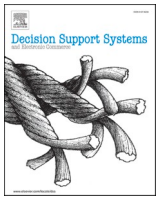
Decision Support Systems 177 (2024) 114100

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss



A hybrid decision support system for adaptive trading strategies: Combining a rule-based expert system with a deep reinforcement learning strategy

Yuhee Kwon, Zoonky Lee *

Graduate School of Information, Yonsei University, Yonsei-ro 50, Seodaemun-gu, Seoul 03722, Republic of Korea

ARTICLE INFO

Keywords:
Intelligent hybrid trading system
Intelligent decision support system
Adaptive trading strategy
Reinforcement learning
Rule-based system

ABSTRACT

Stock trading strategies pose challenging applications of machine learning for significant commercial yields in the finance industry, drawing the attention of both economists and computer scientists. Until now, many researchers have proposed various methods to implement intelligent trading strategy systems that can support decisions regarding stock trading. Some studies have shown that the problem of trading strategies can be successfully addressed by applying hybrid approaches. Motivated by this, we propose a hybrid decision support system for adaptive trading strategies that combines a rule-based system with deep reinforcement learning to self-improve by learning with human expertise. This study overcomes the limitations of previous hybrid models that mainly have focused on optimizing trading decisions and improving forecasting accuracy. The proposed hybrid model combines decision-making information from a rule-based model to enable the agent of reinforcement learning to capture more trading opportunities. In addition, the investor's available balance states facilitate adaptive learning by interacting with the environment. Moreover, the proposed trading mechanism adjusts the volume size using the policy gradient algorithm's action probabilities, resulting in improved risk-adjusted returns. The proposed hybrid model has the potential to be a reliable trading system in real-world applications through its ability to adapt to different market scenarios, withstand stressful market conditions, reduce transaction costs, scale to various index funds, and extend the proposed hybrid structure. This study highlights the applicability of more advanced machine learning in financial areas, and we also suggest expanding this approach to adaptive decision-making systems in other fields.

1. Introduction

The recent increase in data and ease of access to market information have increased the utility of algorithm trading in stock investments [9]. According to international media, the volume of transactions traded by algorithmic trading is increasing yearly, constituting about 80% of the daily trading volume of the U.S. stock exchange [1,2]. One of the most widely used algorithmic trading approaches is the rule-based (RB) model, which is a financial expertise-based approach that requires minimal human intervention [62]. Meanwhile, with the development of machine learning research, recent studies have focused on predicting price movements by applying more advanced techniques [25,37]. As both approaches have their pros and cons, many studies have been conducted to combine these two techniques [35]. The hybrid approach has gained prominence as it leverages the strengths of each model [6]. Nevertheless, the research area of the hybrid approach for trading is still in its infancy and faces many challenges.

A good number of researchers have attempted to hybridize RB and machine learning techniques. One side of the research is to use the results of the RB technique as features in supervised learning to increase forecasting accuracy [11,28,31,62]. The other side of the research is to optimize trade decisions in RB systems by filtering rule signals using the machine learning method [5,51]. Although these hybrid approaches have been reported to have performance increases over one model only, they are not without any limitations. First, the hybrid methods to improve prediction accuracy using supervised learning still require labeling data, which are often criticized for being affected by many external factors and uncertainties, making it difficult to establish a single ground truth logic [25,37]. Second, the hybrid methods that aim to optimize trading rules lack the ability to adapt to markets [26,35];

* Corresponding author.
E-mail addresses: yuheekwon@yonsei.ac.kr (Y. Kwon), zlee@yonsei.ac.kr (Z. Lee).

https://doi.org/10.1016/j.dss.2023.114100
Received 29 November 2022; Received in revised form 1 September 2023; Accepted 27 September 2023
Available online 28 September 2023

therefore, they are often criticized for being unsuitable for addressing changing market situations [6,25,35]. Third, both of these approaches may be difficult to apply in various market scenarios [25,37], including highly volatile or significantly fluctuant conditions. Finally, despite practical trading constraints such as trading volume and current balances that can significantly impact the performance of trading systems in real-world trading [27,53], many studies have rarely considered these constraints [23]. Therefore, there are ample opportunities for the development of hybrid models that use a self-improving trading strategy by adapting to the changing market environment.

To address the aforementioned issues, we study a hybrid decision support system for an adaptive trading strategy. The proposed hybrid model combines a deep reinforcement learning (RL) algorithm with an RB system to facilitate agents to learn efficiently. We propose to build an intelligent model for learning human expertise by incorporating decision information from RB systems into the state space of RL. Additionally, we provide a state space representing investors' available assets status to allow interaction with the environment and assist agents in learning to adapt to the market. Furthermore, we design an effective investment mechanism to adjust the trading volume by utilizing the action probability of the policy gradient (PG) algorithm [66]. Finally, we demonstrated the effectiveness of the proposed hybrid method and constraints and showed the adaptability by analyzing occurrence signals. We also show the reliability of the model in various ways. A summary of our contributions to the machine learning and finance fields is as follows:

- To the best of our knowledge, this is the first study to combine an RB expert system with deep RL to develop a financial model. The RL algorithm learns from the decision-making information of the RB system to autonomously identify additional trading opportunities and adjust the risk. The proposed approach has the potential to effectively mitigate investment risk, by drawing on information derived from decisions made by the RB system.
- The investor's available balance status has been incorporated into the hybrid model as a state space to facilitate interaction with the environment, enabling market-adaptive learning and improving cumulative returns.
- To efficiently capture trading opportunities, we designed a volume-adjusted mechanism that utilizes the action probabilities of the PG algorithm, resulting in improved risk-adjusted returns.
- By comparing and analyzing the differences in the occurrence of trading signals, our model demonstrated the potential to alleviate the positive false bias, resulting in the reduction of trading costs. Therefore, it has exhibited higher overall returns while generating fewer trade signals compared to other models.
- Our model consistently outperformed benchmark indexes and other hybrid models across various market scenarios. It notably presented superior performance during market crashes and effectively followed the long-term tendency during upward trends.

To ensure the practicality of the proposed hybrid model as a trading decision support system, we conducted extensive experiments on six index funds. The consistent empirical findings from these experiments indicate the robustness of our model.

2. Hybrid approaches that combine RB and machine learning techniques in trading

Many previous studies have indicated that the hybridization of two different methods, when properly combined, can lead to reasonably better performance than using only one method [6,25,35,37]. In this study, we examine the emphasis placed on hybrid studies that attempt to combine RB with machine learning techniques. The two main research streams in the field are: (1) improving the performance of supervised learning algorithms by incorporating RB models into machine learning techniques, and (2) optimizing trading decisions by combining machine learning techniques. Therefore, we critically review these two fields of research and suggest a new way of combining them.

2.1. Improving performance by combining an RB model with machine learning

One way of incorporating the RB model into the machine learning technique is to derive input values based on the RB model and input the features into the machine learning model. Tsaih et al. (1998) [62] employed fuzzy logic to define trading rules and provided training input variables to the Reasoning Neural Network (RN). They designed an RB model to capture both the linear and nonlinear characteristics of the S&P 500 index. The study results showed that the hybrid AI system outperformed both backpropagation networks and perceptron models in terms of forecasting accuracy. Wen et al. (2010) [65] developed an automatic stock decision support system using box theory and the support vector machine (SVM) algorithm. They employed box theory to identify trend movements, which helped address the challenges of uncertainty and nonlinearity in trading. The SVM algorithm was then used to make predictions based on the identified information. Chang (2016) [11] proposed a hybrid model that combines the Takagi–Sugeno (TS) fuzzy model with support vector regression (SVR) for stock trading forecasting. The TS fuzzy model is an RB system that captures both linear and nonlinear relationships, and the SVR can learn underlying patterns in the data and make predictions. The study results showed that the proposed hybrid model outperformed other models, including single models. Several studies have been conducted to explore the integration of machine learning techniques with technical indicators (TIs) derived from predefined rules or simple mathematical formulas. These studies have demonstrated the effectiveness of the approach in predicting dynamic stock market movements [28,31,40,48].

Another method of using RB models is to determine the machine learning parameters, such as weights and thresholds. Chang et al. (2011) [10] developed a model that used Piecewise Linear Representations (PLRs) and artificial neural networks (ANNs) to analyze the nonlinear relationships between stock prices and various TIs. They further explored the generation of dynamic threshold bounds that provided guidance for triggering decisions when the ANN-predicted trading signal went above or below the specified threshold bounds. Their results showed that the proposed model outperformed the PLR-Backpropagation Network (PLR-BPN) model. Sermpinis et al. (2012) [56] investigated the use of the Psi Sigma Neural Network (PSN) architecture for forecasting the EUR/USD exchange rate and explored the utility of Kalman Filters (KFs) in combining neural network forecasts. They also applied a time-varying leverage strategy based on RiskMetrics volatility forecasts to further improve the forecasting performance of their model. They showed that the combination of forecasts using KFs with the time-varying strategy outperformed its benchmarks. Most studies have shown that combining models leads to better forecasting accuracy than using a single model alone.

However, this approach may face challenges due to the intrinsic characteristics of supervised learning methods, which require labeled data. In financial markets, defining criteria for labeling market movements can be difficult due to their dynamic nature in real-time changing of markets. Therefore, studies have indicated that the predictive results may vary depending on changes in the market trend or the size of the sliding window used in the training [25,37].

2.2. Optimizing trading decisions by combining machine learning techniques

Many studies have attempted to optimize RB decisions using machine learning techniques. While some studies have used evolutional algorithms such as genetic algorithms (GA) or particle swarm optimization (PSO) to optimize the trading rules [3,41,47,64], we focus more

on the neural network (NN)-based techniques as they are the main interest of our study. The study by [39] aimed to filter the high-frequency signals of an RB foreign exchange trading strategy through an NN-based intelligent selection mechanism. The study employed two networks: the first network filtered out noise from the input data and the second network generated the final trading signal. The results showed that the proposed approach improved trading profitability compared to the traditional trading approach. Ayala et al. (2021) [5] proposed combining technical strategies with a machine learning approach to produce trading decisions. They demonstrated the performance of four machine learning techniques to select the most suitable one: linear model (LM), ANN, random forests (RF), and SVR. The study demonstrated that the addition of machine learning techniques to technical strategies improved the trading signals and the competitiveness of the proposed trading rules. Park et al. (2022) [51] developed a multi-task model that integrates Long Short-Term Memory (LSTM) and RF to prevent overfitting. The focus of their study was to improve the predictive performance and combine the results to trade according to specific trading rules. The results showed that the proposed multi-task model outperformed the single-task model. Therefore, these studies have focused on optimizing trading decisions by using machine learning techniques to remove noise from complex stock markets, which were effective in improving profitability.

Although some studies have shown the effectiveness of RB systems in optimizing trading decisions, they have indicated the challenges in adapting these results to other markets [26,35,37]. These challenges may be due to the inherent nature of RB systems, which are difficult to adapt to changes in market conditions and tend to overfit in specific situations.

In general, the reviewed papers have focused on optimizing the performance of RB systems or improving the predictive accuracy of supervised classifiers. While these approaches have shown promise for enhancing financial market profitability, there still remains a need to further investigate the capabilities and limitations of these hybrid techniques. The lack of studies that explore generating consistently profitable trading [26,35], particularly in highly volatile or fluctuant situations, indicates the necessity for further exploration and discussion in this area, which is the motivation for this paper.

More recently, studies have indicated that these challenges can be successfully addressed by applying RL algorithms to develop automatic trading systems with adaptive trading strategies [8,19,21,46,67]. Nevertheless, the study on how to hybridize the RL algorithm with another method remains open for further exploration. Therefore, developing a hybrid model using RL is the core of the proposed approach. To address the limitations of existing methods and be motivated by the combination of an RB expert system and RL algorithm, this work presents a novel trading strategy approach. The idea is to focus more on providing state spaces that include decision-making information of RB systems based on finance expertise.

3. Methodology

3.1. RB trading system

We adopt a trend-following strategy as the most representative of the RB system [35,50]. Turtle trading strategy, widely used as a trend-following RB algorithm, was officially introduced by Faith [20]. The first decision of trading is when to buy or sell. Turtles use related system trading signals based on the channel breakout system by Donchian [17]. A breakout is defined as the price exceeding the high or low of a particular number of days, with buy positions following N days of high price and sell positions being N/2 days of low price. An N-day breakout would be defined as exceeding the high or low of the preceding N days.

The second decision is about how much to buy or sell. Turtle trading gradually invests when prices continue along the trend to increase profits. The Turtle system measures market risk by using volatility, to enhance profit opportunities, Turtles use this measurement to construct positions in increments that represent a constant amount of volatility. The Turtle trading strategy uses a position-sizing method based on volatility: the referred average true range (ATR). The ATR is calculated using Eq. (1), which defines the true range.

$$TR_t = max[(high_t - low_t), abs(high_t - close_{t-1}), abs(low_t - close_{t-1})] \quad (1)$$

Then, the ATR is calculated in Eq. (2) as follows:

$$ATR_1 = \frac{1}{N}\sum_{t=1}^{N} TR_t,\ ATR_t = \frac{ATR_{t-1}x(N-1) + TR_t}{N} \quad where\ t = 2, \ldots \quad (2)$$

After the occurrence of the long position, there is a gradual investment when it increases by 1/2 ATR in an upward trend. These rules control the total risk of investment, and these limits minimize losses during prolonged losing periods as well as abnormal price fluctuations. The last decision is to predefine the point of exit to reduce losses. In the event of a 2% loss compared to the total assets currently held, all shareholdings will be exited.

3.2. PG trading strategy

3.2.1. RL

Sutton (1999) [4] presents RL as a self-taught process that can be represented with a Markov decision process (MDP). An MDP is used to define sequential decision problems with uncertainty. For MDP issues, the agent receives the state of the environment (State, S_t) per time step. The state space describes the environment and the agent's world. The state space $S = (s_0, s_1, \ldots)$ represents an MDP where the transition between states depends only on the information of the previous state, as reflected in Eq. (3).

$$Pr(s_{t+1}|s_0, a_0, s_1, a_1, \ldots, s_t, a_t) = Pr(s_{t+1}|s_t, a_t) \quad (3)$$

The agent selects the action (A_t) from the received state (S_t) and delivers it back to the environment. The environment delivers the reward (Reward, $R_{t+1} = r(s_t, a_t)$) and the next state (S_{t+1}) to the agent based on this action (A_t). By repeating this process, the agent creates a trajectory of interaction with the environment.

The RL is to repeat the MDP process and maximize the expected return (G_t) defined for each time step. G_t reflects an expected value of the accumulated reward in Eq. (4).

$$G_t = \sum_{t=0}^{T} \gamma^t \bullet R_{t+1}\ where\ \gamma \in (0,1) : discount\ factor \quad (4)$$

By repeating this process, the agent creates a policy (π_θ). Therefore, the ultimate goal of RL is to find the optimal policy (π^*), represented by Eq. (5), to decide what action to take in the given states.

$$\pi^*(a|s) = P(A_t = a|S_t = s) \quad (5)$$

3.2.2. PG

The goal of RL in the model-free policy-based method is to find an optimal policy (θ^*) that maximizes the expected value of the cumulative rewards. The PG [66], called the REINFORCE algorithm, parameterizes the policy directly as θ to find the optimal policy (θ^*). It uses a method goal to find an optimal policy (θ^*) that maximizes the expected value of the cumulative reward (objective function, $J(\theta)$) which is the goal of RL ($\theta^* = argmaxJ(\theta)$). The gradient ascent is used as a method for maximizing the $J(\theta)$. In this way, a PG is a method of updating a policy using a gradient of the $J(\theta)$, as in Eq. (7). In Eq. (6), $p_\theta(\tau)$ is a probability density function of the trajectory (τ) generated by the policy (π_θ), which is used to calculate the expected value of G_t.

$$Objective\ function: J(\theta) = \mathbb{E}_{\tau\sim p_\theta(\tau)}\left[\sum_{t=0}^{T}\gamma^t \bullet R_{t+1}\right], p_\theta(\tau)$$
$$= p(s_0) \bullet \prod_{t=0}^{T}\pi_\theta(a_t|s_t)p(s_{t+1}|s_t, a_t) \quad (6)$$

$$Objective\ function\ Gradient: \nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim p_\theta(\tau)}\left[\sum_{t=0}^{T}(\nabla_\theta log\pi_\theta(a_t|s_t) \bullet G_t)\right] \quad (7)$$

Therefore, this approach is useful for designing stochastic policies ($a_t \sim \pi_\theta(a_t|s_t)$) as it can directly learn policies and output probability distributions for actions. In the PG method, the policy is parameterized as a neural network called a policy neural network, and the policy parameter θ represents all weights of the neural network. We use LSTM [34] neural networks to model policy neural networks. Many recent studies have applied LSTM to financial time series [22,63] and showed superior performance in modeling daily financial data.

3.2.3. RL trading strategy

When applied to trading, some modifications should be implemented to employ RL. The quadruple definitions for the RL trading application are as follows. The first step is to define the agent [4], for which we employ a PG learning approach to train the agent. The agent selects one of the action sets to make trading decisions based on the learned policy. The second step is to define the state space [23]. In this study, we utilized the state space of historical market prices as basic information in stock trading and included the current profit-loss ratio as expressed in Eq. (8). The third step is to define the action space [36], where we employ the discrete action space A = (buy, sell, hold) as $A_t = (0, 1, 2)$. The last step is to define a reward function [23], which is commonly expressed as the accumulated return.

$$current\ asset\ value_t = balance_t + number\ of\ stocks_t \times close_t$$

$$\%current\ profit - loss\ ratio = \frac{current\ asset\ value_t}{current\ asset\ value_{t-1}} \quad (8)$$

4. Proposed hybrid trading system

4.1. The hybrid decision-making approach

We propose a hybrid model based on deep RL to adaptively make trading decisions and generate profits in dynamic financial markets. Our proposed hybrid method resembles the learning process of human investment experts. This approach is based on the observation that human experts analyze the actions of RB systems designed using past price trends and improve their behavior accordingly. The RB trading system involves basic investment principles and detailed physical mechanisms of observed actions. The expertise in stock price trends and volatility is already embedded within its decision-making processes. Moreover, our approach resembles real-world trading practices. Our system takes into account the investor's available asset status and adjusts the trading volume according to confidence. By integrating these components, our hybrid model enables comprehensive decision-making. We anticipate that this approach will lead to more practical and effective algorithmic trading systems. Fig. 1 illustrates the architecture of our research model.

4.2. Hybrid trading system

4.2.1. The RB system's decision state space

One important issue that must be addressed when designing an RL algorithm is the representation of states, which has a significant effect on trading performance. We designated the RB system's decision results as a learning space so that the hybrid model learns price movement trends. Specifically, the RB system's trading results—such as the sell prices, buy prices, position signal, and position size— are used to represent the price movement trends in detail. In our study, the RB system adopts a typically

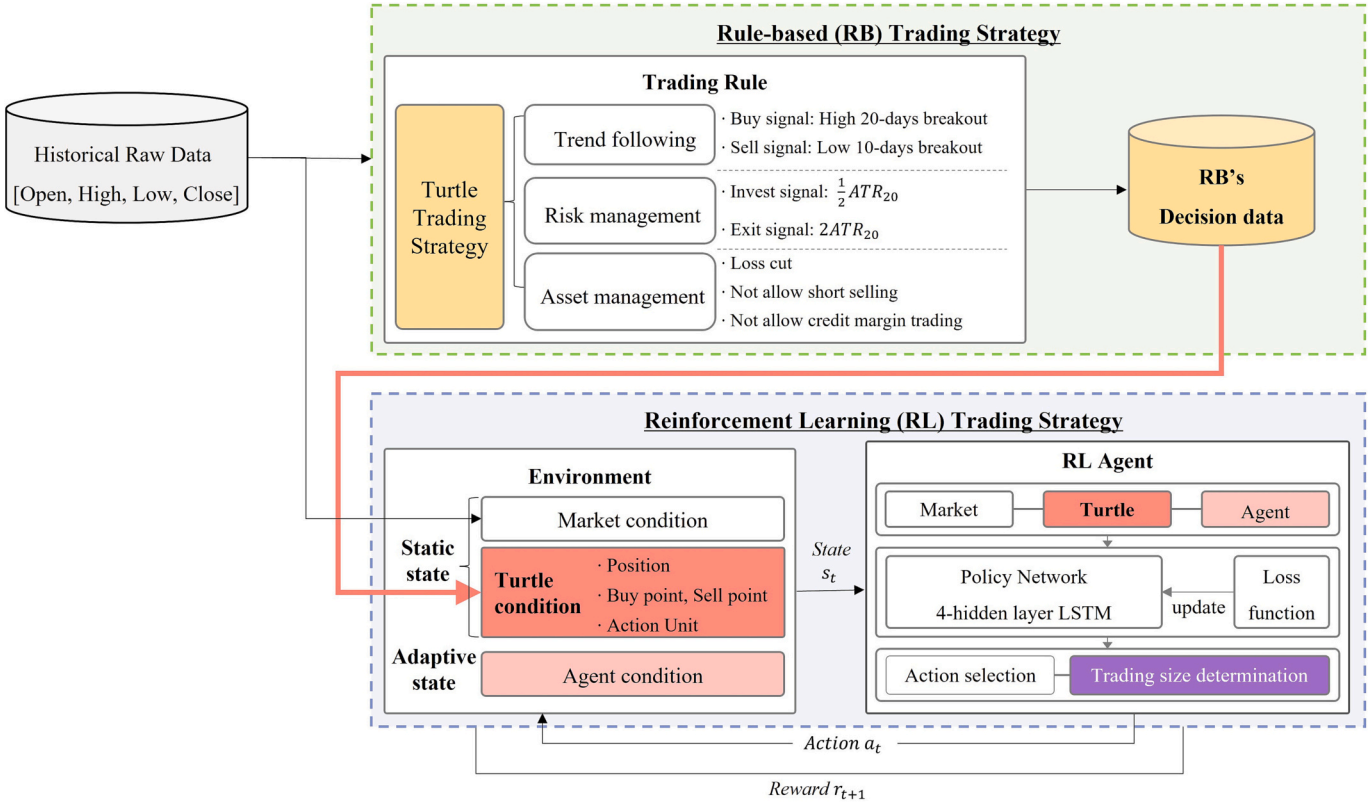


Fig. 1. The architecture of the proposed hybrid system for an adaptive trading strategy.

used trend-following strategy [20] that buys when high prices break out in the previous twenty days and sells when low prices break out in the previous ten days. As mentioned in Section 3.1, the RB system considers the magnitude of volatility when trading along trends. The trading prices and position signals reflect market trends, while the position sizes of shares reflect market volatility. Therefore, the state space of the hybrid RL algorithm reflects both the trend and volatility of the market.

4.2.2. Investor's available asset states

System trading assumes that the agent's trades hardly significantly impact the market price [24] and that historical stock market information is insufficient to predict the future; it serves only as a reference for investment decisions [59]. The definition of a state space in which an agent can interact with the environment is crucial because it allows the agent to plan its next action based on its current state and goals. In real-world trading, the investor's available balance status has a significant impact on their returns on investment. This state space provides information on how the following states change with the agent's actions, unlike the stock market price state space, which remains unaffected by the agent's actions. In other words, even if the movement of the stock market is predictable, individuals can have different responses based on their available balances. In the real world, we trade with limited assets. Even though market movements can be accurately predicted, investment perspectives may vary based on an investor's available asset state such as the current shareholding ratio and fluctuation rate of the average purchase prices. Therefore, investor's available asset states, which drive adaptive investment in various conditions, are valuable information for supporting their decision-making. In our proposed method, we introduce the fluctuation rate of the average buy price and the shareholding ratio as new current state representations of investor's available asset states.

The fluctuation rate of the average buy price is calculated by considering the value that varies according to the k-th additional purchase up to time step t and by calculating the fluctuation rate of the close price relative to that value, which is shown in Eq. (9).

$$fluctuation\ rate\ of\ average\ buy\ price = \frac{close_t}{\frac{1}{k} \bullet \sum_{i=1}^{t} buy\ price_i} - 1 \tag{9}$$

If the fluctuation rate decreases, the average down strategy is implemented, and if this value increases, the trend-following strategy is performed.

The shareholding ratio measures how much stock is currently held relative to the asset value at time step t, as shown in Eq. (10). If the value of the ratio is zero, we do not hold any stocks; if it is one, we hold the maximum number of stocks.

$$shareholding\ ratio = \frac{number\ of\ stocks_t \times close_t}{current\ asset\ value_t} \tag{10}$$

In other words, the ratio value helps the agent invest from the perspective of buying if the number of shares is too small and from the perspective of selling if the number of shares is too large.

4.2.3. Determining trading volume using the action probabilities

Trading volume is powerful in predicting the direction of future price movements, so it greatly influences returns on stock investments [27,32]. It helps the agent make the transaction when the right trading points are captured by having an adequate balance without utilizing too much volume all at once. In particular, we utilize a characteristic of the PG algorithm to determine the trading volume. One of the PG method's main characteristics is the decision action output obtained with a softmax [4] probability through the policy neural network. We use the action probability to adjust the trading volume to reflect the properties of the PG. We increase the trading volume with the probability value of the action determined by the policy neural network because a higher action probability value can enable the agent to buy and sell more with confidence in the determined action. As the episodes are repeatedly trained, the policy evolves, and the confidence of the action probability values increases. We set the maximum trading volume to minimize volatility and specifically determine it, as shown in Eq. (11).

$$\begin{aligned} added\ trading\ volume_t &= softmax_t(p) \\ &\quad \times (maxtrading\ volume - mintrading\ volume) \\ trading\ volume_t &= mintrading\ volume + added\ trading\ volume_t \end{aligned} \tag{11}$$

In the manner suggested, determining the trading volume within the algorithm affects the weight update of the policy neural network. As a result, it can help with the adaptive trading strategy. Fig. 2 shows the proposed trading mechanism.

5. Experiments

We provide details on the experiments conducted. The experiments were implemented in Python 3.6.13 – Keras2.3.1 – TensorFlow1.15.0 – NVIDIA GPU version.

5.1. Raw dataset and splitting into training–test sets

We conducted experiments on the proposed models using daily data from January 2007 to July 2022 for the representative market, the S&P 500 index fund. We obtained the dataset from Yahoo Finance. Table 1 shows the raw dataset of the S&P 500.

We employed hold-out validation, which is a commonly used approach for validating machine learning models. We divided the datasets with a 7:3 ratio, as indicated in Table 2.

5.2. Hybrid model state space

We used the close price and volume as the market state. To construct the hybrid model, we considered the RB's decision-making results as static states and the investor's available assets as current states. The proposed model has a total of nine units in the input layer, and the state of the hybrid model is shown in Table 3. All variables except the current state were standardized.

5.3. Training schemes

The parameter initialization significantly affects the performance of an RL model, as the initialization determines the starting point for the optimization process. To ensure common performance in trading problems, a warm start method [58] is used, which initializes the policy parameters with pre-trained values. The approach functions to initialize the policy with a set of parameters that are close to the optimal solution to speed up convergence and avoid getting stuck in poor local minima. The warm start can be achieved by using the parameters from a previous run of the same or similar policy [14,67], or by setting initial values of the policy parameters based on domain knowledge. The use of the warm start in PG can improve convergence speed and lead to more robust results, particularly in complex or challenging tasks.

The hyperparameter values for modeling are listed in Table 4. The parameter gamma is the discount rate and was set to 0.9. In our study, where agents trade with limited assets, we fixed the initial investments at 100,000. We applied the minimum trading cost at 0.01% per transaction [45,69], and short selling and margin trading were not allowed. We employed a four-hidden-layer LSTM network comprising 256, 128, 64, and 32 units to model the policy network. We used Adam [42] as the optimizer and ReLu [55] as the activation function of the hidden layer. The step size was set to 5, and the learning rate was set to 0.001. The output layer was three units, and softmax was used as the activation function to obtain the decision action as probability values. Using these probability values, we determined the trading volume using the

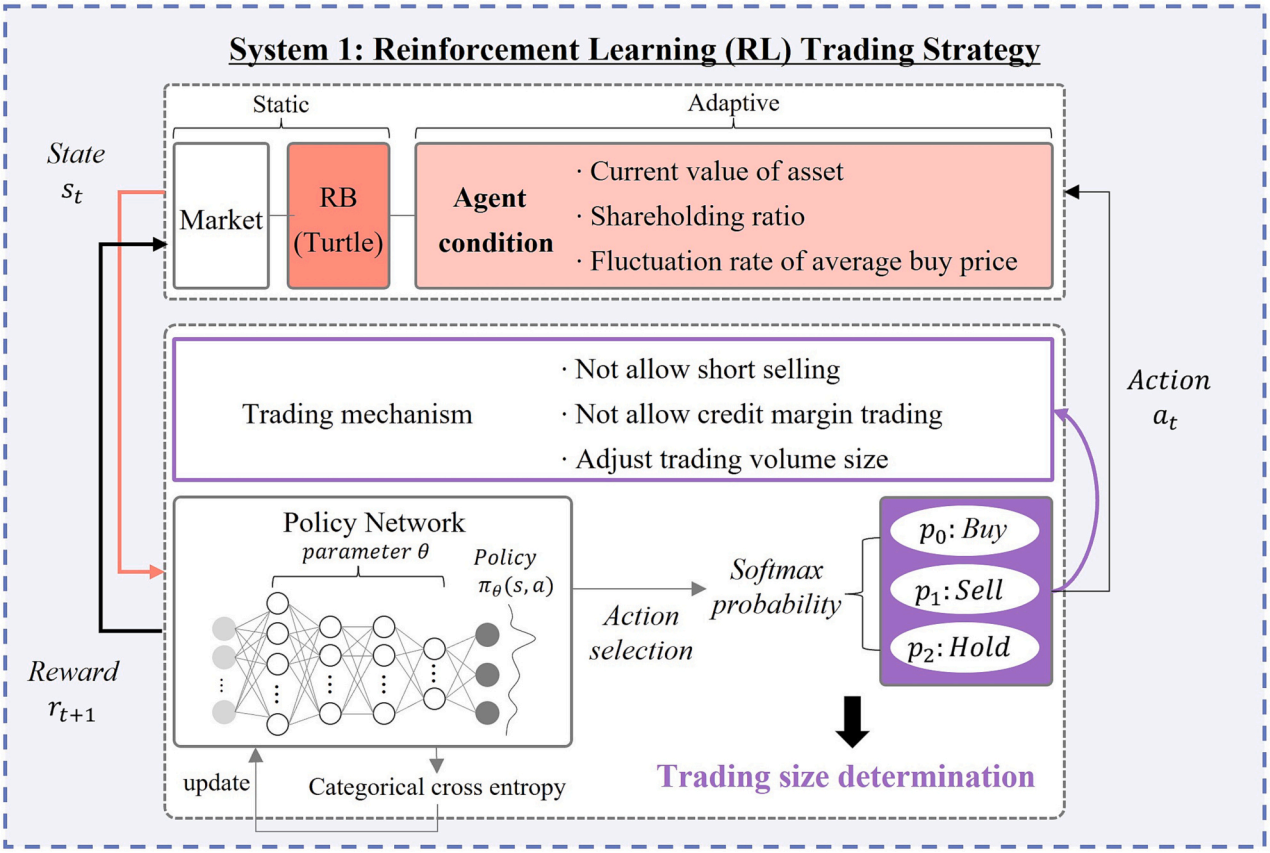


Fig. 2. Deep RL trading mechanism for the adaptive trading strategy: the state space for considering the investor's available balances and the adjusting of trading volume reflecting the probability of action.

Table 1
The raw dataset of the S&P500 index.

Date	*Open*	*High*	*Low*	*Close*	*Volume*
2007-01-03	*1418.0*	*1429.4*	*1407.9*	*1416.6*	*3,429,160,000*
…	…	…	…	…	…
2022-07-29	*4087.3*	*4140.2*	*4079.2*	*4130.3*	*3,817,740,000*

Table 2
Split of train and test datasets.

Dataset	*Start*	*End*	*Days*
Train	*2007-01-03*	*2017-12-29*	*2769*
Test	*2018-01-02*	*2022-07-29*	*1152*

proposed method in Section 4.2.3. To prevent overfitting on the training dataset, we used common normalization methods used in deep learning, such as dropout [60] and batch normalization [54]. Furthermore, for more rigorous verification, we measured the performance of the hybrid model by setting all parameters equally as other models except for adding the proposed components.

The RL approach involves a random factor, such as random exploration during the initial stage of training. The purpose of this randomness is to encourage the agent to explore its environment and learn an optimal policy. Although the results of random exploration can vary each time it is executed, our algorithm is designed to produce consistent results by reducing the amount of random exploration as training progresses. We initialized the exploration rate to 1 and gradually decreased it by repeating episodes until the reward value converged. The convergence refers to the point at which the policy has stabilized and no longer changes significantly [30]. We used the constantly increasing reward and its plateau as an indication of convergence, and we determined that the model had converged when the cost function of the policy network consistently decreased and reached its minimum.

Table 3
State spaces of the proposed hybrid model.

Date	*Market*		*RB*				*Investor's balance*		
	Close	*Volume*	*Signal*	*Buy price*	*Sell price*	*Trading size*	*Current PL*[1]	*SH*[2]	*FR of BP*[3]
2007-01-03	*1416.6*	*3.4e*^{10}	*Hold*	*1427.1*	*1410.8*	*0*	*0*	*0*	*0*
…	…	…	…	…	…	…	…	…	…
2022-07-29	*4130.29*	*3.8e*^{10}	*Buy*	*4072.4*	*3830.8*	*2*	–	–	–

[1] Current PL: current profit–loss ratio.
[2] SH: shareholding ratio.
[3] FR of BP: fluctuation rate of the average buy price

Table 4
Hyperparameter settings of trading models in the S&P500 index fund.

Model	*Learning rate*	*Optimizer*	*Step size*	*Drop out*	*γ*	*Initial balance*	*Max trading volume*	*Episode*
B&H	–	–	–	–	–	*100,000*	–	–
RB	–	–	–	–	–	*100,000*	–	–
RL	*0.001*	*Adam*	*5*	*0.1*	*0.9*	*100,000*	–	*50*
RB + RL	*0.001*	*Adam*	*5*	*0.1*	*0.9*	*100,000*	–	*150*
RB + RL + C1	*0.001*	*Adam*	*5*	*0.1*	*0.9*	*100,000*	–	*100*
RB + RL + C1 + C2	*0.001*	*Adam*	*5*	*0.1*	*0.9*	*100,000*	*10*	*100*

5.4. Verification methods

5.4.1. Ablation studies

Ablation studies are used to identify the important aspects of a system [49,61]. We conducted thorough ablation studies by controlling system components—RB's decision state, the investor's available balance states (C1), and adjusted trading volume mechanism (C2)—to ensure their effectiveness on the performance. For the ablation studies, the baseline model was composed of a single RB as Turtle strategy and a single RL as PG. We also compared the performance of each model with the B&H strategy [8].

To examine the effectiveness of the proposed components, we employed two methods.

At the macro level, we measured the performance evaluation using the six metrics [8] as follows.

- Accumulated return (%AR): measures the change compared to the initial investment.
- Average annual return: a metric for referring to the annual average return.
- Average daily return: the rate of change in asset value compared to the previous day.
- Maximum drawdown (MDD): refers to the maximum losses that can be borne during trading, and the lower the value, the better the performance of the trading strategy.
- Standard deviation (SD): a metric for measuring the volatility of trading strategies.
- Sharpe ratio (SR): measures the risk-adjusted rate of annual returns for the investment.

At the micro level, we examined the actual number of signals generated [12,14,26], as it is important for evaluating the efficiency of a trading strategy to capture optimal market opportunities at the right time.

5.4.2. Comparative studies

To clarify the novelty of our work, we conducted comparisons between our model and state-of-the-art studies that integrated machine learning techniques with multiple TIs. We selected benchmarks by prioritizing hybrid studies that employ RB models as input features for machine learning, as these models are closely related to our approach.

- TI + SVM [31]: This study predicted price trends using useful TIs and the SVM model.
- TI + RF [40]: They used multiple TIs and RF which algorithms are known as ensemble.
- TI + LSTM [48]: They considered important TIs and augmented the predictive power by constructing four deep learning-based regression models using LSTM networks.
- TI + XGBoost+CNN + LSTM [28]: They proposed to generate many features using TIs and apply the XGBoost model for feature engineering. The different CNN layers and LSTM layers have been designed to perform classification.

To evaluate the trading profitability after they predicted returns or price movements, we employed the trading rules used by [51].

5.4.3. Reliability studies

To verify the reliability and usefulness of our proposed hybrid model in real-world applications, we conducted replication experiments [15,33] in four aspects.

- Adaptability [26,29]: the ability of the model to perform on unseen new data. We examined the potential of the model to adapt to various market conditions.
- Robustness [7,57]: refers to the model's ability to perform in the face of extreme changes or noise in data. We evaluated the capability of the model in a market crash.
- Scalability [6,35]: the ability of the model to handle diverse datasets. We investigated the model trades effectively across different markets.
- Extensibility [18]: refers to the ability of a theory or system to be extended. Without changing the structure of our hybrid method, we examined the ability of the system to extend by modifying the RB model.

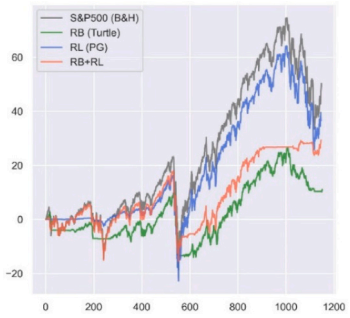
6. Results and discussions

6.1. Verification of the effectiveness of the proposed components for the hybrid trading model

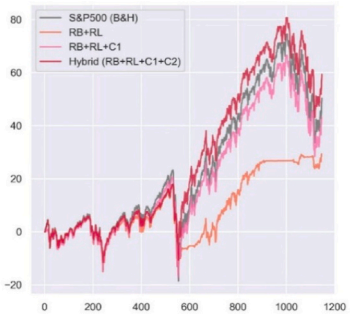
6.1.1. RB decision-making results improve the dynamic behavior and reduce the investment risk of RL's agent

As depicted in Fig. 3 (a), the performance of the RB model was poor, which we attribute to the high volatility and significant fluctuations that occurred during the test period. As the RB expert trading system relies on predefined rules, it struggles to adapt to dynamically changing market situations. On the other hand, the performance of the RL model was relatively better than the RB model. This performance difference can be attributed to the RL model's ability to identify complex relationships and patterns that may not be immediately obvious to human traders. Furthermore, it can improve its trading behavior over time through trial-and-error experience in the market, leading to better trading decisions. However, as shown in Table 5 and Fig. 3 (a), the RL model tended to excessively follow the market trend. The RL model generated few selling signals because it learned from historical price data that had an overall long-term uptrend. This is because the state space of the RL model, which was defined solely as historical price data, was insufficient for the agent to capture short-term positions. As shown in the results, the RL model can be biased or poorly learned depending on the defined environmental state space, which can make an agent unable to adapt to new markets. Therefore, we combined decision-making information from an RB model into a learning space, enabling the RL algorithm to identify and exploit short-term trading opportunities.

The RB + RL generated signals more dynamically than the RL and incremented the frequency of signals. We observed that the RB + RL generated similar signals to the RB, as detailed in Table 6, and had movements in %AR similar to the RB, as shown in Fig. 3 (a). This finding suggests that the RB's decision-making states serve as a trigger that dynamically generates signals from the RL by providing information to



(a) Comparison of the RB, RL, and RB+RL models



(b) Comparison of the RB+RL, RB+RL+C1, and RB+RL+C1+C2 models

Fig. 3. The %AR curves of the RB, RL, RB + RL, RB + RL + C1, hybrid (RB + RL + C1 + C2) models, and B&H strategy in the S&P500 index fund.

Table 5
The number of signals from each model during the test period.

Signal	*The number of signals*				
	RB	*RL*	*RB + RL*	*RB + RL + C1*	*RB + RL + C1 + C2*
Buy	*153*	*185*	*269*	*191*	*94*
Sell	*111*	*0*	*228*	*54*	*82*
Stop	*3*	–	–	–	–

capture short-term positions. Furthermore, the SD value of the RB + RL was smaller than that of the RL, as detailed in Table 7, which suggests that the RB + RL is able to reduce the investment risk of the RL by learning the RB's decision information.

The results are attributed to the inclusion of important short-term trends and RB's decision patterns, enabling RL to effectively capture trading opportunities. These findings suggest that our hybrid approach enhances the decision-making capabilities of the RL agent. However, this state space just provides information that allows the RL agent to explore the decision of the RB and does not provide feedback on the actions taken based on that information to the next states. In the next section, we cover the experiments conducted on the models that considered the state space that can interact with the environment.

6.1.2. The investor's available balance states facilitate market-adaptive learning of an agent for improvement %AR

By including C1, i.e., the investor's available balance in the state representation, the agent can make more informed trading decisions and interact with the environment more effectively. According to Table 5, we confirmed that the RB + RL + C1 generates fewer signals than the RB + RL, and as described in Table 6, we found that the RB + RL + C1 more frequently generates the same signals as the RB than the RB + RL. Therefore, the RB + RL + C1 is indicated to behave similarly to the RB, be autonomous, and learn to adapt to the market compared with the RB + RL. As shown in Fig. 3 (b), the %AR of the RB + RL + C1 was better than the RB + RL and was more similar to the movement of the index fund return growth rate. The experimental results demonstrate that C1 has a clear effect on interacting with the given environment.

These findings suggest that the agent is more adaptable to the market because C1 helps it make decisions based on its trading progress. Nevertheless, only providing the agent with state spaces is still insufficient for developing a more effective trading strategy that is similar to real-world trading mechanisms. In the following section, we present the results of the experiment on the adjusting trading volume mechanism and investigate its effects.

6.1.3. The adjusting trading volume mechanism enables the agent to efficiently trade and improves the SR

C2 helps the agent in identifying obscure market trends and enhancing trading decisions. The actions with higher probabilities under the current policy are given more weight in the gradient calculation, increasing their likelihood of selection in the next episode of learning. Therefore, using the action probabilities of the PG to adjust trading volume facilitates the capture of optimal trading signals. The RB + RL + C1 + C2 generated fewer signals than the RB + RL + C1, as demonstrated in Table 5, and had signals most similar to those of the RB model, as detailed in Table 6. The results indicate that C2 enables the agent to learn more effectively from the RB's decision results. Additionally, the RB + RL + C1 + C2 outperformed the RB + RL + C1 in terms of %AR, as shown in Fig. 3 (b). According to Table 7, the SD value of the RB + RL + C1 + C2 is lower than that of the RB + RL + C1. Moreover, we found that

Table 7
Evaluation of the performance of each model was measured using six metrics.

Model	*%AR*	*Average Annual Return*	*Annual Standard Deviation*	*Average Daily Return*	*MDD*	*Annual Sharpe Ratio*
B&H	*50.32*	*11.98*	*25.21*	*0.0439*	*−3.92*	*0.47*
RB	*11.05*	*2.63*	***12.08***	*0.0096*	*−3.43*	*0.21*
RL	*35.54*	*8.46*	*21.90*	*0.0317*	*−3.78*	*0.37*
RB + RL	*29.32*	*6.98*	*15.32*	*0.0256*	*−3.26*	*0.46*
RB + RL + C1	*43.16*	*10.27*	*23.03*	*0.0377*	*−3.30*	*0.45*
RB + RL + C1 + C2	***59.37***	***14.14***	*20.89*	***0.0673***	***−2.82***	***0.68***

Note: The bold text represents the best value of each metric.

Table 6
Comparison of the frequency with which each model generates the same signals as RB.

Signal	*The number of times the same signal occurs at the same time as RB*							
	RL		*RB + RL*		*RB + RL + C1*		*RB + RL + C1 + C2*	
Buy	*60 (0.32)*	*(0.32)*	*121 (0.45)*	*(0.42)*	*94 (0.49)*	*(0.53)*	*87* ***(0.93)***	***(0.74)***
Sell	*0 (0.00)*		*86 (0.37)*		*36 (0.67)*		*44* ***(0.53)***	

Note: The values in brackets represent the ratio of the number of times the same signal as the RB occurred (Table 6) to the total number of signals (Table 5).
Note: The bold text represents the best value of each metric.

compared with other models, the RB + RL + C1 + C2 had the smallest MDD value and the largest SR value. These findings suggest that incorporating C2 into the algorithm can help adjust the trading strategy's risk by capturing profitable opportunities with adequate trading volume.

6.2. Verification of the reliability of the proposed hybrid model

6.2.1. Adaptability to various market conditions

We examine the adaptability of the model in both uptrend and downtrend market conditions in this section and investigate the robustness of the model during periods of a market crash in Section 6.2.2. We further divided the test dataset into one-year intervals, as shown in Table 8, and distinguished it into three scenarios: market crash, uptrend, and downtrend.

In the uptrend scenario, the hybrid model demonstrated the ability to identify the overall long-term upward trend of the test period. The results showed that the hybrid model achieved performance similar to the B&H strategy, as shown in Table 10. As shown in Fig. 4 (b), a significant number of buying signals were generated at the beginning of the investment period. On the other hand, as illustrated in Fig. 4 (e), the RB employed a loss-cut rule, which resulted in missed potential earnings when the market rapidly shifted. As a result, the RB resulted in a limited %AR performance, as shown in Fig. 4 (h).

In the downtrend scenario, we observed that our hybrid model captured profitable trading opportunities in the short term, as presented in Fig. 4 (c). Our model showed a relatively better performance than other models, as detailed in Table 10. Additionally, the proposed hybrid model generated fewer signals than other models, as shown in Table 9. On the contrary, due to the significant fluctuations within the test period, the RB frequently executed the loss-cut rule, resulting in missed profitable opportunities as shown in Fig. 4 (f). As a result, we confirmed that the RB resulted in a relatively limited %AR performance, as illustrated in Fig. 4 (i).

In general, market conditions comprise not only the overall long-term trend during the investment period but also numerous short-term fluctuations within it. However, the rules of the RB system are not sufficient to adaptively respond to significantly fluctuating market conditions, which can result in missed opportunities or poor decision-making. In contrast, our hybrid model is designed to adapt to dynamically changing market conditions by considering both the overall long-term trend in market prices and short-term trends. The results suggest that our hybrid model consistently generates profitability across various market scenarios.

6.2.2. Robustness in market crash periods

The results, as presented in Fig. 4 (a) and (d), show that the hybrid model made different decisions compared with the RB that employed a loss-cutting rule during the market crash. Although the loss-cutting rule of the RB reduced losses during the recession period, it resulted in stagnant returns and slow recovery until another sell signal occurred. In contrast, the hybrid model generated buying signals at low prices after sell signals occurred during the recession period, leading to a faster recovery than the RB. We conjecture that the hybrid model may have performed better trading strategies because the agent could explore alternative actions without relying overly on the RB's decision results. Moreover, the proposed model achieved adjusted-risk returns compared with other models during recession and recovery periods, as depicted in Fig. 4 (g). These findings suggest that the agent in the hybrid model can take more profitable autonomous actions while learning the decision information of the RB, enabling it to capture potential yields and reduce risks. The study demonstrates that our hybrid model has the ability for robustness under extreme market conditions.

Table 8
Separation of datasets to distinguish market conditions.

Dataset	*Start*	*End*
Market Crash	*2019-07-01*	*2020-07-31*
Up Trend	*2020-07-03*	*2021-07-30*
Down Trend	*2021-07-02*	*2022-07-29*

6.2.3. Comparison with other machine learning based on hybrid systems from previous studies

According to the results presented in Table 11, our model overall outperformed in terms of the %AR and SR compared to previous models. In general, system trading has shown challenges in outperforming the returns achieved by the B&H strategy [16]. Similarly, previous models underperformed the B&H strategy across all market conditions, as demonstrated in Table 11. While they only showed better performance in the uptrend with relatively lower volatility, they considerably underperformed during the market crash with high volatility and the downtrend with significant fluctuations, as shown in Table 11. This is attributed to the difficulty of making short-term predictions in highly volatile or frequently fluctuating situations [52], resulting in challenges in rapidly adapting and accurately predicting in such market conditions. Moreover, these models encounter false positive issues in prediction when the market faces unforeseen events [13], resulting in missed opportunities for promising profitable trading.

In contrast, our model achieved similar or superior performance to the B&H strategy across all market conditions. The RL proves valuable for addressing sequential decision problems, making it more suitable for adaptive decision-making in dynamic environments [8,67]. Therefore, our study employs a hybrid approach combining RL, demonstrating that our model's adaptability outperforms others, particularly in highly fluctuating and volatile scenarios. These are attributed ability of our model to update policies for maximizing returns and establishing a unique trading strategy by incorporating RB trading decision information. This finding indicates that our study is capable of making better decisions compared to previous hybrid studies.

6.2.4. Scalability across various index fund markets

We selected index fund markets with a large market capitalization and total trading volume, such as the NYSE Composite, DAX Performance Index, CAC40, Hang Seng Index, and KOSPI Composite Index, which are among Yahoo Finance's top ten global index funds. To ensure a more rigorous verification, we measured the performance by setting the parameters to be equal to those used in the experiments on the S&P500 index fund. We observed similar results in other index fund markets as those obtained in the S&P 500 index fund experiments, as follows. First, the RB + RL exhibited more dynamic behavior than the RL in all index funds, as detailed in Table 12. Second, the SD was lower for the RB + RL compared with the RL, as confirmed in Table 13. Third, the RB + RL + C1 had higher %AR values than the RB + RL in the majority of fund markets. Fourth, the RB + RL + C1 + C2 had higher %AR and SR values than the RB + RL in the majority of funds. Finally, as shown in Table 11, the number of signals decreased sequentially as follows: (RB + RL > RB + RL + C1 > RB + RL + C1 + C2). The consistent findings of our proposed hybrid model in other index fund markets demonstrate its scalability to different datasets.

6.2.5. Extensibility to another RB model

We employed the mean-reversion trading strategy [43] as an alternative to the RB model. Similar to the other experiments, we trained and tested the hybrid model using the same parameters and periods and performed it on the S&P500 index fund. As shown in Tables 14 and 15, the results revealed five findings, as discussed in Section 6.2.4. Therefore, these findings suggest that the hybrid structure has potential applications in various financial and other fields.

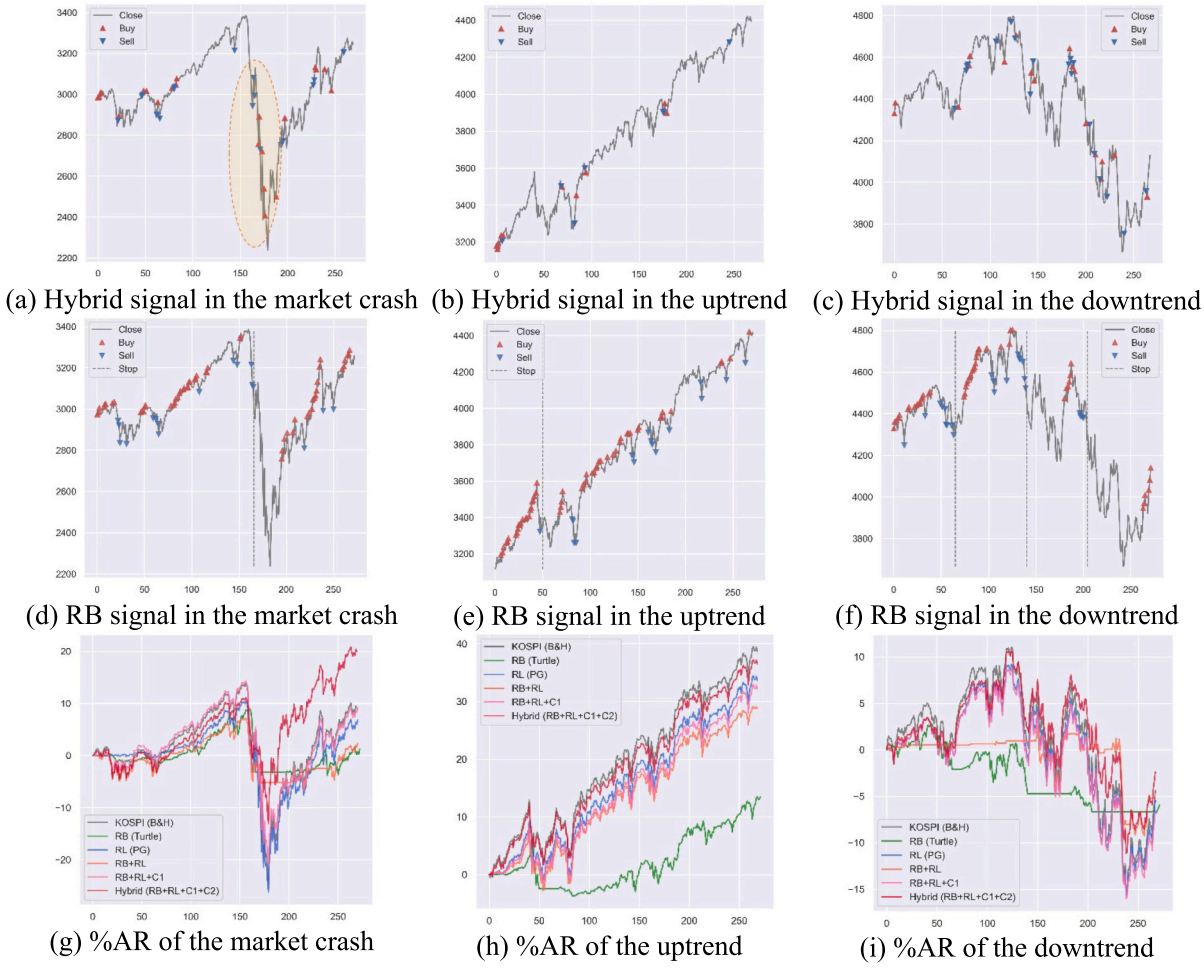


Fig. 4. Comparison of signals and %AR in different market conditions.

Table 9
The number of signals from each model in various market conditions.

Market	*Signal*	*The number of signals*				
		RB	*RL*	*RB + RL*	*RB + RL + C1*	*RB + RL + C1 + C2*
Market Crash	*Buy*	*56*	*39*	*110*	*87*	*37*
	Sell	*17*	*0*	*91*	*55*	*17*
	Stop	*1*	–	–	–	–
Up Trend	*Buy*	*55*	*21*	*108*	*80*	*29*
	Sell	*16*	*0*	*51*	*28*	*17*
	Stop	*1*	–	–	–	–
Down Trend	*Buy*	*49*	*16*	*73*	*83*	*19*
	Sell	*23*	*0*	*105*	*24*	*26*
	Stop	*3*	–	–	–	–

7. Implications

7.1. Theoretical implications

The proposed model and research findings make three major theoretical contributions. First, this study developed a hybrid model that combines an RB system and an RL algorithm to establish adaptive trading strategies. This model overcomes the limitations of previous methods [25,35,37,68] by enabling self-improvement of decisions and allowing direct trading without labeled data. This study utilized the decision information from the RB expert system to define the state space of the RL environment, facilitating the agent's dynamic behavior and more effective risk management. Therefore, our study successfully demonstrated the learning of trading strategies founded not only on market conditions but also on feedback from the RB's decision.

Second, previous studies on RL for trading have focused mostly on the state of market prices [8,12,67] and have rarely explored investor's status [23]. We investigated the impact of defining investors' available balance as a state space that can interact with the environment in RL trading and found that it adds value in improving yields. These findings suggest that the proposed state space can facilitate adaptive learning for agents in the way to utilize the current balance information as a signal for enhancing or adjusting current trading strategies. Finally, we proposed a mechanism that determines the trading volume using the action probabilities [30,66] of the PG algorithm. Although the trading volume in investment transactions has been reported to significantly affect yield improvement [27,32], many previous studies on RL trading have paid little attention to this aspect. From our experimental results, we confirmed that the mechanism improves the SR and reduces the number of false positive trading signals. These findings indicate that the proposed trading mechanism can make more reliable trading strategies when faced with volatile markets.

7.2. Managerial implications

The research findings derived from experiments make three significant practical contributions for managers in the financial domain. Firstly, our model outperformed the benchmark index funds as well as single models. Moreover, our hybrid model surpassed the majority of machine learning-based models examined in previous studies [6,11,35,37,65], indicating the potential of our model as a promising

Table 10
Evaluation of the performance of each model in various market conditions.

Market Conditions	Model	%AR	Standard Deviation	Average Daily Return	MDD	Sharpe Ratio
Market Crash	B&H	9.08	30.21	0.0338	−4.75	0.31
	RB	1.23	**9.68**	0.0063	**−4.32**	0.13
	RL	6.72	27.71	0.0252	−4.66	0.24
	RB + RL	2.34	12.65	0.0087	−5.51	0.18
	RB + RL + C1	9.13	26.87	0.0339	−5.47	0.34
	RB + RL + C1 + C2	**19.97**	20.89	**0.0743**	−4.72	**0.96**
Up Trend	B&H	**38.65**	17.27	**0.1442**	−3.10	**2.24**
	RB	13.43	**10.95**	0.0492	**−3.02**	1.23
	RL	33.66	16.13	0.1255	−3.23	2.09
	RB + RL	28.75	14.08	0.1073	−3.17	2.04
	RB + RL + C1	33.12	15.94	0.1198	−3.09	2.08
	RB + RL + C1 + C2	36.53	16.60	0.1364	−3.16	2.21
Down Trend	B&H	−4.51	19.67	−0.0165	−3.74	−0.23
	RB	−5.95	**6.84**	−0.0218	**−2.24**	−0.87
	RL	−5.48	18.49	−0.0205	−3.26	−0.29
	RB + RL	−5.21	9.02	−0.0195	−3.03	−0.58
	RB + RL + C1	−5.74	17.32	−0.0212	−2.48	−0.33
	RB + RL + C1 + C2	**−2.39**	16.67	**−0.0089**	−2.39	**−0.14**

Note: The bold text represents the best value of each metric.

trading system. Secondly, we have demonstrated that our model can achieve consistently profitable returns across various market conditions [28,48]. In the upward-trending market condition, our model showed strong performance with minimal signal occurrence, effectively aligning with substantial long-term uptrend. Furthermore, during the market crash, our model exhibited a relatively swift response, which suggests its capability to react effectively in demanding market conditions. Thirdly, our model showed the potential of combining RB with RL models in developing new trading strategies. One of the major problems with the RB model is its inability to adjust to the changing market conditions. While we employed the turtle model as an exemplar of the RB model, various other RB models could be experimented with. When properly trained, these combined models can significantly minimize false positive buy or sell signals, as demonstrated in our model. This will provide valuable guidance for making decision signals that are more robust across various market situations.

7.3. Limitations and future directions

While this study has academic and managerial implications, it has several limitations. Firstly, the RL algorithm requires large amounts of data to train effectively, and in practical industrial applications, it is necessary to use more data to ensure model stability [38]. Secondly, the variations in the deep neural networks of RL agents and modifying the policy update method are beyond the scope of this study. However, real industry applications might require additional policy updates and deep neural network variations [14,67] to ensure superior performance. Thirdly, while we demonstrate how the proposed components affect the agent's decision-making and performance, this does not guarantee interpretability and transparency [44]. Finally, our system is based on the assumption that the agent's actions have a rarely significant impact on market prices, which is a prerequisite for systematic trading [24]. Therefore, the system is only suitable for large-cap stocks such as the index fund market, and not appropriate for small-value stocks, which require trading decisions based on corporate value. We suggest future directions to explore models that recommend stocks that can be applied to our system. Furthermore, the proposed hybrid approach can be extended to the asset allocation algorithm.

8. Conclusion

In this study, a novel hybrid model that combines an RB system and RL algorithm is proposed. In contrast to previous studies, the proposed model does not require labels and has the ability to generate more adaptive trading signals. This approach enables the agent to adapt to changing market conditions and manage risks by establishing its trading strategies through learning expertise-based information. This study shows that combining these two approaches can reduce investment risk by improving its comprehensive thinking ability. Moreover, we examined that defining investors' available balance states that interact with the environment can facilitate adaptive learning for agents and improve the %AR. Additionally, we propose a trading mechanism that adjusts the

Table 12
The number of signals generated by each model across the five index fund markets.

Market	Signal	The number of signals				
		RB	RL	RB + RL	RB + RL + C1	RB + RL + C1 + C2
NYSE	Buy	94	250	139	158	56
	Sell	89	0	194	45	70
	Stop	3	–	–	–	–
DAX	Buy	91	46	146	60	135
	Sell	75	0	189	57	197
	Stop	7	–	–	–	–
CAC40	Buy	111	151	106	74	73
	Sell	107	0	151	44	103
	Stop	4	–	–	–	–
HSI	Buy	84	68	143	185	134
	Sell	70	0	228	264	120
	Stop	5	–	–	–	–
KOSPI	Buy	131	310	199	106	68
	Sell	107	116	301	188	151

Table 11
Comparison of our proposed model with previous hybrid models.

Test Period	Entire		Market Crash		Up Trend		Down Trend	
Metric	%AR	SR	%AR	SR	%AR	SR	%AR	SR
B&H	50.32	0.47	9.08	0.31	38.65	2.24	−4.51	−0.23
TI + SVM	23.53	0.27	2.23	0.16	22.65	1.98	−8.46	−0.45
TI + RF	32.38	0.34	4.47	0.21	27.42	2.02	−7.56	−0.41
TI + LSTM	40.50	0.41	6.78	0.24	34.13	2.11	−6.59	−0.36
TI + XGBoost+CNN + LSTM	45.69	0.45	7.14	0.28	36.45	2.20	−5.42	−0.28
RB + RL + C1 + C2 (Ours)	59.37	0.68	19.97	0.96	36.53	2.21	−2.39	−0.14

Table 13
Evaluation of the performance of each model was measured using six metrics across five index fund markets.

Market	Model	%AR	Average Annual Return	Annual Standard Deviation	Average Daily Return	MDD	Annual Sharpe Ratio
NYSE	B&H	16.82	4.01	19.83	0.0147	−3.09	0.20
	RB	8.96	2.13	**9.23**	0.0078	−2.92	0.23
	RL	16.69	3.97	17.99	0.0146	−3.09	0.22
	RB + RL	9.56	2.04	12.18	0.0075	−2.87	0.16
	RB + RL + C1	17.84	4.25	17.26	0.0156	−2.64	0.24
	RB + RL + C1 + C2	**29.58**	**7.04**	15.09	**0.0232**	**−2.58**	**0.47**
DAX	B&H	0.74	0.18	19.77	0.0006	−3.71	0.01
	RB	−16.01	−3.81	9.04	−0.0138	−3.91	−0.42
	RL	−2.88	−0.68	18.12	−0.0025	−2.88	−0.03
	RB + RL	0.89	0.21	**2.70**	0.0007	**−2.85**	0.07
	RB + RL + C1	8.04	1.91	20.39	0.0069	−3.86	0.09
	RB + RL + C1 + C2	**15.79**	**3.76**	14.89	**0.0126**	−3.43	**0.25**
CAC40	B&H	17.51	4.17	20.29	0.0149	−3.41	0.21
	RB	−1.02	−0.24	9.26	−0.0009	−3.21	−0.03
	RL	19.66	4.68	19.87	0.0168	−3.88	0.24
	RB + RL	−2.05	−0.49	**6.31**	−0.0018	−4.76	−0.08
	RB + RL + C1	23.82	5.67	17.14	0.0204	−3.94	0.33
	RB + RL + C1 + C2	**36.02**	**8.57**	10.48	**0.0299**	**−2.23**	**0.82**
HSI	B&H	−34.77	−8.28	17.82	−0.0309	−2.60	−0.46
	RB	−12.47	−2.96	7.96	−0.0111	−2.58	−0.37
	RL	−32.74	−7.79	16.52	−0.029	−2.61	−0.47
	RB + RL	−9.63	−2.29	7.74	−0.0086	−2.88	−0.29
	RB + RL + C1	−8.53	−2.03	11.87	−0.0076	−2.27	−0.17
	RB + RL + C1 + C2	**20.83**	**4.95**	**6.42**	**0.0186**	**−2.04**	**0.78**
KOSPI	B&H	−2.46	−0.58	17.36	−0.0022	−2.76	−0.03
	RB	**21.22**	**5.05**	**11.13**	**0.0189**	−2.95	**0.45**
	RL	6.71	1.59	16.68	0.0059	−2.88	0.09
	RB + RL	10.64	2.53	11.49	0.0095	−3.38	0.22
	RB + RL + C1	8.40	2.04	12.56	0.0075	−3.69	0.16
	RB + RL + C1 + C2	10.92	2.62	13.24	0.0097	**−2.61**	0.21

Note: The bold text represents the best value of each metric.

Table 14
The number of signals from each model applied with an alternative RB model.

Signal	The number of signals				
	RB	RL	RB + RL	RB + RL + C1	RB + RL + C1 + C2
Buy	62	185	137	110	74
Sell	59	0	108	94	67

Table 15
Evaluation of the performance of each model applied with an alternative RB model was measured using six metrics.

Model	%AR	Annual AR	Annual SD	DR	MDD	Annual SR
B&H	50.32	11.98	25.21	0.0439	−3.92	0.47
RB	47.11	11.22	**13.22**	0.0409	**−2.64**	**0.84**
RL	25.09	5.97	21.64	0.0224	−3.78	0.27
RB + RL	40.21	9.57	14.61	0.0351	−3.14	0.66
RB + RL + C1	51.65	12.30	19.01	0.0435	−3.18	0.65
RB + RL + C1 + C2	**62.28**	**14.83**	18.57	**0.0543**	−2.91	0.80

trading volume using the action probability of the PG algorithm, which improves the agent's trading efficiency.

While our study has some limitations, such as the lack of interpretability and transparency in its learning processes, it has significant academic and managerial implications. We advance adaptive decision-making problems within the financial domain by showing better SR across various market scenarios, in comparison to both prior hybrid models and the benchmark index funds. Furthermore, this study demonstrates the potential to reduce transaction costs by improving returns with fewer trading signals than other models. Finally, the reliability of the model was examined in terms of scalability across different markets, and the extensibility of the proposed hybrid structure was evaluated, indicating its potential as a promising practical solution that can be applied to a real-world trading system. The study suggests that combining RB and RL techniques can leverage the strengths of both approaches and overcome their weaknesses, resulting in effective risk-adjusted trading strategies. Therefore, our model can be useful for professional traders, financial analysts, and individual investors in the future. This study highlights new directions for advanced machine-learning approaches in financial research. Finally, future research can extend the proposed hybrid structure to address other fields' problems caused by adaptive decision-making.

CRediT authorship contribution statement

Yuhee Kwon: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data curation, Writing – original draft, Validation, Visualization, Project administration. **Zoonky Lee:** Conceptualization, Validation, Writing – review & editing, Supervision, Resources.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] Algo trading dominates 80% of stock market, in. https://seekingalpha.com/article/4230982-algo-trading-dominates-4230980-percent-of-stock-market, 2019.

[2] What Percentage Of Trading Is Algorithmic? (Algo Trading Volume), in. https://therobusttrader.com/what-percentage-of-trading-is-algorithmic/, 2022.

[3] F. Allen, R. Karjalainen, Using genetic algorithms to find technical trading rules, J. Financ. Econ. 51 (2) (1999) 245–271, https://doi.org/10.1016/S0304-405X(98)00052-X.

[4] A.M. Andrew, Reinforcement learning: an introduction, in: Richard S. Sutton, Andrew G. Barto (Eds.), Adaptive Computation and Machine Learning series, MIT Press (Bradford Book), Cambridge, Mass, 1998 xviii+ 322 pp, ISBN 0–262–19398-1,(hardback,£ 31.95), Robotica, 17(2) (1999) 229–235.

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

[6] A. Bahrammirzaee, A comparative survey of artificial intelligence applications in finance: artificial neural networks, expert system and hybrid intelligent systems, Neural Comput. & Applic. 19 (8) (2010) 1165–1195, https://doi.org/10.1007/s00521-010-0362-z.

[7] W. Blaschke, M.T. Jones, G. Majnoni, M.S. Martinez Peria, Stress *T*esting of *F*inancial *S*ystems: *A*n *O*verview of *I*ssues, *M*ethodologies, and FSAP *E*xperiences, 2001, https://doi.org/10.5089/9781451851168.001.

[8] S. Carta, A. Ferreira, A.S. Podda, D.R. Recupero, A. Sanna, Multi-DQN: An ensemble of *d*eep Q-learning agents for stock market forecasting, Expert Syst. Appl. 164 (2021) 113820, https://doi.org/10.1016/j.eswa.2020.113820.

[9] A.P. Chaboud, B. Chiquoine, E. Hjalmarsson, C. Vega, Rise of the machines: *a*lgorithmic trading in the foreign exchange market, J. Financ. 69 (5) (2014) 2045–2084, https://doi.org/10.1111/jofi.12186.

[10] P.-C. Chang, T.W. Liao, J.-J. Lin, C.-Y. Fan, A dynamic threshold decision system for stock trading signal detection, Appl. Soft Comput. 11 (5) (2011) 3998–4010, https://doi.org/10.1016/j.asoc.2011.02.029.

[11] P.-C. Chang, J.-L. Wu, J.-J. Lin, A Takagi–Sugeno fuzzy model combined with a support vector regression for stock trading forecasting, Appl. Soft Comput. 38 (2016) 831–842, https://doi.org/10.1016/j.asoc.2015.10.030.

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

[13] M.L. De Prado, Advances in *F*inancial *M*achine *L*earning, John Wiley & Sons, 2018.

[14] Y. Deng, F. Bao, Y. Kong, Z. Ren, Q. Dai, Deep direct reinforcement learning for financial signal representation and trading, IEEE Trans. Neural Networks Learn. Syst. 28 (3) (2016) 653–664, https://doi.org/10.1109/TNNLS.2016.2522401.

[15] W.G. Dewald, J.G. Thursby, R.G. Anderson, Replication in empirical economics: *t*he journal of money, credit and banking project, Am. Econ. Rev. (1986) 587–603. http://www.jstor.org/stable/1806061.

[16] H. Dichtl, Investing in the S&P 500 index: *c*an anything beat the buy-and-hold strategy? Rev. Financ. Econ. 38 (2) (2020) 352–378, https://doi.org/10.1002/rfe.1078.

[17] R.D. Donchian, Commodities: *h*igh finance in copper, Financ. Anal. J. 16 (6) (1960) 133–142, https://doi.org/10.2469/faj.v16.n6.133.

[18] L. Duboc, D. Rosenblum, T. Wicks, A framework for characterization and analysis of software system scalability, in: Proceedings of the the 6th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering, 2007, pp. 375–384, https://doi.org/10.1145/1287624.1287679.

[19] D. Eilers, C.L. Dunis, H.-J. von Mettenheim, M.H. Breitner, Intelligent trading of seasonal effects: *a* decision support algorithm based on reinforcement learning, Decis. Support. Syst. 64 (2014) 100–108, https://doi.org/10.1016/j.dss.2014.04.011.

[20] C. Faith, Way of the Turtle: The Secret Methods that Turned Ordinary People into Legendary Traders: The Secret Methods that Turned Ordinary People into Legendary Traders, McGraw Hill Professional, 2007.

[21] S. Feuerriegel, H. Prendinger, News-based trading strategies, Decis. Support. Syst. 90 (2016) 65–74, https://doi.org/10.1016/j.dss.2016.06.020.

[22] T. Fischer, C. Krauss, Deep learning with long short-term memory networks for financial market predictions, Eur. J. Oper. Res. 270 (2) (2018) 654–669, https://doi.org/10.1016/j.ejor.2017.11.054.

[23] T.G. Fischer, Reinforcement learning in financial markets-a survey, in: FAU Discussion Papers in Economics, 2018. *http://hdl.handle.net/10419/183139*.

[24] V. François-Lavet, P. Henderson, R. Islam, M.G. Bellemare, J. Pineau, An introduction to deep reinforcement learning, Found. Trends® in Machine Learn. 11 (3–4) (2018) 219–354, https://doi.org/10.1561/2200000071.

[25] D.P. Gandhmal, K. Kumar, Systematic analysis and review of stock market prediction techniques, Comp. Sci. Rev. 34 (2019) 100190, https://doi.org/10.1016/j.cosrev.2019.08.001.

[26] E.A. Gerlein, M. McGinnity, A. Belatreche, S. Coleman, Evaluating machine learning classification for financial trading: An empirical approach, Expert Syst. Appl. 54 (2016) 193–207, https://doi.org/10.1016/j.eswa.2016.01.018.

[27] S. Gervais, R. Kaniel, D.H. Mingelgrin, The high-volume return premium, J. Financ. 56 (3) (2001) 877–919, https://doi.org/10.1111/0022-1082.00349.

[28] M. Ghahramani, H.E. Najafabadi, Compatible Deep Neural Network Framework with Financial Time Series Data, Including Data Preprocessor, Neural Network Model and Trading Strategy, arXiv preprint *arXiv:2205.08382*, 2022, https://doi.org/10.48550/arXiv.2205.08382.

[29] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT press, 2016.

[30] A. Gosavi, A reinforcement learning algorithm based on policy iteration for average reward: *e*mpirical results with yield management and convergence analysis, Mach. Learn. 55 (2004) 5–29, https://doi.org/10.1023/B:MACH.0000019802.64038.6c.

[31] H. Grigoryan, Stock market trend prediction using support vector machines and variable selection methods, in: 2017 International Conference on Applied Mathematics, Modelling and Statistics Application (AMMSA 2017), Atlantis Press, 2017, pp. 210–213, https://doi.org/10.2991/ammsa-17.2017.45.

[32] S. Gupta, D. Das, H. Hasim, A.K. Tiwari, The dynamic relationship between stock returns and trading volume revisited: *a* MODWT-VAR approach, Financ. Res. Lett. 27 (2018) 91–98, https://doi.org/10.1016/j.frl.2018.02.018.

[33] M. Hindman, Building better models: *p*rediction, replication, and machine learning in the social sciences, Ann. Am. Acad. Pol. Soc. Sci. 659 (1) (2015) 48–62, https://doi.org/10.1177/0002716215570279.

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

[35] Y. Hu, K. Liu, X. Zhang, L. Su, E. Ngai, M. Liu, Application of evolutionary computation for rule discovery in stock algorithmic trading: *a* literature review, Appl. Soft Comput. 36 (2015) 534–551, https://doi.org/10.1016/j.asoc.2015.07.008.

[36] C.Y. Huang, Financial Trading as a Game: A Deep Reinforcement Learning Approach, arXiv preprint, 2018. *arXiv:1807.02787*. 10.48550/arXiv.1807.02787.

[37] W. Jiang, Applications of deep learning in stock market prediction: recent progress, Expert Syst. Appl. 184 (2021) 115537, https://doi.org/10.1016/j.eswa.2021.115537.

[38] M.I. Jordan, T.M. Mitchell, Machine learning: *t*rends, perspectives, and prospects, Science 349 (6245) (2015) 255–260, https://doi.org/10.1126/science.aaa8415.

[39] A. Kayal, A neural networks filtering mechanism for foreign exchange trading signals, in: 2010 IEEE international conference on intelligent computing and intelligent systems, IEEE, 2010, pp. 159–167, https://doi.org/10.1109/ICICISYS.2010.5658495.

[40] L. Khaidem, S. Saha, S.R. Dey, Predicting the direction of stock market prices using random forest, arXiv preprint *arXiv:1605.00003*, 2016, https://doi.org/10.48550/arXiv.1605.00003.

[41] Y. Kim, W. Ahn, K.J. Oh, D. Enke, An intelligent hybrid trading system for discovering trading rules for the futures market using rough sets and genetic algorithms, Appl. Soft Comput. 55 (2017) 127–140, https://doi.org/10.1016/j.asoc.2017.02.006.

[42] D.P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, arXiv preprint *arXiv:1412.6980*, 2014, https://doi.org/10.48550/arXiv.1412.6980.

[43] T.S.-T. Leung, X. Li, Optimal Mean Reversion Trading: Mathematical Analysis and Practical Applications, World Scientific, 2015.

[44] G. Liu, O. Schulte, W. Zhu, Q. Li, Toward interpretable deep reinforcement learning with linear model u-trees, in: Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2018, Dublin, Ireland, September 10–14, 2018, Proceedings, Part II 18, Springer, 2019, pp. 414–429, https://doi.org/10.48550/arXiv.1807.05887.

[45] H. Liu, Optimal consumption and investment with transaction costs and multiple risky assets, J. Financ. 59 (1) (2004) 289–338, https://doi.org/10.1111/j.1540-6261.2004.00634.x.

[46] P. Liu, Y. Zhang, F. Bao, X. Yao, C. Zhang, Multi-type data fusion framework based on deep reinforcement learning for algorithmic trading, Appl. Intell. (2022) 1–24, https://doi.org/10.1007/s10489-022-03321-w.

[47] X. Liu, H. An, L. Wang, X. Jia, An integrated approach to optimize moving average rules in the EUA futures market based on particle swarm optimization and genetic algorithms, Appl. Energy 185 (2017) 1778–1787, https://doi.org/10.1016/j.apenergy.2016.01.045.

[48] S. Mehtab, J. Sen, A. Dutta, Stock price prediction using machine learning and LSTM-based deep learning models, in: Machine Learning and Metaheuristics Algorithms, and Applications: Second Symposium, SoMMA 2020, Chennai, India, October 14–17, 2020, Revised Selected Papers 2, Springer, 2021, pp. 88–106, https://doi.org/10.1007/978-981-16-0419-5_8.

[49] R. Meyes, M. Lu, C.W. de Puiseau, T. Meisen, Ablation Studies in Artificial Neural Networks, arXiv preprint *arXiv:1901.08644*, 2019, https://doi.org/10.48550/arXiv.1901.08644.

[50] T.J. Moskowitz, Y.H. Ooi, L.H. Pedersen, Time series momentum, J. Financ. Econ. 104 (2) (2012) 228–250, https://doi.org/10.1016/j.jfineco.2011.11.003.

[51] H.J. Park, Y. Kim, H.Y. Kim, Stock market forecasting using a multi-task approach integrating long short-term memory and the random forest framework, Appl. Soft Comput. 114 (2022) 108106, https://doi.org/10.1016/j.asoc.2021.108106.

[52] I.R. Parray, S.S. Khurana, M. Kumar, A.A. Altalbe, Time series data analysis of stock price movement using machine learning techniques, Soft. Comput. 24 (2020) 16509–16517, https://doi.org/10.1007/s00500-020-04957-x.

[53] R.K. Raut, N. Das, R. Mishra, Behaviour of individual investors in stock market trading: *e*vidence from India, Glob. Bus. Rev. 21 (3) (2020) 818–833, https://doi.org/10.1177/0972150918778915.

[54] S. Santurkar, D. Tsipras, A. Ilyas, A. Madry, How does batch normalization help optimization? Adv. Neural Inf. Proces. Syst. 31 (2018) https://doi.org/10.48550/arXiv.1805.11604.

[55] J. Schmidt-Hieber, Nonparametric regression using deep neural networks with ReLU activation function, Ann. Stat. 48 (4) (2020) 1875–1897, https://doi.org/10.1214/19-AOS1875.

[56] G. Sermpinis, C. Dunis, J. Laws, C. Stasinakis, Forecasting and trading the EUR/USD exchange rate with stochastic *n*eural *n*etwork combination and time-varying leverage, Decis. Support. Syst. 54 (1) (2012) 316–329, https://doi.org/10.1016/j.dss.2012.05.039.

[57] S. Shalev-Shwartz, S. Ben-David, Understanding *M*achine *L*earning: From *T*heory to *A*lgorithms, Cambridge University Press, 2014, https://doi.org/10.1017/CBO9781107298019.
[58] A. Silva, M. Gombolay, Encoding human domain knowledge to warm start reinforcement learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2021, pp. 5042–5050, https://doi.org/10.1609/aaai.v35i6.16638.
[59] P. Slovic, Psychological study of human judgment: *i*mplications for investment decision making, J. Financ. 27 (4) (1972) 779–799, https://doi.org/10.2307/2978668.
[60] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, Journal Machine Learn. Res. 15 (1) (2014) 1929–1958, https://doi.org/10.5555/2627435.2670313.
[61] X. Sun, P. Wu, S.C. Hoi, Face detection using deep learning: An improved faster RCNN approach, Neurocomputing 299 (2018) 42–50, https://doi.org/10.1016/j.neucom.2018.03.030.
[62] R. Tsaih, Y. Hsu, C.C. Lai, Forecasting S&P 500 stock index futures with a hybrid AI system, Decis. Support. Syst. 23 (1998) 161–174, https://doi.org/10.1016/S0167-9236(98)00028-1.
[63] A. Tsantekidis, N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, A. Iosifidis, Using deep learning to detect price change indications in financial markets, in: 2017 25th European Signal Processing Conference (EUSIPCO), IEEE, 2017, pp. 2511–2515, https://doi.org/10.23919/EUSIPCO.2017.8081663.
[64] F. Wang, L. Philip, D.W. Cheung, Combining technical trading rules using particle swarm optimization, Expert Syst. Appl. 41 (6) (2014) 3016–3026, https://doi.org/10.1016/j.eswa.2013.10.032.
[65] Q. Wen, Z. Yang, Y. Song, P. Jia, Automatic stock decision support system based on box theory and SVM algorithm, Expert Syst. Appl. 37 (2) (2010) 1015–1022, https://doi.org/10.1016/j.eswa.2009.05.093.
[66] R.J. Williams, Simple statistical gradient-following algorithms for connectionist reinforcement learning, Mach. Learn. 8 (3) (1992) 229–256, https://doi.org/10.1023/A:1022672621406.
[67] X. Wu, H. Chen, J. Wang, L. Troiano, V. Loia, H. Fujita, Adaptive stock trading strategies with deep reinforcement learning methods, Inf. Sci. 538 (2020) 142–158, https://doi.org/10.1016/j.ins.2020.05.066.
[68] Y. Yoon, T. Guimaraes, G. Swales, Integrating artificial neural networks with rule-based expert systems, Decis. Support. Syst. 11 (5) (1994) 497–507, https://doi.org/10.1016/0167-9236(94)90021-3.
[69] Y. Zhang, P. Zhao, Q. Wu, B. Li, J. Huang, M. Tan, Cost-sensitive portfolio selection via deep reinforcement learning, IEEE Trans. Knowl. Data Eng. 34 (1) (2020) 236–248, https://doi.org/10.1109/TKDE.2020.2979700.

Yuhee Kwon is pursuing a Ph.D. in the Department of Information Systems at Yonsei University in Korea. She received a BS degree from Hankuk University of Foreign Studies, Korea, in August 2014 and an MS degree from Ewha Womans University, Korea, in August 2016. She was a business consultant at a member firm of KPMG International from 2019 to 2021. Her research interests are e-business strategy in data science, artificial intelligence and its applications, and machine learning in finance.

Zoonky Lee is a professor at the Graduate School of Information at Yonsei University in Korea. He holds a Ph.D. from the University of Southern California and a master's degree from the University of Michigan in Statistics and Carnegie Mellon University in Social and Decision Sciences, respectively. Before joining Yonsei University, he was an Assistant Professor at the University of Nebraska, Lincoln, from 1999 to 2003 and a business consultant at Coopers and Lybrand from 1991 to 1994. His research interests include IT impact on business strategy and e-business strategy in data science and pricing. He has published in various IT journals, including Communications of the ACM, Communications of AIS, Computers and Security, Information and Management, Journal of Information Technology, Data Base, Journal of Organizational Computing and Electronic Commerce, and Journal of Business Strategies.