stock daily trading strategy," *Mathematical Problems in Engineering*, vol. 2019, pp. 1–30, Apr. 2019, doi: 10.1155/2019/7816154.
[14] A. S. Saud and S. Shakya, "3-way gated recurrent unit network architecture fore stock price prediction," *Indian Journal of Computer Science and Engineering*, vol. 12, no. 2, pp. 421–427, Apr. 2021, doi: 10.21817/indjcse/2021/v12i2/211202011.
[15] G. Ding and L. Qin, "Study on the prediction of stock price based on the associated network model of LSTM," *International Journal of Machine Learning and Cybernetics*, vol. 11, no. 6, pp. 1307–1317, Jun. 2020, doi: 10.1007/s13042-019-01041-1.
[16] M. Qiu and Y. Song, "Predicting the direction of stock market index movement using an optimized artificial neural network model," *PLOS ONE*, vol. 11, no. 5, May 2016, doi: 10.1371/journal.pone.0155133.
[17] G. Kumar and V. Sharma, "Stock market index forecasting of nifty 50 using machine learning techniques with ANN approach," *International Journal of Modern Computer Science*, vol. 4, no. 3, pp. 22–27, 2016.
[18] R. Dash and P. K. Dash, "A hybrid stock trading framework integrating technical analysis with machine learning techniques," *The Journal of Finance and Data Science*, vol. 2, no. 1, pp. 42–57, Mar. 2016, doi: 10.1016/j.jfds.2016.03.002.
[19] A. Atkins, M. Niranjan, and E. Gerding, "Financial news predicts stock market volatility better than close price," *The Journal of Finance and Data Science*, vol. 4, no. 2, pp. 120–137, Jun. 2018, doi: 10.1016/j.jfds.2018.02.002.
[20] V. S. Pagolu, K. N. Reddy, G. Panda, and B. Majhi, "Sentiment analysis of Twitter data for predicting stock market movements," in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, Oct. 2016, pp. 1345–1350, doi: 10.1109/SCOPES.2016.7955659.
[21] T. B. Shahi, A. Shrestha, A. Neupane, and W. Guo, "Stock price forecasting with deep learning: A comparative study," *Mathematics*, vol. 8, no. 9, Aug. 2020, doi: 10.3390/math8091441.
[22] Y. Shynkevich, T. M. McGinnity, S. A. Coleman, A. Belatreche, and Y. Li, "Forecasting price movements using technical indicators: Investigating the impact of varying input window length," *Neurocomputing*, vol. 264, pp. 71–88, Nov. 2017, doi: 10.1016/j.neucom.2016.11.095.
[23] A. W. Li and G. S. Bastos, "Stock market forecasting using deep learning and technical analysis: A systematic review," *IEEE Access*, vol. 8, pp. 185232–185242, 2020, doi: 10.1109/ACCESS.2020.3030226.
[24] Y. Xia, Y. Liu, and Z. Chen, "Support vector regression for prediction of stock trend," in *2013 6th International Conference on Information Management, Innovation Management and Industrial Engineering*, Nov. 2013, pp. 123–126, doi: 10.1109/ICIII.2013.6703098.
[25] T. M. Sands, D. Tayal, M. E. Morris, and S. T. Monteiro, "Robust stock value prediction using support vector machines with particle swarm optimization," in *2015 IEEE Congress on Evolutionary Computation (CEC)*, May 2015, pp. 3327–3331, doi: 10.1109/CEC.2015.7257306.
[26] Q. Zhuge, L. Xu, and G. Zhang, "LSTM neural network with emotional analysis for prediction of stock price," *Engineering Letters*, vol. 25, no. 2, pp. 167–175, 2017.
[27] X. Li, L. Yang, F. Xue, and H. Zhou, "Time series prediction of stock price using deep belief networks with intrinsic plasticity," in *2017 29th Chinese Control And Decision Conference (CCDC)*, May 2017, pp. 1237–1242, doi: 10.1109/CCDC.2017.7978707.
[28] D. L. Minh, A. Sadeghi-Niaraki, H. D. Huy, K. Min, and H. Moon, "Deep learning approach for short-term stock trends prediction based on two-stream gated recurrent unit network," *IEEE Access*, vol. 6, pp. 55392–55404, 2018, doi: 10.1109/ACCESS.2018.2868970.
[29] M. Wen, P. Li, L. Zhang, and Y. Chen, "Stock market trend prediction using high-order information of time series," *IEEE Access*, vol. 7, pp. 28299–28308, 2019, doi: 10.1109/ACCESS.2019.2901842.
[30] B. Yang, Z.-J. Gong, and W. Yang, "Stock market index prediction using deep neural network ensemble," in *2017 36th Chinese Control Conference (CCC)*, Jul. 2017, pp. 3882–3887, doi: 10.23919/ChiCC.2017.8027964.
[31] A. H. Moghaddam, M. H. Moghaddam, and M. Esfandyari, "Stock market index prediction using artificial neural network," *Journal of Economics, Finance and Administrative Science*, vol. 21, no. 41, pp. 89–93, Dec. 2016, doi: 10.1016/j.jefas.2016.07.002.
[32] I. Gurrib, "Performance of the average directional index as a market timing tool for the most actively traded USD based currency pairs," *Banks and Bank Systems*, vol. 13, no. 3, pp. 58–70, Aug. 2018, doi: 10.21511/bbs.13(3).2018.06.
[33] C. Faijareon and O. Sornil, "Evolving and combining technical indicators to generate trading strategies," *Journal of Physics: Conference Series*, vol. 1195, Apr. 2019, doi: 10.1088/1742-6596/1195/1/012010.
[34] B. S. Alhashel, F. W. Almudhaf, and J. A. Hansz, "Can technical analysis generate superior returns in securitized property markets? Evidence from East Asia markets," *Pacific-Basin Finance Journal*, vol. 47, pp. 92–108, Feb. 2018, doi: 10.1016/j.pacfin.2017.12.005.
[35] Y. S. Abu-Mostafa and A. F. Atiya, "Introduction to financial forecasting," *Applied Intelligence*, vol. 6, pp. 205–213, 1996, doi: 10.1007/BF00126626.
[36] E. Guresen, G. Kayakutlu, and T. U. Daim, "Using artificial neural network models in stock market index prediction," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10389–10397, Aug. 2011, doi: 10.1016/j.eswa.2011.02.068.
[37] Y. Lin, H. Guo, and J. Hu, "An SVM-based approach for stock market trend prediction," in *The 2013 International Joint*

Conference on Neural Networks (IJCNN), Aug. 2013, pp. 1–7, doi: 10.1109/IJCNN.2013.6706743.
[38] W. Chen, C. K. Yeo, C. T. Lau, and B. S. Lee, "Leveraging social media news to predict stock index movement using RNN-boost," *Data & Knowledge Engineering*, vol. 118, pp. 14–24, Nov. 2018, doi: 10.1016/j.datak.2018.08.003.
[39] Z. Hu, Y. Zhao, and M. Khushi, "A survey of forex and stock price prediction using deep learning," *Applied System Innovation*, vol. 4, no. 1, Feb. 2021, doi: 10.3390/asi4010009.
[40] J. W. Wilder, *New concepts in technical trading systems hardcover*. Trend Research, 1978.
[41] "Nepal stock exchange Ltd," *Nepal stock exchange Ltd*. http://nepalstock.com.np/stockWisePrices (accessed Apr. 15, 2021).
[42] I. Stocks, "S&P BSE-100," *Indian Stocks*. https://www.investing.com/equities/india (accessed Apr. 15, 2021).
[43] Yahoo, "Yahoo finance," *Yahoo*. https://finance.yahoo.com/ (accessed Apr. 15, 2021).
[44] A. S. Saud and S. Shakya, "Analysis of look back period for stock price prediction with RNN variants: A case study on banking sector of NEPSE," *Procedia Computer Science*, vol. 167, pp. 788–798, 2020, doi: 10.1016/j.procs.2020.03.419.
[45] F. Schmid and R. Schmidt, "Statistical Inference for Sharpe Ratio," in *Interest Rate Models, Asset Allocation and Quantitative Techniques for Central Banks and Sovereign Wealth Funds*, London: Palgrave Macmillan UK, 2010, pp. 337–357.

BIOGRAPHIES OF AUTHORS

Arjun Singh Saud received MSc degree in Computer Science and Information Technology from Tribhuvan University, Kathmandu, Nepal, in 2006. He is currently pursuing his PhD degree from the Central Department of Computer Science and IT, Tribhuvan University. He is also a faculty member of the same department. His research interest includes artificial intelligence, machine learning, deep learning, and data science. Email: arjunsaud@cdcsit.edu.np.

Subarna Shakya received MSc and PhD degrees in Computer Engineering from the Lviv Polytechnic National University, Ukraine, in 1996 and 2000, respectively. Currently, he is a professor at the Department of Electronics and Computer Engineering, Pulchowk Campus, Institute of Engineering, Tribhuvan University. His research interest includes e-government system, distributed and cloud computing, and software engineering and information system, Deep Learning, and Data Science. Email: drss@ioe.edu.np.