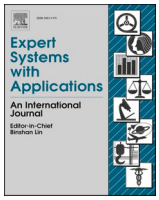
Expert Systems With Applications 254 (2024) 124465

Contents lists available at ScienceDirect

Expert Systems With Applications

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



Pro Trader RL: Reinforcement learning framework for generating trading knowledge by mimicking the decision-making patterns of professional traders

Da Woon Jeong [a,1], Yeong Hyeon Gu [b,*]

[a] Departments of Computer Engineering, Sejong University, Neungdong-ro, Gwangjin-gu, Seoul, Republic of Korea
[b] Department of Artificial Intelligence and Data Science, Sejong University, Neungdong-ro, Gwangjin-gu, Seoul, Republic of Korea

ARTICLE INFO

Keywords:
Deep Reinforcement learning
Artificial intelligence
Trader knowledge
Trading policy
Stock trading
Stock trading strategy

ABSTRACT

This study proposes a novel reinforcement learning (RL) framework, professional trader RL (Pro Trader RL), which mimics the decision-making patterns and trading philosophy of professional traders in stock trading. By exploiting the characteristics of RL, the framework aims to learn efficient trading strategies while mimicking the trading philosophy and risk management methods of professional traders. The framework takes into account the complex nature of the stock market and presents an integrated approach to RL, from data pre-processing to buying, selling and stop-loss. Pro Trader RL consists of four main modules. Data Preprocessing, Buy Knowledge RL, Sell Knowledge RL and Stop Loss Rule, each of which plays the role of professional traders knowledge. The results of three experiments show that the framework achieves high returns and Sharpe ratio regardless of market conditions and has stable performance with low maximum drawdown (MDD), which is superior to the state-of-the-art research. The proposed framework provides a novel approach to applying RL to the stock market and is expected to be useful and applicable in real-world trading settings.

1. Introduction

For decades, traders and academics have explored various methodologies to predict market movements and create optimal trading strategies (Shleifer, 2000). Traditional stock trading research has primarily focused on fundamental and technical analysis (Murphy, 1999). Fundamental analysis involves assessing whether a stock is overvalued or undervalued based on company financial statements, industry trends and economic indicators (Graham et al., 1962), while technical analysis attempts to predict future stock price movements based on historical data and trading volume (Edwards et al., 2018; Pring, 2021). On the other hand, traditional asset allocation research has focused on balancing risk and return based on correlations and expected returns among various assets (Markowitz, 1952). However, these traditional methodologies have struggled to cope with the complexity, irregularity and rapid changes in the stock market (Lo, 2004).

In particular, they cannot fully reflect the characteristics of modern markets, such as globalization, technological advances and the diversification of financial instruments (Stulz, 2009). To overcome these limitations, research centered on reinforcement learning (RL) has received significant attention in recent years (Silver et al., 2016; Deng et al., 2016). RL is a machine learning methodology where an agent learns reward-maximizing behaviors through interaction with its environment (Sutton & Barto, 2018). In stock trading, RL has mainly been used to optimize buy and sell timing and portfolio allocation strategies (Jiang et al., 2017). Various algorithms and models have interacted with market data to explore optimal trading strategies (Bao et al., 2017). Despite various models interacting with market data to explore optimal trading strategies, existing RL-based stock trading research has been limited in its ability to comprehensively incorporate the trading philosophy and risk management strategies of professional traders.

The Professional Trader RL (Pro Trader RL) framework proposed in this study is designed to transcend the limitations of traditional RL approaches in financial trading. Unlike previous works that simplify and consolidate the decision-making process into a singular model, Pro Trader RL meticulously replicates the complex and nuanced decision-making patterns of professional traders. This framework not only employs RL to model trading actions but is also specifically engineered

* Corresponding author.
E-mail addresses: chris410@sju.ac.kr (D.W. Jeong), yhgu@sejong.ac.kr (Y.H. Gu).
[1] ORCID: https://orcid.org/0000-0001-8913-3798.

https://doi.org/10.1016/j.eswa.2024.124465
Received 27 October 2023; Received in revised form 28 May 2024; Accepted 7 June 2024
Available online 8 June 2024

around the sophisticated trading philosophy, portfolio management strategies and risk management techniques used by professionals. It achieves this by structuring the internal components of the RL system to independently model the separate facets of a traders decision-making process.

Pro Trader RL does not limit state inputs to specific stocks or financial indicators. Instead, it encompasses a broader spectrum of market data and trader insights, representing the holistic view that professional traders typically consider. Instead of relying solely on profit or asset value, the rewards in Pro Trader RL are defined relative to various possible outcomes, closely aligning with how traders evaluate different trading scenarios. This approach allows the model to learn from specific cases in a manner that professional traders would, emphasizing learning from both successes and non-ideal outcomes. Actions within Pro Trader RL are not just buy, sell, or hold they involve intricate judgments that incorporate a traders trading philosophy at each decision point, mimicking the real-time decision-making process of seasoned traders. Additionally, the framework integrates a stop-loss rule directly derived from common practices in professional trading, ensuring that risk management is not an afterthought but a fundamental component of the decision-making process. The main contributions of this research can be summarized as follows.

1. Pro Trader RL uniquely structures each component of the professional traders decision-making process into discrete, interconnected modules. This novel approach allows each segment, from market analysis to trade execution, to be individually optimized, improving overall trading efficiency.
2. The framework provides a reinforcement learning environment and agent that segments decision-making processes from a professional trader's perspective and applies tailored learning methods to each process. This customization ensures that strategies are optimally adapted to the specific needs of different trading decisions.
3. Pro Trader RL introduces a novel method for calculating relative rewards, which are based on performance metrics for each decision process. This approach allows for more precise reinforcement signals, improving the learning efficiency and effectiveness of the trading strategies.
4. By integrating advanced risk management protocols directly into the RL modules, Pro Trader RL not only seeks to maximize profits, but also places a strong emphasis on sustainable trading by effectively managing potential financial risks.
5. Rigorously tested in a variety of market scenarios, the framework demonstrates its ability to maintain stability and deliver high returns even under volatile conditions, outperforming existing state-of-the-art models.

The remainder of this paper is organized as follows. Section 2 presents related background and discusses related work. Section 3 describes the proposed Pro Trader RL framework. Section 4 presents the experimental results. Finally, Section 5 concludes the paper and discusses future work.

2. Background

2.1. Professional traders trading rules

The trading strategies of professional traders are based on clear trading principles, risk management and meticulous market analysis (Douglas, 2001; Murphy, 1999; Covel, 2009; Elder, 2002). They strive for stability and sustainability in their trading and their principles also serve as useful guidelines for retail investors and academic researchers (Tharp et al., 2007). Professional traders conduct in-depth analyses of financial conditions, market trends and economic traders to establish detailed trading rules for various financial instruments (Murphy, 1999; Elder, 2002). These rules drive trading decisions by providing direction for setting clear goals, managing risk and securing consistent returns.

These popular, well-known and actively researched trading rules are relatively accessible (Schwager, 2012; Covel, 2009). However, even if the average trader follows the same trading rules, their results may differ significantly from those of professional traders. This disparity arises because professional traders augment and utilize trading rules with their own trading philosophy, shaped by years of experience (Taleb, 2007; Steenbarger, 2015). Additionally, their personalized risk management strategies effectively limit losses across their entire portfolio (Tharp et al., 2007).

Consequently, the difference between casual traders and professional traders lies in their trading philosophy, experience and ability to use various technical analysis tools to precisely determine when to buy and sell (Elder, 2002). This precision is contingent on how systematic and rational each trader is in their approach to trading.

Pro Trader RL, as currently implemented, primarily utilizes technical indicators and market price data, which are directly observable and quantifiable. This approach is based on the hypothesis that these factors predominantly influence short-term trading decisions. This approach was selected to initially validate the framework's ability to effectively learn and make decisions based on data that is readily available and commonly used in trading algorithms. However, we acknowledge that professional traders incorporate a broader spectrum of data, including macroeconomic indicators, company fundamentals and more comprehensive market analysis, all of which significantly enrich their decision-making framework. The exclusion of these data types in the current version is primarily because we aim to build a robust model for understanding and replicating the simple elements of trading decisions based on technical analysis.

2.2. Reinforcement learning (RL)

RL is a branch of machine learning where learners acquire optimal behavioral strategies through interaction with their environment, based on reward and punishment signals (Sutton & Barto, 2018; Mnih et al., 2015). It is closely related to decision optimization problems and finds applications across various fields including game theory, robotics, finance and healthcare (Kochenderfer, 2015). The subject of RL, the agent, interacts with the environment and tries to discover an optimal policy (Silver et al., 2017).

A state is information that describes the current situation in the environment and an action is an action that an agent can take based on the state. Rewards are the immediate feedback of the agent's actions and policies are the rules or strategies that determine actions based on the state (Duan et al., 2016). The agent understands the current state of the environment and selects the action that can maximize the reward among several possible actions. Through this process, RL is effectively utilized to solve complex problems in various domains (See Fig. 1).

The use of RL in financial trading can be seen in a variety of applications, including portfolio optimization, algorithmic trading and credit scoring (Moody & Saffell, 2001; Deng et al., 2016; Jiang et al., 2017; Bertoluzzo & Corazza, 2012; Xiong et al., 2018). RL is used to learn trading strategies and maximize rewards within a given market

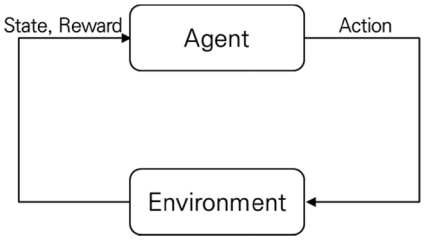


Fig. 1. Basic architecture of reinforcement learning (RL): an agent, environment, state, action and reward.

environment, which is actively researched as an automated decision process to reduce the burden on traders and generate high returns at the same time. In the financial trading domain, RL agents act as a kind of algorithmic trader or portfolio manager, deciding on actions such as buy, hold and sell (Nevmyvaka et al., 2006). The environment is the financial market and the state is provided by various technical indicators and variables of the market. While the return from a transaction is primarily used as a reward, other risk-adjusted returns, such as the Sharpe ratio and Sortino ratio, can be used as rewards to learn behavioral strategies that maximize them. RL has also recently been used to develop trading strategies such as high-frequency trading (HFT), pair trading and portfolio optimization (Chakraborty & Kearns, 2011). Financial strategies using RL focus on achieving high returns and low risk at the same time (Gu et al., 2016). As such, RL technology continues to be applied in the financial domain, leading to the development of a variety of strategies that effectively achieve both return generation and risk management.

2.3. Related works

In the evolving landscape of stock trading and asset allocation, deep learning (DL) techniques are increasingly pivotal in maximizing returns and minimizing risks. This section explores the integration of DL within stock trading strategies, which has led to more sophisticated and predictive models, moving beyond the traditional analytical frameworks.

Deep learning has revolutionized the ability to process and interpret vast amounts of data in the stock market. The use of convolutional neural networks (CNNs) and recurrent neural networks (RNNs) has particularly enhanced the predictive capabilities of trading systems. Lee et al. (2023) effectively employed CNNs combined with natural language processing (NLP) to analyze market sentiment, which significantly improved the accuracy of market trend predictions. Soni et al. (2022) systematically review machine learning approaches, including deep learning models, to predict stock prices. Their research highlights the effectiveness of these methods in understanding short and long-term market trends. Kumbure et al. (2022) provide a literature review on machine learning techniques including deep learning for stock market forecasting, demonstrating how these methods have evolved to handle complex prediction tasks. Nabipour et al. (2020) employed CNNs to analyze financial news, demonstrating significant improvements in market trend predictions. Similarly, Jiang (2021) reviews the applications of deep learning in stock market prediction, highlighting the effectiveness of various deep learning architectures under different market conditions. The incorporation of multimodal methods has also been a game changer in the field, allowing for the integration of diverse data types structured and unstructured. Corizzo and Rosen (2024) proposed a framework that harnesses this capability, merging market data with news articles and social media content to derive a comprehensive understanding of market dynamics. Beyond single-model applications, the hybridization of deep learning techniques with other machine learning methodologies has addressed some of the traditional constraints of predictive models. Zou & Herremans (2023) introduced a hybrid model that combines supervised deep learning with dynamic data inputs to refine decision-making processes under various market conditions. Chong et al. (2017) explored various deep learning models and data representations for stock market analysis, providing a comprehensive methodology that enhances predictive capabilities. These developments underscore the capacity of deep learning to integrate multimodal data sources such as market data, news and financial reports into robust analytical models.

In the field of stock trading and asset allocation, RL algorithms are used to maximize returns while minimizing risk in the market environment. In the field of stock trading and asset allocation, RL algorithms can be broadly categorized into stock trading strategies and RL and portfolio optimization and asset allocation.

Stock trading strategies and RL covers research related to stock trading, mainly focusing on how to develop effective trading strategies using RL techniques. These studies explore strategies to maximize profits and minimize risks in the stock market using various RL models and technical indicators. They provide new insights and strategies for stock trading and show superior performance compared to traditional models. The papers in this group focus on the application of RL techniques to stock trading. Their main research goal is to develop effective trading strategies that maximize profits and minimize risks in the stock market. They mainly address the following features.

Ye and Schuller (2023) propose an algorithm that mimics the behavior of human traders in stock trading. The model combines RL and imitation learning to develop better trading strategies and further align its behavior with human traders. Ye and Schuller (2023) model and the Pro Trader RL framework are similar in that they mimic traders. However, this method relies on a combination of imitation learning and, similar to existing research, focuses on learning a single decision-making process for executing buy, sell and hold actions. In contrast, our proposed Pro Trader RL systematically structures the decision-making process in each trading position of a real Pro Trader into distinct, interconnected RL modules. Our framework systematically replicates the entire decision-making process inherent in a pro trader using strategically interconnected RL modules. Our proposed framework provides an improved framework for mimicking the pro trader by creating a framework that organizes and integrates the decision-making process that occurs during each trade into RL modules. However, the disadvantage of this approach is that it requires more specialized knowledge and complexity to modify the proposed framework. Aloud and Alkhamees (2021) propose a stock trading strategy based on Directional Change (DC), which is optimized using Q-learning. The paper demonstrates high returns and improvements in Sharpe ratios, with roaring performance in the stock market. Ma et al. (2021) introduce an algorithm that analyzes current market data and long-term market trends in parallel. Experimental results demonstrate that the algorithm outperforms other state-of-the-art algorithms. Yu et al. (2023) emphasize the importance of designing stock trading agents and introduce a method for dynamically generating trading decisions in multiple stock market environments. In experiments, the method showed high returns and low risk on US, Japanese and UK stocks. Li et al. (2022) propose a method for learning stock trading strategies using deep RL models. Experimental results demonstrate that higher returns can be obtained in the Chinese stock market and the S&P 500 stock market. Wu et al. (2020) use Gated Recurrent Unit (GRU) to analyze stock market data and introduce two trading strategies. Gated Deep Q-learning trading strategy (GDQN) and Gated Deterministic Policy Gradient trading strategy (GDPG). Experimental results show that these strategies have stable performance against stock market volatility. Tan et al. (2011) propose a new artificial intelligence trading system by combining Adaptive Network Fuzzy Inference System (ANFIS) and RL. Initial experimental results show that it outperforms other methodologies and performs well on long periods of stock market data. Théate and Ernst (2021) present the Trading Deep Q-Network (TDQN) algorithm, a novel approach based on Deep Reinforcement Learning (DRL) to determine optimal trading positions in the stock market. TDQN is inspired by the famous Deep Q-Network (DQN) algorithm and modified for specific trading problems in the stock market. A new methodology for evaluating the performance of this algorithm is also proposed and the TDQN algorithm shows promising results. Yang. (2023) proposes an RL-based Task-Context Mutual Actor-Critic (TC-MAC) algorithm for portfolio management, which integrally considers the contextual information of assets and portfolios. Experimental results show that it outperforms existing methodologies.

Portfolio Optimization and Asset Allocation combines modern portfolio theory and RL to provide traders with optimal asset allocation strategies. It explores how to adjust portfolios to account for correlations and risks among assets and how to use RL to learn dynamic asset allocation strategies. They suggest how traders can pursue stable returns

while minimizing risk. This group covers research related to equity portfolio optimization and asset allocation. The research goal of this group is to find ways for traders to effectively diversify their assets and achieve maximum returns while managing risk. It is characterized by the following features.

Zhao et al. (2023) introduce a novel RL approach for modeling complex correlations between assets. It is experimentally validated on several financial datasets and provides an effective solution to the portfolio selection problem. Jang and Seong (2023) propose a novel method to solve the stock portfolio optimization problem by combining Modern portfolio theory (MPT) and RL. It deals with multi-modal problems and dynamically adjusts weights according to market conditions. Song et al. (2023) tackle the portfolio optimization problem using a stochastic RL framework and demonstrate excellent performance during the 2020 stock market crash. Wu et al. (2021) developed a Portfolio Management System (PMS) using RL, convolutional neural network (CNN) and recurrent neural network (RNN). By introducing the Sharpe ratio reward function, the return increased by 39.0 % and the loss decreased by 13.7 %. The PMS showed high profitability in most datasets. Jeong et al. (2023) propose a safety asset allocation reinforcement learning (AARL) framework that combines several protective dynamic asset allocation strategies (PDAS) and demonstrates superior performance in dynamic asset allocation by minimizing risk and providing stable returns.

Each group's research is constantly evolving to bring new perspectives and methods to stock trading and portfolio management and to help traders make better decisions. Each group makes unique contributions to stock market research in different aspects and has a significant impact on the field of RL and portfolio optimization in finance.

3. Proposed framework: Pro Trader RL

This paper proposes a new framework, Pro Trader RL, for generating stock trading knowledge similar to that of professional traders. The framework is implemented by combining RL and stop-loss rules from two distinct learning methodologies. The following Fig. 2 is a flowchart of the Pro Trader RL Framework.

Pro Trader RL consists of four main modules as illustrated in Fig. 3. The Data Preprocessing module generates trading strategy signals and prepares data for the stock market environment to which RL is applied. This includes sub-modules for dataset trading strategy signal generation, variable generation and data normalization. The Knowledge RL module is divided into two parts. The Buy Knowledge RL module identifies and provides stocks with a high probability of trading success. It comprises an RL agent and an RL environment, with sub-modules including the Donchian Channel strategy, Function Train Mode and Test Mode. The Sell Knowledge RL module determines and provides the optimal time to sell each day based on the time of purchase. It also consists of an RL Agent and an RL Environment, with sub-modules for the 120 days after buy Function, Reward Function, Train Mode and Test Mode. The Stop Loss Rule module assists the Sell Knowledge RL in effectively managing the stop loss point. Sub-modules include Stop Loss on Dips and Stop Loss on Sideways.

3.1. Data preprocessing

3.1.1. Datasets

In order to learn when to look at trading similar to traders, we constructed a dataset that is not limited to a specific stock and is not affected by the stock's market capitalization, volatility, price, etc. Our dataset encompasses all stocks listed in the S&P 500 Index, S&P MidCap 400 Index and S&P SmallCap 600 Index, representing a diverse spectrum from large to small market capitalizations. Data were collected daily from Yahoo Finance and any stocks that did not have at least two years of data for generating technical indicators were excluded. This rigorous selection process resulted in a final sample of 1,465 stocks for our analysis.

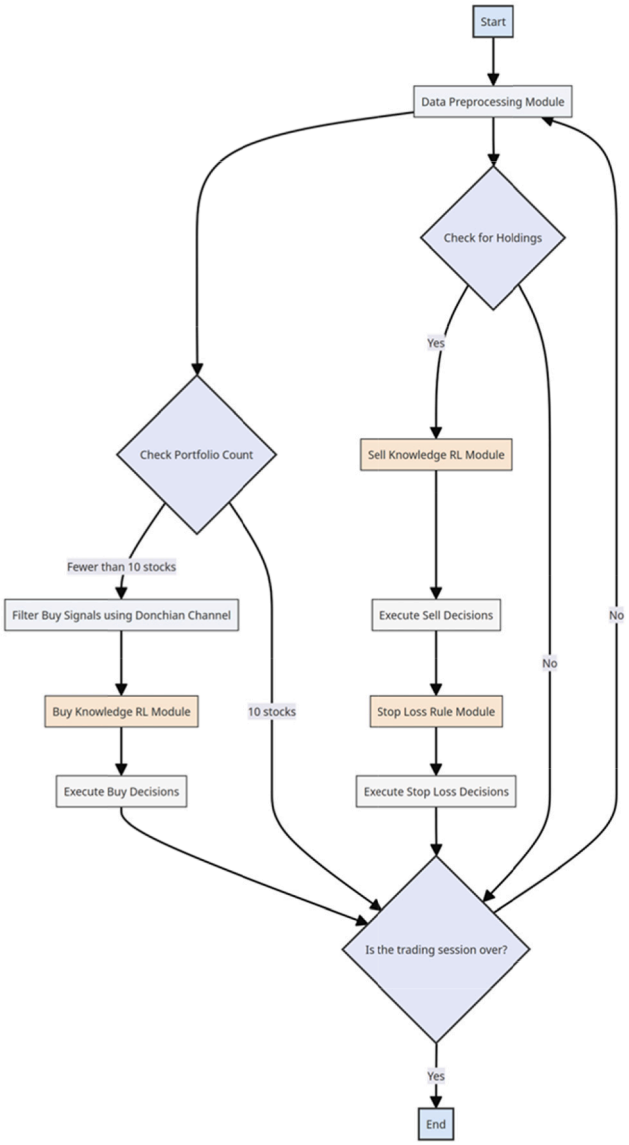


Fig. 2. Flowchart of the Pro Trader RL Framework.

3.1.2. Trading strategy signal generation

Even if an average investor and an expert investor use the same trading strategy, their results may differ significantly due to differences in their experience. In this study, we aim to reduce the difference in experience between investors by learning buying and selling knowledge based on trading strategies. Although many trading strategies exist, the Donchian Channel strategy was chosen for this study due to its simplicity and effectiveness. This strategy generates a single technical indicator to generate buy and sell signals and operates on relatively straightforward principles. The strategy, developed by Richard Donchian, utilizes specific indicators used in market trading (Donchian, 1960). Specifically, it involves buying when the price exceeds the Upper Channel, which is set as the highest price in the last 20 days and selling when the price falls below the Lower Channel, set as the lowest price in the last 20 days. A buy signal is generated when the current high price exceeds the Upper Channel and no buy signal has occurred the previous day. Similarly, a sell signal is generated when the low price falls below the Lower Channel after a buy signal occurs. These trading strategy signals are utilized to construct the dataset and calculate the input variables and rewards in two RL environments. Specifically, only data generating buy signals are

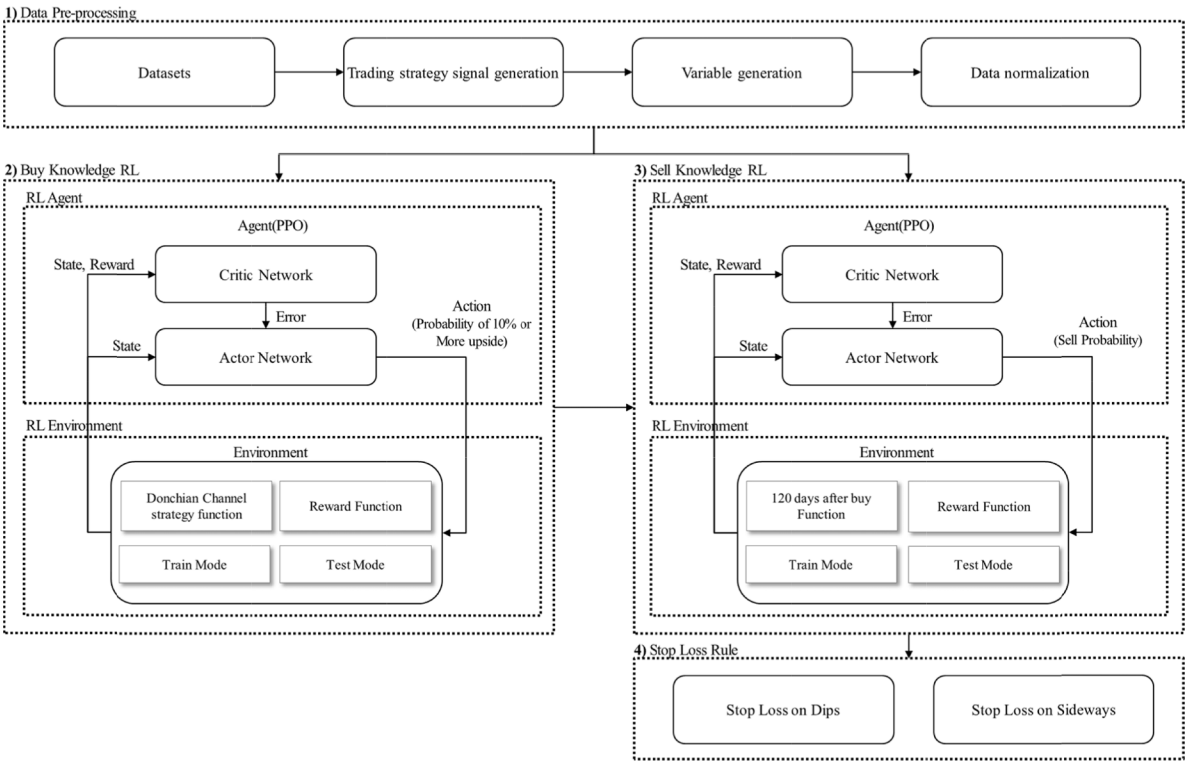


Fig. 3. Pro Trader RL architecture. The proposed framework includes four major modules, data pre-processing, buy knowledge RL, sell knowledge RL and stop loss rules.

Table 1
Default variables.

Basic Variables (9)				
Open	High	Low	Close	Volume
HA Open	HA High	HA Low	HA Close	–.

Table 2
Technical indicator variables.

Technical Indicator Variables (21)				
Return	ATR	Stock 1–12 (12 pieces)	Super Trend 14, 21 (2)	MFI
RSI	Donchian Upper, Lower (2)	AVG Stock	–.	–.

Table 3
Stock Index variables.

Stock Index Variables (13)				
DJI ATR	Indexes 1–12 (12)	–.	–.	–.

Table 4
Stock index versus stock variable.

Stock Index versus Stock Variables. (26)				
RS	Indexes 1–12 (12)	rs avg 2,4,6,8,10,12 (6)	RS Rate	RS Rate 5, 10, 20, 40 (4)
Up Stock	Down Stock	–.	–.	–.

used within the RL environment. This selective usage is crucial for creating an environment where the RL system can learn from both successful and unsuccessful trading outcomes, akin to professional traders. Particularly, data that triggered a buy signal is used as input for the Buy Knowledge RL, which assesses whether the trade was successful and calculates the corresponding rewards to facilitate the learning process. Similarly, Sell Knowledge RL leverages the initial buy signal data to secure 120 days of trading data surrounding each buy signal, using this information to compute rewards based on the outcomes of these longer-term trading decisions.

Thus, the data and rewards that shape the learning environment in each RL module are derived from signals generated by the Donchian Channel strategy. By integrating the Donchian Channel strategy into the RL environment, the system utilizes historical data to improve trading strategies. Additionally, it learns to use strategies similar to those of professional traders.

3.1.3. Variable generation

Tables 1, 2, 3 and 4 detail the list of variables used to configure the RL environment. These variables will be utilized as input variables in the future and are derived from all stock data. The 69 variables are categorized into four categories. Basic variables, technical indicator variables, stock index variables and stock index versus stock variables.

The choice of basic variables such as Open, High, Low, Close and Volume is fundamental to any financial analysis. These variables provide the backbone for most trading strategies and are indispensable for the construction of price and movement charts. The inclusion of Heikin Ashi (HA) candlesticks, which are calculated as averages that smooth out price fluctuations, provides a clearer view of market trends and reduces the noise often seen with traditional candlesticks. This modification is crucial for the RL model to detect and follow trends more accurately.

Technical indicators are vital for developing actionable trading signals. The 21 technical indicators selected are among the most widely used in stock analysis, interpreting price and volume data to predict future market movements. These indicators provide insights into trends, momentum, volatility and market strength, enabling the RL model to make informed predictions and strategic trading decisions.

- Return: This represents the return, which is the difference between the previous day's close and the current close.
- Average True Range (ATR): This indicator measures the volatility of a stock's price, measuring the actual change in price over a period of N days and averaging it to show how risky the current market is. Here, N is set to 10 days.
- Stock (N): This compares the current ATR to the ATR from N months ago. N is set to 1 to 12, or a total of one year.
- Super Trend (N): This is a trend-following indicator based on the ATR, which is used to detect trends and volatility to identify changes in trend direction. The time periods used to calculate Super Trend are 14 and 21 and the multipliers are 2 and 1.
- Money Flow Index (MFI): This indicator is utilized to measure the relative strength of buying and selling and the variable Time period is set to 14.
- Relative Strength Index (RSI): The RSI is an indicator that helps to predict when a stock trend is about to turn by showing the strength of the current trend as a percentage. It typically uses the sum of the gains and losses over a 14-day period to show relative strength and uses volume instead of stock price.
- Donchian Channel: This is a channel indicator that uses the maximum high value over a period of time N as the Upper Channel and the minimum low value as the Lower Channel. N is set to 20 days.
- Average (AVG) Stock: This is the average value of Stock(N), meaning the average value of 1 month, 3 months, 6 months and 12 months. It is utilized as an indicator to understand the volatility of a stock over the short, medium and long term over the course of a year.

Stock Index Variables are calculated using data from the globally recognized Dow Jones Index, chosen for its broad representation of the U.S. stock market. The inclusion of stock index variables allows the model to consider broader economic indicators and market sentiments, which influence individual stock movements and overall market trends.

- Dow Jones Index Average True Range (DJI ATR): This is an ATR indicator to measure the volatility of the Dow Jones Index. In this case, 10 days is used as the N value.
- Index (N): This indicator is used to compare the current DJI ATR value with the DJI ATR value N months ago. Where N ranges from 1 to 12, which means a total of one year.

The stock index versus stock variables that compare stocks to the index or to each other are crucial for assessing relative performance. These comparisons help identify stocks that are outperforming or underperforming the market or their peers, which is valuable for portfolio management and risk assessment.

- Relative Strength (RS): RS is used to identify how much a stock outperforms the overall market or a specific benchmark and to compare it to other stocks. It is measured as the ratio of Stock (N), calculated for a stock, to Index (N), calculated for a stock index, where N is set from 1 to 12, utilizing a total of one year of data.
- RS Average (AVG): RS AVG represents the average value of N RSs and is a measure of a stock's volatility by comparing its short-term, medium-term and long-term volatility to the stock index. Where N is a multiple of 2, it can be set up to 12 to analyze a total of one year's worth of data.
- RS Rate: The RS Rate is used to compare RS to other stocks by converting it to a number between 0 and 100.
- RS Rate(N): RS Rate(N) represents the moving average of the RS Rate over N periods and is used for comparison between stocks. N is set to 5, 10, 20, or 40.
- Up Stock: Up Stock indicates the number of times each stock's Return value is positive and is used to determine whether the overall stock market is in a bull market or not.
- Down Stock: Conversely, Down Stock indicates the number of stocks with a negative Return value and is used to determine if the overall stock market is in a bear market.

A total of 69 variables are initially sent to the two RL environments and the data normalization module in an unnormalized state. These variables play an important role in calculate reward values in the Buy Knowledge RL environment and the Sell Knowledge RL environment and are also used as key inputs in the data normalization module.

The variables affect reward and learning in each RL environment, helping the RL algorithm effectively learn Pro Trader's buying and selling knowledge.

3.1.4. Data normalization

Data normalization is essential in processing data involving a diverse range of instruments and variables into a uniform range of standardized values. This standardization enables effective comparisons among multidimensional data, which significantly improves the learning stability and convergence speed of the model. Although general methods like MinMax Scaler, Standard Scaler, MaxAbs Scaler and Robust Scaler are commonly employed, they are not optimal for stock data due to their limitation of only normalizing data within a specific range. Stock data often exhibits extreme fluctuations; for example, a stock price might rise from $1 to $1000 or fall to $0.1.

In our research, we apply specific normalization formulas tailored for stock data to ensure consistency across different time periods. These formulas maintain data values within a defined range, taking into account the significant variations common in stock data. The methods defined in Eqs. (1)–(8) normalize data by exploiting differences in other variables.

$$DonchianUpper_{new} = \frac{DonchianUpper}{High} \tag{1}$$

$$DonchianLower_{new} = \frac{DonchianLower}{Low} \tag{2}$$

$$Close_{new} = \frac{DonchianUpper}{High} \tag{3}$$

$$Low_{new} = \frac{DonchianUpper}{High} \tag{4}$$

$$High_{new} = \frac{DonchianUpper}{High} \tag{5}$$

$$HAClose_{new} = \frac{DonchianUpper}{High} \tag{6}$$

$$HALow_{new} = \frac{DonchianUpper}{High} \tag{7}$$

$$HAHigh_{new} = \frac{DonchianUpper}{High} \tag{8}$$

The formulas defined in Eqs. (9)–(10) normalize the data using the difference from the previous day's data. Where t represents the current time and t-1 represents the previous day.

$$DJIATR_{new} = \frac{DJIATR_t}{DJIATR_{t-1}} \tag{9}$$

$$ATR_{new} = \frac{ATR_t}{ATR_{t-1}} \tag{10}$$

The formulas defined in Eqs. (11)–(15) normalize the data by utilizing the minimum and maximum values of each variable.

$$Index_{new} = \frac{Index - Index_{\min(Index1\ 12)}}{Index_{\max(Index1\ 12)} - Index_{\min(Index1\ 12)}} \tag{11}$$

$$Stock_{new} = \frac{Stock - Stock_{\min(Stock1\ 12)}}{Stock_{\max(Stock1\ 12)} - Stock_{\min(Stock1\ 12)}} \tag{12}$$

$$AVGStock_{new} = \frac{AVGStock - Stock_{\min(Stock1\ 12)}}{Stock_{\max(Stock1\ 12)} - Stock_{\min(Stock1\ 12)}} \tag{13}$$

$$RS_{new} = \frac{RS - RS_{\min((RS1\ 12),(RSAVG2,4,6,8,10,12))}}{RS_{\max((RS1\ 12),(RSAVG2,4,6,8,10,12)))} - RS_{\min((RS1\ 12),(RSAVG2,4,6,8,10,12))}} \tag{14}$$

$$RSAVG_{new} = \frac{RSAVG - RS_{\min((RS1\ 12),(RSAVG2,4,6,8,10,12))}}{RS_{\max((RS1\ 12),(RSAVG2,4,6,8,10,12)))} - RS_{\min((RS1\ 12),(RSAVG2,4,6,8,10,12))}} \tag{15}$$

The formulas defined in expressions (16)–(18) normalize variables with percentage values.

$$RSRate_{new} = RSRate \times 0.01 \tag{16}$$

$$MFI_{new} = MFI \times 0.01 \tag{17}$$

$$RSI_{new} = RSI \times 0.01 \tag{18}$$

The Super Trend (14, 21), Return, Up Stock and Down Stock variables do not undergo a separate normalization process because they already contain normalized values. Open and HA Open, which are used for normalization and Volume, which has a complex normalization, are excluded from the RL input variables.

3.2. Buy knowledge RL

3.2.1. RL environment

Professional traders analyze their past trades to learn from both their successes and failures. They apply this knowledge to their future trades. To build an RL environment that mimics this approach, we utilize four functions, the Donchian Channel Strategy Function, Reward Function, Train Mode and Test Mode. The environment accepts and processes inputs in the form of both unnormalized and normalized data and interacts with the RL Agent to communicate the status and rewards of a given action.

The Donchian Channel Strategy Function uses unnormalized data to execute the Donchian Channel Strategy, integrating the generated buy and sell signals. The market price on the following day of these buy and sell signals is then used to calculate the return using the method defined in Eq. (19). This calculated return is utilized by the Reward Function to determine the reward and the normalized data corresponding to the buy signal is provided to the RL Agent as a state.

$$SignalReturn = \frac{SellSignal_{Open(t+1)} - BuySignal_{Open(t+1)}}{SellSignal_{Open(t+1)}} \tag{19}$$

Professional traders build their buying knowledge based on past failed and successful trading. The criteria for distinguishing between failure and success depend on each traders trading philosophy. Most consider trading that have earned more than a certain rate of return to be successful and they analyze the market conditions surrounding them in depth. Based on this knowledge, they select stocks that they believe will rise from among the many buy signals and purchase them. To apply this approach to RL in our study, we defined an Action and Reward Function. We categorized trading with a return of 10 % or more as successful and those with a return of less than 10 % as unsuccessful. We defined actions with a probability of 10 % or more as Action 1 and those with a probability of less than 10 % as Action 2.

The Reward Function uses the state determined by the output of the RL Agent as input to determine the reward. Rewards are defined in four scenarios, depending on how the action corresponds with the actual return, as follows.

- Scenario 1: If the probability of action 1 is high out of two action probabilities and the return is 10 % or more: +1 point.
- Scenario 2: If the probability of action 1 is high out of two action probabilities and the return is 10 % less than: 0 point.
- Scenario 3: If the probability of action 2 is high out of two action probabilities and the return is 10 % less than: +1 point.
- Scenario 4: If the probability of action 2 is high out of two action probabilities and the return is 10 % or more: 0 point.

The RL environment is divided into Train Mode and Test Mode, each of which aims to evaluate the learning and performance of the Agent. In Train Mode, the RL Agent is trained using only the data that generated a buy signal via the Donchian Channel Strategy. Specifically, buy signals with a return of 10 % or more are used in the same proportion as those that do not achieve this return. This approach ensures that the learning is balanced and not skewed in one direction. As a result of the training, it is possible that the ACTION on up to 1,465 stocks at one point in time may be predicted to rise by 10 % or more. To avoid overly optimistic forecasts, Test Mode includes a module to select stocks. In Test Mode, the trained RL Agent is loaded and actual testing is conducted. The input data is the same as in Train Mode, using data containing buy signals derived using the Donchian Channel Strategy. The RL Agent outputs the probability of a return of 10 % or more based on the input state. The output results are combined for stocks with the same date and up to 10 stocks are selected using the stock selection module. Stocks with a higher probability of rising by 10 % or more are prioritized and then the top 10 stocks are selected by sorting them in order of increasing probability.

3.2.2. RL agent

The Buy Knowledge RL's RL Agent embodies the buying knowledge of professional traders who study past trading records and apply this knowledge to future trades. At this time, the agent determines one of two behaviors based on the given state and reward, a probability of 10 % or more and a probability of less than 10 %. There are two actions, one associated with a probability of 10 % or more and one with a probability of less than 10 %. The learning process is based on the proximal policy optimization (PPO) algorithm, which takes the normalized state received from the RL environment as input and outputs two probabilities, each represented as a decimal value between 0 and 1. The structure of the policy network used in the study is based on a Deep Neural Network (DNN) and consists of three hidden layers with 69, 40 and 2 hidden units. The structure is identical for both the Actor Network and the Critic Network.

3.3. Sell Knowledge RL

3.3.1. RL environment

Professional traders decide whether to hold or sell at each point after a purchase. Even if a trading strategy has a clear sell rule, traders may ignore it when presented with data from similar cases in the past. To build an RL environment that mimics this approach, we use four functions, the 120 days after buy Function, the Reward Function, Train Mode and Test Mode. As with Buy Knowledge RL, we take as input both unnormalized and normalized data. The 120 days after buy Function utilizes 120 days of data from the buy signal generated by running the Donchian Channel Strategy on the unnormalized data. In this case, the value of the current state after the buy is added to the state value using the formula defined in Equation (20). The calculated return is then combined with the data normalized to a state value that is comparable to the price at the time of the buy and the current price and provided to the RL Agent as a state input.

$$SellReturn = \frac{Open_{(t)}}{BuySignal_{Open(t+1)}} \tag{20}$$

Professional traders develop their selling knowledge based on their past trading experiences. Unlike building knowledge for buying, when considering selling, traders deem trading that outperform their expectations as successful and analyze the market conditions at that specific time. Based on this knowledge, traders decide whether to sell or hold the stock at each point in time, guided by their trading philosophy, regardless of the sell signals from the prescribed strategy. In this study, we set up an Action and Reward Function to apply this approach to RL. We categorized trading with a return of more than 10 % as successful and those with a return of less than 10 % as unsuccessful. Accordingly, we defined the probability of selling as Action 1 and the probability of holding as Action 2.

The Reward Function determines the reward based on the Action output by the RL Agent, using the state as input. Unlike Buy Knowledge RL, the reward is calculated by incrementing the date by one day over a 120-day period corresponding to a single buy point. Among the states that achieve a return of 10 %, the highest return is awarded +2 points and the lowest return is awarded +1 point. This is calculated using the formula defined in Eq. (21), which determines the relative rewards.

$$= \frac{SellKnowledgeRLReward}{\frac{Rankedindescendingorderofreturnof10\%ormore}{Totalnumberofreturnsof10\%ormore}} + 1 \tag{21}$$

Rewards are defined in four scenarios, depending on how the Action and Return are matched, as follows.

- Scenario 1: If the probability of action 1 is high out of two action probabilities and the return is 10 % or more: RL Reward point.
- Scenario 2: If the probability of action 1 is high out of two action probabilities and the return is 10 % less than: −1 point.
- Scenario 3: If the probability of action 2 is high out of two action probabilities and the return is 10 % less than: +0.5 point.
- Scenario 4: If the probability of action 2 is high out of two action probabilities and the return is 10 % or more: −1 point.

The RL environment is divided into Train Mode and Test Mode, each using different settings for training and testing. Train Mode is the training phase for the RL Agent, utilizing all 120 days of data following a buy signal generated by the Donchian Channel Strategy. Given the additional 120 days of data post-buy signal, we accumulate 120 times the data compared to Buy Knowledge RL. However, we exclude data that does not show a return of at least 10 % as we consider it unnecessary. This exclusion is based on Buy Knowledge RL's focus on outputs expected to increase by more than 10 % during the learning process. The objective is to generate probabilities for selling and holding for 120 days from the buy date. Test Mode, on the other hand, is where the trained RL Agent is tested and includes an additional module for selecting the optimal time to sell. The RL Agent outputs the probability of selling based on the given state over 120 days. Thus, choosing the right time to sell is crucial. The optimal sell time is defined as the moment when the difference between the sell and hold probabilities exceeds 0.85 and the sell probability is greater than the hold probability.

3.3.2. RL agent

Sell Knowledge RL's RL Agent embodies the sell knowledge of professional traders, analyzing whether to sell at each point after a purchase based on past trades and applying this to future trades. The RL Agent continuously interacts with the environment and determines its behavior based on the status and rewards it receives. Its behavior is defined by two types of probabilities, sell probability and hold probability. The agent's learning process is based on the PPO algorithm and utilizes the normalized state received from the RL environment as input. It outputs two probabilities, which are decimal values between 0 and 1. The structure of the policy network used in the study is based on a DNN and consists of three hidden layers with 70, 40 and 2 hidden units. This structure is identical for both the Actor Network and the Critic Network.

3.4. Stop Loss Rules

In addition to simply selling, professional traders manage risk through stop-loss rules. These rules are set according to the investor's philosophy. Most traders decide to initiate a stop-loss when the price of a stock falls below a certain return, or if it does not rise above a certain return over a set period of time. While Sell Knowledge RL possesses the knowledge to sell at the best yield, it lacks an independent risk management function; therefore, this gap is bridged by stop-loss rules. Since stop-losses require consistent rules rather than subjective judgment, we have implemented a rule-based approach.

The Stop Loss Rules module consists of two sub-modules, Stop Loss on Dips and Stop Loss on Sideways. Both of these submodules work in parallel with Sell Knowledge RL to dictate behavior at each point in time and their results override the results of Sell Knowledge RL.

- Stop Loss on Dips: If the return at any point in time falls below 10 %, you will be stopped out at the next day's open price to manage risk.
- Stop Loss on Sideways: If you detect 20 days with a return of 10 % or less in a 120-day trading period, you will be stopped out at the open price on the 21st day.

4. Experiments

To validate the Pro Trader RL framework, we designed three experiments. The first experiment focused on comparing performance against traditional strategies, global indices and machine learning algorithms over the period from October 16, 2017, to October 15, 2023, including bull, sideways and bear markets. The second experiment analyzed the performance of the framework against the global equity market in each market condition during that period. The last experiment evaluated the performance of the framework against the latest RL-based stock trading research.

4.1. Performance evaluation method

To systematically evaluate the performance of the Pro Trader RL framework, we used the following key metrics, annual and cumulative returns, Sharpe ratio and MDD. Annual returns is a metric that shows how much an investor's wealth has effectively grown over the course of a year, representing the average growth rate of assets over a given period. It is a weighted average of asset returns over that period. Cumulative returns reflect the overall return from the start of the trading to the present, giving you a bird's eye view of your overall trading performance. It is calculated by subtracting the initial value from the final value and then dividing the result by the initial value. The Sharpe ratio is a metric that considers both the return and risk of an trading, with a high Sharpe ratio indicating an trading strategy that achieves high returns while having relatively low risk. This allows investors to determine how efficient their trading strategy is. It is calculated by subtracting the annualized risk-free rate from the annualized return and then dividing the result by the annualized volatility. MDD is an indicator of the maximum drawdown experienced over the course of an trading, which is very important in assessing the risk level of an trading strategy. A low MDD value indicates that the trading strategy is stable and has generated steady returns without significant losses. Trading Count indicates the total number of trades executed during the simulation. This metric helps to illustrate the activity level of a trading strategy. Accuracy represents the percentage of trades that were profitable by 10 % or more. This is a crucial metric for understanding the effectiveness of a trading strategy in making successful predictions. Trading Count and accuracy metrics are utilized in the first experiment to compare machine learning and deep learning algorithms.

4.2. Experimental environment

The details of the experimental environment utilized in this study are summarized in Table 5. The RL environment was set up via OpenAI Gym (Vidyadhar et al., 2021) and the RL agent was implemented using PyTorch (Paszke et al., 2019) and Stable-Baselines3 (Raffin et al., 2021). The model training period and number of epochs were determined based on specific criteria. For both Buy Knowledge RL and Sell Knowledge RL, an early stop function was applied to shorten the training time. Buy Knowledge RL took an average of 74.12 s per epoch and a total of 5,967 epochs took about 122 h to train. On the other hand, Sell Knowledge RL took about 152.25 s per epoch, totaling 195 h to train 4,622 epochs.

To ensure that our experiments accurately reflect practical trading conditions, we have incorporated several constraints into our experimental setup. The setup includes a realistic initial budget, a limit on the number of stocks that can be held simultaneously and standard trading fees. These parameters are essential for assessing the performance of the Pro Trader RL framework under realistic financial constraints.

Initial Budget: Each trading simulation starts with an initial budget of $10,000. This budget is utilized to purchase stocks, with the success of the trading strategy measured by the growth of this initial amount through strategic trading decisions.

Maximum Shares: To mimic common restrictions faced by individual investors, we have imposed a maximum limit of 10 stocks that can be held in the portfolio at any given time. Additionally, the amount invested in any one stock is capped at 10 % of the total budget. This constraint challenges the RL agent to optimize its selections and prioritize trades that potentially offer the highest returns.

Trading Fees: A trading fee of 0.1 % is applied to each transaction, affecting both purchases and sales. This fee is subtracted from the budget, adding a layer of complexity to the trading strategy as it directly impacts the net profit or loss from each trade.

These constraints are integrated into the RL environment to provide a more accurate assessment of how the Pro Trader RL framework might perform in real-world trading scenarios. The inclusion of these practical constraints is designed to enhance the robustness and applicability of our trading strategies, ensuring that the system can operate effectively under common market limitations.

To evaluate each configuration of hyperparameters, the network must be fully trained and its performance tested, a process that is notably time-consuming. Furthermore, the number of hyperparameters to be tuned is considerable, and they are interconnected in complex ways. Altering one hyperparameter may cause others to become suboptimal. Consequently, conducting a systematic grid search to identify the optimal set of hyperparameters was not feasible with the resources available. Therefore, we employed default hyperparameter values in our study. The reinforcement learning hyperparameter values used are presented in Table 6.

4.3. Performance comparison during test periods

To objectively validate the performance of Pro Trader RL, Experiment 1 compared its performance against a traditional investment strategy, a global index and a machine learning algorithm. The assets used consisted of stocks from the S&P 500 Index, S&P MidCap 400 Index and S&P SmallCap 600 Index. The training data covered the period from February 25, 2005 to October 15, 2017, while the test data was set from October 16, 2017 to October 15, 2023. For reference, the traditional strategy used in this study was the Donchian Channel strategy. As for the machine learning algorithms, we selected Decision Tree (Cramer et al., 1976), Random Forest (Breiman, 2001), AdaBoost (Ying et al., 2013) and Gaussian Naïve Bayes (Ontivero-Ortega et al., 2017) as they have excellent performance in classification. For deep learning algorithms, we chose Deep Neural Network (Liu et al., 2017), Long Short-Term Memory (Hochreiter & Schmidhuber, 1997) and Convolutional Neural Network (Gu et al., 2018). In the study, supervised learning was applied to all selected algorithms by labeling data based on whether the return was over 10 % or not. Trading decisions were then made according to the Donchian Channel strategy sell signal. This technique uses price channel breakouts to determine optimal selling points, effectively coupling the predictive power of machine learning models with time-tested technical trading rules. In addition to these machine learning and deep learning approaches, we also integrated three benchmark reinforcement learning algorithms into our study to further enhance the comparison and analysis of trading strategies. These included Proximal Policy Optimization (PPO), Deep Q-Networks (DQN), and Advantage Actor-Critic (A2C). Each of these algorithms employs a distinct reinforcement learning method, yet they share a common operational framework for action decisions typical in trading studies. Distinctively, while these benchmark reinforcement learning algorithms used the same dataset as the other models for consistency in evaluation, they were specifically designed to perform buying, selling, and holding actions in a manner consistent with traditional reinforcement learning trading studies. We compared the performance of all these models to the Dow Jones Index, a global index that includes the dataset and each of the modules proposed in the study. This comprehensive approach allowed us to assess the effectiveness of various trading strategies under different market conditions, providing insights into the adaptability and efficiency of both traditional and advanced trading models. To ensure a thorough evaluation of these strategies, the test data used in our experiments was carefully selected to reflect a diverse range of market conditions. This included periods characterized by two sideways movements, a bear market, and a bull market. All experimental results, capturing the impact of these varied conditions on the performance of each model, are summarized in Table 7. Pro Trader RL significantly outshone all other strategies and systems, recording an annualized return of 65.284 % and a cumulative return of 1936.801 %. It also showcased the highest Sharpe ratio at 4.584 and the lowest MDD at 8.372 %, indicating optimal risk-adjusted performance. The Dow Jones Index had moderate success with an annualized return of 6.687 % and a cumulative return of 47.459 %, reflecting typical market performance. Among traditional strategies, the Donchian Channel had an annualized return of 7.346 % and a cumulative return of 52.858 %, showing marginal improvement over the Dow Jones Index. Gaussian Naive Bayes was the best-performing machine learning algorithm with a commendable annualized return of 31.434 % and a cumulative return of 415.407 % but

Table 5
Experimental Environment for all experiments.

Device	Specifications
OS	Windows 10
CPU	Intel Core i9-9900KF 3.6 GHz
GPU	NVIDIA GeForce RTX 2080Ti × 2
RAM	128 GB
Storage	2 TB SSD
Language	Python 3.6.13, PyTorch = 1.8.1, Stable-Baselines3 = 1.3.0, Tensorflow = 2.1.0, OpenAI gym = 0.19.0

Table 6
Hyperparameters of the PPO algorithm selected for Pro Trader RL.

Hyperparameter	Value	Description
Learning Rate	0.0001	The learning rate
N Steps	2,048	The number of steps to run for each environment per update
Batch size	64	Minibatch size
Entropy coefficient	0.01	Entropy coefficient for the loss calculation
Gamma	0.99	Discount factor
Gae Lambda	0.95	Factor for trade-off of bias vs variance for Generalized Advantage Estimator
Clip Range	0.2	Clipping parameter
Value function coefficient	0.5	Value function coefficient for the loss calculation

Table 7
Performance comparisons during testing periods for Experiment 1.

Categorizing Models	System	Annual Return	Cumulative Returns	Sharpe Ratio	MDD	Accuracy	Trading Count
Index	Dow Jones Index	6.687	47.459	0.464	35.367	–	–
Strategy	Donchian channel	7.346	52.858	0.828	49.344	17.181	518
Supervised Learning (SL)	Decision Tree	2.835	18.143	0.764	48.428	14.919	496
SL	Random Forest	0.798	4.872	0.434	**8.252**	37.5	8
SL	AdaBoost	0.736	4.475	−0.078	17.569	22.727	22
SL	Gaussian Naive Bayes	31.434	415.407	1.526	36.855	21.94	433
SL	Deep Neural Network	3.102	20.102	0.857	12.361	37.5	24
SL	Long Short-Term Memory	−1.262	−7.344	−0.512	20.975	30.769	26
SL	Convolution Neural Network	2.889	18.553	0.668	51.001	16.264	455
Reinforcement Learning (RL)	Proximal Policy Optimization	−12.722	−55.844	0.059	59.667	2.064	4,069
RL	Deep Q-Network	−3.788	−20.76	0.949	59.965	1.541	7,268
RL	Advantage Actor-Critic	1.057	6.404	1.005	47.429	3.289	2,493
RL	Buy Knowledge RL (Ours)	46.861	902.319	2.46	22.926	23.094	446
RL	Sell Knowledge RL (Ours)	11.904	96.171	0.819	43.847	**38.596**	171
Rule	Stop Loss Rule (Ours)	13.757	116.489	2.101	30.36	20.701	314
RL	**Pro Trader RL (Ours)**	**65.284**	**1936.801**	**4.584**	8.372	28.409	616

still lagged significantly behind Pro Trader RL. Deep learning models showed varied results, with the LSTM model performing poorly with a negative annualized return of −1.262 % and a cumulative loss of −7.344 %. The Convolutional Neural Network had a modest performance, similar to traditional algorithms. PPO struggled in the tested market conditions, resulting in an annual return of −12.722 % and a cumulative return of −55.844 %, which highlights the challenges it faces in unstable markets. DQN also experienced difficulties, with an annual return of −3.788 % and a cumulative return of −20.76 %, though its Sharpe ratio of 0.949 indicates some potential in risk management. A2C showed a marginal gain, with an annual return of 1.057 % and a cumulative return of 6.404 %, suggesting a better adaptation to the market dynamics compared to PPO and DQN. Components of the Pro Trader RL framework such as Buy Knowledge RL and Stop Loss Rule also demonstrated strong individual performance, significantly contributing to the framework's overall efficacy. Buy Knowledge RL itself recorded a high annualized return of 46.861 % and a cumulative return of 902.319 %.

This comprehensive analysis underscores the superiority of the Pro Trader RL framework over traditional investment strategies, global indices and both machine learning and deep learning models in terms of return, risk management and stability. The integration of Buy Knowledge RL, Sell Knowledge RL and Stop Loss Rule modules within the Pro Trader RL framework enhances its efficiency and efficacy, making it a robust choice for stock trading across various market conditions.

Fig. 4 shows a graph of the log cumulative return of each strategy over the test period. The reason for the logarithm is that the large difference in returns makes other comparisons difficult to see. Using the Dow Jones Industrial Average, we can see that the market as a whole remained sideways from February 2018 to October 2019, followed by a bear market from January to July 2020 with a 23.201 % decline, which was followed by a bull market. From February 2022 to May 2023, the market went sideways again. Given the volatility of the market, most algorithms followed the downtrend of the market, but Pro Trader RL managed to stay positive even in a downward market. This shows that Pro Trader RL was either less sensitive to market volatility or made more effective trades.

The results show that Pro Trader RL outperforms traditional strategies and other machine learning algorithms on the key metrics of Annual return, Cumulative return, Sharpe ratio and MDD. Pro Trader RL provides higher returns and stability compared to traditional strategies and other algorithms, demonstrating the superiority of the proposed method.

To thoroughly assess the adaptability and effectiveness of the Pro Trader RL framework, we conducted a detailed performance evaluation across various industry sectors. This analysis helps to identify which

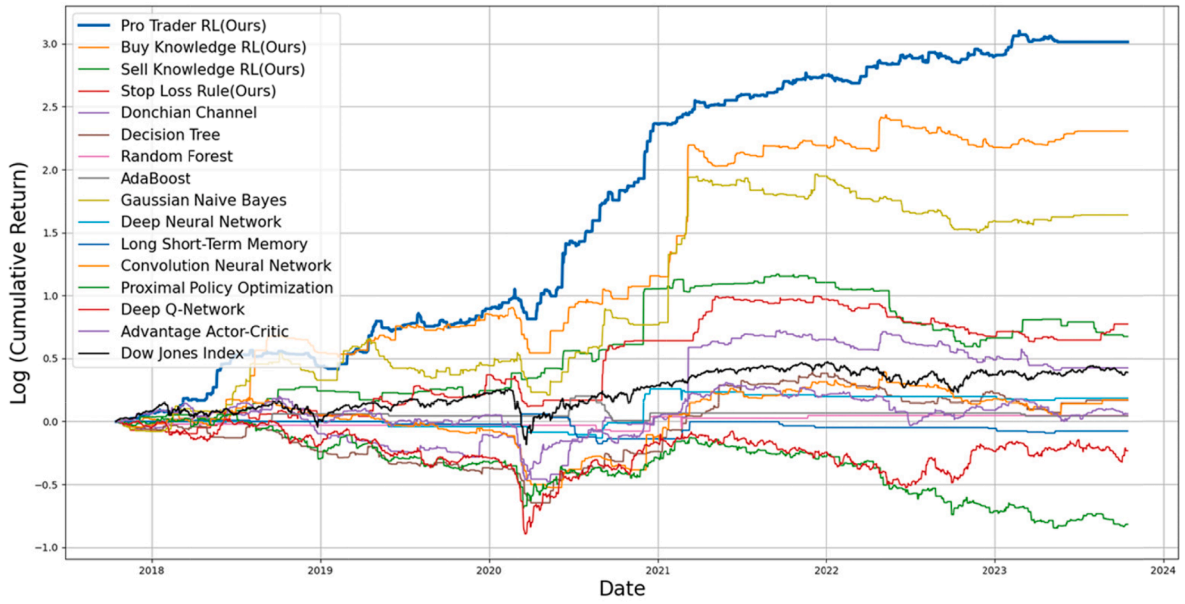


Fig. 4. Cumulative return graph per strategy during the test period for Experiment 1.

sectors the model performs best in and offers insights into the potential reasons for varying performances. The following Table 8 illustrates the trading performance of our strategy across different sectors, detailing the total and average returns, as well as the number of stocks analyzed within each sector.

The Energy sector stands out with the highest average returns of 14.583 %. This may indicate that the Pro Trader RL framework is particularly effective in markets characterized by high volatility and dynamic price movements, which are common in the energy market. Similarly, sectors like Basic Materials and Healthcare also showed robust performance, suggesting that the framework can capitalize on sector-specific trends and volatilities. Sectors such as Technology and Industrials showed good performance with average returns of 8.275 % and 9.086 %, respectively. These sectors often involve a mix of stable and high-growth companies, indicating that our model can effectively navigate different sub-sectors to optimize returns. The Utilities sector, known for its stability and lower volatility, exhibited the lowest average returns of 1.682 %. This suggests that the Pro Trader RL framework may be less effective in sectors where price movements are minimal, thus offering fewer opportunities for substantial gains. Financial Services also showed lower average returns, which could be indicative of the model's challenges in dealing with highly regulated environments or markets with less pronounced trends.

This sector-based performance not only validates the effectiveness of the Pro Trader RL framework in various market conditions but also highlights its potential limitations in less volatile sectors.

4.4. Validate performance in different market conditions

In Experiment 2, we validate the performance of the proposed framework under different market conditions. The tests were conducted in sideways, bear and bull markets. We selected six major indices as benchmarks. Dow Jones Index, Nikkei 225 Index, SSE Index, FTSE 100 Index, DAX Index and CAC 40 Index to evaluate the performance. Each module proposed in the study was also evaluated for comparison.

4.4.1. Validate performance during sideways periods

The market's sideways movement from February 15, 2018, to October 23, 2019, was driven by a combination of factors, including the U.S.-China trade war, the Federal Reserve's interest rate adjustments, a slowing global economy, declining corporate profitability and geopolitical instability. These factors increased market uncertainty and made it difficult to determine a clear direction for the stock market. The results of the experiment can be seen in Table 9. Dow Jones Index showed a moderate performance with an annualized return of 5.103 % and a cumulative return of 9.372 %, along with a Sharpe Ratio of 0.449 and an MDD of 18.543 %. Nikkei 225 Index and DAX Index had lower returns with 3.528 % and 1.831 % annualized returns, respectively, indicating stable but conservative gains. SSE Index experienced a negative performance, declining annually by −4.169 %, showing the challenges in the associated market. FTSE 100 Index and CAC 40 Index demonstrated very modest gains, with the CAC 40 almost matching the Dow Jones in performance metrics. The Donchian Channel strategy underperformed significantly, with an alarming, annualized return of −31.412 % and a cumulative return of −49.323 %, accompanied by the highest MDD in the table at 49.272 %, reflecting its high risk and poor performance in the evaluated period. PPO registered a slight decline with an annual return of −0.909 % and a cumulative return of −1.728 %, but it managed a positive Sharpe Ratio of 0.477, indicating some resilience in risk-adjusted returns. DQN, despite a decline in annual return of −4.015 % and cumulative return of −7.204 %, achieved a relatively high Sharpe Ratio of 0.590, suggesting that while the model underperformed in returns, it maintained a level of risk efficiency. Conversely, A2C demonstrated promising results with an annual return of 3.739 % and a cumulative return of 6.724 %, coupled with a solid Sharpe Ratio of 0.733 and a lower MDD of 12.821, reflecting its effective balance between risk and return. Buy Knowledge RL offered a strong performance with an annualized return of 13.696 % and a cumulative return of 25.866 %, supported by a respectable Sharpe Ratio of 1.07. Sell Knowledge RL was the standout performer with an impressive, annualized return of 35.208 % and a cumulative return of 71.938 %, indicating excellent profitability and risk management, as evidenced by a very low MDD of 4.888 %. Stop Loss Rule showed minimal gains, reflecting its conservative nature designed to protect against losses rather than to generate significant returns. The integrated Pro Trader RL framework outperformed all individual components and indices, achieving the highest annualized return of 46.134 % and a cumulative return of 97.752 %. The framework also exhibited strong risk-adjusted returns with a Sharpe Ratio of 2.38 and an MDD of 13.488 %, showcasing its effectiveness in maximizing returns while controlling for downside risks. The Pro Trader RL framework's superiority over traditional indices and strategies, particularly in terms of its robust annual and cumulative returns and its effective management of market volatility and risk. The individual components of Pro Trader RL Buy Knowledge RL and Sell Knowledge RL also demonstrate significant contributions to the framework's overall success, with Sell Knowledge RL notably excelling in both profitability and risk management. The Pro Trader RL framework offers a comprehensive and reliable approach for navigating diverse market conditions, achieving superior returns and minimizing potential losses, making it a highly effective tool for advanced trading strategies.

Fig. 5 is a graph of the cumulative returns during the first sideways period from March 2018 to November 2019. We can see that Pro Trader RL and Buy Knowledge RL are clearly outperforming, with a significant difference when compared to major global indices. While most indices tend to move up or down over a given period of time, Pro Trader RL has been on a continuous uptrend, with noticeably higher returns.

From February 1, 2022 to May 1, 2023, the stock market traded sideways due to a variety of factors. In early 2023, rising inflationary pressures led investors to anticipate a possible interest rate hike by the Federal Reserve. As a result, many investors began to pull their money out of the stock market, causing the market to decline. In addition, poor earnings from some companies also contributed to the market's decline. The combined effect of these factors led to a sideways movement of the stock market. The results of the experiment can be seen in Table 10. Dow Jones Index experienced a slight decline with an annualized return of −1.276 % and a cumulative return of −1.656 %, highlighting challenges during the sideways market period. Nikkei 225 Index showed strong resilience and growth with an annualized return of 6.756 % and a cumulative return of 8.87 %, outperforming other indices. SSE Index also faced a downturn, similar to Dow Jones, with an annualized return of −2.822 % and a cumulative return of −3.653 %. FTSE 100 Index and CAC 40 Index performed well, achieving annualized returns of 3.745 % and 4.978 % respectively, indicating robustness in less volatile conditions. DAX Index had moderate performance with an annualized return of 1.87 %. The Donchian Channel strategy significantly underperformed with an alarming, annualized return of −19.322 % and a cumulative

Table 8
Sector-specific trading performance of the Pro Trader RL framework.

Sector	Total Return	Average Return	Count
Basic Materials	358.503	10.864	33
Communication Services	59.752	2.490	24
Consumer Cyclical	242.500	4.491	54
Consumer Defensive	136.162	3.095	44
Energy	**977.042**	**14.583**	67
Financial Services	132.028	1.760	75
Healthcare	709.874	7.170	99
Industrials	545.155	9.086	60
Delisted (None)	58.375	4.490	13
Real Estate	171.792	2.490	69
Technology	537.855	8.275	65
Utilities	21.868	1.682	13

Table 9
Validation of performance during the first sideways period of Experiment 2.

Categorizing Models	System	Annual Return	Cumulative Returns	Sharpe Ratio	MDD
Index	Dow Jones Index	5.103	9.372	0.449	18.543
Index	Nikkei 225 Index	3.528	6.44	0.323	20.814
Index	SSE Index	−4.169	−7.38	−0.144	26.517
Index	FTSE 100 Index	0.344	0.621	0.091	16.411
Index	DAX Index	1.831	3.319	0.204	20.449
Index	CAC 40 Index	5.085	9.338	0.455	17.682
Strategy	Donchian channel	−31.412	−49.323	−3.859	49.272
RL	Proximal Policy Optimization	−0.909	−1.728	0.477	22.722
RL	Deep Q-Network	−4.015	−7.204	0.59	20.183
RL	Advantage Actor-Critic	3.739	6.724	0.733	12.821
RL	Buy Knowledge RL (Ours)	13.696	25.866	1.07	16.173
RL	Sell Knowledge RL (Ours)	35.208	71.938	**3.115**	**4.888**
Rule	Stop Loss Rule (Ours)	0.179	0.222	0.223	27.649
RL	**Pro Trader RL (Ours)**	**46.134**	**97.752**	2.38	13.488

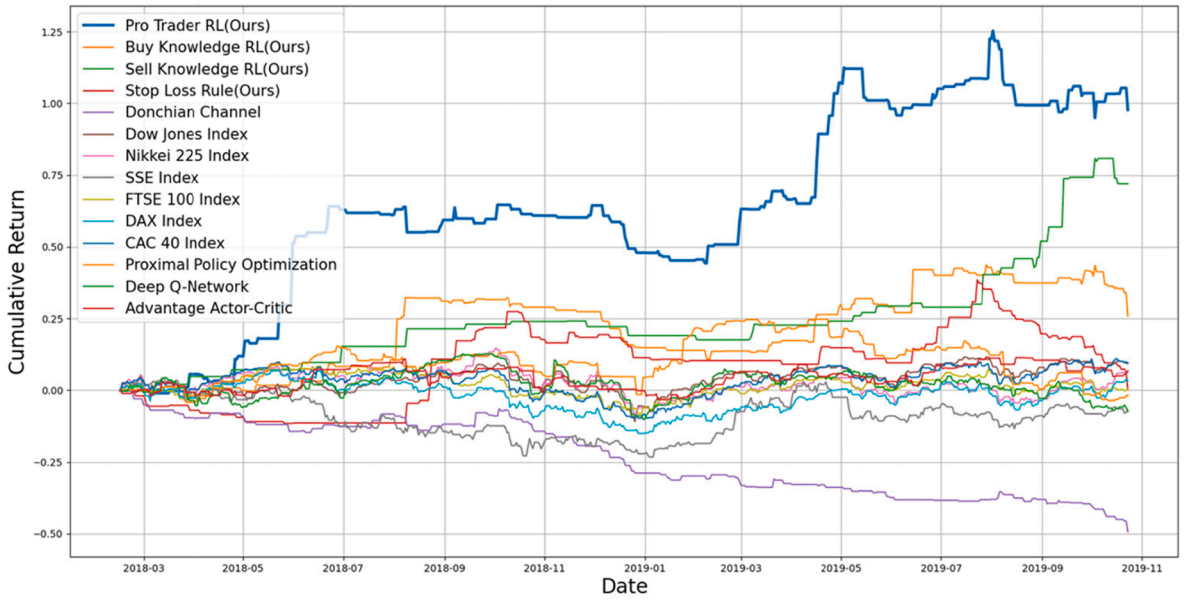


Fig. 5. Cumulative return graph for the first sideways period in Experiment 2.

Table 10
Verify performance during the second sideways period of Experiment 2.

Categorizing Models	System	Annual Return	Cumulative Returns	Sharpe Ratio	MDD
Index	Dow Jones Index	−1.276	−1.656	0.01	19.295
Index	Nikkei 225 Index	6.756	8.87	0.51	11.533
Index	SSE Index	−2.822	−3.653	−0.117	17.995
Index	FTSE 100 Index	3.745	4.895	0.326	**11.029**
Index	DAX Index	1.87	2.438	0.198	23.821
Index	CAC 40 Index	4.978	6.519	0.341	20.291
Strategy	Donchian channel	−19.322	−24.43	−1.712	25.307
RL	Proximal Policy Optimization	−23.025	−28.851	−0.422	34.601
RL	Deep Q-Network	−11.448	−14.705	0.235	26.8
RL	Advantage Actor-Critic	−9.863	−12.715	−0.141	22.051
RL	Buy Knowledge RL (Ours)	−11.758	−15.093	−0.924	22.194
RL	Sell Knowledge RL (Ours)	5.701	7.366	**0.742**	**8.556**
Rule	Stop Loss Rule (Ours)	−23.412	−29.373	−2.814	29.957
RL	**Pro Trader RL (Ours)**	**6.689**	**8.673**	0.576	14.916

return of −24.43 %, indicating its ineffectiveness during the sideways market period. PPO struggled considerably in the sideways market, resulting in an annual return of −23.025 % and a cumulative return of −28.851 %, with a negative Sharpe Ratio of −0.422, indicating a significant risk relative to the returns. Similarly, DQN faced challenges but performed slightly better than PPO with an annual return of −11.448 % and a cumulative return of −14.705 %, while maintaining a positive Sharpe Ratio of 0.235. A2C also encountered difficulties but showed a less drastic decline with an annual return of −9.863 % and a cumulative return of −12.715 %. Buy Knowledge RL encountered significant challenges, posting a negative annualized return of −11.758 % and a cumulative return of −15.093 %. Sell Knowledge RL, however, proved to be highly effective with an annualized return of 5.701 % and a cumulative return of 7.366 %. It also achieved the highest Sharpe Ratio of

0.742 among all systems and the lowest MDD of 8.556 % in this group, showcasing its efficient risk management. Stop Loss Rule showed the most considerable losses in this scenario, with an annualized return of −23.412 % and a cumulative return of −29.373 %, reflecting its high sensitivity to sideways market conditions. Pro Trader RL itself recorded a robust performance, with an annualized return of 6.689 % and a cumulative return of 8.673 %. This indicates its capability to navigate through and adapt effectively in a sideways market, highlighted by a respectable Sharpe Ratio of 0.576 and an MDD of 14.916 %. During sideways periods, Pro Trader RL outperformed the major global stock indices. Despite the underperformance of Buy Knowledge RL and Stop Loss Rules, the Pro Trader RL framework achieved positive results. This is due to the synergistic effect of the integration of these component modules, especially the integration with Sell Knowledge RL, which, as in the previous sideways market, improved the overall performance. The sideways market shows that the Sell Knowledge RL module plays an important role in increasing the profitability and risk management effectiveness of the Pro Trader RL system.

Fig. 6 is a graph of the cumulative returns during the second sideways period from February 2022 to May 2023. You can see that the Pro Trader RL strategy has a clear upward trend during this period, with a particularly large uptick from late 2022 to early 2023. In contrast, the other indices seem to be more volatile, with some indices trending sharply lower.

4.4.2. Validate performance during a bear markets

There are many reasons why the stock market bear from January 1, 2020 to July 1, 2020. Since the beginning of 2020, the economic impact of the COVID-19 pandemic has had a significant impact on the stock market. Investors started pulling their money out of the stock market, which led to a decline in the stock market. In addition, some companies performed worse than expected, which also contributed to the decline in the stock market. These factors combined to cause the stock market to fall. The results of the experiment can be seen in Table 11. Dow Jones Index suffered considerable losses with an annualized return of −17.049 % and a cumulative return of −8.922 %. FTSE 100 Index and CAC 40 Index faced the steepest declines among the indices, with annualized returns of −33.873 % and −31.548 % respectively. SSE Index showed relative resilience, with the smallest decline among the indices at −2.955 % annualized return. The Donchian Channel strategy struggled significantly, with an annualized return of −21.604 %, indicating its vulnerability in rapidly changing market conditions. Similarly, the newly integrated PPO and A2C exhibited severe downturns, with PPO recording an annualized return of −45.437 % and A2C closely matching at −45.441 %. Both strategies demonstrated high sensitivity to the bear market, as reflected in their substantial negative Sharpe Ratios and high MDD. DQN also faced challenges but performed slightly better than PPO and A2C, with an annualized return of −22.643 %. While still negative, DQN's Sharpe Ratio of −0.087 suggests a somewhat more stable performance compared to the other reinforcement learning strategies, though it still incurred a high MDD of 47.371. Buy Knowledge RL, part of the Pro Trader RL framework, also struggled significantly in this environment, posting an annualized return of −25.940 %. Sell Knowledge RL stood out as an exception with a positive annualized return of 13.501 %, demonstrating its capability to perform well even during market downturns. Stop Loss Rule experienced extreme losses, the most substantial among all systems, with an annualized return of −80.673 %, indicating that its settings may not have effectively mitigated the steep market downturn. Despite the adverse market conditions, Pro Trader RL managed an impressive, annualized return of 55.996 % and a cumulative return of 24.773 %. This performance was significantly bolstered by its high Sharpe ratio of 0.992 and a relatively low maximum drawdown (MDD) of 26.327 %, underscoring its robustness and effectiveness in managing risks during volatile periods. During the bear market, Pro Trader RL outperformed other major global equity indices. The Pro Trader RL framework with all modules integrated performs significantly better across all metrics than when each module is used individually. This means that the positive contribution of Sell Knowledge RL is not only beneficial on its own, but also improves the performance of the integrated framework. Integration improves overall performance by mitigating the risks and shortcomings of other modules.

Fig. 7 is a graph of cumulative returns during the bear market period from January 2020 to July 2020. Pro Trader RL has been a standout performer, rising significantly since mid-June, while most other strategies and major indices have seen a general downward trend with significant volatility during this period.

4.4.3. Validate performance during bull markets

The main reasons for the stock market's sustained rise from July 2020 to January 2022 were as follows: The development and large-scale deployment of COVID-19 vaccines raised expectations that economic

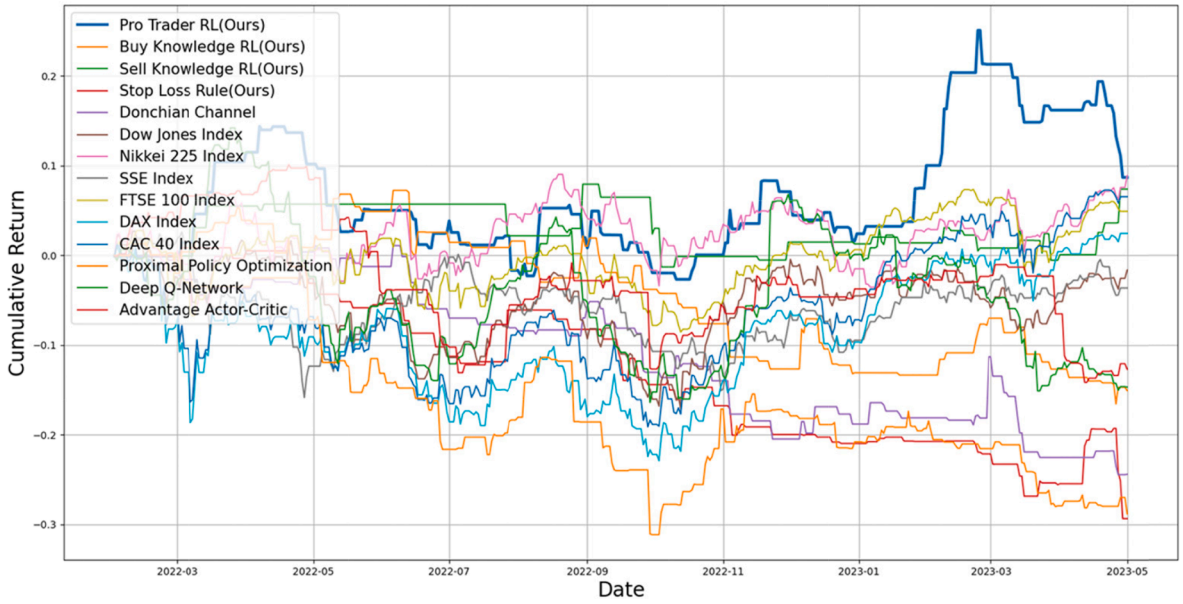


Fig. 6. Cumulative return graph for the second sideways period in Experiment 2.

Table 11
Verifying performance during a bear market in Experiment 2.

Categorizing Models	System	Annual Return	Cumulative Returns	Sharpe Ratio	MDD
Index	Dow Jones Index	−17.049	−8.922	−0.346	35.367
Index	Nikkei 225 Index	−11.69	−6.027	−0.269	31.252
Index	SSE Index	−2.955	−1.489	0.005	14.198
Index	FTSE 100 Index	−33.873	−18.682	−0.94	34.93
Index	DAX Index	−13.211	−6.839	−0.121	38.321
Index	CAC 40 Index	−31.548	−17.264	−0.807	37.201
Strategy	Donchian channel	−21.604	−11.547	−0.658	30.465
RL	Proximal Policy Optimization	−45.437	−26.207	−0.791	38.431
RL	Deep Q-Network	−22.643	−12.135	−0.087	47.371
RL	Advantage Actor-Critic	−45.441	−26.21	−1.109	36.907
RL	Buy Knowledge RL (Ours)	−25.94	−14.028	−0.832	38.313
RL	Sell Knowledge RL (Ours)	13.501	6.43	0.311	**9.868**
Rule	Stop Loss Rule (Ours)	−80.673	−56.082	−8.496	56.038
RL	**Pro Trader RL (Ours)**	**55.996**	**24.773**	**0.992**	26.327

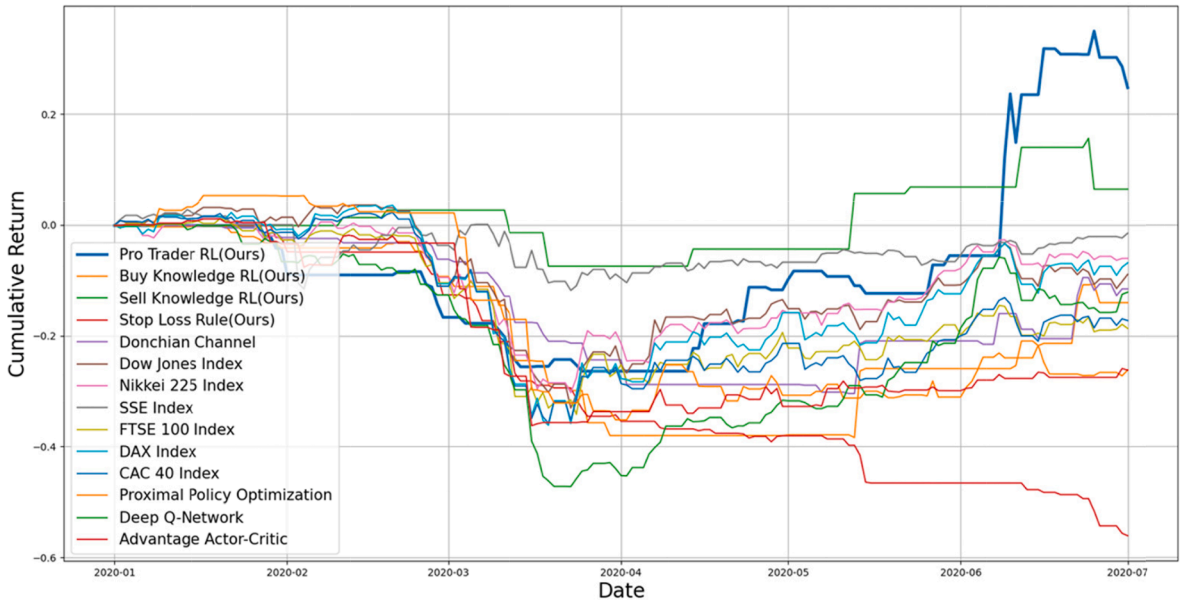


Fig. 7. Graph of cumulative returns during the bear market period in Experiment 2.

activity would resume and normalize. As a result, many countries began to implement various stimulus programs and policies to mitigate the economic impact of the coronavirus. As a result, 2021 saw the majority of companies return to pre-coronavirus levels of earnings and growth. In addition, major central banks continued to provide monetary policy support, such as low interest rate policies and quantitative easing, to boost market liquidity. Finally, technology-related stocks outperformed as demand for remote work and the digitization of learning and entertainment increased. Together, these factors contributed to the bull run in the stock market. The results of the experiment can be seen in Table 12. Dow Jones Index demonstrated robust growth with an annualized return of 26.703 % and a cumulative return of 42.62 %, supported by a Sharpe Ratio of 1.816. CAC 40 Index led the indices with the highest annualized return of 27.671 % and a cumulative return of 44.258 %, indicating strong market participation in Europe. DAX Index and Nikkei 225 Index also performed well, posting annualized returns of 18.433 % and 18.453 %, respectively, showcasing solid growth in both European and Asian markets. FTSE 100 Index had the lowest performance among the major indices but still managed a positive annualized return of 12.238 %. The Donchian Channel strategy had an annualized return of 12.768 %, indicating it was somewhat effective during the market upturn. PPO achieved a moderate performance with an annualized return of 12.608 % and a cumulative return of 19.377 %, along with a Sharpe Ratio of 1.003. This shows that while it did capture some of the upward trend, its potential was not fully realized compared to other strategies. DQN, on the other hand, struggled significantly in this environment, returning a marginal annualized gain of 1.094 % and a cumulative return of 1.544 %, which may indicate its limitations in strongly trending markets, as evidenced by its relatively low Sharpe Ratio and extremely high MDD of 31.444. Conversely, A2C outperformed both PPO and DQN with a robust annualized return of 35.861 % and a cumulative return of 58.201 %, supported by an impressive Sharpe Ratio of 1.913 and a manageable MDD of 14.072. This showcases A2C's superior ability to adapt and capitalize on market conditions. Buy Knowledge RL outshone all other strategies with an exceptional annualized return of 95.246 % and a cumulative return of 172.546 %, reflecting its effectiveness in capturing upward market trends. Sell Knowledge RL also showed outstanding performance with an annualized return of 50.284 % and the highest Sharpe Ratio of all evaluated systems at 4.418, suggesting excellent return for the given risk level. Stop Loss Rule had a moderate performance compared to other Pro Trader RL components, reflecting the less critical role of stop-loss measures in bullish conditions. Pro Trader RL performed excellently with an annualized return of 90.331 % and a cumulative return of 162.32 %. It maintained a high Sharpe Ratio of 3.997 and the lowest MDD among all systems at 5.188 %, underscoring its ability to maximize returns while effectively managing risks during the bull market. During the bull market, Pro Trader RL and Buy Knowledge RL outperformed other major global stock indices and

Table 12
Validate performance during the bull market in Experiment 2.

Categorizing Models	System	Annual Return	Cumulative Returns	Sharpe Ratio	MDD
Index	Dow Jones Index	26.703	42.62	1.816	8.971
Index	Nikkei 225 Index	18.453	28.919	1.183	10.451
Index	SSE Index	14.361	22.297	0.884	9.836
Index	FTSE 100 Index	12.238	18.908	0.835	11.369
Index	DAX Index	18.433	28.887	1.092	14.093
Index	CAC 40 Index	27.671	44.258	1.501	11.899
Strategy	Donchian channel	12.768	19.631	1.282	10.48
RL	Proximal Policy Optimization	12.608	19.377	1.003	8.498
RL	Deep Q-Network	1.094	1.544	0.592	31.444
RL	Advantage Actor-Critic	35.861	58.201	1.913	14.072
RL	Buy Knowledge RL (Ours)	**95.246**	**172.546**	1.798	13.164
RL	Sell Knowledge RL (Ours)	50.284	84.05	**4.418**	**2.261**
Rule	Stop Loss Rule (Ours)	12.777	19.645	0.935	18.933
RL	**Pro Trader RL (Ours)**	90.331	162.32	3.997	5.188

strategies. In a bull market, the Pro Trader RL framework and its individual modules are well suited to capitalize on uptrends while effectively managing risk. Buy Knowledge RL succeeds in capturing uptrends, while Sell Knowledge RL contributes significantly to risk-adjusted performance and loss minimization. By integrating these different strategies, the Pro Trader RL framework provides a balanced approach to maximize profits while controlling risk, proving its robustness in rising market conditions.

Fig. 8 is a graph of cumulative returns during the bull market period from July 2020 to January 2022. Pro Trader RL had the most noticeable upward trend during this period, followed by Buy Knowledge RL. Notably, both RLs made a big jump with a sharp uptick in the early days, but Pro Trader RL maintained a steady uptrend afterward. On the other hand, other global indices saw relatively modest gains during this period.

Overall, Pro Trader RL is an algorithm that is based on the investment philosophy and risk management strategy of professional traders and it has shown stable and excellent performance in a variety of market conditions. This indicates that Pro Trader RL can provide a similarly effective investment methodology to professional traders in any market condition. In this study, Buy Knowledge RL offers more upside in terms of returns and is particularly suited to bull markets. However, Sell Knowledge RL plays a key role in improving risk-adjusted returns and minimizing potential losses and contributes significantly to the overall effectiveness of Pro Trader RL. By combining Buy Knowledge RL's ability to identify rising stocks with Sell Knowledge RL's efficient and conservative strategies and stop-loss rules, Pro Trader RL exhibits a positive synergy in maximizing returns while effectively managing risk.

4.5. Comparison of the latest research

In Experiment 3, we compare the performance of our proposed framework with the state-of-the-art research on trading with RL. Table 13 compares the performance of the proposed framework with the same test period as that in the paper. The experiments were conducted in different market environments and assets, but we assume that this is irrelevant since our goal is to compare investment performance over the same time period. Wu et al. (2021) introduced a portfolio management system (PMS) employing two neural network architectures a Convolutional Neural Network (CNN) and a Recurrent Neural Network (RNN) to manage portfolios using the Sharpe ratio as a performance metric to mitigate risk. This system was evaluated over approximately two years. PMS_CNN reported a cumulative return of 103.041 %, showcasing significant effectiveness. PMS_RNN, however, yielded a lower cumulative return of 37.735 %. Compared to these, Pro Trader RL achieved a superior cumulative return of 127.609 %, with a much higher Sharpe ratio of 3.597, demonstrating more stable and higher returns.

Théate and Ernst (2021) present an innovative approach based on deep reinforcement learnig (DRL) to solve the algorithmic trading problem of when to determine the optimal trading position during trading activity in the stock market. Denominated the Trading Deep Q-Network algorithm (TDQN), this new DRL approach is inspired by the

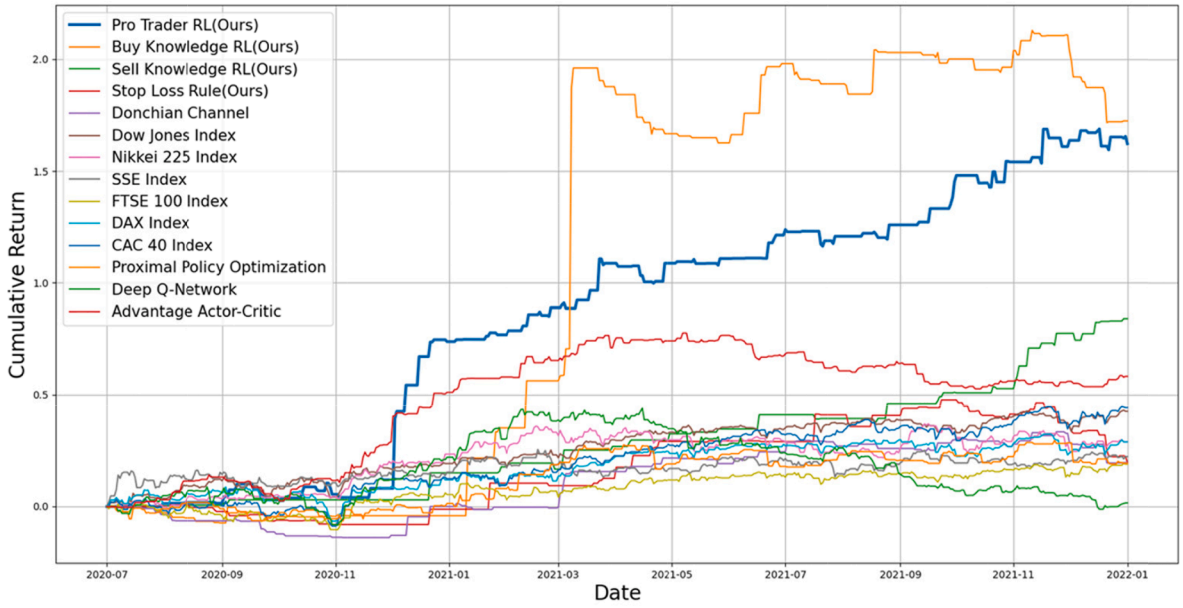


Fig. 8. Graph of cumulative returns during the bull market period in Experiment 2.

Table 13
Performance indicators for state-of-the-art frameworks and the proposed framework in Experiment 3.

System	Annual Return	Cumulative Returns	Sharpe Ratio	MDD
Test period: 2017-08-01 ~ 2019-07-31				
PMS_CNN (Wu 2021)	48.888	103.041	0.578	17.843
PMS_RNN (Wu 2021)	17.903	37.735	0.431	**9.028**
Buy Knowledge RL (Ours)	41.215	99.216	1.893	15.875
Sell Knowledge RL (Ours)	24.866	55.759	2.793	9.753
Stop Loss Rule (Ours)	5.915	12.067	0.725	12.952
Pro Trader RL (Ours)	**50.943**	**127.609**	**3.597**	14.17
Test period: 2018-01-01 ~ 2019-12-31				
TDQN (Théate & Ernst, 2021)	32.81	100.288	1.484	17.31
Buy Knowledge RL (Ours)	33.184	77.202	1.705	15.484
Sell Knowledge RL (Ours)	34.514	80.758	2.612	**11.718**
Stop Loss Rule (Ours)	15.307	32.825	0.694	25.554
Pro Trader RL (Ours)	**42.746**	**103.561**	**2.703**	12.191
Test period: 2018-01-01 ~ 2019-12-31				
TC-MAC (Yang, 2023)	23.2	51.688	2.578	21.9
Buy Knowledge RL (Ours)	33.184	77.202	1.705	15.484
Sell Knowledge RL (Ours)	34.514	80.758	2.612	**11.718**
Stop Loss Rule (Ours)	15.307	32.825	0.694	25.554
Pro Trader RL (Ours)	**42.746**	**103.561**	**2.703**	12.191
Test period: 2018-07-01 ~ 2022-03-31				
Safety AARL (Jeong 2023)	10.259	44.229	1.35	**3.828**
Buy Knowledge RL (Ours)	36.752	218.322	1.487	27.945
Sell Knowledge RL (Ours)	33.088	187.656	2.968	8.012
Stop Loss Rule (Ours)	0.469	1.643	0.267	33.234
Pro Trader RL (Ours)	**83.048**	**836.275**	**4.998**	7.753

famous DQN algorithm and significantly adapted to the specific algorithmic trading problem. The comparison is made on about two years of data. TDQN generated a cumulative return of 100.288 % over a similar two-year period. In contrast, Pro Trader RL surpassed TDQN with a cumulative return of 103.561 %, nearly double the Sharpe ratio at 2.703 and a lower MDD, indicating more efficient risk-adjusted performance.

Yang et al. (2023) propose the Task-Context Mutual Actor-Critic (TC-MAC) algorithm to address the problem that state-of-the-art algorithms using RL for portfolio management ignore the overall contextual information of the portfolio. TC-MAC simultaneously encodes the asset characteristics and the overall dynamic context of the portfolio and learns the optimal portfolio policy through a mutual information loss function. The comparison is performed on about two years of data. TDQN generated a cumulative return of 100.288 % over a similar two-year period. In contrast, Pro Trader RL surpassed TDQN with a cumulative return of 103.561 %, nearly double the Sharpe ratio at 2.703 and a lower MDD, indicating more efficient risk-adjusted performance.

Jeong et al. (2023) address dynamic asset allocation, which adjusts asset weights based on performance to reduce risk according to market conditions. They propose a novel framework called Safety asset allocation reinforcement learning (Safety AARL), which combines multiple protective dynamic asset allocation strategies (PDAS) to minimize risk through RL. The framework integrates eight validated PDAS and develops a new RL environment and agent to invest in PDAS. The comparison covers approximately 3 years and 8 months of data. Safety AARL maintained stability with a low risk, yielding a 10 % annualized return and a cumulative return of 44.229 %. Pro Trader RL exhibited higher risk but delivered an exceptionally high cumulative return of 836.275 %, demonstrating its capability to achieve substantial growth over extended periods. The individual contributions from Buy Knowledge RL and Sell Knowledge RL were pivotal, with Sell Knowledge RL notably enhancing the risk-adjusted performance. The Stop Loss Rule, though less profitable, significantly contributed to overall risk management.

In our tests, Pro Trader RL showed high cumulative returns and excellent stability when compared to various state-of-the-art studies. With higher returns and Sharpe ratio, it was evaluated to operate with relatively low risk. These results demonstrate that Pro Trader RL provides excellent performance in stock trading based on RL and indicate the superiority of the proposed RL learning methodology. The superior performance of the Pro Trader RL framework is attributed to a balanced approach that integrates the high return capabilities of the Buy and Sell Knowledge modules with the effective risk control of the Stop Loss rule. This synergy results in outperformance over the latest research in all market conditions.

5. Conclusion and future work

Considering the diverse and complex nature of modern stock markets, this study proposes a novel RL framework, Pro Trader RL, that mimics the decision-making patterns and investment philosophies of professional traders. This framework enables the learning and implementation of efficient investment strategies that closely mimic those of real professional traders. Pro Trader RL consists of four main modules. The data preprocessing module analyzes complex data from the market to create a ready dataset to be applied to RL. It has several sub-modules associated with it, such as investment strategy signal generation, variable generation and data normalization. Then, two RL modules, Buy Knowledge RL and Sell Knowledge RL, take on the role of a professional trader's knowledge to determine when to buy and sell stocks. Each of these two modules contains various sub-modules, reflecting different situations and strategies in the real-world investment environment. Finally, the stop loss rule module plays an important role in managing the risk of the investment, effectively determining the stop loss on dips and stop loss on sideways market. Three experiments were conducted to validate the proposed framework.

Experimental results show that the proposed framework performs similarly to professional traders, minimizing risk and ensuring high returns regardless of market conditions. The results also demonstrate the superiority of the methodology compared to the latest research on reinforcement learning trading. Through our experiments, we found that the outstanding performance of the Pro Trader RL framework can be attributed to its integrated approach, which combines the Buy Knowledge RL and Sell Knowledge RL modules with the Stop Loss Rules module. This integration allows traders to execute trades with precision to increase profit and manage risk to balance profit and risk. In particular, the Sell Knowledge RL module enhances the framework's ability to quickly adapt to market changes, contributing to effective decision-making in all market conditions. In our future research, we plan to enhance the Pro Trader RL framework by developing a more advanced RL agent. This agent will integrate multiple investment strategies to improve decision-making on when to buy and sell and incorporate hyper-parameter optimization to accelerate learning and enhance

overall model performance. Additionally, we will apply various modern models and advanced RL algorithms to further boost the system's effectiveness. This approach will broaden the framework's capabilities, enabling it to more accurately emulate a diverse array of professional trading strategies and adapt more effectively to dynamic market conditions.

CRediT authorship contribution statement

Da Woon Jeong: Conceptualization, Methodology, Software, Validation, Investigation, Writing – original draft, Writing – review & editing. **Yeong Hyeon Gu:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Yeong Hyeon Gu reports financial support was provided by Information & communications Technology Planning & Evaluation (IITP). Da Woon Jeong reports financial support was provided by Information & communications Technology Planning & Evaluation (IITP). If there are other authors, they 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 work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 1711160571, MLOps Platform for Machine learning pipeline automation).

References

Aloud, M. E., & Alkhamees, N. (2021). Intelligent algorithmic trading strategy using reinforcement learning and directional change. *IEEE Access, 9*(11), 4659–114671.
Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PLoS One1, 12*(7), e0180944.
Bertoluzzo, F., & Corazza, M. (2012). Testing different reinforcement learning configurations for financial trading: Introduction and applications. *Procedia Economics and Finance, 3*, 68–77.
Breiman, L. (2001). Random forests. *Machine Learning, 45*, 5–32.
Chakraborty, T., & Kearns, M. (2011, June). Market making and mean reversion. In Proceedings of the 12th ACM conference on electronic commerce (pp. 307-314).
Chong, E., Han, C., & Park, F. C. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications, 83*, 187–205.
Corizzo, R., & Rosen, J. (2024). Stock market prediction with time series data and news headlines: A stacking ensemble approach. *Journal of Intelligent Information Systems, 62*(1), 27–56.
Covel, M. W. (2009). *Trend following: Learn to make millions in up or down markets*. FT Press.
Cramer, G. M., Ford, R. A., & Hall, R. L. (1976). Estimation of toxic hazard-a decision tree approach. *Food and Cosmetics Toxicology, 16*(3), 255–276.
Deng, Y., Bao, F., Kong, Y., Ren, Z., & Dai, Q. (2016). Deep direct reinforcement learning for financial signal representation and trading. *IEEE Transactions on Neural Networks and Learning Systems, 28*(3), 653–664.
Donchian, R. D. (1960). Commodities: High finance in copper. *Financial Analysts Journal, 16*(6), 133–142.
Douglas, M. (2001). *Trading in the zone: Master the market with confidence, discipline, and a winning attitude*. Penguin.
Duan, Y., Chen, X., Houthooft, R., Schulman, J., & Abbeel, P. (2016). Benchmarking deep reinforcement learning for continuous control. In *International conference on machine learning* (pp. 1329–1338). PMLR.
Edwards, R. D., Magee, J., & Bassetti, W. C. (2018). *Technical analysis of stock trends*. CRC Press.
Elder, A. (2002). *Come into my trading room: A complete guide to trading*. John Wiley & Sons.
Graham, B., Dodd, D. L. F., Cottle, S., & Tatham, C. (1962). *Security analysis: Principles and technique*. New York: McGraw-Hill.
Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition, 77*, 354–377.
Gu, S., Lillicrap, T., Ghahramani, Z., Turner, R. E., & Levine, S. (2016). Q-prop: Sample-efficient policy gradient with an off-policy critic. arXiv preprint arXiv:1611.02247.
Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.
Jang, J., & Seong, N. (2023). Deep reinforcement learning for stock portfolio optimization by connecting with modern portfolio theory. *Expert Systems with Applications, 218*, Article 119556.
Jeong, D. W., Yoo, S. J., & Gu, Y. H. (2023). Safety AARL: Weight adjustment for reinforcement-learning-based safety dynamic asset allocation strategies. *Expert Systems with Applications, 227*, Article 120297.
Jiang, W. (2021). Applications of deep learning in stock market prediction: Recent progress. *Expert Systems with Applications, 184*, Article 115537.
Jiang, Z., Xu, D., & Liang, J. (2017). A deep reinforcement learning framework for the financial portfolio management problem. arXiv preprint arXiv:1706.10059.
Kochenderfer, M. J. (2015). *Decision making under uncertainty: Theory and application*. MIT Press.
Kumbure, M. M., Lohrmann, C., Luukka, P., & Porras, J. (2022). Machine learning techniques and data for stock market forecasting: A literature review. *Expert Systems with Applications, 197*, Article 116659.
Lee, T. W., Teisseyre, P., & Lee, J. (2023). Effective exploitation of macroeconomic indicators for stock direction classification using the multimodal fusion transformer. *IEEE Access, 11*, 10275–10287.
Li, Y., Liu, P., & Wang, Z. (2022). *Stock trading strategies based on deep reinforcement learning*. Scientific Programming.
Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., & Alsaadi, F. E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing, 234*, 11–26.
Lo, A. W. (2004). The adaptive markets hypothesis: Market efficiency from an evolutionary perspective. *Journal of Portfolio Management, Forthcoming*.
Ma, C., Zhang, J., Liu, J., Ji, L., & Gao, F. (2021). A parallel multi-module deep reinforcement learning algorithm for stock trading. *Neurocomputing, 449*, 290–302.
Markowitz, H. (1952). Portfolio selection. *The Journal of Finance, 7*(1), 77–91.
Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature, 518*(7540), 529–533.
Moody, J., & Saffell, M. (2001). Learning to trade via direct reinforcement. *IEEE transactions on neural Networks, 12*(4), 875–889.
Murphy, J. J. (1999). *Technical analysis of the financial markets: A comprehensive guide to trading methods and applications*. Penguin.
Nabipour, M., Nayyeri, P., Jabani, H., Mosavi, A., & Salwana, E. (2020). Deep learning for stock market prediction. *Entropy, 22*(8), 840.
Nevmyvaka, Y., Feng, Y., & Kearns, M. (2006, June). Reinforcement learning for optimized trade execution. In Proceedings of the 23rd international conference on machine learning (pp. 673-680).
Ontivero-Ortega, M., Lage-Castellanos, A., Valente, G., Goebel, R., & Valdes-Sosa, M. (2017). Fast Gaussian Naïve Bayes for searchlight classification analysis. *NeuroImage, 163*, 471–479.
Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... & Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. Advances in Neural Information Processing Systems, 32.
Pring, M. J. (2021). Technical analysis explained: The successful investor's to spotting investment trends turning points.
Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., & Dormann, N. (2021). Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research, 22*(268), 1–8.
Schwager, J. D. (2012). *Market wizards, updated: Interviews with top traders*. John Wiley & Sons.
Shleifer, A. (2000). *Inefficient markets: An introduction to behavioral finance*. Oup Oxford.
Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., & Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature, 529*(7587), 484–489.
Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., ... & Hassabis, D. (2017). Mastering chess and shogi by self-play with a general reinforcement learning algorithm. arXiv preprint arXiv:1712.01815.
Song, Z., Wang, Y., Qian, P., Song, S., Coenen, F., Jiang, Z., & Su, J. (2023). From deterministic to stochastic: An interpretable stochastic model-free reinforcement learning framework for portfolio optimization. *Applied Intelligence, 53*(12), 15188–15203.
Soni, P., Tewari, Y., & Krishnan, D. (2022). Machine Learning approaches in stock price prediction: A systematic review. In Journal of Physics: Conference Series (Vol. 2161, No. 1, p. 012065). IOP Publishing.
Steenbarger, B. N. (2015). *Trading Psychology 2.0: From best practices to best processes*. John Wiley & Sons.
Stulz, R. M. (2009). Securities laws, disclosure, and national capital markets in the age of financial globalization. *Journal of Accounting Research, 47*(2), 349–390.
Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.
Taleb, N. N. (2007). *The black swan: The impact of the highly improbable*. Random house.
Tan, Z., Quek, C., & Cheng, P. Y. (2011). Stock trading with cycles: A financial application of ANFIS and reinforcement learning. *Expert Systems with Applications, 38*(5), 4741–4755.
Tharp, V. K., Chabot, C., & Tharp, K. (2007). *Trade your way to financial freedom* (p. 343). New York, NY, USA: McGraw-Hill.

Théate, T., & Ernst, D. (2021). An application of deep reinforcement learning to algorithmic trading. *Expert Systems with Applications, 173*(114632), 4.
Vidyadhar, V., Nagaraj, R., & Ashoka, D. V. (2021). NetAI-Gym: Customized environment for network to evaluate agent algorithm using reinforcement learning in open-AI gym platform. *International Journal of Advanced Computer Science and Applications, 12*(4).
Wu, M. E., Syu, J. H., Lin, J. C. W., & Ho, J. M. (2021). Portfolio management system in equity market neutral using reinforcement learning. *Applied Intelligence*, 1–13.
Wu, X., Chen, H., Wang, J., Troiano, L., Loia, V., & Fujita, H. (2020). Adaptive stock trading strategies with deep reinforcement learning methods. *Information Sciences, 538*, 142–158.
Xiong, Z., Liu, X. Y., Zhong, S., Yang, H., & Walid, A. (2018). Practical deep reinforcement learning approach for stock trading. arXiv preprint arXiv:1811.07522, 1-7.
Yang, S. (2023). Deep reinforcement learning for portfolio management. *Knowledge-Based Systems, 278*, Article 110905.
Ye, Z. J., & Schuller, B. W. (2023). Human-aligned trading by imitative multi-loss reinforcement learning. *Expert Systems with Applications, 234*, Article 120939.
Ying, C., Qi-Guang, M., Jia-Chen, L., & Lin, G. (2013). Advance and prospects of AdaBoost algorithm. *Acta Automatica Sinica, 39*(6), 745–758.
Yu, X., Wu, W., Liao, X., & Han, Y. (2023). Dynamic stock-decision ensemble strategy based on deep reinforcement learning. *Applied Intelligence, 53*(2), 2452–2470.
Zhao, T., Ma, X., Li, X., & Zhang, C. (2023). Asset correlation based deep reinforcement learning for the portfolio selection. *Expert Systems with Applications, 221*, Article 119707.
Zou, Y., & Herremans, D. (2023). PreBit—A multimodal model with Twitter FinBERT embeddings for extreme price movement prediction of Bitcoin. *Expert Systems with Applications, 233*, Article 120838.