AlphaStock: A Buying-Winners-and-Selling-Losers Investment Strategy using Interpretable Deep Reinforcement Attention Networks

Jingyuan Wang[1,4], Yang Zhang[1], Ke Tang[2], Junjie Wu[3,4,∗], Zhang Xiong[1]

1.MOE Engineering Research Center of Advanced Computer Application Technology,
School of Computer Science Engineering, Beihang University, Beijing, China
2.Institute of Economics, School of Social Sciences, Tsinghua University, Beijing China
3.Beijing Key Laboratory of Emergency Support Simulation Technologies for City Operations,
School of Economics and Management, Beihang University, Beijing, China
4.Beijing Advanced Innovation Center for BDBC, Beihang University, Beijing, China. ∗ Corresponding author.

ABSTRACT

Recent years have witnessed the successful marriage of finance innovations and AI techniques in various finance applications including quantitative trading (QT). Despite great research efforts devoted to leveraging deep learning (DL) methods for building better QT strategies, existing studies still face serious challenges especially from the side of finance, such as the balance of risk and return, the resistance to extreme loss, and the interpretability of strategies, which limit the application of DL-based strategies in real-life financial markets. In this work, we propose *AlphaStock*, a novel reinforcement learning (RL) based investment strategy enhanced by interpretable deep attention networks, to address the above challenges. Our main contributions are summarized as follows: *i*) We integrate deep attention networks with a Sharpe ratio-oriented reinforcement learning framework to achieve a risk-return balanced investment strategy; *ii*) We suggest modeling interrelationships among assets to avoid selection bias and develop a cross-asset attention mechanism; *iii*) To our best knowledge, this work is among the first to offer an interpretable investment strategy using deep reinforcement learning models. The experiments on long-periodic U.S. and Chinese markets demonstrate the effectiveness and robustness of AlphaStock over diverse market states. It turns out that AlphaStock tends to select the stocks as winners with high long-term growth, low volatility, high intrinsic value, and being undervalued recently.

CCS CONCEPTS

• **Applied computing** → **Economics**; • **Computing methodologies** → *Reinforcement learning*; *Neural networks*.

KEYWORDS

Investment Strategy, Reinforcement Learning, Deep Learning, Interpretable Prediction

1 INTRODUCTION

Given the ability in handling large scales of transactions and offering rational decision-makings, quantitative trading (QT) strategies have long been adopted in financial institutions and hedge funds and have achieved spectacular successes.Traditional QT strategies are usually based on specific financial logics. For instance, the *momentum* phenomenon found by Jegadeesh and Titman in the stock market [14] was used to build momentum strategies. The *mean reversion* [20] proposed by Poterba and Summers believes that asset price tends to move to the average over time, so the bias of asset prices to their means could be used to select investment targets. The *multi-factor* strategy [7] uses factor-based asset valuations to select assets. Most of these traditional QT strategies, though equipped with solid financial theories, can only leverage some specific characteristic of financial markets, and therefore might be vulnerable to complex markets with diverse states.

In recent years, deep learning (DL) emerges as an effective way to extract multi-aspect characteristics from complex financial signals. Many supervised deep neural networks are proposed in the literature to predict asset prices using various factors, such as frequency of prices [11], economic news [12], social media [27], and financial events [4, 5]. Deep neural networks are also adopted in reinforcement learning (RL) frameworks to enhance traditional shallow investment strategies [3, 6, 16]. Despite the rich studies above, applying DL to real-life financial markets still faces several challenges:

Challenge 1: Balancing return and risk. Most existing supervised deep learning models in finance focus on price prediction without risk awareness, which is not in line with fundamental investment principles and may lead to suboptimal performance [8]. While some RL-based strategies [8, 17] have considered this problem, how to adopt state-of-the-art DL approaches into risk-return-balanced RL frameworks, is yet not well studied.

Challenge 2: Modeling interrelationships among assets. Many financial tools in the market can be used to derive risk-aware profits from the interrelationship among assets, such as hedging, arbitrage, and the BWSL strategy used in this work. However, existing DL/RL-based investment strategies paid little attention to this important information.

Challenge 3: Interpreting investment strategies. There is a long-standing voice arguing that DL-based systems are "unexplainable black boxes" and therefore cannot be used in crucial applications like medicine, investment and military [9]. RL-based strategies with deep structures make it even worse. How to extract interpretable rules from DL-enabled strategies remains an open problem.

In this paper, we propose *AlphaStock*, a novel reinforcement learning based strategy using deep attention networks, to overcome the above challenges. AlphaStock is essentially a *buying winners and selling losers* (BWSL) strategy for stock assets. It consists of three components. The first is a *Long Short-Term Memory with History state Attention* (LSTM-HA) network, which is used to extract asset representations from multiple time series. The second component is a *Cross-Asset Attention Network* (CAAN), which can fully model the interrelationships among assets as well as the asset price rising prior. The third is a portfolio generator, which gives the investment proportion of each asset according to the output winner scores of the attention networks. We use a RL framework to optimize our model towards a return-risk-balanced objective, *i.e.,* maximizing the Sharpe Ratio. In this way, the merit of representation learning via deep attention models and the merit of risk-return balance via Sharpe ratio targeted reinforcement learning are integrated naturally. Moreover, to gain interpretability for AlphaStock, we propose a sensitivity analysis method to unveil how our model selects an asset to invest according to its multi-aspect features.

Extensive experiments on long-periodic U.S. stock markets demonstrate that our AlphaStock strategy outperforms some state-of-the-art competitors in terms of a variety of evaluation measures. In particular, AlphaStock shows excellent adaptability to diverse market states (enabled by RL and Sharpe ratio) and exceptional ability for extreme loss control (enabled by CAAN). Extended experiments on Chinese stock markets further confirm the superiority of Alpha-Stock and its robustness. Interestingly, the interpretation analysis results reveal that AlphaStock selects assets by following a principle as "selecting the stocks as winners with high long-term growth, low volatility, high intrinsic value, and being undervalued recently".

2 PRELIMINARIES

In this section, we first introduce the financial concepts used throughout this paper, and then formally define our problem.

2.1 Basic Financial Concepts

DEFINITION 1 (HOLDING PERIOD). *A holding period is a minimum time unit to invest an asset. We divide the time axis as sequential holding periods with fixed length, such as one day or one month. We call the starting time of the t-th holding period as the time t.*

DEFINITION 2 (SEQUENTIAL INVESTMENT). *A sequential investment is a sequence of holding periods. For the t-th holding period, a strategy uses original capital to invest in assets at time t, and gets profits (could be negative) at time $t + 1$. The capitals plus profits of the*

t-th holding period are used as the original capitals of the $(t + 1)$-th holding period.

DEFINITION 3 (ASSET PRICE). *The price of an asset is defined as a time series $\boldsymbol{p}^{(i)} = \{p_1^{(i)}, p_2^{(i)}, \dots, p_t^{(i)}, \dots\}$, where $p_t^{(i)}$ denotes the price of asset i at time t.*

In this work, we use a stock as an asset to describe our model, which could be extended to other types of assets by taking asset specificities and transaction rules into consideration.

DEFINITION 4 (LONG POSITION). *The long position is the trading operation that buys an asset at time t_1 first and then sells it at t_2. The profit of a long position during the period from t_1 to t_2 for asset i is $u_i(p_{t_2}^{(i)} - p_{t_1}^{(i)})$, where u_i is the buying volume of asset i.*

In the long position, traders expect an asset will rise in price, so they buy the asset first and wait for the price rise to earn profits.

DEFINITION 5 (SHORT POSITION). *A short position is the trading operation that sells an asset at t_1 first and then buys it back at t_2. The profit of a short position during the period from t_1 to t_2 for asset i is $u_i(p_{t_1}^{(i)} - p_{t_2}^{(i)})$, where u_i is the selling volume of asset i.*

Short position is a reverse operation of the long position. Traders' expectation in short position is that the price will drop, so they sell at a price higher than the price at which they buy it back later. In the stock market, a short position trader borrows stocks from a broker and sells them at t_1. At t_2, the trader buys the sold stocks back and returns them to the broker.

DEFINITION 6 (PORTFOLIO). *Given an asset pool with I assets, a portfolio is defined as a vector $\boldsymbol{b} = (b^{(1)}, \dots, b^{(i)}, \dots, b^{(I)})^\top$, where $b^{(i)}$ is the proportion of the investment on asset i, with $\sum_{i=1}^{I} b^{(i)} = 1$.*

Assume we have a collection of portfolios $\{\boldsymbol{b}^{(1)}, \dots, \boldsymbol{b}^{(j)}, \dots, \boldsymbol{b}^{(J)}\}$. The investment on portfolio $\boldsymbol{b}^{(j)}$ is $M^{(j)}$ with $M^{(j)} \geq 0$ when taking a long position on $\boldsymbol{b}^{(j)}$, and $M^{(j)} \leq 0$ when taking a short position. We then have the following important definition.

DEFINITION 7 (ZERO-INVESTMENT PORTFOLIO). *A zero-investment portfolio is a collection of portfolios that has a net total investment of zero when the portfolios are assembled. That is, for a zero-investment portfolio containing J portfolios, the total investment $\sum_{j=1}^{J} M^{(j)} = 0$.*

For instance, an investor may borrow \$1,000 worth of stocks in one set of companies and sell them as a short position, and then use the proceeds of short selling to purchase \$1,000 stocks in another set of companies as a long position. The assemble of the long and short positions is a zero-investment portfolio. Note that while the name is "zero-investment", there still exists a budget constraint to limit the overall worth of stocks that can be borrowed from the broker. Also, we ignore real-world transaction costs for simplicity.

2.2 The BWSL Strategy

In this paper, we adopt the *buy-winners-and-sell-losers* (BWSL) strategy for stock trading [14], the key of which is to buy the assets with high price rising rate (winners) and sell those with low price rising rate (losers). We execute the BWSL strategy as a zero-investment portfolio consisting of two portfolios: a long portfolio for buying winners and a short portfolio for selling losers. Given a

sequential investment with T periods, we denote the short portfolio for the t-th period as $\boldsymbol{b}_t^-$ and the long portfolio as $\boldsymbol{b}_t^+$, $t = 1, \ldots, T$.

At time t, given a budget constraint $\tilde{M}$, we borrow the "loser" stocks from brokers according to the investment proportion in $\boldsymbol{b}_t^-$. The volume of stock i that we can borrow is

$$u_t^{-(i)} = \tilde{M} \cdot b_t^{-(i)} / p_t^{(i)}, \tag{1}$$

where $b_t^{-(i)}$ is the proportion of stock i in $\boldsymbol{b}_t^-$. Next, we sell the "loser" stocks we borrowed and get the money $\tilde{M}$. After that, we use $\tilde{M}$ to buy the "winner" stocks according to the long portfolio $\boldsymbol{b}_t^+$. The volume of stock i that we can buy at time t is

$$u_t^{+(i)} = \tilde{M} \cdot b_t^{+(i)} / p_t^{(i)}. \tag{2}$$

The money $\tilde{M}$ we used to buy winner stocks is the proceeds of short selling, so the net investment on the portfolio $\{\boldsymbol{b}_t^+, \boldsymbol{b}_t^-\}$ is zero.

At the end of the t-th holding period, we sell stocks in the long portfolio. The money we can get is the proceeds of selling stocks using new prices at $t + 1$ for all stocks, i.e.,

$$M_t^+ = \sum_{i=1}^{I} u_t^{+(i)} p_{t+1}^{(i)} = \sum_{i=1}^{I} \tilde{M} \cdot b_t^{+(i)} \frac{p_{t+1}^{(i)}}{p_t^{(i)}}. \tag{3}$$

Next, we buy the stocks in the short portfolio back and return them to the broker. The money we spend on buying the short stocks is

$$M_t^- = \sum_{i=1}^{I'} u_t^{-(i)} p_{t+1}^{(i)} = \sum_{i=1}^{I'} \tilde{M} \cdot b_t^{-(i)} \frac{p_{t+1}^{(i)}}{p_t^{(i)}}. \tag{4}$$

The ensemble profit earned by the long and short portfolios is $M_t = M_t^+ - M_t^-$. Let $z_t^{(i)} = p_{t+1}^{(i)} / p_t^{(i)}$ denote the price rising rate of stock i in the t-th holding period. Then, the rate of return of the ensemble portfolio is calculated as

$$R_t = \frac{M_t}{\tilde{M}} = \sum_{i=1}^{I} b_t^{+(i)} z_t^{(i)} - \sum_{i=1}^{I'} b_t^{-(i)} z_t^{(i)}. \tag{5}$$

Insight I. As shown in Eq. (5), a positive profit, i.e., $R_t > 0$, means the average price rising rate of stocks in the long portfolio is higher than that in the short portfolio, i.e.,

$$\sum_{i=1}^{I} b_t^{+(i)} z_t^{(i)} > \sum_{i=1}^{I'} b_t^{-(i)} z_t^{(i)}. \tag{6}$$

A profitable BWSL strategy must ensure the stocks in the portfolio $\boldsymbol{b}^+$ have a higher average price rising rate than the stocks in $\boldsymbol{b}^-$. That is to say, even the prices of all stocks in the market are falling, as long as we can ensure the price falling of stocks in $\boldsymbol{b}^+$ is slower than that in $\boldsymbol{b}^-$, we can still get profits. On the contrary, even the prices of all stocks are rising, if the rising of stocks in $\boldsymbol{b}^-$ is faster than that in $\boldsymbol{b}^+$, our strategy still lose money. This characteristic implies that the absolute price rising or falling of stocks is not the main concern of our strategy; rather, the relative price relations among stocks are much more important. As a consequence, we must design a mechanism to describe the interrelationships of stock prices in our model for the BWSL strategy.

2.3 Optimization Objective

In order to ensure that our strategy considers both return and risk of an investment, we adopt the *Sharpe ratio*, a risk-adjusted
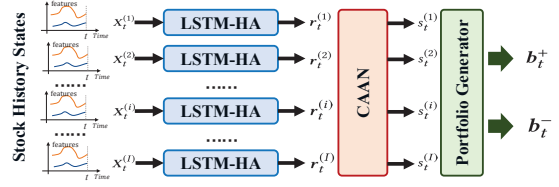

Figure 1: The framework of the AlphaStock model.

return developed by the Nobel laureate William F. Sharpe [21] in 1994, to measure the performance of our strategy.

DEFINITION 8 (SHARPE RATIO). *The Sharpe ratio is the average return in excess of the risk-free return per unit of volatility. Given a sequential investment that contains T holding periods, its Sharpe ratio is calculated as*

$$H_T = \frac{A_T - \Theta}{V_T}, \tag{7}$$

where A_T is the average rate of return per period for the investment, V_T is the volatility that is used to measure risk of the investment, Θ is a risk-free return rate, such as the return rate of bank.

Given a sequential investment with T holding periods, A_T is calculated as

$$A_T = \frac{1}{T} \sum_{t=1}^{T} R_t - TC_t, \tag{8}$$

where TC_t is a transaction cost in the t-th period. The volatility V_T in Eq. (7) is defined as

$$V_T = \sqrt{\frac{\sum_{t=1}^{T} (R_t - \bar{R}_t)^2}{T}}, \tag{9}$$

where $\bar{R}_t = \sum_{t=1}^{T} R_t / T$ is the average of R_t.

For a T-period investment, the optimization objective of our strategy is to generate the long and short portfolio sequences $\boldsymbol{B}^+ = \{\boldsymbol{b}_1^+, \ldots \boldsymbol{b}_T^+\}$ and $\boldsymbol{B}^- = \{\boldsymbol{b}_1^-, \ldots, \boldsymbol{b}_T^-\}$ that can maximize the Sharpe ratio of the investment as

$$\underset{\{\boldsymbol{B}^+, \boldsymbol{B}^-\}}{\arg\max} H_T\left(\boldsymbol{B}^+, \boldsymbol{B}^-\right). \tag{10}$$

Insight II. The Sharpe ratio evaluates the performance of a strategy from both profit and risk perspectives. This profit-risk balance characteristic requires our model not only focuses on maximizing return rate R_t for each period, but also considers the long-term volatility of R_t across all periods in an investment. In other words, designing a far-sighted steady investment strategy is more valuable than a short-sighted strategy with short-term high profits.

3 THE ALPHASTOCK MODEL

In this section, we propose a reinforcement learning (RL) based model called *AlphaStock* to implement a BWSL strategy with the Sharpe ratio defined in Eq. (7) as the optimization objective. As shown in Fig. 1, AlphaStock contains three components. The first component is a LSTM with History state Attention network (LSTM-HA). For each stock i, we use the LSTM-HA model to extract a stock representation $\boldsymbol{r}^{(i)}$ from its history states $\boldsymbol{X}^{(i)}$. The second component is a Cross-Asset Attention Network (CAAN) to describe interrelationships among the stocks. The CAAN takes as input the

representations ($r^{(i)}$) of all stocks, and estimates a winner score $s^{(i)}$ for every stock. The $s^{(i)}$ is a score to indicate the degree of stock i belonging to a winner. The third component is a portfolio generator, which calculates the investment proportions in b^+ and b^- according to the scores ($s^{(i)}$) of all stocks. We use reinforcement learning to end-to-end optimize the three components as a whole, where the Sharpe ratio of a sequential investment is maximized through a far-sighted way.

3.1 Raw Stock Features

The stock features used in our model contains two categories. The first category is the *trading features*, which describes the trading information of a stock. At time t, the trading features include:

• **Price Rising Rate (PR)**: The price rising rate of a stock during the last holding period. It is defined as $\left(p_t^{(i)} / p_{t-1}^{(i)} \right)$ for stock i.

• **Fine-grained Volatility (VOL)**: A holding period can be further divided into many sub-periods. We set one month as a holding period in our experiment, thus a sub-period can be a trading day. VOL is defined as the standard deviation of the prices of all sub-periods from $t - 1$ to t.

• **Trade Volume (TV)**: The total quantity of stocks traded from $t - 1$ to t. It reflects the market activity of a stock.

The second category is the company features, which describe the financial condition of the company that issues a stock. At time t, the company features include:

• **Market Capitalization (MC)**: For stock i, it is defined as the product of the price $p_t^{(i)}$ and the outstanding shares of the stock.

• **Price-earnings Ratio (PE)**: It is the ratio of the market capitalization of a company to its annual earnings.

• **Book-to-market Ratio (BM)**: It is the ratio of the book value of a company to its market value.

• **Dividend (Div)**: It is the reward from company's earnings to stock holders during the $(t - 1)$-th holding period.

Since the values of these features are not in the same scale, we standardize them into Z-scores.

3.2 Stock Representations Extraction

The performance of a stock has close relations with its history states. In the AlphaStock model, we propose a *Long Short-Term Memory with History state Attention (LSTM-HA)* model to learn the representation of a stock from its history features.

The sequential representation. In the LSTM-HA network, we use the vector $\tilde{x}_t$ to denote the history state of a stock at time t, which consists of the stock features given in Section 3.1. We name the last K historical holding periods at time t, *i.e.,* the period from time $t - K$ to time t, as a *look-back window* of t. The history states of a stock in the look-back window are denoted as a sequence $X = \{x_1, \ldots, x_k, \ldots, x_K\}$ [1], where $x_k = \tilde{x}_{t-K+k}$. Our model uses a Long Short-Term Memory (LSTM) network [10] to recursively encode X into a vector as

$$h_k = \text{LSTM}(h_{k-1}, x_k), \ k \in [1, K] \tag{11}$$

---
[1]We also use X to denote the matrix (x_k), the two definitions are interchangeable.

where h_k is the hidden state encoded by LSTM at step k. The h_K at the last step is used as a representation of the stock. It contains the sequential dependence among elements in X.

The history state attention. The h_K can fully exploit the sequential dependence of elements in X, but the global and long-range dependence among X are not effectively modeled. Therefore, we adopt a history state attention to enhance h_K using all middle hidden states h_k. Specifically, following the standard attention [22], the history state attention enhanced representation, denoted as r, is calculated as

$$r = \sum_{k=1}^{K} \text{ATT}(h_K, h_k) h_k, \tag{12}$$

where $\text{ATT}(\cdot, \cdot)$ is an attention function defined as

$$
\begin{aligned}
\text{ATT}(h_K, h_k) &= \frac{\exp(\alpha_k)}{\sum_{k'=1}^{K} \exp(\alpha_{k'})}, \\
\alpha_k &= w^\top \cdot \tanh\left(W^{(1)} h_k + W^{(2)} h_K\right).
\end{aligned}
\tag{13}
$$

Here, w, $W^{(1)}$ and $W^{(2)}$ are the parameters to learn.

For the i-th stock at time t, the history state attention enhanced representation is denoted as $r_t^{(i)}$. It contains both the sequential and global dependences of stock i's history states from time $t - K + 1$ to time t. In our model, the representation vectors for all stocks are extracted by the same LSTM-HA network. The parameters w, $W^{(1)}$, $W^{(2)}$ and those of the LSTM network in Eq. (11) are shared by all stocks. In this way, the representations extracted by LSTM-HA are relatively stable and general for all stocks rather than for a particular one.

Remark. A major advantage of LSTM-HA is that it can learn both the sequential and global dependences from stock history states. Compared with the existing studies that only use a recurrent neural network to extract the sequential dependence in history states [3, 17] or directly stack history states as an input vector of MLP [16] to learn the global dependence, our model describes stock histories more comprehensively. It is worth mentioning that LSTM-HA is also an open framework. The representations learned from other types of information sources, such as news, events and social media [4, 12, 27], could also be concatenated or attended with $r_t^{(i)}$.

3.3 Winners and Losers Selection

In the traditional RL-based strategy models, the investment portfolio is often directly generated from the stock representations through a softmax normalization [3, 6, 16]. The drawback of this type of methods is that it does not fully exploit the interrelationships among stocks, which however is very important for the BWSL strategy as analyzed in *Insight I* of Section 2.2. In light of this, we propose a *Cross-Asset Attention Network* (CAAN) to describe the interrelationships among stocks.

The basic CAAN model. The CAAN model adopts the self-attention mechanism proposed by Ref. [24] to model the interrelationships among stocks. Specifically, given the stock representation $r^{(i)}$ (we omit time t without loss of generality), we calculate a query vector $q^{(i)}$, a key vector $k^{(i)}$ and a value vector $v^{(i)}$ for stock i as

$$q^{(i)} = W^{(Q)} r^{(i)}, \ k^{(i)} = W^{(K)} r^{(i)}, \ v^{(i)} = W^{(V)} r^{(i)}, \tag{14}$$

where $W^{(Q)}$, $W^{(K)}$ and $W^{(V)}$ are the parameters to learn. The interrelationship of stock j to stock i is modeled as using the $q^{(i)}$ of the stock i to query the key $k^{(j)}$ of stock j, i.e.,

$$\beta_{ij} = \frac{q^{(i)\top} \cdot k^{(j)}}{\sqrt{D_k}}, \qquad (15)$$

where D_k is a re-scale parameter setting following Ref. [24]. Then, we use the normalized interrelationships $\{\beta_{ij}\}$ as weights to sum the values $\{v^{(j)}\}$ of other stocks into an attenuation score:

$$a^{(i)} = \sum_{j=1}^{I} \text{SATT}\left(q^{(i)}, k^{(j)}\right) \cdot v^{(j)}, \qquad (16)$$

where the self-attention function $\text{SATT}(\cdot, \cdot)$ is a softmax normalized interrelationships of β_{ij}, i.e.,

$$\text{SATT}\left(q^{(i)}, k^{(j)}\right) = \frac{\exp\left(\beta_{ij}\right)}{\sum_{j'=1}^{I} \exp\left(\beta_{ij'}\right)}. \qquad (17)$$

We use a fully connected layer to transform the attention vector $a^{(i)}$ into a winner score as

$$s^{(i)} = \text{sigmoid}\left(w^{(s)\top} \cdot a^{(i)} + e^{(s)}\right), \qquad (18)$$

where $w^{(s)}$ and $e^{(s)}$ are the connection weights and the bias to learn. The winner score $s_t^{(i)}$ indicates the degree of stock i being a winner in the t-th holding period. A stock with a higher score is more likely to be a winner.

Incorporating price rising rank prior. In the basic CAAN, the interrelationships modeled by Eq. (15) are directly learned from data. In fact, we could use priori knowledge to help our model to learn the stock interrelationships. We use $c_{t-1}^{(i)}$ to denote the rank of price rising rate of stock i in the last holding period (from $t-1$ to t). Inspired by the method for modeling positional information from the NLP field, we use the relative positions of stocks in the coordinate axis of $c_{t-1}^{(i)}$ as a priori knowledge of the stock interrelationships. Specifically, given two stocks i and j, we calculate their discrete relative distance in the coordinate axis of $c_{t-1}^{(i)}$ as

$$d_{ij} = \left\lfloor \left| c_{t-1}^{(i)} - c_{t-1}^{(j)} \right| \Big/ Q \right\rfloor, \qquad (19)$$

where Q is a preset quantization coefficient. We use a lookup matrix $L = (l_1, \ldots, l_L)$ to represent each discretized value of d_{ij}. Using the d_{ij} as the index, the corresponding column vector $l_{d_{ij}}$ is an embedding vector of the relative distance d_{ij}.

For a pair of stocks i and j, we calculate a priori relation coefficient ψ_{ij} using $l_{d_{ij}}$ as

$$\psi_{ij} = \text{sigmoid}\left(w^{(L)\top} l_{d_{ij}}\right), \qquad (20)$$

where $w^{(L)}$ is a learnable parameter. The relationship between i and j estimated by Eq. (15) is rewritten as

$$\beta_{ij} = \frac{\psi_{ij}\left(q^{(i)\top} \cdot k^{(j)}\right)}{\sqrt{D}}. \qquad (21)$$

In this way, the relative positions of stocks in price rising rate rank are introduced as a weight to enhance or weaken the attention coefficient. The stocks have similar history price rising rates will have a stronger interrelationship in the attention and then have similar winner scores.

Remark. As shown in Eq. (16), for each stock i, the winner score $s^{(i)}$ is calculated according to the attention of all other stocks. In this way, the interrelationships among all stocks are involved into CAAN. This special attention mechanism meets the model design requirement of *Insight I* in Section 2.2.

3.4 Portfolios Generator

Given the winner scores $\{s^{(1)}, \ldots, s^{(i)}, \ldots, s^{(I)}\}$ of I stocks, our AlphaStock model generally buys the stocks with high winner scores and sells those with low winner scores. Specifically, we first sort the stocks in descending order by their winner scores and obtain the sequence number $o^{(i)}$ for each stock i. Let G denote the preset size of portfolio b^+ and b^-. If $o^{(i)} \in [1, G]$, stock i will enter the portfolio $b^{+(i)}$, with the investment proportion calculated as

$$b^{+(i)} = \frac{\exp\left(s^{(i)}\right)}{\sum_{o^{(i')} \in [1, G]} \exp\left(s^{(i')}\right)}. \qquad (22)$$

If $o^{(i)} \in (I-G, I]$, stock i will enter $b^{-(i)}$ with a proportion

$$b^{-(i)} = \frac{\exp\left(1 - s^{(i)}\right)}{\sum_{o^{(i')} \in (I-G, I]} \exp\left(1 - s^{(i')}\right)}. \qquad (23)$$

The rest stocks are unselected for the lack of clear buy/sell signals. For simplicity, we can use one vector to record all the information of the two portfolios. That is, we form the vector b^c of length I, with $b^{c(i)} = b^{+(i)}$ if $o^{(i)} \in [1, G]$, or $b^{c(i)} = b^{-(i)}$ if $o^{(i)} \in (I-G, I]$, or 0 otherwise, $i = 1, \ldots, I$. In what follows, we use b^c and $\{b^+, b^-\}$ interchangeably as the return of our AlphaStock model for clarity.

3.5 Optimization via Reinforcement Learning

We frame the AlphaStock strategy into a RL game with discrete agent actions to optimize the model parameters, where a T-period investment is modeled as a state-action-reward trajectory π of a RL agent, i.e., $\pi = \{state_1, action_1, reward_1, \ldots, state_t, action_t, reward_t, \ldots, state_T, action_T, reward_T\}$. The $state_t$ is the history market state observed at t, which is expressed as $\mathcal{X}_t = (X_t^{(i)})$. The $action_t$ is an I-dimensional binary vector, of which the element $action_t^{(i)} = 1$ when the agent invests stock i at t, and 0 otherwise[2]. According to $state_t$, the agent has a probability $\Pr(action_t^{(i)} = 1)$ to invest stock i, which is determined by AlphaStock as

$$\Pr\left(action_t^{(i)} = 1 \Big| \mathcal{X}_t^n, \theta\right) = \frac{1}{2}\mathcal{G}^{(i)}(\mathcal{X}_t^n, \theta) = \frac{1}{2}b_t^{c(i)}, \qquad (24)$$

where $\mathcal{G}^{(i)}(\mathcal{X}_t^n, \theta)$ is part of AlphaStock that generates $b_t^{c(i)}$, θ denotes the model parameters, and $1/2$ is to ensure $\sum_{i=1}^{I} \Pr(action_t^{(i)} = 1) = 1$. Let H_π denote the Sharpe ratio of π, then $reward_t$ is the contribution of $action_t$ to H_π, with $\sum_{t=1}^{T} reward_t = H_\pi$.

For all possible π, the average reward of the RL agent is

$$J(\theta) = \int_\pi H_\pi \Pr(\pi|\theta)\mathrm{d}\pi, \qquad (25)$$

---

[2]In the RL game, the actions of an agent are discrete states with the probability $b_t^{c(i)}/2$ indicating whether to invest stock i. In the real investment, we allocate capitals to stocks i according the continuous proportion $b_t^{c(i)}$. This approximation is for the sake of problem solving.

where $\Pr(\pi|\theta)$ is the probability of generating π from θ. Then, the objective of the RL model optimization is to find the optimal parameters $\theta^* = \arg\max_\theta J(\theta)$.

We use the gradient ascent approach to iteratively optimize θ at round τ as $\theta_\tau = \theta_{\tau-1} + \eta \nabla J(\theta)|_{\theta=\theta_{\tau-1}}$, where η is a learning rate. Given a training dataset that contains N trajectories $\{\pi_1, \ldots, \pi_n, \ldots, \pi_N\}$, $\nabla J(\theta)$ can be approximately calculated as [23]

$$\nabla J(\theta) = \int_\pi H_\pi \Pr(\pi|\theta) \nabla \log \Pr(\pi|\theta) d\pi .$$

$$\approx \frac{1}{N} \sum_{n=1}^{N} \left(H_{\pi_n} \sum_{t=1}^{T_n} \sum_{i=1}^{I} \nabla_\theta \log \Pr\left(action_t^{(i)} = 1 \middle| \mathcal{X}_t^{(n)}, \theta\right) \right), \quad (26)$$

The gradient $\nabla_\theta \log \Pr(action_t^{(i)} = 1|\mathcal{X}_t^{(n)}, \theta) = \nabla_\theta \log \mathcal{G}^{(i)}(\mathcal{X}_t^n, \theta)$, which is calculated by the Back Propagation algorithm.

In order to ensure the proposed model can beat the market, we introduce the threshold method [23] into our reinforcement learning. Then the gradient $\nabla J(\theta)$ in Eq. (26) is rewritten as

$$\nabla J(\theta) = \frac{1}{N} \sum_{n=1}^{N} \left((H_{\pi_n} - H_0) \sum_{t=1}^{T_n} \sum_{i=1}^{I} \nabla_\theta \log \mathcal{G}^{(i)}(\mathcal{X}_t^n, \theta) \right), \quad (27)$$

where the threshold H_0 is set as the Sharpe ratio of the overall market. In this way, the gradient ascent only encourages the parameters that can outperform the market.

Remark. The Eq. (27) uses $(H_{\pi_n} - H_0)$ to integrally weight the the gradients $\nabla_\theta \log \mathcal{G}$ of all holding periods in π_n. The reward is not directly given to any isolated step in π_n but given to all steps in π_n. This feature of our model meets the far-sight requirement of *Insight II* in Section 2.2.

4 MODEL INTERPRETATION

In the AlphaStock model, the LSTM-HA and CAAN networks cast the raw stock features as winner scores. The final investment portfolios are directly generated from the winner scores. A natural follow-up question is: what kind of stocks would be selected as winners by AlphaStock? To answer this question, we propose a sensitivity analysis method [1, 25, 26] to interpret how the history features of a stock influence its winner score in our model.

We use $s = \mathcal{F}(X)$ to express the function of history features X of a stock to its winner score s. In our model, $s = \mathcal{F}(X)$ is a combined network of LSTM-HA and CAAN. We use x_q to denote an element of X which is the value of one feature (defined in Section 3.1) at a particular time period of the look-back window, *e.g.,* the price rising rate of a stock at the time of three months ago.

Given the history state X of a stock, the influence of x_q to its winner score s, *i.e.,* the sensitivity of s to x_q, is expressed as

$$\delta_{x_q}(X) = \lim_{\Delta x_q \to 0} \frac{\mathcal{F}(X) - \mathcal{F}\left(x_q + \Delta x_q, X_{\neg x_q}\right)}{x_q - \left(x_q + \Delta x_q\right)} = \frac{\partial \mathcal{F}(X)}{\partial x_q}, \quad (28)$$

where $X_{\neg x_q}$ denotes the elements of X except x_q.

For all possible stock states in a market, the average influence of the stock state feature x_q to the winner score s is

$$\bar{\delta}_{x_q} = \int_{D_X} \Pr(X) \delta_{x_q}(X) \, d_\sigma . \quad (29)$$

where $\Pr(X)$ is the probability density function of X, and $\int_{D_X} \cdot d_\sigma$ is an integral over all possible value of X. According to the Large

Number Law, given a dataset that contains history states of I stocks in N holding periods, the $\bar{\delta}_{x_q}$ is approximated as

$$\bar{\delta}_{x_q} = \frac{1}{I \times N} \sum_{n=1}^{N} \sum_{i=1}^{I} \delta_{x_q} \left(X_n^{(i)} \middle| \mathcal{X}_n^{(\neg i)} \right), \quad (30)$$

where $X_n^{(i)}$ is the history state of the i-th stock at the n-th holding period, and $\mathcal{X}_n^{(\neg i)}$ denotes the history states of other stocks that are concurrent with the history state of i-th stock.

We use $\bar{\delta}_{x_q}$ to measure the overall influence of a stock feature x_q to the winner score. A positive value of $\bar{\delta}_{x_q}$ indicates that our model tends to take a stock as a winner when x_q is large, and vice versa. For example, in the experiment to follow, we obtain $\bar{\delta} < 0$ for the fine-grained volatility feature, which means that our model trends to select low volatility stocks as winners.

5 EXPERIMENT

In this section, we empirically evaluate our AlphaStock model by the data in the U.S. markets. The data in the Chinese stock markets are also used for robustness check.

5.1 Data and Experimental Setup

The data of U.S. stock market used in our experiments are obtained from Wharton Research Data Services (WRDS) [3]. The time range of the data is from Jan. 1970 to Dec. 2016. This long time range covers several well-known market events, such as the dot-com bubble from 1995 to 2000 and the subprime mortgage crisis from 2007 to 2009, which enables the evaluation over diverse market states. The stocks are from four markets: NYSE, NYSE American, NASDAQ, and NYSE Arca. The number of valid stocks is more than 1000 per year. We use the data from Jan. 1970 to Jan. 1990 as the training and validation set, and the rest as the test set.

In the experiment, the holding period is set to one month, and the number of holding periods T in an investment is set to 12, *i.e.,* the Sharpe ratio reward is calculated every 12 months for RL. The look-back window size K is set to 12, *i.e.,* we look back on the 12-month history states of stocks. The size G of the portfolios is set as 1/4 of number of all stocks.

5.2 Baseline Methods

AlphaStock is compared with a number of baselines including:
- *Market*: the uniform Buy-And-Hold strategy [13];
- Cross Sectional Momentum (*CSM*) [15] and Time Series Momentum (*TSM*) [18]: two classic momentum strategies;
- Robust Median Reversion (*RMR*): a newly reported reversion strategy [13];
- Fuzzy Deep Direct Reinforcement (*FDDR*): a newly reported RL-based BWSL strategy [3];
- AlphaStock-NC (*AS-NC*): the AlphaStock model without the CAAN, where the outputs of LSTM-HA are directly used as the inputs of the portfolio generator.
- AlphaStock-NP (*AS-NP*): the AlphaStock model without price rising rank prior, where we use the basic CAAN in our model.

The baselines TSM/CSM/RMR represent the traditional financial strategies. TSM and CSM are based on the momentum logic and

---

[3] https://wrds-web.wharton.upenn.edu/wrds/

RMR is based on the reversion logic. FDDR represents the state-of-the-art RL-based BWSL strategy. AS-NC and AS-NP are used as a contrast to verify the effectiveness of the CAAN and price rising rank prior. The Market is used to indicate states of the market.

5.3 Evaluation Measures

The most standard evaluation measure for investment strategies is Cumulative Wealth, which is defined as

$$CW_T = \prod_{t=1}^{T} (R_t + 1 - TC), \qquad (31)$$

where R_t is the rate of return defined in Eq. (5) and the transaction cost TC is set to 0.1% in our experiments according to Ref. [3].

The preferences of different investors are varied. Therefore, we also use some other evaluation measures including:

1) *Annualized Percentage Rate (APR)* is an annualized average of return rate. It is defined as $APR_T = A_T \times N_Y$, where N_Y is the number of holding periods in a year.

2) *Annualized Volatility (AVOL)* is an annualized average of volatility. It is defined as $AVOL_T = V_T \times \sqrt{N_Y}$ and is used to measure the average risk of a strategy during an unit time period.

3) *Annualized Sharpe Ratio (ASR)* is the risk-adjusted annualized return based on APR and AVOL. The formalized definition of ASR is $ASR_T = APR_T / AVOL_T$.

4) *Maximum DrawDown (MDD)* is the maximum loss from a peak to a trough of a portfolio, before a new peak is attained. It is the other way to measure the investment risk. The formalized definition of MDD is

$$MDD_T = \max_{\tau \in [1,T]} \left(\max_{t \in [1,\tau]} \left(\frac{APR_t - APR_\tau}{APR_t} \right) \right). \qquad (32)$$

5) *Calmar Ratio (CR)* is the risk-adjusted APR based on Maximum DrawDown. It is calculated as $CR_T = APR_T / MDD_T$.

6) *Downside Deviation Ratio (DDR)* measures the downside risk of a strategy as the average of returns when it falls below a minimum acceptable return (MAR). It is the risk-adjusted APR based on Downside Deviation. The formalized definition of DDR is given as

$$DDR_T = \frac{APR_T}{\text{Downside Deviation}} = \frac{APR_T}{\sqrt{\mathbb{E}[\min(R_t, MAR)]^2}}, \ t \in [1, T]. \qquad (33)$$

In our experiment, the MAR is set to zero.

5.4 Performance in U.S. Markets

Fig. 2 is a cumulative wealth comparison of AlphaStock and the baselines. In general, the performance of AlphaStock (AS) is much better than other baselines, which verifies the effectiveness of our model. Some interesting observations are highlighted as follows:

1) The performance of AlphaStock is better than AlphaStock-NP and the performance of AlphaStock-NP is better than AlphaStock-NC, which indicates that the stock rank priors and interrelationships modeled by CAAN are very helpful for the BWSL strategy.

2) The FDDR is also a kind of deep RL investment strategy, which extracts the fuzzy representations of stocks using a recurrent deep neural network. In our experiment, the performance of AlphaStock-NC is better than FDDR, indicating the advantage of our LSTM-HA network in the stock representation learning.
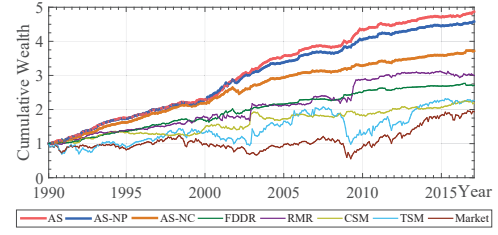

Figure 2: The Cumulative Wealth in U.S. markets.

Table 1: Performance comparison on U.S. markets.

	APR	AVOL	ASR	MDD	CR	DDR
Market	0.042	0.174	0.239	0.569	0.073	0.337
TSM	0.047	0.223	0.210	0.523	0.090	0.318
CSM	0.044	0.096	0.456	0.126	0.350	0.453
RMR	0.074	0.134	0.551	0.098	1.249	0.757
FDDR	0.063	0.056	1.141	0.070	0.900	2.028
AS-NC	0.101	**0.052**	1.929	0.068	1.492	1.685
AS-NP	0.133	0.065	2.054	0.033	3.990	4.618
AS	**0.143**	0.067	**2.132**	**0.027**	**5.296**	**6.397**

3) The TSM strategy performs well in the bull market but very poorly in the bear market (the financial crisis in 2003 and 2008), while the RMR has an opposite performance. This implies the traditional financial strategies can only adapt to a certain type of market state without an effective forward-looking mechanism. This defect is greatly addressed by the RL strategies, including AlphaStock and FDDR, which perform much stably across different market states.

The performances evaluated by other measures are listed in Table 1. For the measures underlined (AVOL, MDD), the lower value indicates the better performance, while the situation is opposite for the other measures. As shown in Table 1, the performances of AlphaStock, AlphaStock-NP and AlphaStock-NC are better than other baselines with all measures, confirming the effectiveness and robustness of our strategy. The performances of AlphaStock, AlphaStock-NP and AlphaStock-NC are close in terms of ASR, which might be due to all of these models are optimized for maximizing the Sharpe ratio. The profits of AlphaStock and AlphaStock-NP measured by APR are higher than that of AlphaStock-NC, at the cost of a little bit higher volatility.

More interestingly, the performance of AlphaStock measured by MDD, CR and DDR is much better than that of AlphaStock-NP. The similar results could be observed by comparing MDD, CR and DDR of AlphaStock-NP and AlphaStock-NC. The three measures are used to indicate the extreme loss in an investment, *i.e.,* the maximum draw down and the returns below the minimum acceptable threshold. The results suggest that the extreme loss control ability of the three models are AlphaStock > AlphaStock-NP > AlphaStock-NC, which highlights the contribution of the CAAN component and the price rising rank prior. Indeed, CAAN with price rising rank priors fully exploits the ranking relationship among stocks. This mechanism can protect our strategy from the error of "buying losers and selling winners", and therefore can greatly avoid extreme losses in investments. In summary, AlphaStock is a very competitive strategy for investors with different types of preferences.
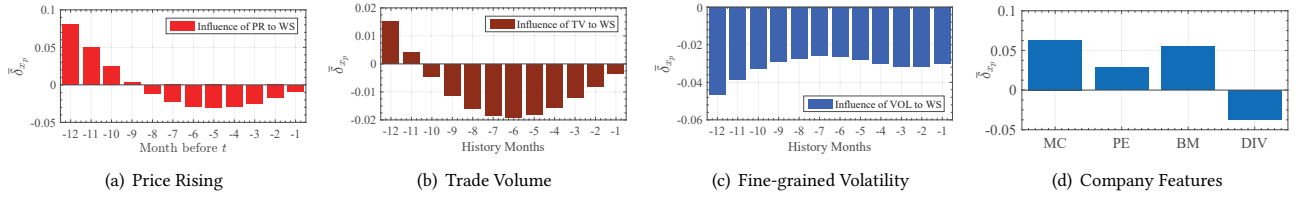
(a) Price Rising

(b) Trade Volume

(c) Fine-grained Volatility

(d) Company Features

Figure 3: Influence of history trading features to winner scores.

Table 2: Performance comparison on Chinese markets.

	APR	AVOL	ASR	MDD	CR	DDR
Market	0.037	0.260	0.141	0.595	0.062	0.135
TSM	0.078	0.420	0.186	0.533	0.147	0.225
CSM	0.023	0.392	0.058	0.633	0.036	0.064
RMR	0.079	0.279	0.282	0.423	0.186	0.289
FDDR	0.084	0.152	0.553	0.231	0.365	0.801
AS-NC	0.104	0.113	0.916	0.163	0.648	1.103
AS-NP	0.122	0.105	1.163	0.136	0.895	1.547
AS	**0.125**	**0.103**	**1.220**	**0.135**	**0.296**	**1.704**

5.5 Performance in Chinese Markets

In order to further testify the robustness of our model, we run the back-test experiments of our model and baselines over the Chinese stock markets, which contains two exchanges: Shanghai Stock Exchange (SSE) and Shenzhen Stock Exchange (SZSE). The data are obtained from the WIND database[4]. The stocks are the RMB priced ordinary shares (A-share) and the total number of stocks used for experiment is 1,131. The time range of our data is from Jun. 2005 to Dec. 2018, with the period from Jun. 2005 – Dec. 2011 used as the training/validation set and the rest as the test set. Since the Chinese markets cannot short sell, so we only use the b^+ portfolio in the experiment.

The experimental results are given in Table 2. From the table we can see that the performances of AlphaStock, AlphaStock-NP and AlphaStock-NC are better than that of other baselines again. This verifies the effectiveness of our model over the Chinese markets. By further comparing Table 2 with Table 1, it turns out that the risk of our model measured by AVOL and MDD in the Chinese markets is higher than that in the U.S. markets. This might be attributable to the market faultiness of emerging countries like China, with more speculative capital but less effective governance. The lack of short sell mechanism also contributes to the imbalance of market forces. The AVOL and MDD of the Market and other baselines in the Chinese markets are also higher than that in the U.S. markets. Compared with these baselines, the risk control ability of our model is still competitive. To sum up, the experimental results in Table 2 indicate the robustness of our model over emerging markets.

5.6 Investment Strategies Interpretation

Here, we try to interpret the underlying investment strategies of AlphaStock, which is crucial for practitioners to better understanding this model. To this end, we use $\bar{\delta}_{x_p}$ in Eq. (30) to measure the influence of the stock features defined in Section 3.1 to AlphaStock's winner selection. Figures 3(a)-3(b) plot the influences from the trading features. The vertical axis denotes the influence

strengths indicated by $\bar{\delta}_{x_q}$, and the horizontal axis denotes how many months before the trading time. For example, the bar indexed by "-12" of the horizontal axis in Fig. 3(a) denotes the influence of stock price rising rate (PR) at the time of twelve months ago.

As shown in Fig. 3(a), the influence of history price rising rate is heterogeneous along the time axis. The PR in long-term months, *i.e.,* 9 to 11 months ahead, has positive influence to winner scores, but for the short-term months, *i.e.,* 1 to 8 months ahead, the influence becomes negative. This result indicates that our model tends to buy the stocks with long-term rapid price increase (valid excellence) or with short-term rapid price retracement (over undervalued). This implies that AlphaStock behaviors like a long-term momentum but short-term reversion mixed strategy. Moreover, since price rising is usually accompanied by frequent stock trading, Fig. 3(b) shows that the $\bar{\delta}_{x_p}$ of trading volumes (TV) has a similar tendency with the price rising rate (PR). Finally, as shown in Fig. 3(c), the volatilities (VOL) have negative influence to winner scores for all history months. It means that our model trends to select low volatility stocks as winners, which indeed explains why AlphaStock can adapt to diverse market states.

Fig. 3(d) further exhibits the average influences of different company features to the winner score, *i.e.,* the $\bar{\delta}_{x_p}$ averaged on all history months. It turns out that Market Capitalization (MC), Price-earnings Ratio (PE), and Book-to-market Ratio (BM) have positive influences. The three features are important company valuation factors for a listed company, which indicates that AlphaStock tends to select companies with sound fundamental values. In contrast, dividends mean a part of company values are returned to shareholders and could reduce the intrinsic value of a stock. That is why the influence of Dividends (DIV) is negative in our model.

To sum up, while AlphaStock is an AI-enabled investment strategy, the interpretation analysis proposed in Section 4 can help to extract investment logics from AlphaStock. Specifically, AlphaStock suggests selecting the stocks as winners with *high long-term growth*, *low volatility*, *high intrinsic value*, and *being undervalued recently*.

6 RELATED WORKS

Our work is related to the following research directions.

Financial Investment Strategy: Classic financial investment strategy includes Momentum, Mean Reversion, and Multi-factors. In the first work of BWSL [14], Jegadeesh and Titman found "momentum" could be used to select winners and losers. The momentum strategy buys assets that have had high returns over a past period as winners, and sells those that have had poor returns over the same period. Classic momentum strategies include the Cross Sectional Momentum (CSM) [15] and the Time Series Momentum (TSM) [18].

---

[4]http://www.wind.com.cn/en/Default.html

The mean reversion strategy [20] considers asset prices always return to their mean over a past period, so it buys assets with a price under their historical mean and sells above the historical mean. The multi-factor model [7] uses factors to compute a valuation for each asset and buys/sells those assets with price under/above their valuations. Most of these financial investment strategies can only exploit a certain factor of financial markets and thus might fail in complex market environments.

Deep Learning in Finance: In recent years, deep learning approaches begin to be applied in the financial areas. In the literature, L. Zhang *et al.* proposed to exploit frequency information to predict stock prices [11]. News and social media were used in price prediction in Refs. [12, 27]. Information about events and corporation relationships were used to predict stock prices in Ref. [2, 4]. Most of these works focus on price prediction rather than end-to-end investment portfolio generation like us.

Reinforcement Learning in Finance: The RL approaches used in investment strategies fall in two categories: the value-based and the policy-based [8]. The value-based approaches learn a critic to describe the expected outcomes of markets to trading actions. Typical value-based approaches in investment strategies include Q-learning [19] and deep Q-learning [16]. A defect of value-based approaches is the market environment is too complex to be approximated by a critic. Therefore, policy-based approaches are considered as more suitable to financial markets [8]. The AlphaStock model also belongs to this category. A classic policy-based RL algorithm in investment strategy is the Recurrent Reinforcement Learning (RRL) [17]. The FDDR [3] model extends the RRL framework using deep neural networks. In the Investor-Imitator model [6], a policy-based deep RL framework was proposed to imitate the behaviors of different types of investors. Compared with RRL and its deep learning extensions, which focus on exploiting sequential dependence in financial signals, our AlphaStock model pays more attention to the interrelationships among assets. Moreover, deep RL approaches are often hard to deployed in real-life applications for unexplainable deep network structures. The interpretation tools offered by our model can solve this problem.

7 CONCLUSIONS

In this paper, we proposed a RL-based deep attention network to design a BWSL strategy called AlphaStock. We also designed a sensitivity analysis method to interpret the investment logics of our model. Compared with existing RL-based investment strategies, AlphaStock fully exploits the interrelationship among stocks, and opens a door for solving the "black box" problem of using deep learning models in financial markets. The back-testing and simulation experiments over U.S. and Chinese stock markets showed that AlphaStock performed much better than other competing strategies. Interestingly, AlphaStock suggests buying stocks with high long-term growth, low volatility, high intrinsic value, and being undervalued recently.

ACKNOWLEDGMENTS

REFERENCES

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. 2018. Sanity checks for saliency maps. In *NIPS'18*. 9525–9536.

[2] Yingmei Chen, Zhongyu Wei, and Xuanjing Huang. 2018. Incorporating Corporation Relationship via Graph Convolutional Neural Networks for Stock Price Prediction. In *CIKM'18*. ACM, 1655–1658.

[3] Yue Deng, Feng Bao, Youyong Kong, Zhiquan Ren, and Qionghai Dai. 2017. Deep direct reinforcement learning for financial signal representation and trading. *IEEE TNNLS* 28, 3 (2017), 653–664.

[4] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. 2015. Deep learning for event-driven stock prediction.. In *IJCAI'15*. 2327–2333.

[5] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. 2016. Knowledge-driven event embedding for stock prediction. In *COLING'16*. 2133–2142.

[6] Yi Ding, Weiqing Liu, Jiang Bian, Daoqiang Zhang, and Tie-Yan Liu. 2018. Investor-Imitator: A Framework for Trading Knowledge Extraction. In *KDD'18*. ACM, 1310–1319.

[7] Eugene F Fama and Kenneth R French. 1996. Multifactor explanations of asset pricing anomalies. *J. Finance* 51, 1 (1996), 55–84.

[8] Thomas G Fischer. 2018. *Reinforcement learning in financial markets-a survey*. Technical Report. FAU Discussion Papers in Economics.

[9] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. 2018. A survey of methods for explaining black box models. *ACM Computing Surveys (CSUR)* 51, 5 (2018), 93.

[10] Sepp Hochreiter and Jurgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (1997), 1735–1780.

[11] Hao Hu and Guo-Jun Qi. 2017. State-Frequency Memory Recurrent Neural Networks. In *ICML'17*. 1568–1577.

[12] Ziniu Hu, Weiqing Liu, Jiang Bian, Xuanzhe Liu, and Tie-Yan Liu. 2018. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. In *WSDM'18*. ACM, 261–269.

[13] Dingjiang Huang, Junlong Zhou, Bin Li, Steven CH Hoi, and Shuigeng Zhou. 2016. Robust median reversion strategy for online portfolio selection. *IEEE TKDE* 28, 9 (2016), 2480–2493.

[14] Narasimhan Jegadeesh and Sheridan Titman. 1993. Returns to buying winners and selling losers: Implications for stock market efficiency. *J. Finance* 48, 1 (1993), 65–91.

[15] Narasimhan Jegadeesh and Sheridan Titman. 2002. Cross-sectional and time-series determinants of momentum returns. *RFS* 15, 1 (2002), 143–157.

[16] Olivier Jin and Hamza El-Saawy. 2016. *Portfolio Management using Reinforcement Learning*. Technical Report. Stanford University.

[17] John Moody, Lizhong Wu, Yuansong Liao, and Matthew Saffell. 1998. Performance functions and reinforcement learning for trading systems and portfolios. *Journal of Forecasting* 17, 5-6 (1998), 441–470.

[18] Tobias J Moskowitz, Yao Hua Ooi, and Lasse Heje Pedersen. 2012. Time series momentum. *J. Financial Economics* 104, 2 (2012), 228–250.

[19] Ralph Neuneier. 1995. Optimal Asset Allocation using Adaptive Dynamic Programming. In *NIPS'95*.

[20] James M Poterba and Lawrence H Summers. 1988. Mean reversion in stock prices: Evidence and implications. *J. Financial Economics* 22, 1 (1988), 27–59.

[21] William F Sharpe. 1994. The sharpe ratio. *JPM* 21, 1 (1994), 49–58.

[22] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. *NIPS'14* (2014), 3104–3112.

[23] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS'17*. 5998–6008.

[25] Jingyuan Wang, Qian Gu, Junjie Wu, Guannan Liu, and Zhang Xiong. 2016. Traffic speed prediction and congestion source exploration: A deep learning method. In *ICDM'16*. IEEE, 499–508.

[26] Jingyuan Wang, Ze Wang, Jianfeng Li, and Junjie Wu. 2018. Multilevel wavelet decomposition network for interpretable time series analysis. In *KDD'18*. ACM, 2437–2446.

[27] Yumo Xu and Shay B Cohen. 2018. Stock movement prediction from tweets and historical prices. In *ACL'18*, Vol. 1. 1970–1979.